Manual de FreeBSD

Manual de FreeBSD

Revisión: 43184
Febrero 1999 por .
Copyright © 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010 The FreeBSD Documentation Project

Resumen

¡Bienvenido a FreeBSD! Este manual cubre la instalación y uso diario de *FreeBSD 10.0-RELEASE* y *FreeBSD 9.3-RELEASE*. Este manual está en *constante evolución* y es el resultado del trabajo de muchas personas. Algunas secciones no están completas y otras necesitan ser actualizadas. Si está interesado en colaborar en este proyecto envíe un mensaje de correo electrónico a lista de correo del proyecto de documentación de FreeBSD. La última versión de este documento está siempre disponible en el sitio web de FreeBSD. También puede encontrarla en diferentes formatos y opciones de compresión en el servidor FTP de FreeBSD o en las numerosas réplicas. Si prefiere una copia en papel de este manual puede comprarla en FreeBSD Mall. También es posible hacer búsquedas en este manual.

La redistribución y uso en fuente (SGML DocBook) y formatos "compilados" (SGML, HTML, PDF, PostScript, RTF y cualquier otro) con o sin modificación, esta permitida siempre y cuando se cumplan las siguientes condiciones:

1. La redistribución del código fuente (SGML DocBook) debe contener esta información de copyright, esta lista de condiciones y el siguiente texto como las primeras líneas del fichero sin modificar.

2. La redistribución en formatos compilados (transformado a otros DTDs, convertidos a PDF, PostScript, RTF y otros formatos) debe reproducir el siguiente texto de copyright, esta lista de condiciones y el siguiente texto en la documentación y/u otros materiales entregados con la distribución.

Importante

THIS DOCUMENTATION IS PROVIDED BY THE FREEBSD DOCUMENTATION PROJECT "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FREEBSD DOCUMENTATION PROJECT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

FreeBSD is a registered trademark of the FreeBSD Foundation.

3Com and HomeConnect are registered trademarks of 3Com Corporation.

3ware is a registered trademarks of 3ware Inc.

ARM is a registered trademark of ARM Limited.

Adaptec is a registered trademark of Adaptec, Inc.

Adobe, Acrobat, Acrobat Reader, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Apple, AirPort, FireWire, Mac, Macintosh, Mac OS, Quicktime, and TrueType are trademarks of Apple Computer, Inc., registered in the United States and other countries.

Corel and WordPerfect are trademarks or registered trademarks of Corel Corporation and/or its subsidiaries in Canada, the United States and/or other countries.

Sound Blaster is a trademark of Creative Technology Ltd. in the United States and/or other countries.

CVSup is a registered trademark of John D. Polstra.

Heidelberg, Helvetica, Palatino, and Times Roman are either registered trademarks or trademarks of Heidelberger Druckmaschinen AG in the U.S. and other countries.

IBM, AIX, OS/2, PowerPC, PS/2, S/390, and ThinkPad are trademarks of International Business Machines Corporation in the United States, other countries, or both.

IEEE, POSIX, and 802 are registered trademarks of Institute of Electrical and Electronics Engineers, Inc. in the United States.

Intel, Celeron, EtherExpress, i386, i486, Itanium, Pentium, and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Intuit and Quicken are registered trademarks and/or registered service marks of Intuit Inc., or one of its subsidiaries, in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

LSI Logic, AcceleRAID, eXtremeRAID, MegaRAID and Mylex are trademarks or registered trademarks of LSI Logic Corp.

M-Systems and DiskOnChip are trademarks or registered trademarks of M-Systems Flash Disk Pioneers, Ltd.

Macromedia, Flash, and Shockwave are trademarks or registered trademarks of Macromedia, Inc. in the United States and/or other countries.

Microsoft, IntelliMouse, MS-DOS, Outlook, Windows, Windows Media and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Netscape and the Netscape Navigator are registered trademarks of Netscape Communications Corporation in the U.S. and other countries.

GateD and NextHop are registered and unregistered trademarks of NextHop in the U.S. and other countries.

Motif, OSF/1, and UNIX are registered trademarks and IT DialTone and The Open Group are trademarks of The Open Group in the United States and other countries.

Oracle is a registered trademark of Oracle Corporation.

PowerQuest and PartitionMagic are registered trademarks of PowerQuest Corporation in the United States and/or other countries.

RealNetworks, RealPlayer, and RealAudio are the registered trademarks of RealNetworks, Inc.

Red Hat, RPM, are trademarks or registered trademarks of Red Hat, Inc. in the United States and other countries.

SAP, R/3, and mySAP are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world.

Sun, Sun Microsystems, Java, Java Virtual Machine, JDK, JRE, JSP, JVM, Netra, Solaris, StarOffice and SunOS are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Symantec and Ghost are registered trademarks of Symantec Corporation in the United States and other countries.

MATLAB is a registered trademark of The MathWorks, Inc.

SpeedTouch is a trademark of Thomson.

U.S. Robotics and Sportster are registered trademarks of U.S. Robotics Corporation.

VMware is a trademark of VMware, Inc.

Waterloo Maple and Maple are trademarks or registered trademarks of Waterloo Maple Inc.

Mathematica is a registered trademark of Wolfram Research, Inc.

XFree86 is a trademark of The XFree86 Project, Inc.

Ogg Vorbis and Xiph.Org are trademarks of Xiph.Org.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this document, and the FreeBSD Project was aware of the trademark claim, the designations have been followed by the “™” or the “®” symbol.

Tabla de contenidos

Prefacio xix
I. Primeros pasos 1
1. Introducción 5
1.1. Sinopsis 5
1.2. Bienvenido a FreeBSD 5
1.3. Acerca del Proyecto FreeBSD 9
2. Instalación de FreeBSD 17
2.1. Sinopsis 17
2.2. Requisitos de hardware 18
2.3. Tareas anteriores a la instalación 20
2.4. Inicio de la instalación 28
2.5. ?Qué es sysinstall? 35
2.6. Asignación de espacio en disco 40
2.7. Elección de qué instalar 56
2.8. Elección del medio de instalación 59
2.9. El punto sin retorno 61
2.10. Después de la instalación 63
2.11. Solución de problemas 97
2.12. Guía avanzada de instalación 101
2.13. Cómo preparar su propio medio de instalación 103
3. Conceptos básicos de Unix 109
3.1. Sinopsis 109
3.2. Consolas virtuales y terminales 109
3.3. Permisos 113
3.4. Estructura de directorios 119
3.5. Organización de disco 122
3.6. Montaje y desmontaje de sistemas de ficheros 131
3.7. Procesos 134
3.8. Dæmons, señales y cómo matar procesos 136
3.9. Shells 139
3.10. Editores de texto 142
3.11. Dispositivos y nodos de dispositivos 143
3.12. Formatos binarios 143
3.13. Más información 145
4. Instalación de aplicaciones: «packages» y ports 149
4.1. Sinopsis 149
4.2. Aproximación a la instalación de software 149
4.3. Cómo encontrar aplicaciones 152
4.4. Uso del sistema de packages 153
4.5. Uso de la colección de ports 155
4.6. Después de instalar un port 168
4.7. Ports que no funcionan 169
5. El sistema X Window 171
5.1. Sinopsis 171

5.2. Entender X 172
5.3. Instalar X11 175
5.4. Configuración de X11 176
5.5. Uso de tipos en X11 181
5.6. El gestor de pantalla X 186
5.7. Entornos de escritorio 189
II. Tareas comunes 195
6. Aplicaciones de escritorio 199
6.1. Sinopsis 199
6.2. Navegadores 200
6.3. Productividad 203
6.4. Visores de documentos 206
6.5. Finanzas 208
6.6. Resumen 210
7. Multimedia 211
7.1. Sinopsis 211
7.2. Configuración de la tarjeta de sonido 212
7.3. Sonido MP3 216
7.4. Reproducción de vídeo 219
7.5. Configuración de tarjetas de TV 229
7.6. Escáneres de imágenes 230
8. Configuración del kernel de FreeBSD 237
8.1. Sinopsis 237
8.2. ?Qué razones hay para compilar un kernel personalizado? 237
8.3. Inventario de hardware del sistema 238
8.4. Controladores del kernel, subsistemas y módulos 240
8.5. Compilación e instalación de un kernel personalizado 240
8.6. El fichero de configuración 244
8.7. Qué hacer si algo va mal 260
9. Imprimir 263
9.1. Sinopsis 263
9.2. Introducción 264
9.3. Configuración básica 265
9.4. Configuración avanzada de impresoras 265
9.5. Cómo utilizar impresoras 265
9.6. Alternativas a LPD 265
9.7. Solución de problemas 265
10. Compatibilidad binaria con Linux 267
10.1. Sinopsis 267
10.2. Instalación 268
10.3. Instalación de Mathematica® 272
10.4. Instalación de Maple™ 274
10.5. Instalación de MATLAB® 276
10.6. Instalación de Oracle® 280
10.7. Instalación de SAP® R/3® 284
10.8. Temas avanzados 308

III. Administración del sistema. 311
11. Configuración y Adaptación del Sistema 317
11.1. Configuración de Tarjetas de Red 317
11.2. “Arrancar servicios” 317
11.3. “Soft Updates” 317
11.4. Añadir espacio swap 317
12. El proceso de arranque en FreeBSD 319
12.1. Sinopsis 319
12.2. El problema que representa arrancar el sistema 319
12.3. El RMA y las etapas de arranque uno, dos y tres 321
12.4. Interacción con el kernel durante el arranque 326
12.5. Device Hints 326
12.6. Init: inicialización del proceso de control 327
12.7. Secuencia de apagado 328
13. Usuarios y administración básica de cuentas 329
13.1. Sinopsis 329
13.2. Introducción 329
13.3. La cuenta superusuario 331
13.4. Cuentas de sistema 332
13.5. Cuentas de usuario 332
13.6. Modificación de cuentas 332
13.7. Limitar a los usuarios 339
13.8. Personalizar a los usuarios 342
13.9. Grupos 342
14. Seguridad 345
14.1. Sinopsis 345
14.2. Introducción 346
14.3. Asegurar FreeBSD 348
14.4. DES, MD5 y Crypt 357
14.5. Contraseñas de un solo uso 358
14.6. TCP Wrappers 364
14.7. KerberosIV 368
14.8. Kerberos5 376
14.9. OpenSSL 386
14.10. VPN sobre IPsec 390
14.11. OpenSSH 404
14.12. Listas de control de acceso a sistemas de ficheros 411
14.13. Monitorización de fallos de seguridad de aplicaciones 413
14.14. FreeBSD Security Advisories 414
14.15. Contabilidad de procesos 417
15. Jaulas 419
15.1. Sinopsis 419
15.2. Términos relacionados con las jaulas 419
15.3. Introducción 420
15.4. Creación y gestión de jaulas 421
15.5. Administración y personalización a fondo 424

15.6. Uso de las jaulas 425
16. Mandatory Access Control 433
16.1. Sinopsis 433
16.2. Términos clave en este capítulo 433
16.3. Explicación de MAC 433
16.4. Las etiquetas MAC 433
16.5. Configuración de módulos 433
16.6. El módulo MAC ifoff 433
16.7. El módulo MAC portacl 433
16.8. Políticas de etiquetas MAC 434
16.9. El módulo MAC partition 434
16.10. El módulo de seguridad multinivel MAC 434
16.11. El módulo MAC Biba 434
16.12. El módulo MAC LOMAC 434
16.13. Implementación de un entorno seguro con MAC 434
16.14. Otro ejemplo: Uso de MAC para restringir un servidor web 434
16.15. Depuración de errores en MAC 434
17. Auditoría de eventos de seguridad 435
17.1. * 435
18. Almacenamiento 437
18.1. Sinopsis 437
18.2. Nombres de dispositivo 437
18.3. Añadir discos 438
18.4. RAID 441
18.5. Dispositivos de almacenamiento USB 447
18.6. Creación y uso de medios ópticos (CD) 449
18.7. Crear y utilizar medios ópticos (DVDs) 456
18.8. Creación y uso de disquetes (floppies) 462
18.9. Creación y uso de cintas de datos 464
18.10. Respaldos en disquetes 468
18.11. Bases para respaldos 469
18.12. Sistemas de ficheros en red, memoria y respaldados en fichero ... 478
18.13. Instantáneas (“snapshots”) de sistemas de ficheros 483
18.14. Cuotas en sistemas de ficheros 484
18.15. Cifrado de particiones de disco 488
19. GEOM: Marco de trabajo modular de transformación de discos 495
19.1. Sinopsis 495
19.2. Introducción a GEOM 495
19.3. RAID0 - Distribución por bandas 496
19.4. RAID1 - Replicación 497
20. El Gestor de Volúmenes Vinum 503
20.1. Sinopsis 503
20.2. Los Discos son Demasiado Pequeños 503
20.3. Cuellos de Botella en el Acceso 503
20.4. Integridad de Datos 504
20.5. Objetos Vinum 504

20.6. Ejemplos 505
20.7. Esquema de Nombres de los Objetos 508
20.8. Configuración de Vinum 508
20.9. Uso de Vinum en el Sistema de Ficheros Raíz 509
21. Virtualización 511
21.1. * 511
22. Localización - Uso y configuración de I18N/L10N 513
22.1. Sinopsis 513
22.2. Lo básico 513
22.3. Uso de la localización 514
22.4. Compilación de programas con soporte para I18N 521
22.5. Localización de FreeBSD a idiomas específicos 522
23. Lo último de lo último 527
23.1. Sinopsis 527
23.2. FreeBSD-CURRENT vs. FreeBSD-STABLE 527
23.3. Sincronización de su código fuente 527
23.4. Uso de `make world` 527
23.5. Redes pequeñas 527
IV. Comunicaciones en red 529
24. Comunicaciones serie 533
24.1. Sinopsis 533
24.2. Introducción 533
24.3. Terminales 539
24.4. Servicio dial-in 545
24.5. Servicio dial-out 553
24.6. Configurando la consola serie 557
25. PPP y SLIP 569
25.1. Sinopsis 569
25.2. Uso de User PPP 570
25.3. Uso de Kernel PPP 586
25.4. Uso de PPP sobre Ethernet (PPPoE) 594
25.5. Uso de PPP sobre ATM (PPPoA) 596
25.6. Uso de SLIP 599
26. Cortafuegos 613
26.1. * 613
27. Correo Electrónico 615
27.1. Sinopsis 615
27.2. Utilización del correo electrónico 616
27.3. Configuración de sendmail 619
27.4. Sustitución del Agente de Transferencia de Correo 622
27.5. Depuración de Problemas 626
27.6. Conceptos Avanzados 630
27.7. SMTP con UUCP 633
27.8. Configuración para sólamente enviar correo 636
27.9. Utilización del correo con una conexión mediante módem analógico (dial-up) 637

27.10. Autentificación utilizando SMTP 638
27.11. Agente de Correo de Usuario 640
27.12. Manejo de fetchmail 648
27.13. Uso de procmail 649
28. Servidores de red 651
28.1. * 651
29. Networking avanzado 653
29.1. Resumen 653
29.2. Pasarelas y “routers” 654
29.3. Redes sin cables (“wireless”) 661
29.4. Bluetooth 668
29.5. Puenteado 678
29.6. NFS 681
29.7. Ejecución sin disco duro 689
29.8. RDSI 700
29.9. NIS/YP 705
29.10. DHCP 727
29.11. DNS 733
29.12. NTP 748
29.13. Traducción de direcciones de red 751
29.14. El “Superservidor” inetd 756
29.15. Línea IP paralela (PLIP) 761
29.16. IPv6 763
29.17. ATM en FreeBSD 5.X 769
V. Apéndices 773
30. Cómo obtener FreeBSD 777
30.1. Servidores FTP 777
30.2. Uso de CVSup 777
31. Bibliografía 779
31.1. Libros y revistas específicas sobre FreeBSD 779
31.2. Guías de usuario 779
31.3. Guías de administrador 780
31.4. Guías de programadores 780
31.5. El sistema operativo por dentro 781
31.6. Referencia de seguridad 782
31.7. Referencia de hardware 782
31.8. Historia de UNIX 782
31.9. Diarios y revistas 783
A. Recursos en Internet 785
A.1. Listas de correo 785
A.2. Grupos de noticias de Usenet 797
A.3. Servidores WWW 798
A.4. Direcciones de correo electrónico 801
A.5. Cuentas shell 801
32. PGP keys 803
32.1. Responsables 803

32.2. Miembros del Core Team 804
Índice 813

Lista de figuras

2.1. Ejemplo de resultado de prueba de dispositivos 33
2.2. Menú de selección de país 34
2.3. Salir de Sysinstall 35
2.4. Selección de «Usage» en el menú principal de sysinstall 36
2.5. Selección del menú de documentación 36
2.6. El menú de documentación de sysinstall 37
2.7. Menú principal de sysinstall 38
2.8. Menú de esquemas de teclado de sysinstall 38
2.9. Menú principal de sysinstall 39
2.10. Opciones de sysinstall 39
2.11. Comenzar una instalación estándar 40
2.12. Elija en qué unidad usar FDisk 44
2.13. Un ejemplo de particionamiento típico con FDisk 45
2.14. Partición con FDisk usando el disco completo 46
2.15. Menú de gestores de arranque de sysinstall 47
2.16. Salir de la selección de unidad 48
2.17. El editor Disklabel 52
2.18. Editor Disklabel con valores por omisión 53
2.19. Liberar espacio para la partición raíz 54
2.20. Edición del tamaño de la partición raíz 54
2.21. Elegir el tipo de partición raíz 55
2.22. Elegir el punto de montaje del sistema de ficheros raíz 55
2.23. Editor Disklabel 56
2.24. Elección de distribuciones 57
2.25. Confirmar la elección de distribuciones 59
2.26. Choose Installation Media 60
2.27. Selección de un dispositivo Ethernet 64
2.28. Configuración de ed0 65
2.29. Edición de `inetd.conf` 67
2.30. Edición del mensaje de bienvenida de FTP 70
2.31. Edición de `exports` 71
2.32. Opciones de configuración de la consola del sistema 72
2.33. Opciones del salvapantallas 73
2.34. Retardo del salvapantallas 73
2.35. Salida del menú de configuración de la consola del sistema 74
2.36. Elección de región 75
2.37. Elección de país 75
2.38. Elección de zona horaria 76
2.39. Elija el protocolo que usa el ratón 77
2.40. Ajuste del protocolo del ratón 78
2.41. Configuración del puerto del ratón 78
2.42. Configuración del puerto del ratón 79
2.43. Arranque del dæmon del ratón 79
2.44. Prueba del dæmon del ratón 80

2.45. Categorías de «packages» 81
2.46. Selección de «packages» 81
2.47. Instalación de «packages» 82
2.48. Confirmación previa a la instalación de «packages» 83
2.49. Selección de «Añadir un usuario» 84
2.50. Añadir la información del usuario 84
2.51. Salir de la gestión de usuarios y grupos 86
2.52. Salir de la instalación 87
2.53. Primer nivel de servicios de red 89
2.54. Elegir el MTA del sistema 90
2.55. Configuración de Ntpdate 91
2.56. Segundo nivel de configuración de red 92
20.1. Organización Concatenada 503
20.2. Organización con “Striping” 504
20.3. Organización en RAID-5 504
20.4. Un Volumen Vinum Sencillo 505
20.5. Un Volumen Vinum Replicado 506
20.6. Un Volumen Vinum en “Striping” 507
20.7. Un Volumen Vinum Replicado y en “Striping” 508

Lista de tablas

2.1. Ejemplo de inventario de dispositivos 20
2.2. Estructura de particiones del primer disco 48
2.3. Esquema de particiones para varios discos 50
2.4. Nombres y funciones de las imágenes de FreeBSD 6.*X* and 7.*X* 104
3.1. Códigos de dispositivos de disco 129
18.1. Convenciones para nombrar discos físicos 438
29.1. Cableado de una conexión de cable paralelo para redes 761
29.2. Direcciones IPv6 reservadas 765

Lista de ejemplos

2.1. Uso de una partición sin cambiar nada 22
2.2. Reducir una partición existente 23
3.1. Ejemplo de nombres de disco, «slice» y partición 130
3.2. Modelo conceptual de un disco 130
4.1. Descarga de un paquete manualmente e instalación en nuestro sistema 153
12.1. Pantalla `boot0` 321
12.2. Pantalla de `boot2` 322
12.3. Una consola insegura en /etc/ttys 327
13.1. Configuración de adduser 333
13.2. Eliminación interactiva de cuenta con rmuser 336
13.3. chpass interactivo ejecutado por el superusuario 336
13.4. chpass interactivo ejecutado por un usuario normal 337
13.5. Cambio de tu contraseña 338
13.6. Cambio de la contraseña de otro usuario como superusuario 338
13.7. Añadir un grupo usando pw(8) 342
13.8. Añadir a alguien a un grupo usando pw(8) 343
13.9. Determinar pertenencia a grupos con id(1) 343
14.1. Uso de SSH para crear un túnel seguro para SMTP 409
18.1. Utilizando `dump` a través de ssh 470
18.2. Uso de `dump` a través de ssh con `RSH` configurada 470
18.3. Un “script” para la creación de discos flexibles de arranque 474
18.4. Uso de vnconfig para montar una imagen de un sistema de ficheros bajo FreeBSD 4.X 478
18.5. Creación de una imagen nueva de un sistema de ficheros respaldado en un archivo con `vnconfig` 479
18.6. Uso de `mdconfig` para montar una imagen de un sistema de ficheros en FreeBSD 5.X 480
18.7. Creación de un disco respaldado en fichero con `mdconfig` 480
18.8. Configurar y montar un disco basado en un fichero con `mdmfs` 481
18.9. Disco de memoria md en FreeBSD 4.X 481
18.10. Creación de un nuevo disco basado en memoria con `mdconfig` 482
18.11. Creación de un nuevo disco basado en memoria con `mdmfs` 482
24.1. Agregando entradas de terminal a `/etc/ttys` 542
27.1. Configuración de la base de datos de acceso de sendmail 619
27.2. Mail Aliases 620
27.3. Ejemplo de asociación de correo de dominio virtual 622
29.1. Anclaje de una exportación utilizando amd 686
29.2. Sucursal o red doméstica 703
29.3. Oficina central u otra LAN 704
29.4. Envío de una señal HANGUP a inetd 758

Prefacio

A quién va dirigido este texto

La primera sección de este libro está pensada para guiar a los recién llegados a FreeBSD durante el proceso de instalación y presentarles los conceptos y convenciones que dan entidad a UNIX®. Para sacar provecho de esta sección lo único que el lector necesitará son ganas de explorar y habilidad para ir interiorizando nuevos conceptos a medida que le van planteando.

La siguiente sección, la segunda, mucho más extensa, es una referencia completa de todo tipo de temas de interés para administradores de sistemas FreeBSD. Es posible que alguno de esos capítulos requiera que previamente lea otros textos; si es el caso esto se anuncia en la sinopsis situada al principio de cada capítulo.

Hay una lista de fuentes de información adicionales en el Capítulo 31, *Bibliografía*.

Cambios desde la segunda edición

Esta tercera edición es la culminación de más de dos años de trabajo de los miembros del FreeBSD Documentation Project. Estos son los cambios más importantes que encontrará en esta nueva edición:

- Capítulo 11, *Configuración y Adaptación del Sistema*, configuración y adaptación del sistema, ha sido ampliado con nuevos datos sobre el sistema de gestión de energía y recursos ACPI, la utilidad del sistema `cron` y más opciones de personalización del kernel.

- Capítulo 14, *Seguridad*, seguridad, ha sido ampliado con información sobre redes privadas virtuales (VPN), listas de control de accesos al sistema de ficheros (más conocidas por ACL) e información sobre avisos de seguridad.

- Capítulo 16, *Mandatory Access Control*, Mandatory Access Control (MAC), es un capítulo nuevo. En él se expone qué es MAC y de qué forma puede utilizarse en FreeBSD como complemento de la seguridad del sistema.

- Capítulo 18, *Almacenamiento*, almacenamiento, se ha ampliado con información sobre dispositivos USB de almacenamiento, instantáneas (“snapshots”) de sistemas de ficheros, sistemas de ficheros basados en ficheros y en red, y particiones de disco cifradas.

- Capítulo 20, *El Gestor de Volúmenes Vinum*, Vinum, es un capítulo nuevo. Describe el uso de Vinum, un gestor de volúmenes lógicos que permite disponer de discos lógicos independientes de dispositivo, así como de RAID-0, RAID-1 y RAID-5 por software.

- Se ha añadido una sección de depuración de problemas al Capítulo 25, *PPP y SLIP*, PPP y SLIP.

- Capítulo 27, *Correo Electrónico*, correo electrónico, ha sido ampliado con información sobre el uso de MTA alternativos, autentificación SMTP, UUCP, fetchmail, procmail así como otros temas avanzados relacionados con el correo.

- Capítulo 28, *Servidores de red*, servidores de red, es un capítulo nuevo. En él se incluye información sobre la configuración de Apache HTTP Server, ftpd y cómo configurar Samba para su uso con clientes Microsoft® Windows®. Ciertas partes de Capítulo 29, *Networking avanzado*, "networking" avanzado, se han trasladado a este capítulo para dar más solidez al contenido.

- Capítulo 29, *Networking avanzado*, "networking" avanzado, incluye en esta edición nuevos datos sobre el uso de dispositivos Bluetooth® en FreeBSD, la configuración de redes "wireless" y el "networking" ATM (Asynchronous Transfer Mode).

- Se ha creado un glosario para disponer de una ubicación centralizada donde encontrar definiciones de términos técnicos que se utilizan a lo largo del libro.

- Las tablas e imágenes que aparecen en el libro han experimentado diversas mejoras est´ticas.

Cambios desde la primera edición

La segunda edición fue la culminación de más de dos años de trabajo de los miembros del FreeBSD Documentation Project. Estos son los principales cambios que encontrará en esta edición:

- Se ha creado un índice completo.

- Todos los esquemas ASCII han sido reemplazados por diagramas gráficos.

- Se ha añadido una sinopsis estándar a cada capítulo donde se ofrece un breve sumario del contenido del capítulo y qué se espera que sepa el lector para poder sacarle provecho.

- Se ha reorganizado de forma lógica el contenido en tres partes: "primeros pasos", "administración del sistema " y "apéndices".

- El Capítulo 2, *Instalación de FreeBSD* ("instalación de FreeBSD") ha sido reescrito completamente e incluye gran cantidad de capturas de pantalla para facilitar su comprensión a los nuevos usuarios.

- El Capítulo 3, *Conceptos básicos de Unix* ("conceptos básicos de UNIX®") se ha ampliado con más información sobre procesos, dæmons y señales.

- El Capítulo 4, *Instalación de aplicaciones: «packages» y ports* (“instalación de aplicaciones”) incluye información sobre la gestión de paquetes binarios.
- El Capítulo 5, *El sistema X Window*, (“El sistema X Window”) ha sido reescrito totalmente, con especial énfasis en el uso de modernas tecnologías de escritorio como KDE y GNOME en XFree86™ 4.X.
- El Capítulo 12, *El proceso de arranque en FreeBSD* (“El proceso de arranque de FreeBSD”) ha sido ampliado.
- El Capítulo 18, *Almacenamiento* (“Almacenamiento”) ha sido el fruto de refundir el contenido de lo eran dos capítulos sobre “discos” y “copias de seguridad”. Pensamos que estos temas eran más fáciles de entender si se trataban en un mismo capítulo. Se ha añadido también una sección sobre RAID (tanto hardware como software).
- El Capítulo 24, *Comunicaciones serie* (“communicaciones serie”) ha sido reorganizado y actualizado completamente con información sobre FreeBSD 4.X y 5.X.
- El Capítulo 25, *PPP y SLIP* (“PPP y SLIP”) ha sido actualizado en gran parte.
- Se han añadido muchas secciones nuevas al Capítulo 29, *Networking avanzado* (“Networking avanzado”).
- El Capítulo 27, *Correo Electrónico* (“correo electrónico” incluye ahora más información sobre la configuración de sendmail.
- El Capítulo 10, *Compatibilidad binaria con Linux* (“compatibilidad conLinux®”) se ha ampliado con información sobre la instalación de Oracle® y SAP® R/3®.
- En esta segunda edición se incluyen también estos nuevos contenidos:
  - Configuración y adaptación del sistema (Capítulo 11, *Configuración y Adaptación del Sistema*).
  - Multimedia (Capítulo 7, *Multimedia*)

Cómo se organiza este libro

Este libro se divide en cinco secciones lógicamente distintas. La primera, *primeros pasos*, trata sobre la instalación y el uso más básico de FreeBSD. Está pensada para que el lector los recorra en secuencialmente, aunque puede saltarse algunos si tratan sobre temas que le resulten familiares. La segunda sección, *Tareas comunes*, está dedicada a las características de FreeBSD que suelen utilizarse con más frecuencia. Esta sección, así como las siguientes, pueden leerse en cualquier orden. Cada capítulo comienza con una sinopsis sucinta en la que se describe el contenido del capítulo y qué es lo que se espera que sepa el lector antes de leerlo. Esto se hace para dar ocasión al lector ocasional a que elija el capí-

tulo que más pueda interesarle. La tercera sección, *Administración del sistema*, trata sobre diversos aspectos de la administración del sistema. La cuarta sección, *Redes y comunicaciones*, está dedicada al "networking" y temas relacionados con los servidores. La quinta sección dispone de apéndices con información de referencia.

Capítulo 1, Introducción, *Presentación*
: Este capítulo, como su nombre indica, presenta FreeBSD al usuario. En él se cuenta la historia del Proyecto FreeBSD, sus objetivos y su modelo de desarrollo.

Capítulo 2, Instalación de FreeBSD, *Instalación*
: Guía al usuario a través de un proceso completo de instalación. Se explican también algunos aspectos avanzados como la instalación a través de una consola serie.

Capítulo 3, Conceptos básicos de Unix, *Conceptos básicos de UNIX®*
: Se explican los comandos más básicos y el funcionamiento del sistema operativo FreeBSD. Si ya se tiene experiencia con Linux® u otro tipo de UNIX® posiblemente sea este un capítulo que no deba leerse.

Capítulo 4, Instalación de aplicaciones: «packages» y ports, *Instalación de aplicaciones*
: Se explica la instalación de software desarrollado por personas ajenas al proyecto, tanto mediante la innovadora "Colección de Ports" de FreeBSD como mediante paquetes binarios estándar ("packages").

Capítulo 5, El sistema X Window, *El sistema X Window*
: Describe el sistema X Windows en general y su uso en FreeBSD en particular. Describe también entornos de escritorio de uso común como KDE y GNOME.

Capítulo 6, Aplicaciones de escritorio, *Aplicaciones de escritorio*
: Se enumeran algunas aplicaciones de escritorio muy comunes, como navegadores web y suites ofimáticas, y cómo instalar estas aplicaciones en FreeBSD.

Capítulo 7, Multimedia, *Multimedia*
: Trata sobre la configuración la reproducción de sonido y vídeo en su sistema. Describe también algunas aplicaciones de sonido y vídeo.

Capítulo 8, Configuración del kernel de FreeBSD, *Configuración del kernel de FreeBSD*
: Explica bajo qué circunstancias tendrá que configurar un nuevo kernel y facilita instrucciones detalladas para la configuración, compilación e instalación de un kernel personalizado.

Capítulo 9, Imprimir, *Imprimir*
: Describe la gestión de impresoras en FreeBSD, abarcando aspectos diversos como las páginas "banner", las cuentas de impresión así como la configuración inicial.

Capítulo 10, Compatibilidad binaria con Linux, *Compatibilidad binaria con Linux®*
: Describe las características de compatibilidad con Linux® de FreeBSD. Incluye también instrucciones detalladas de instalación de varias aplicaciones Linux® muy populares, como Oracle®, SAP® R/3® y Mathematica®.

Capítulo 11, Configuración y Adaptación del Sistema, Configuración y adaptación del sistema
Describe los parámetros que los administradores de sistemas tienen a su alcance para hacer que FreeBSD rinda al máximo. Describe también los diversos ficheros de configuración que se usan en FreeBSD y dónde están.

Capítulo 12, El proceso de arranque en FreeBSD, El proceso de arranque en FreeBSD
Describe el proceso de arranque de FreeBSD y explica cómo controlar este proceso mediante opciones de configuración.

Capítulo 13, Usuarios y administración básica de cuentas, Usuarios y administración básica de cuentas
Describe la creación y gestión de cuentas de usuario. Trata también sobre la limitación de recursos que puede aplicarse sobre los mismos, así como otras tareas administrativas.

Capítulo 14, Seguridad, Seguridad
Describe las abudantes herramientas diferentes que pueden ayudar a que su sistema FreeBSD esté y permanezca seguro. Entre ellas encontrará a Kerberos, IPsec y OpenSSH.

Capítulo 15, Jaulas, Jaulas
Describe el uso de jaulas en FreeBSD y el avance que este "framework" supone respecto al tradicional uso de chroot que se hacía en FreeBSD.

Capítulo 16, Mandatory Access Control, Mandatory Access Control
Explica qué es Mandatory Access Control (MAC) y cómo puede usarse este mecanismo para hacer más seguro FreeBSD

Capítulo 17, Auditoría de eventos de seguridad, Auditoría de eventos de seguridad
Describe qué es la auditoría de eventos en FreeBSD cómo instalarla y configurarla y cómo pueden inspeccionarse y monitorizarse dichas auditorías.

Capítulo 18, Almacenamiento, Almacenamiento
Describe cómo gestionar medios de almacenamiento y sistemas de ficheros en FreeBSD, tanto discos físicos, arreglos RAID, medios ópticos o en cinta, como discos en memoria y sistemas de ficheros en red.

Capítulo 19, GEOM: Marco de trabajo modular de transformación de discos, GEOM
Trata sobre el "framework" "GEOM" de FreeBSD y cómo configurar con él alguno de los diversos niveles de RAID que admite.

Capítulo 20, El Gestor de Volúmenes Vinum, Vinum
Describe cómo usar Vinum, un gestor de volúmenes lógicos que permite tanto el uso de discos lógicos independientes de dispositivo, como RAID-0, RAID-1 y RAID-5 por software.

Capítulo 21, Virtualización, Virtualización
Describe los sistemas de virtualización disponibles en FreeBSD y cómo utilizarlos.

Capítulo 22, Localización - Uso y configuración de I18N/L10N, *Localización*
Describe el uso en FreeBSD de idiomas distintos del inglés. Trata tanto la localización del sistema como de las aplicaciones.

Capítulo 23, Lo último de lo último, *Lo último de lo último*
Explica las diferencias existentes entre FreeBSD-STABLE, FreeBSD-CURRENT y las releases de FreeBSD. Describe también qué tipos de usuario pueden beneficiarse de seguir el desarrollo contínuo del sistema en su propia máquina y muestra el proceso a seguir.

Capítulo 24, Comunicaciones serie, *Comunicaciones serie*
Explica cómo conectar terminales serie y módems a su sistema FreeBSD tanto para conexiones entrantes como salientes.

Capítulo 25, PPP y SLIP, *PPP y SLIP*
Describe cómo usar en FreeBSD PPP, SLIP o PPP sobre Ethernet para conexiones a sistemas remotos.

Capítulo 27, Correo Electrónico, *Correo electrónico*
Detalla los distintos elementos que componen un servidor de correo electrónico y explica diversos aspectos sencillos de la configuración del servidor de correo más extendido: sendmail.

Capítulo 28, Servidores de red, *Servidores de red*
En este capítulo encontrará instrucciones detalladas y ficheros de configuración de ejemplo que le permitirán configurar su sistema FreeBSD como servidor NFS, servidor de nombres, servidor NIS o de sincronización de hora.

Capítulo 26, Cortafuegos, *Cortafuegos*
Explica la filosofía que sustenta los cortafuegos por software y facilita información detallada sobre la configuración de los distintos cortafuegos disponibles en FreeBSD.

Capítulo 29, Networking avanzado, *"Networking" avanzado*
Describe diversos temas relacionados con el "networking", desde compartir la conexión a Internet con otras máquinas de su LAN a diversos aspectos avanzados del encaminamiento de tráfico, pasando por las redes "wireless", Bluetooth®, ATM, IPv6 y mucho más.

Capítulo 30, Cómo obtener FreeBSD, *Cómo conseguir FreeBSD*
Enumera las diferentes fuentes desde la que puede conseguirse FreeBSD en CDROM o DVD, así como los numerosos sitios de Internet desde los que puede descargar e instalar FreeBSD.

Capítulo 31, Bibliografía, *Bibliografía*
Este libro toca tantos temas que es muy posible que el lector se quede a falta de una explicación más detallada. En la bibliografía muchos libros estupendos relacionados con el contenido de este texto.

Apéndice A, Recursos en Internet, *Recursos en Internet*
: Describe los abundantes foros de que disponen los usuarios de FreeBSD para enviar preguntas y participar en conversaciones técnicas sobre FreeBSD.

Capítulo 32, PGP keys, *PGP Keys*
: Lista las claves públicas PGP de varios desarrolladores de FreeBSD.

Convenciones usadas en este libro

Con el ánimo de mantener la consistencia y facilitar la lectura del texto se siguen varias convenciones a lo largo del libro.

Convenciones tipográficas

Cursiva
: Se usa un tipo de letra*cursiva* cuando se citan nombres de fichero, URL, texto en el que se quiere hacer énfasis y cuando un término técnico aparece por primera vez en el texto.

`Tipografía de máquina de escribir`
: Se usa un tipo de letra `de máquina de escribir` cuando se muestran mensajes de error, comandos, variables de entorno, nombres de “ports”, nombres de máquina, nombres de usuario o de grupo, nombres de dispositivo, variables y cuando Se usa un tipo de letra `monospaced` cuando se muestran mensajes de error, comandos, variables de entorno, nombres de “ports”, nombres de máquina, nombres de usuario o de grupo, nombres de dispositivo, variables y cuando se muestran fragmentos de código.

Negrita
: Se usa un tipo de letra negrita en el nombre de aplicaciones, comandos y cuando se muestran claves.

Datos que introduce el usuario

Las claves se muestran en negrita para distinguirlas de cualquier otro texto. Las combinaciones de teclas que implican que sean pulsadas simultáneamente se muestran con el símbolo `+' entre una y otra, como en:

Ctrl+Alt+Supr

Esto significa que el usuario debe pulsar las teclas Ctrl, Ctrl, Alt y Supr al mismo tiempo.

Las teclas que deben pulsarse secuencialmente se separan con comas, como en este ejemplo:

Ctrl+X, Ctrl+S

Esto significa que el usuario debe pulsar simultáneamente las teclas Ctrl y X y después pulsar simultáneamente Ctrl y S.

Ejemplos

Un comienzo como `E:\>` indica un ejemplo de comando de MS-DOS®. A menos que se especifique otra cosa, estos comandos deben ejecutarse en una terminal "Command Prompt" de un sistema Microsoft® Windows® moderno.

```
E:\> tools\fdimage floppies\kern.flp A:
```

Si hay un `#` indica que el comando debe ejecutarse como superusuario en FreeBSD. Puede acceder al sistema como `root` y ejecutar el comando o bien con su usuario habitual y utilizar su(1) para disponer de privilegios de superusuario.

```
# dd if=kern.flp of=/dev/fd0
```

Si el ejemplo comienza con un `%` indica que el comando puede ejecutarse en una cuenta de usuario normal. Salvo que se indique otra cosa se usa la sintaxis de C-shell para asignar valores a variables de entorno u otros comandos.

```
% top
```

Agradecimientos

Este libro representa el esfuerzo de muchos cientos de personas del mundo entero. Tanto si fue en forma de correción de errores de contenido como gramaticales, o bien fue un capítulo entero lo que enviaron, toda contribución ha sido muy valiosa.

Varias compañías han colaborado en el desarrollo de este documento pagando a tiempo completo a varios autores, financiando la publicación, etc. En especial BSDi (posteriormente adquirida por Wind River Systems) pagaron a varios miembros del FreeBSD Documentation Project por trabajar en la mejora lo que fue la primera edición de este libro, que apareció en marzo de 2000 (ISBN 1-57176-241-8). Además de esto, Wind River Systems aportó el dinero para que otros autores pudieran realizar gran cantidad de mejoras en la estructura de creación de material listo para impresión y para pudieran añadir nuevos capítulos. Este trabajo culminó con la publicación de la segunda edición impresa en noviembre de 2001 (ISBN 1-57176-303-1). En 2003 y 2004 FreeBSD Mall, Inc, pagó a varios autores para que se dedicaran a lo que será la tercera edición.

Parte I. Primeros pasos

Esta parte del manual de FreeBSD es para usuarios y administradores nuevos en FreeBSD. El cometido de estos capítulos es:

- Ofrecer una introducción a FreeBSD.
- Guiar a través de una instalación de FreeBSD.
- Explicar conceptos básica de Unix.
- Explicar cómo instalar la gran cantidad de software de terceros disponible para FreeBSD.
- Presentar una introducción al manejo de X Window, el sistema de ventanas de UNIX®, y detallar cómo configurar un entorno de escritorio más productivo.

Se ha intentado minimizar el número de referencias a otras secciones de este documento para evitar el salto entre páginas y facilitar la lectura continuada.

Tabla de contenidos

1. Introducción 5
    1.1. Sinopsis 5
    1.2. Bienvenido a FreeBSD 5
    1.3. Acerca del Proyecto FreeBSD 9
2. Instalación de FreeBSD 17
    2.1. Sinopsis 17
    2.2. Requisitos de hardware 18
    2.3. Tareas anteriores a la instalación 20
    2.4. Inicio de la instalación 28
    2.5. ?Qué es sysinstall? 35
    2.6. Asignación de espacio en disco 40
    2.7. Elección de qué instalar 56
    2.8. Elección del medio de instalación 59
    2.9. El punto sin retorno 61
    2.10. Después de la instalación 63
    2.11. Solución de problemas 97
    2.12. Guía avanzada de instalación 101
    2.13. Cómo preparar su propio medio de instalación 103
3. Conceptos básicos de Unix 109
    3.1. Sinopsis 109
    3.2. Consolas virtuales y terminales 109
    3.3. Permisos 113
    3.4. Estructura de directorios 119
    3.5. Organización de disco 122
    3.6. Montaje y desmontaje de sistemas de ficheros 131
    3.7. Procesos 134
    3.8. Dæmons, señales y cómo matar procesos 136
    3.9. Shells 139
    3.10. Editores de texto 142
    3.11. Dispositivos y nodos de dispositivos 143
    3.12. Formatos binarios 143
    3.13. Más información 145
4. Instalación de aplicaciones: «packages» y ports 149
    4.1. Sinopsis 149
    4.2. Aproximación a la instalación de software 149
    4.3. Cómo encontrar aplicaciones 152
    4.4. Uso del sistema de packages 153
    4.5. Uso de la colección de ports 155
    4.6. Después de instalar un port 168
    4.7. Ports que no funcionan 169
5. El sistema X Window 171
    5.1. Sinopsis 171
    5.2. Entender X 172
    5.3. Instalar X11 175

5.4. Configuración de X11 176
5.5. Uso de tipos en X11 181
5.6. El gestor de pantalla X 186
5.7. Entornos de escritorio 189

Capítulo 1. Introducción

Reesctructurado, reorganizado y con algunas partes reescritas por Jim Mock.

1.1. Sinopsis

Gracias por su interés en FreeBSD. El siguiente capítulo trata varios temas relativos al Proyecto FreeBSD, como su historia, objetivos, modelo de desarrollo, etc.

Después de leer este capítulo sabrá:

- Qué relación guarda FreeBSD con otros sistemas operativos.
- La historia del Proyecto FreeBSD.
- Los objetivos del Proyecto FreeBSD.
- Los fundamentos del modelo de desarrollo de código abierto de FreeBSD.
- Y por supuesto: de dónde procede el nombre "FreeBSD".

1.2. Bienvenido a FreeBSD

FreeBSD es un sistema operativo basado en 4.4BSD-Lite para ordenadores Intel (x86 e Itanium®), AMD64, Alpha™ y Sun UltraSPARC®. Se está trabajando también en versiones para otras arquitecturas. También puede leer sobre la historia de FreeBSD, o sobre la distribución actual. Si cree que puede ayudar al proyecto de algún modo (desarrollando código, donando hardware, dinero, etc) consulte el artículo Contribuir a FreeBSD.

1.2.1. ?Qué puede hacer FreeBSD?

FreeBSD tiene muchas características notables. Algunas de ellas son:

- *Multitarea expropiativa* con prioridades dinámicamente ajustadas para asegurar que distintas aplicaciones y usuarios compartan los recursos del sistema de un modo equitativo, incluso bajo la mayor de las cargas.
- *Servicios multiusuario* que permiten a mucha gente usar un sistema FreeBSD simultáneamente para distintas cosas. Ésto significa, por ejemplo, que los periféricos del sistema como impresoras y dispositivos de cinta son compartidos adecuadamente por varios usuarios del sistema o la red, y que pueden establecerse límites sobre recursos concre-

tos para usuarios o grupos de usuarios, protegiendo los recursos críticos del sistema de un uso abusivo.

- Conexión de *redes TCP/IP* muy robusta, con soporte para estándares industriales como SCTP, DHCP, NFS, NIS, PPP, SLIP, IPSec e IPv6. Esto quiere decir que su sistema FreeBSD puede interactuar fácilmente con otros sistemas y hacer de servidor en una empresa, proporcionando servicios clave como NFS (acceso a ficheros remotos) y servicios de correo electrónico, o proporcinando la presencia en Internet de su organización mediante WWW, FTP, servicios de encaminamiento y cortafuegos.

- La *protección de memoria* garantiza que las aplicaciones (o los usuarios) no se estorben los unos a los otros. Un error catastrófico en una aplicación no afecta al resto.

- FreeBSD es un sistema operativo de *32-bits* (de *64-bits* en Alpha, Itanium®, AMD64, y UltraSPARC®) y fue diseñado como tal desde el principio.

- *X Window System* (X11R6), estándar de la industria, dota a los usuarios una interfaz gráfica (GUI) por el coste de una tarjeta VGA y un monitor comunes, y viene con los fuentes completos.

-

- Hay en Internet miles y miles de aplicaciones *listas para su uso*. FreeBSD es compatible a nivel de código fuente con la mayoría de sistemas UNIX® comerciales; por tanto la mayoría de aplicaciones requieren poco o ningún cambio para compilar en FreeBSD.

- En Internet hay miles de aplicaciones *fáciles de portar*. El código fuente de FreeBSD es compatible con el de los sistemas UNIX® comerciales más populares y por ello la mayoría de las aplicaciones tan sólo necesitan pocos cambios, si es que necesitan alguno, para compilar.

- El diseño de la *memoria virtual* con paginación bajo demanda y de la "caché unificada de VM/buffer" satisface a aplicaciones que requieren grandes cantidades de memoria de forma eficiente aun dando respuestas interactivas a otros usuarios.

- Soporte para *SMP* en máquinas con múltiples CPUs.

-

- Disponer del *código fuente* del sistema entero significa contar con el mayor nivel de control posible sobre su entorno. ?Para qué atarse a una solución propietaria a merced de un fabricante cuando puede tener un verdadero sistema abierto?

- Documentación exhaustiva *en línea*.

- *¡Y mucho más!*

FreeBSD está basado en la versión 4.4BSD-Lite del Computer Systems Research Group (CSRG) de la Universidad de California en Berkeley, y continúa la distinguida tradición de desarrollo de sistemas BSD. Además del excelente trabajo del CSRG, el Proyecto FreeBSD ha invertido miles de horas en ajustar el sistema para conseguir un rendimiento y una fiabilidad máximas en situaciones de carga reales. Mientras que muchos de los gigantes comerciales se esfuerzan en dotar a los sistemas operativos para PC de esas características, rendimiento y fiabilidad, FreeBSD puede ofrecerlas ¡*ya*!

Los usos que pueda darle a FreeBSD se ven limitados tan sólo por su imaginación. Desde el desarrollo de programas hasta la automatización de fábricas, desde control de inventarios hasta corrección de azimut de antenas de satélites remotos; Si puede hacerse con un UNIX® comercial lo más seguro es que también pueda llevarse a cabo con FreeBSD. FreeBSD también hace buen uso de las literalmente miles de aplicaciones de alta calidad que se desarrollan en centros de investigación y universidades de todo el mundo, frecuentemente disponibles por poco o ningún coste. También existen aplicaciones comerciales, cuyo número aumenta cada dia.

Dado que el código fuente de FreeBSD está disponible para todo el mundo el sistema puede personalizarse en un grado nunca visto para aplicaciones o proyectos especiales, y de maneras generalmente imposibles con los sistemas operativos de la mayoría de los fabricantes comerciales. Aquí damos tan sólo una muestra de aplicaciones en las que se está usando actualmente FreeBSD:

- *Servicios de Internet:* La robusta conectividad TCP/IP integrada en FreeBSD hace de este sistema una plataforma ideal para servicios de Internet como:

  - Servidores FTP
  - Servidores web (estándares o seguros [SSL])
  -
  - Servidores de correo electrónico
  - USENET y BBSs
  - y muchos más.

  Con FreeBSD puede empezar fácilmente con un pequeño y económico PC de tipo 386 e ir actualizando su equipo hasta un tetraprocesador Xeon con almacenamiento RAID a medida que su proyecto crezca.

- *Educación:* ?Es estudiante de informática o de algún campo relacionado con la ingeniería? No hay mejor modo de estudiar sistemas operativos, arquitectura de computado-

res y redes que la experiencia a bajo nivel que FreeBSD puede aportar. Para aquéllos cuyo principal interés en los ordenadores no es otro que el de poder realizar su trabajo existe una serie de paquetes gratuitos de CAD, matemáticas y diseño gráfico que hacen de FreeBSD un sistema verdaderamente útil.

- *Investigación:* Encontrándose disponible el código fuente del sistema entero FreeBSD es una excelente plataforma para la investigación en sistemas operativos y otras ramas de la informática. El hecho de que FreeBSD esté disponible gratuítamente hace posible que grupos remotos puedan colaborar con ideas o compartan desarrollos sin tener que preocuparse de acuerdos de licencias especiales o de limitaciones acerca de lo que puede discutirse en foros públicos.
-
-
- *Desarrollo de software:* El sistema base de FreeBSD incluye una completa colección de herramientas de desarrollo que incluyen el famoso compilador y depurador de C/C++ de GNU.

Puede conseguir FreeBSD tanto en forma de código fuente como binaria por FTP anónimo o en CDROM. Por favor, consulte el Capítulo 30, *Cómo obtener FreeBSD* para más información.

1.2.2. ?Quién usa FreeBSD?

Algunos de los mayores sitios web de Internet utilizan FreeBSD; he aquí algunos de ellos:

- Yahoo!
- Apache
- Blue Mountain Arts
- Pair Networks
- Sony Japón
- Netcraft
- Weathernews
-

Supervalu

- TELEHOUSE America
- Sophos Anti-Virus
- JMA Wired

y muchos más.

1.3. Acerca del Proyecto FreeBSD

En la siguiente sección se explican ciertos aspectos básicos del Proyecto, una breve historia, sus objetivos y el modelo de desarrollo del mismo.

1.3.1. Breve historia de FreeBSD

Elaborado por Jordan Hubbard.

La génesis del proyecto FreeBSD se remonta a comienzos de 1993, en parte como una extensión del "Unofficial 386BSD Patchkit" debida a los tres últimos coordinadores del patchkit: Nate Williams, Rod Grimes y yo mismo.

Nuestro objetivo original era producir una instantánea de 386BSD intermedia para arreglar una serie de problemas que no se podían solventar con uno de nuestros parches. Quizás haya quien recuerde que el primer nombre del proyecto fue "386BSD 0.5" o "386BSD Interim" debido a esto.

386BSD era el sistema operativo de Bill Jolitz, que hasta ese punto había estado sufriendo severamente las consecuencias de prácticamente un año que más valdría olvidar. A medida que el "patchkit" se iba haciendo haciendo más incómodo cada dia que pasaba, así que acordamos ayudar a Bill con una instantánea del sistema. Estos planes se vieron bruscamente interrumpidos cuando Bill Jolitz decidió repentinamente retirar su aprobación al proyecto sin dejar ninguna indicación clara de qué debía hacerse a continuación.

No tardamos mucho en decidir que el objetivo seguía valiendo la pena, aun sin el soporte de Bill, así que adoptamos el nombre de "FreeBSD", una idea de David Greenman. Nuestros objetivos iniciales se fijaron tras consultar a los usuarios del sistema y cuando quedó claro que el proyecto estaba en marcha y que podía llegar a ser una realidad contacté con Walnut Creek CDROM con idea de mejorar los canales de distribución de FreeBSD y hacer m´s fácil llegar a aquellas personas que no tenían la suerte de tener acceso a Internet. Walnut Creek CDROM no solo nos ayudó con la idea de distribuir FreeBSD en CD; también facilitó al Proyecto una máquina en la que trabajar y una conexión rápida a Internet. Sin

la fe casi sin precedentes que tuvo Walnut Creek CDROM en lo que era en aquél momento un proyecto completamente desconocido, es bastante improbable que FreeBSD hubiera logrado tanto y tan rápido como ha logrado hasta el dia de hoy.

La primera distribución en CDROM (y disponible por la red) fue FreeBSD 1.0, publicado en diciembre de 1993. Estaba basado en la cinta de U.C. Berkeley del 4.3BSD-Lite ("Net/2"), con bastantes componentes de 386BSD y de trabajos proveniente de la Free Software Foundation. Fue un logro bastante apreciable para una primera versión; pronto le siguió FreeBSD 1.1 en mayo de 1994, que tuvo un gran éxito.

Por entonces se formaron unos inesperados nubarrones en el horizonte ya que Novell y la Universidad de Berkeley resolvieron el largo juicio acerca del estatus legal de la cinta de Berkeley Net/2. Una condición del acuerdo fue la concesión por parte de Berkeley de que una gran parte de Net/2 era código "gravado" y propiedad de Novell, quien a su vez lo había adquirido de AT&T anteriormente. Berkeley obtuvo a cambio de Novell el "beneplácito" para que 4.4BSD-Lite, cuando saliera, fuera declarado como "no gravado" y se instara a los usuarios de Net/2 a cambiar. Esto repercutió sobre el Proyecto FreeBSD, a quienes se dio hasta julio de 1994 para dejar de sacar su producto basado en Net/2. Bajo los términos de aquel acuerdo se permitía al Proyecto sacar una última versión antes de la fecha límite: esa versión fue FreeBSD 1.1.5.1.

FreeBSD tuvo entonces que acometer la ardua tarea de (literalmente) reinventarse a sí mismo a partir de partes nuevas y bastante incompletas de 4.4BSD-Lite. Las versiones "Lite" eran ligeras en parte porque el CSRG de Berkeley quitó grandes partes del código necesario para construir un sistema que pudiera arrancar (debido a diversos requisitos legales) y porque la versión del 4.4 para Intel era muy incompleta. Hasta noviembre de 1994 el proyecto al fin realizó esa transición; apareció FreeBSD 2.0 en la red y (a finales de diciembre) en CDROM. A pesar de no estar suficientemente pulida esta distribución fue un éxito significativo, al cual siguió el más robusto y fácil de instalar FreeBSD 2.0.5; era junio de 1995.

Sacamos FreeBSD 2.1.5 en Agosto de 1996; pareció ser suficientemente popular entre ISPs y otras comunidades comerciales como para que mereciera otra versión de la rama 2.1-STABLE. Fue FreeBSD 2.1.7.1, publicada en febrero de 1997, que marcó el final de la línea principal de desarrollo en 2-1.STABLE. Una vez en puesta en mantenimiento, en esa rama (RELENG_2_1_0) sólo se harían ya mejoras en seguridad y se corregirían errores críticos.

FreeBSD se ramificó desde la línea principal de desarrollo ("-CURRENT") en noviembre de 1996 como la rama RELENG_2_2, y la primera versión completa (2.2.1) salió en abril de 1997. Se hicieron más versiones de la rama 2.2 en verano y otoño de 1997, la última de las cuales (2.2.8) apareció en noviembre de 1998. La primera versión 3.0 oficial salió en octubre de 1998 y marcó el inicio del fin de la rama 2.2.

El árbol se ramificó de nuevo el 20 de Enero de 1999, dando lugar a las ramas 4.0-CURRENT y 3.X-STABLE. A partir de la 3.X-STABLE salió 3.1 el 15 de febrero de 1999, 3.2 el 15 de mayo de 1999, 3.3 el 16 de septiembre de 1999, 3.4 el 20 de diciembre de 1999 y 3.5 el 24 de

junio de 2000. Pocos dias después de esta apareció una actualización menor, la 3.5.1, que incorporaba parches de seguridad de última hora para Kerberos. Esa fue la última versión de la rama 3.X.

Hubo otra ramificación el 13 de Marzo de 2000 que dio lugar a la rama 4.X-STABLE. Ha habido varias versiones de la misma desde entonces: 4.0-RELEASE salió en marzo de 2000 y la última versión de la rama, 4.11-RELEASE, apareció en enero de 2005.

La largamente esperada 5.0-RELEASE se anunció el 19 de enero de 2003, culminando casi tres años de trabajo; esta versión situó a FreeBSD en el escenario del multiproceso avanzado y el soporte de hilos para las aplicaciones, e introdujo soporte para las plataformas UltraSPARC® y `ia64`. Siguió a esta la versión 5.1 lanzada en Junio de 2003. La última versión de 5.X como la rama -CURRENT fué 5.2.1-RELEASE, que salió en febrero de 2004.

La rama RELENG_5, creada en agosto de 2004, desembocó en 5.3-RELEASE, que marcó el inicio de la rama de versiones 5-STABLE. La versión 5.5-RELEASE más reciente apareció en mayo de 2006. No aparecerán más versiones a partir de la rama RELENG_5.

El árbol se dividió de nuevo en julio de 2005, en esta ocasión para crear RELENG_6. 6.0-RELEASE, la primera versión de la rama 6.X, apareció en noviembre de 2005. 9.3-RELEASE apareció en July 2014. Irán apareciendo más versiones a partir de la rama RELENG_6.

La rama RELENG_7

1.3.2. Objetivos del Proyecto FreeBSD

Texto original de Jordan Hubbard.

Los objetivos del Proyecto FreeBSD son producir software que pueda usarse con cualquier propósito y sin ningún tipo de restricción. Muchos de nosotros participamos de forma significativa en el código (y en el proyecto) y ciertamente no nos importaría recibir una pequeña compensación económica de vez en cuando, pero no vamos a insistir en ello. Creemos que nuestra "misión" más importante y primordial es facilitar el acceso al código a cualquiera, para lo que quiera usarlo y de forma que se use tanto y para sacarle tanto provecho para sea posible. Creo que éste es uno de los objetivos más fundamentales del software libre y algo que nosotros apoyamos con entusiasmo.

El código fuente de nuestro árbol que se halla bajo la GNU General Public License (GPL) o la Library General Public License (LGPL) viene con algunas restricciones más, si bien para garantizar acceso al mismo, y no al contrario como es lo habitual. Debido a las complicaciones adicionales que pueden surgir en el uso comercial de software con licencia GPL preferimos que el software que incluyamos en el árbol venga bajo la licencia BSD, menos restrictiva, siempre y cuando sea una opción razonable.

1.3.3. El modelo de desarrollo de FreeBSD

Texto de Satoshi Asami.

El desarrollo de FreeBSD es un proceso muy abierto y flexible: FreeBSD está literalmente compuesto de partes hechas por centenares de personas de todo el mundo, como puede verse en este texto. la infraestructura de desarrollo del proyecto FreeBSD permite a estos cientos de desarrolladores trabajar a través de Internet. Estamos buscando constantemente de nuevos desarrolladores e ideas. Las personas interesadas en vincularse más al proyecto tienen que ponerse en contacto con nosotros en la lista de correo de discusiones técnicas en FreeBSD. La lista de anuncios importantes del Proyecto FreeBSD está a disposición de quienes deseen dar a conocer a otros usuarios de FreeBSD grandes áreas de trabajo.

Veamos unas cuantas cosas útiles sobre el Proyecto FreeBSD y su proceso de desarrollo, ya sea trabajando de forma independiente o en estrecha cooperación:

Los repositorios SVN y CVS

> El árbol central de código de FreeBSD se ha mantenido mediante CVS (Concurrent Versions System, o sistema concurrente de versiones), una herramienta de control de código totalmente libre que forma parte de FreeBSD. En junio de 2008 el Proyecto adoptó SVN (Subversion). La necesidad de un cambio como este venía de largo a medida que las limitaciones técnicas que el uso de CVS imponía se iban haciendo más y más obvias con la rápida expansión del árbol de código y de la cantidad de historial de cambios que hay que almacenar y gestionar. Aunque el repositorio principal use SVN las aplicaciones del lado del cliente como CVSup y csup, que dependen de la infraestructura anterior basada en CVS, siguen funcionando normalmente; los cambios que se hacen en el repositorio SVN se van replicando en el repositorio CVS con este fin. Actualmente (octubre 2008) solamente el árbol central de código usa SVN. Los repositorios de documentación, WWW y Ports siguen usando still using CVS. El repositorio primario está alojado en una máquina en Santa Clara (California, EEUU), que constituye el original del que todas las réplicas (a las que llamamos a veces “mirrors”) son copias exactas actualizadas cada muy poco tiempo. El árbol SVN, que contiene también los árboles -CURRENT y -STABLE, puede replicarse muy fácilmente en local en su máquina. Consulte la sección Sincronización del árbol de código para más información.

La lista de committers

> Los *committers* son la gente que tienen permisos de *escritura* en el los fuentes de FreeBSD (el término “committer” viene de la orden `commit` de cvs(1), que sirve para hacer cambios en el repositorio CVS). La mejor manera de enviar aportaciones para que sean revisadas por los committers es usar send-pr(1). Si le parece que si pareciera que algo va mal en el sistema también puede enviar correo electrónico a la lista de correo para 'committers' de FreeBSD.

El Core Team de FreeBSD

El *Core Team de FreeBSD* sería el equivalente a una junta directiva si el Proyecto FreeBSD fuese una compañía. La tarea principal del Core Team es la de garantizar que el Proyecto como un todo tenga salud y se mueva en las direcciones adecuadas. Otra de sus funciones es invitar a desarrolladores comprometidos y responsables a que se unan a nuestro equipo, además de reclutar nuevos miembros del Core Team cuando alguno se va. El Core Team actual fue elegido entre un conjunto de candidatos committers en julio de 2008. Se celebran elecciones cada dos años.

Algunos miembros del Core Team tienen también áreas específicas de responsabilidad, pues se encargan de garantizar que grandes secciones del sistema funcionen según lo previsto. Hay una lista completa de desarrolladores de FreeBSD con sus áreas de responsabilidad en la lista de colaboradores. .

Nota

La mayoría de los miembros del Core Team trabaja de forma altruísta en el desarrollo de FreeBSD y no reciben beneficio económico del Proyecto; es por esto que "compromiso" no debe confundirse con "soporte garantizado". La anterior analogía de la "junta directiva" en realidad no es del todo exacta; quizá fuera más acertado decir que esa es la gente que dedica su vida a FreeBSD *en contra de lo que les aconseja su propio sentido común.*

Contribuidores externos

Por último, y no por ello menos importante, el mayor grupo de desarrolladores está formado por los mismos usuarios, quienes constantemente nos aportan comentarios y corrección de errores. La mejor manera de seguir de cerca el desarrollo (al estilo descentralizado de FreeBSD) es suscribirse a la lista de correo de discusiones técnicas en FreeBSD, que es donde se habla de este tipo de cosas. Consulte el Apéndice A, *Recursos en Internet* si necesita más información sobre las diferentes listas de correo de FreeBSD.

La lista de colaboradores es larga y no para de crecer. ?Por qué no apuntarse y hacer algo en FreeBSD hoy mismo?

Aportar código no es la única manera de ayudar al proyecto; hay una lista completa de tareas pendientes en el sitio web del Proyecto FreeBSD.

En resumen, nuestro modelo de desarrollo está organizado como un conjunto de círculos concéntricos. El modelo centralizado está diseñado pensando el la comodidad de los "usuarios" de FreeBSD, que así tienen un modo sencillo de estar al dia con una base de código central y por supuesto *no para excluir a quien quiera ayudar*. Nuestro afán es ofrecer

un sistema operativo estable con un gran conjunto de aplicaciones coherentes que los usuarios puedan instalar y usar fácilmente (y este modelo está dando buenos resultados)

Lo único que pedimos a quienes quieran unirse a nosotros como desarrolladores de FreeBSD es la misma dedicación que los integrantes actuales tienen.

1.3.4. En qué consiste el FreeBSD que distribuimos

FreeBSD es un sistema operativo libre y gratuito que se distribuye con el código fuente íntegro. Está basado en 4.4BSD-Lite y está diseñado para funcionar en sistemas Intel i386™, i486™, Pentium®, Pentium® Pro, Celeron®, Pentium® II, Pentium® III, Pentium® III, Pentium®4 (o compatible), Xeon™, DEC Alpha™ y sistemas basados en SUN UltraSPARC®. Está basado principalmente en software del grupo CSRG de la Universidad de Berkeley (California), y tiene mejoras importadas de NetBSD, OpenBSD, 386BSD y código creado al amparo de la Free Software Foundation.

Desde la versión 2.0 de FreeBSD de finales del 94 el rendimiento, conjunto de funcionalidades, y estabilidad del sistema han mejorado drásticamente. El último cambio consiste en un rediseño del sistema de memoria virtual con una caché unificada de VM/buffer que no solo aumenta el rendimiento sino que reduce el consumo de memoria de FreeBSD, haciendo que una configuración de 5 MB sea un mínimo más aceptable. Otras mejoras incluyen soporte completo para clientes y servidores NIS, soporte para transacciones TCP, llamada bajo demanda PPP, soporte para DHCP integrado, un subsistema SCSI mejorado, soporte para RDSI (ISDN), soporte para ATM, FDDI, adaptadores Fast y Gigabit Ethernet (1000 Mbit), soporte mejorado para los últimos controladores Adaptec y cientos de correcciones de errores.

Además de la distribución base FreeBSD ofrece una colección de software con miles de programas de uso común. *En el momento de escribir esto hay unos 24,000 "Ports"* La lista de Ports comprende desde servidores HTTP (WWW), juegos, lenguajes de programación, editores, y prácticamente cualquier cosa. La colección de Ports completa requiere un espacio de aproximadamente 500 MB, todos ellos expresados como "deltas" de sus fuentes originales. Esto hace que nos sea mucho más fácil actualizar Ports y reduce notablemente el espacio en disco que necesitaba la anterior Colección de Ports 1.0. Para compilar un port uno simplemente se sitúa bajo el directorio del programa que desea instalar, escribe `make install` y deja que el sistema se encargue del resto. La distribución original completa de cada port que compile se descargará dinámicamente de un CDROM o un sitio FTP, de modo que sólo necesita el espacio necesario para compilar los ports que quiera. La mayoría de los ports también están precompilados como "packages", que quienes no quieran compilar Ports pueden instalar con una simple orden: `pkg_add` . En el Capítulo 4, *Instalación de aplicaciones: «packages» y ports* tiene más información sobre los "packages" y Ports.

Algunos documentos que pueden ser de ayuda en el proceso de instalación y al utilizar FreeBSD pueden también encontrarse el directorio `/usr/share/doc` de cualquier máquina con una versión reciente de FreeBSD encontrará varios documentos que pueden serle

de ayuda en el proceso de instalación o al usar FreeBSD. Para poder consultarlos utilice cualquier navegador para seguir estos enlaces:

El *Handbook* de FreeBSD
    /usr/share/doc/handbook/index.html

Las FAQ de FreeBSD
    /usr/share/doc/faq/index.html

También puede ver la copia original (y más frecuentemente actualizada) en http://www.FreeBSD.org/.

Capítulo 2. Instalación de FreeBSD

Reestructurado, reorganizado y con algunas partes reescritas por Jim Mock.
El recorrido por sysinstall, las capturas de pantalla y el texto original son obra de Randy Pratt.

2.1. Sinopsis

FreeBSD dispone de un programa en modo texto muy fácil de usar llamado sysinstall. Es el programa de instalación por omisión en FreeBSD, pero quien decida distribuir FreeBSD tiene todo el derecho de facilitar un sistema de instalación propio si así lo desea. Este capítulo trata sobre cómo usar sysinstall para instalar FreeBSD

Tras leer este capítulo sabrá usted:

- Cómo crear los discos de instalación de FreeBSD
- Cómo interpreta (y subdivide) FreeBSD sus discos duros.
- Cómo arrancar sysinstall.
- Qué preguntas le hará sysinstall, qué significan y cómo responderlas.

Antes de leer este capítulo debería usted:

- Leer la lista de hardware soportado que se suministra con la con la versión de FreeBSD que va a instalar y verificar que su hardware está en dicha lista.

Nota

En general éstas instrucciones de instalación han sido escritas para computadoras de arquitectura i386™ («PC compatible»). En algunos puntos concretos se darán instrucciones específicas para otras plataformas (por ejemplo Alpha). A pesar de que esta guía se intenta mantener todo lo al día que es posible puede que se encuentre con pequeñas diferencias entre el programa de instalación y lo que aquí se le muestra. Le sugerimos que use este capítulo como una guía general más que como un manual literal de instalación.

2.2. Requisitos de hardware

2.2.1. Configuración mínima

La configuración mínima para instalar FreeBSD varía según la versión de FreeBSD y la arquitectura de hardware.

Tiene información sobre la confuración mínima en las Notas de Instalación que encontrará en la sección de Información de Releases del sitio web de FreeBSD. En la siguiente sección se facilita un resumen de dicha información. Dependiendo de cuál sea el método de instalación que elija para instalar FreeBSD necesitará un floppy, un lector de CDROM que pueda utilizar con FreeBSD o quizás un adaptador de red. Todo esto se explica en la Sección 2.3.7, “Preparación del medio de arranque”.

2.2.1.1. FreeBSD/i386 y FreeBSD/pc98

Tanto FreeBSD/i386 como FreeBSD/pc98 necesitan un procesador 486 o superior y un mínimo de 24 MB de RAM. Necesitará también al menos 150 MB de espacio libre en disco, que es lo que necesita la instalación mínima.

Nota

En sistemas muy antiguos la mayoría de las veces será de mucha más ayuda conseguir más RAM y espacio de disco que un procesador más rápido.

2.2.1.2. FreeBSD/alpha

Para instalar FreeBSD/alpha necesitará una plataforma que esté soportada (consulte Sección 2.2.2, “Hardware soportado”) y un disco duro dedicado a FreeBSD. En este momento no es posible compartir un disco con otro sistema operativo. Este disco debe estar necesariamente conectado a una controladora SCSI que esté soportada por el firmware SRM, o si se trata de un disco IDE el SRM de su máquina debe permitir el arranque desde discos IDE.

Necesitará el firmware de la consola SRM de su plataforma. En ciertos casos es posible pasar del firmware AlphaBIOS (o ARC) al SRM. En otros casos no habrá más remedio que descargar un nuevo firmware desde el sito web del fabricante.

Nota

A partir de FreeBSD 7.0 no hay soporte para Alpha. La serie FreeBSD 6.*X* es la última que ofrece soporte para esta arquitectura.

2.2.1.3. FreeBSD/amd64

Hay dos tipos de procesadores capaces de ejecutar FreeBSD/amd64. La primera son los procesadores AMD64, entre los que están los AMD Athlon™64, AMD Athlon™64-FX, AMD Opteron™ y los modelos superiores.

La segunda categoría de procesadores que pueden usar FreeBSD/amd64 es la de los procesadores de arquitectura EM64T de Intel®, por ejemplo las familias de procesadores Intel® Core™ 2 Duo, Quad, y Extreme, y la secuencia de procesadores Intel® Xeon™ 3000, 5000 y 7000.

Si tiene una máquina basada en una nVidia nForce3 Pro-150 *tendrá que usar la configuración de la BIOS* para deshabilitar IO ACPI. Si no tiene la opción de hacerlo tendrá que deshabilitar ACPI. Hay errores en el chipset Pro-150 para los que no hemos encontrado aún una solución.

2.2.1.4. FreeBSD/sparc64

Para instalar FreeBSD/sparc64 necesita una plataforma que esté soportada (consulte la Sección 2.2.2, “Hardware soportado”).

Necesitará un disco dedicado a FreeBSD/sparc64. De momento es imposible compartir un disco duro con otro sistema operativo.

2.2.2. Hardware soportado

Cada versión de FreeBSD incluye una lista de hardware soportado en las «FreeBSD Hardware Notes». Este documento suele estar en un fichero llamado `HARDWARE.TXT` , que está en el directorio raiz del CDROM o distribución FTP, o en el menú de documentación de sysinstall. En este documento se listan los dispositivos de hardware que se sabe que funcionan con cada versión de FreeBSD y para qué arquitectura. En la página de Información de Releases del sitio web de FreeBSD encontrará copias de esta lista para diversas releases y arquitecturas.

2.3. Tareas anteriores a la instalación

2.3.1. Inventario de su sistema

Antes de instalar FreeBSD en su sistema debería hacer un inventario de los componentes de su computadora. El sistema de instalación de FreeBSD le mostrará los componentes (discos duros, tarjetas de red, unidades de CDROM, etc.) con sus datos de modelo y fabricante. FreeBSD tratará también de determinar la configuración correcta para dichos dispositivos, lo que incluye información sobre las IRQ y el uso de puertos IO. A causa de la ingente variedad de hardware para PC este proceso no siempre se puede culminar con éxito y es posible que deba corregir las decisiones de FreeBSD retocando la configuración.

Si ya dispone de otro sistema operativo instalado (como Windows® o Linux) puede usar los recursos que dicho o dichos sistemas operativos le faciliten para determinar exactamente qué hardware tiene y cómo está configurado. Si tiene del todo claro qué configuración está usando una tarjeta de expasión concreta es posible que pueda encontrar esos datos impresos en la propia tarjeta. Es muy habitual el uso de las IRQ 3, 5 y 7 y las direcciones de los puertos IO suelen representarse con números hexadecimales, como 0x330.

Le recomendamos imprimir o tomar nota de todos esos datos antes de instalar FreeBSD. Una tabla como esta puede serle de mucha ayuda:

Tabla 2.1. Ejemplo de inventario de dispositivos

Nombre de dispositivo	IRQ	Puerto(s) IO	Notas
Primer disco duro	N/A	N/A	40 GB, fabricado por Seagate, primer maestro IDE
CDROM	N/A	N/A	Primer esclavo IDE
Segundo disco duro	N/A	N/A	20 GB, fabricado por IBM, segundo maestro IDE
Primera controladora IDE	14	0x1f0	
Tarjeta de red	N/A	N/A	Intel® 10/100
Módem	N/A	N/A	3Com® 56K faxmodem, en COM1
...			

Una vez termine el inventorio de componentes de su sistema debe comprobar si aparecen en la lista de hardware soportado de la versión de FreeBSD que vaya a instalar.

2.3.2. Haga una copia de seguridad de sus datos

Si la máquina en la que va a instalar FreeBSD contiene datos que desea conservar por algún motivo asegúrese de haber hecho una copia de seguridad de los mismos y de que

esa copia es de fiar antes de instalar FreeBSD. El sistema de instalación de FreeBSD le mostrará una advertencia antes de modificar datos en su disco pero una vez que el proceso ha comenzado no hay manera de dar marcha atrás.

2.3.3. Decida dónde instalar FreeBSD

Si quiere que FreeBSD use todo su disco duro puede saltar tranquilamente a la siguiente sección.

Si por el contrario necesita que FreeBSD coexista con otros sistemas operativos tendrá que comprender cómo se almacenan los datos en el disco duro y cómo le afecta esto.

2.3.3.1. Esquemas de disco en FreeBSD/i386™

Un disco de PC puede dividirse en varias partes. Estas partes reciben el nombre de *partitions*. Dado que FreeBSD internamente también tiene particiones la nomenclatura puede ser confusa muy rápidamente, así que estas partes del disco reciben el nombre de «disk slices» o sencillamente «slices» («rebanadas de disco»y «rebanadas» respectivamente). Por ejemplo, la versión de `fdisk` que usará FreeBSD con las particiones de disco de PC usa la palabra «slices» en lugar de «partitions». Debido a limitaciones de diseño la plataforma PC sólo admite cuatro particiones por disco. Dichas particiones reciben el nombre de *particiones primarias*. Esta limitación puede sortearse (y de ese modo disponer de más de cuatro particiones) gracias a que se creó un nuevo tipo de partición, las *particiones extendidas*. Un disco puede contener una única partición extendida. Dentro de ella pueden crearse particiones especiales, que reciben el nombre de *particiones lógicas*.

Cada partición tiene un *identificador de partición* (o *partition ID*), que es un número que se usa para identificar el tipo de datos que alberga la partición. Las particiones FreeBSD tienen como identificador de partición `165`.

Normalmente cada sistema operativo que vaya a utilizar identificará las particiones de un modo propio. Por ejemplo DOS (y sus descendientes, como Windows®) asignan a cada partición primaria y lógica una *letra de unidad* a partir de `C:`.

FreeBSD debe instalarse en una partición primaria. FreeBSD puede albergar todos los datos que necesita, incluyendo cualquier fichero que pueda usted crear, en esta partición. Si tiene usted varios discos duros puede crear particiones para que FreeBSD las use en todos ellos o en algunos nada más. Al instalar FreeBSD debe usar al menos una partición. Puede usar una partición vacía que haya preparado o puede usar también una partición que contenga datos que no desea conservar.

Si está usando todas las particiones de todos sus discos tendrá que dejar libre una de ellas para FreeBSD usando las herramientas del otro sistema operativo que esté usando (por ejemplo `fdisk` en DOS o en Windows®).

Si tiene una partición sobrante puede usarla, pero puede verse en la necesidad de reducir una o más de las particiones que está usando.

Una instalación mínima de FreeBSD cabrí en sólo 100 MB de disco pero tenga en cuenta que apenas quedaría espacio para los ficheros que quiera crear. Un mínimo más realista sería de 250 MB si no pretende usar entorno gráfico y 350 MB o más si quiere usar un interfaz gráfico de usuario. Si quiere instalar gran cantidad de software para usarlo en FreeBSD sin duda necesitará más espacio.

Para ello puede usar herramientas comerciales como PartitionMagic® o libres como GParted para redimensionar sus particiones y hacer sitio para FreeBSD. El directorio `tools` directory del CDROM de instalación contiene dos herramientas libres con las que puede hacer hacer esta redimensión: FIPS y PResizer. En el mismo directorio encontrará documentación de ambas. FIPS, PResizer y PartitionMagic® pueden redimensionar particiones FAT16 y FAT32, que pueden encontrarse desde MS-DOS® hasta Windows® ME. Tanto PartitionMagic® como GParted funcionan también en particiones NTFS. GParted forma parte de diversas distribuciones de «Live CD» de Linux, como SystemRescueCD.

Hay informes de problemas redimensionando particiones de Microsoft® Vista. Le recomendamos tener a mano un disco de instalación de Vista cuando intente hacer esto. Lo dicho para cualquier otra tarea de mantenimiento de discos es válido aquí: tenga una copia de seguridad *fiable* y reciente a mano.

Aviso

El uso incorrecto de estas herramientas puede borrar datos de su disco duro. Recuerde, asegúrese de disponer de copias de seguridad recientes y utilizables antes de usarlas.

Ejemplo 2.1. Uso de una partición sin cambiar nada

Supongamos que tiene una máquina con un sólo disco de 4 GB que ya tiene una versión de Windows® instalada y que ese disco está dividido en dos unidades, `C:` y `D:`, cada una de las cuales tiene un tamaño de 2 GB. Tiene 1 GB de datos en `C:` y 0.5 GB de datos en `D:`.

Esto significa que su disco duro tiene dos particiones, una por cada letra de unidad. Copie todos sus datos de `D:` en `C:`; de este modo vaciará la segunda partició y podrá usarla con FreeBSD.

Ejemplo 2.2. Reducir una partición existente

Suponga que tiene una máquina con un sólo disco de 4 GB que contiene una versión de Windows® instalada. Cuando instaló Windows® creó una gran partición, lo que le dió como resultado una unidad `C:` de 4 GB. Está usando 1.5 GB de espacio y quiere que FreeBSD tenga 2 GB de espacio.

Para poder instalar FreeBSD tendrá que realizar una de las siguientes tareas:

1. Haga una copia de sus datos de Windows® y después reinstale Windows®, eligiendo una partición de 2 GB en el momento de la instalación.
2. Utilice alguna herramienta del estilo de PartitionMagic® que se han descrito antes para reducir el tamaño de su partición de Windows®.

2.3.3.2. Estructura de discos en Alpha

Tendrá que dedicar un disco de su sistema para usar FreeBSD puesto que de momento es imposible compartir un disco con otro sistema operativo. Dependiendo de la la máquina Alpha que tenga el disco podrá ser SCSI o IDE en la medida en que sea posible arrancar desde tales discos.

Siguiendo las normas de los manuales de Digital / Compaq todos los datos suministrados a SRM se muestran en mayúsculas. SRM no distingue entre mayúsculas y minúsculas.

Use `SHOW DEVICE` en la consola de SRM para saber qué tipo de discos hay en su sistema:

```
>>>SHOW DEVICE
dka0.0.0.4.0               DKA0           TOSHIBA CD-ROM XM-57  3476
dkc0.0.0.1009.0            DKC0                      RZ1BB-BS  0658
dkc100.1.0.1009.0          DKC100                SEAGATE ST34501W  0015
dva0.0.0.0.1               DVA0
ewa0.0.0.3.0               EWA0             00-00-F8-75-6D-01
pkc0.7.0.1009.0            PKC0                 SCSI Bus ID 7  5.27
pqa0.0.0.4.0               PQA0                      PCI EIDE
pqb0.0.1.4.0               PQB0                      PCI EIDE
```

Este ejemplo es de una Digital Personal Workstation 433au y muestra tres discos instalados en el sistema. El primer disco es una unidad CDROM llamada `DKA0` y los otros dos reciben los nombres de `DKC0` y `DKC100` respectivamente.

Los discos con nombres tipo `DKx` son discos SCSI. Por ejemplo `DKA100` se refiere a un disco SCSI con el «target ID 1» en el primer bus SCSI (A), mientras que `DKC300` es un disco SCSI con un ID 3 en el tercer bus SCSI (C). Los nombres de dispositivo `PKx` son para adaptadores

de bus SCSI. Como hemos visto en la salida de `SHOW DEVICE` las unidades CDROM SCSI son consideradas iguales a otros discos duros SCSI.

Los discos IDE tienen nombres similares a `DQx`, mientras que `PQx` es la controladora IDE asociada.

2.3.4. Recopile los datos de la configuración de la red

Si pretende conectarse a alguna red durante la instalación de FreeBSD (por ejemplo si pretende hacerlo desde un sitio FTP o mediante un servidor NFS) tendrá que conocer la configuración de su red. Durante la instalación se le pedirán esos datos para que FreeBSD pueda conectarse a la red y realizar la instalación.

2.3.4.1. Conexión a una red Ethernet o a un módem Cable/DSL

Necesitará la siguiente información si va a conectarse a una red Ethernet o si tiene una conexión a Internet a través de una adaptador Ethernet via cable o DSL:

1. Dirección IP
2. Dirección IP de la pasarela («gateway», «puerta de enlace»)
3. Nombre del sistema («hostname»)
4. Dirección IP del servidor DNS
5. Máscara de subred

Si no conoce estos datos póngase en contacto con su administrador de sistemas o con su proveedor de servicios. Es que le digan que tal información se asigna automáticamente mediante *DHCP*. Si es así, anótelo.

2.3.4.2. Conexión mediante módem

Si usted se conecta con su ISP mediante un módem tradicional sigue pudiendo instalar FreeBSD a través de Internet; el problema es que tardará mucho más que por otros medios.

Necesitará saber:

1. El número de teléfono de su ISP a través del que accederá a Internet
2. El COM: el puerto al que está conectado su módem
3. Su nombre de usuario y su contraseña de acceso a Internet

2.3.5. Consulte «FreeBSD Errata»

A pesar de que el proyecto FreeBSD hace todo lo humanamente posible para asegurarse de que cada «release» de FreeBSD es todo lo estable posible a veces algún error logra

entrar en escena. En contadísimas ocasiones esos errores llegan a afectar al proceso de instalación. Cuando esos errores son ubicados y corregidos aparecen en lo que llamamos la FreeBSD Errata, que encontrará en el sitio web de FreeBSD. Debería consultar este texto antes de la instalación para asegurarse de que no hay problemas de última hora de los que deba preocuparse.

Tiene información sobre las «releases», incluyendo la «errata» de cada una de ellas, en la sección de información de «releases» del sitio web de FreeBSD.

2.3.6. Obtención de los ficheros de instalación de FreeBSD

El proceso de instalación de FreeBSD permite instalar FreeBSD desde ficheros ubicados en cualquiera de los siguientes sitios:

- Un CDROM o DVD
- Una partición DOS en la propia computadora
- Una cinta SCSI o QIC
- Discos floppy
- Un sitio FTP, saliendo a través de un cortafuegos o usando un proxy HTTP si fuera necesario
- Un servidor NFS
- Una conexión serie o a través de una cable paralelo

Si ha adquirido FreeBSD en CD o DVD ya tiene todo lo que necesitará, puede pasar a la siguiente sección: (Sección 2.3.7, “Preparación del medio de arranque”).

Si no dispone de los ficheros de instalación de FreeBSD debería consultar la Sección 2.13, “Cómo preparar su propio medio de instalación”, donde se explica cómo preparar la instalación de FreeBSD desde cualquiera de los medios listados anteriormente. Tras leer esa sección puede volver aquí y leer la Sección 2.3.7, “Preparación del medio de arranque”.

2.3.7. Preparación del medio de arranque

El proceso de instalación de FreeBSD comienza por arrancar su sistema mediante el instalador de FreeBSD: no es un programa que pueda ejecutar desde otro sistema operativo. Su sistema suele arrancar usando el sistema operativo que está instalado en su disco duro pero puede también ser configurado para que lo haga desde un floppy «arrancable». Las computadoras más modernas pueden también arrancar desde un CDROM introducido en la unidad CDROM.

Sugerencia

Si tiene FreeBSD en CDROM o DVD (por haberlo comprado o haberlo preparado por usted) y su sistema puede arracar desde CDROM o DVD (suele ser una opción de BIOS llamada «Boot Order» o algo similar) puede saltarse esta sección. Las imágenes de CDROM o DVD de FreeBSD permiten arrancar desde ellas y pueden emplearse para instalar FreeBSD sin ninguna preparación especial.

Siga estos pasos para crear las imágenes que le permitirán arrancar desde floppy:

1. Consiga las imágenes de arranque desde floppy

   Los discos de arranque se encuentran en el directorio `floppies/` del medio de instalación o pueden descargarse desde el directorio correspondiente de `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/<arch>/<version>-RELEASE/floppies/`. Reemplace *<arch>* y *<version>* con la arquitectura y la versión de FreeBSD que quiera instalar. Por ejemplo, las imágenes de arranque desde floppy para FreeBSD 10.0-RELEASE para i386™ están en from `ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/10.0-RELEASE/floppies/` .

   Las imágenes de floppy tienen la extensión `.flp`. El directorio `floppies/` contiene diferentes imágenes y las que usted necesitará dependerán de la versión de FreeBSD que vaya a instalar y, en algunos casos, del hardware en el que lo va a instalar. En la mayoría de de los casos solamente usará dos ficheros: `kern.flp` y `mfsroot.flp` . La instalación en algunos sistemas concretos requerirá controladores de dispositivo adicionales, que se encuentran en la imagen `drivers.flp` . Consulte `README.TXT` en el propio directorio, ahí encontrará la información más reciente sobre las imágenes.

   ## Importante

   Su programa de FTP debe usar *modo binario* para descargar las imágenes. Algunos navegadores web suelen usar el modo *texto* (o *ASCII*). He aquí lo primero a comprobar si no puede arrancar desde los disquetes que ha creado.

2. Preparación de los discos floppy

   Tendrá que preperar un disquete por cada imagen que descargue. Es imprescindible que esos discos carezcan de errores. La forma más sencilla de asegurarlo es formatearlos usted. No confíe en disquetes preformateados. La herramienta de formateo de Windows® no le advertirá del hallazgo de bloques defectuosos, si encuentra alguno sencillamente lo marcará como «defectuoso» y lo ignorará. Le recomendamos que use disquetes nuevos si decide usar este procedimiento de instalación.

   Importante

   Si instenta instalar FreeBSD y el programa de instalación falla, se queda congelado o sucede alguna otra catástrofe uno de las primeras cosas de las que sospechar son los disquetes. Vuelque los ficheros de imagen en discos nuevos e inténtelo de nuevo.

3. Escriba los ficheros de imagen en discos floppy (« disquetes»)

   Los ficheros `.flp` *no* son ficheros normales que puedan copiarse a disco. Son imágenes del contenido completo de los discos. Esto significa que *no puede* simplemente copiar esos ficheros de un disco a otro. Debe usar herramientas especializadas para escribir esas imágenes directamente al disco correspondiente.

   Si va a crear los disquetes de arranque en un sistema en el que se está ejecutando MS-DOS®/Windows® utilice la herramienta `fdimage`.

   Si las imágenes están en el CDROM y su CDROM es la unidad `E:` ejecute lo siguiente:

   ```
   E:\> tools\fdimage floppies\kern.flp A:
   ```

   Repita el proceso con cada fichero `.flp` reemplazando cada vez el disco y recuerde etiquetarlos con el nombre del fichero que ha copiado en cada uno. Modifique la línea del comando donde sea necesario, adaptándola al lugar donde tenga usted los ficheros `.flp`. Puede descargar `fdimage` desde el directorio `tools` del sitio FTP de FreeBSD.

   Si va a crear los disquetes en un sistema UNIX® (por ejemplo otro sistema FreeBSD) puede utilizar dd(1) para volcar las imágenes a los discos. En FreeBSD puede ejecutar algo como:

   ```
   # dd if=kern.flp of=/dev/fd0
   ```

En FreeBSD `/dev/fd0` es la primera unidad de disquetes (la unidad `A:`). `/dev/fd1` sería la unidad `B:` y así sucesivamente. Otras versiones de UNIX® pueden asignar nombres diferentes a las unidades de disquetes; consulte la documentación de su sistema.

Ya podemos instalar instalar FreeBSD.

2.4. Inicio de la instalación

Importante

La instalación no efectúa ningún cambio en su disco o discos duros hasta la aparición del siguiente mensaje:

```
Last Chance: Are you SURE you want continue the
 installation?

If you're running this on a disk with data you wish
 to save then WE
STRONGLY ENCOURAGE YOU TO MAKE PROPER BACKUPS before
 proceeding!

We can take no responsibility for lost disk
 contents!
```

Es decir:

```
Última oportunidad: ?Seguro que quiere seguir
 adelante con la instalación?

¡Si está ejecutando este programa en un disco que
 contenga datos
que quiera conservar LE RECOMENDAMOS ENCARECIDAMENTE
 QUE HAGA
COPIAS DE SEGURIDAD FIABLES antes de proseguir!

¡No podemos responsabilizarnos de datos perdidos!
```

El proceso de instalación puede abandonarse en cualquier momento antes de la advertencia final sin efectuar cambios en el contenido del disco duro. Si advierte que ha configurado algo de forma incorrecta basta con que apague su sistema y no estropeará nada.

2.4.1. El arranque

2.4.1.1. El arranque en i386™

1. Comience con su sistema apagado.

2. Arranque el sistema. Durante el arranque debería mostrarse la opción para entrar en la BIOS, normalmente mediante las teclas F2, F10, Del, o Alt+S. Utilice la tecla o combinación de las mismas que se le indique en pantalla. En algunos casos el sistema puede mostrar un gráfico durante el arranque. Pulsar Esc suele disminuir en esos casos el tamaño del gráfico y le permitirá ver los mensajes del arranque.

3. Encuentre el parámetro que controla desde qué dispositivos arranca el sistema. Normalmente se llama «Boot Order» y suele presentarse como una lista de dispositivos, como `Floppy`, `CDROM`, `First Hard Disk` , etc.

   Si necesita disquetes de arranque asegúrese de que selecciona la unidad correspondiente. Si va a arrancar desde CDROM, seleccione la unidad CDROM. En caso de duda consulte el manual que venía con su computadora y/o el de su placa base.

   Haga los cambios necesarios, guarde los cambios y salga. El sistema debería reiniciarse.

4. Si ha elegido arrancar desde disquete, tal y como se describe en Sección 2.3.7, “Preparación del medio de arranque”, uno de ellos será el primer disco de arranque, probablemente el que contiene `kern.flp` . Introduzca ese disco en su unidad de disquetes.

   Si va a arrancar desde CDROM tendrá que arrancar el sistema e introducir el CDROM en cuanto tenga ocasión.

   Si su sistema arranca normalmente y carga el sistema operativo que ya está instalado puede ocurrir alguna de estas cosas:

   1. Los discos no se introdujeron lo suficientemente pronto en el proceso de arranque. Déjelos insertados y reinicie su sistema.

   2. Los cambios que hizo en la BIOS no han funcionado. Debería repetir los pasos previos hasta que dé con la opción correcta.

   3. Su BIOS en concreto no admite el arranque el arranque desde el medio que ha elegido.

5. FreeBSD comenzará a arrancar. Si está arrancando desde CDROM debería ver algo parecido a esto (se ha omitido la información de número de versión):

```
Verifying DMI Pool Data ........
Boot from ATAPI CD-ROM :
 1. FD 2.88MB  System Type-(00)
Uncompressing ... done
```

```
BTX loader 1.00 BTX version is 1.01
Console: internal video/keyboard
BIOS drive A: is disk0
BIOS drive B: is disk1
BIOS drive C: is disk2
BIOS drive D: is disk3
BIOS 639kB/261120kB available memory

FreeBSD/i386 bootstrap loader, Revision 0.8

/kernel text=0x277391 data=0x3268c+0x332a8 |

|
Hit [Enter] to boot immediately, or any other key for command ↺
prompt.
Booting [kernel] in 9 seconds... _
```

Si arranca desde floppy verá algo parecido a esto (se ha omitido la información de número de versión):

```
Verifying DMI Pool Data ........

BTX loader 1.00  BTX version is 1.01
Console: internal video/keyboard
BIOS drive A: is disk0
BIOS drive C: is disk1
BIOS 639kB/261120kB available memory

FreeBSD/i386 bootstrap loader, Revision 0.8

/kernel text=0x277391 data=0x3268c+0x332a8 |

Please insert MFS root floppy and press enter:
```

Siga las instrucciones y extraiga el disco `kern.flp` disc, inserte el disco `mfsroot.flp` y pulse Intro.

6. Tanto si arranca desde disquete como CDROM el proceso de arranque llegará a este punto:

```
Hit [Enter] to boot immediately, or any other key for command ↺
prompt.
Booting [kernel] in 9 seconds... _
```

Dicho y hecho: espere diez segundos o pulse Enter. Esto lanzará el menú de configuración del kernel.

2.4.1.2. Arranque en Alpha

1. Comience con su sistema apagado.

2. Encienda su computadora y espera un mensaje de arranque en el monitor.

3. Si va a arrancar desde disquetes, tal y como se describe en la Sección 2.3.7, "Preparación del medio de arranque", uno de ellos será el primer disco de arranque, probablemente el que contiene `kern.flp`. Ponga este disco en la unidad de disquetes y escriba el siguiente comando para lanzar el arranque desde el disco (corrija el nombre de su unidad de disquetes si fuera necesario):

   ```
   >>>BOOT DVA0 -FLAGS '' -FILE ''
   ```

   Si va a arrancar desde CDROM introduzca el CDROM en la unidad y escriba el siguiente comando para iniciar la instalación (corrija el nombre de la unidad correcta de CDROM si fuera necesario):

   ```
   >>>BOOT DKA0 -FLAGS '' -FILE ''
   ```

4. FreeBSD comenzará a arrancar. Si está arrancando desde disquete llegado un cierto punto verá usted este mensaje:

   ```
   Please insert MFS root floppy and press enter:
   ```

   Siga las instrucciones del programa de instalación y retire el disco `kern.flp`, inserte el disco `mfsroot.flp` y pulse Intro.

5. Tanto si arrancó desde disquete como desde CDROM el proceso de arranque llegará a este punto:

   ```
   Hit [Enter] to boot immediately, or any other key for command ↺
   prompt.
   Booting [kernel] in 9 seconds... _
   ```

   Dicho y hecho: Espere diez segundos o pulse Enter. Esto iniciará el menú de configuración del kernel.

2.4.1.3. Arranque en Sparc64®

La mayoría de sistemas Sparc64® están configurados para arrancar automáticamente desde disco. Si quiere instalar FreeBSD tendrá que arrancar por red o desde un CDROM, lo que requiere que acceda a la PROM (OpenFirmware).

Reinicie el sistema y espere hasta que aparezca el mensaje de arranque. Depende del modelo, pero debería parecerse a este:

```
Sun Blade 100 (UltraSPARC-IIe), Keyboard Present
Copyright 1998-2001 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.2, 128 MB memory installed, Serial #51090132.
Ethernet address 0:3:ba:b:92:d4, Host ID: 830b92d4.
```

Si en este punto su sistema arranca desde el disco pulse L1+A o Stop+A, o envíe un`BREAK` desde la consola serie serial console (usando, por ejemplo, `~#` en tip(1) o cu(1)) para así recuperar el prompt de PROM. Tiene este aspecto:

```
ok     ❶
ok {0} ❷
```

❶ Este es el prompt que verá en sistemas con una sola CPU.
❷ Este prompt se usa en sistemas SMP; el dígito indica el número de la CPU activa.

Ponga el CDROM dentro de la unidad y teclée `boot cdrom` en el prompt de la PROM.

2.4.2. Revisión de los resultados de la prueba de dispositivos

Es posible revisar los últimos cientos de líneas que se han mostrado en pantalla, pues se almacenan en un búfer con ese propósito.

Pulse Bloq Despl (Scroll Lock) y ya puede revisar el búfer. Para moverse use las flechas o RePág y AvPág (PageUp y PageDown respectivamente). Pulse de nuevo Bloq Despl (Scroll Lock) cuando quiera salir del búfer.

Pruébelo, revise el texto que ha generado el kernel al probar los dispositivos del sistema. Verá un texto muy similar al de la Figura 2.1, “Ejemplo de resultado de prueba de dispositivos”, aunque en su caso concreto se mostrarán los dispositivos que tenga su sistema.

```
avail memory = 253050880 (247120K bytes)
Preloaded elf kernel "kernel" at 0xc0817000.
Preloaded mfs_root "/mfsroot" at 0xc0817084.
md0: Preloaded image </mfsroot> 4423680 bytes at 0xc03ddcd4

md1: Malloc disk
Using $PIR table, 4 entries at 0xc00fde60
npx0: <math processor> on motherboard
npx0: INT 16 interface
pcib0: <Host to PCI bridge> on motherboard
pci0: <PCI bus> on pcib0
pcib1:<VIA 82C598MVP (Apollo MVP3) PCI-PCI (AGP) bridge> at device ↺
1.0 on pci0
pci1: <PCI bus> on pcib1
pci1: <Matrox MGA G200 AGP graphics accelerator> at 0.0 irq 11
isab0: <VIA 82C586 PCI-ISA bridge> at device 7.0 on pci0
isa0: <iSA bus> on isab0
atapci0: <VIA 82C586 ATA33 controller> port 0xe000-0xe00f at device ↺
7.1 on pci0
ata0: at 0x1f0 irq 14 on atapci0
ata1: at 0x170 irq 15 on atapci0
uhci0 <VIA 83C572 USB controller> port 0xe400-0xe41f irq 10 at ↺
device 7.2 on pci
0
usb0: <VIA 83572 USB controller> on uhci0
usb0: USB revision 1.0
uhub0: VIA UHCI root hub, class 9/0, rev 1.00/1.00, addr1
uhub0: 2 ports with 2 removable, self powered
pci0: <unknown card> (vendor=0x1106, dev=0x3040) at 7.3
dc0: <ADMtek AN985 10/100BaseTX> port 0xe800-0xe8ff mem ↺
0xdb000000-0xeb0003ff ir
q 11 at device 8.0 on pci0
dc0: Ethernet address: 00:04:5a:74:6b:b5
miibus0: <MII bus> on dc0
ukphy0: <Generic IEEE 802.3u media interface> on miibus0
ukphy0: 10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
ed0: <NE2000 PCI Ethernet (RealTek 8029)> port 0xec00-0xec1f irq 9 ↺
at device 10.
0 on pci0
ed0 address 52:54:05:de:73:1b, type NE2000 (16 bit)
isa0: too many dependant configs (8)
isa0: unexpected small tag 14
orm0: <Option ROM> at iomem 0xc0000-0xc7fff on isa0
fdc0: <NEC 72065B or clone> at port 0x3f0-0x3f5,0x3f7 irq 6 drq2 on ↺
isa0
fdc0: FIFO enabled, 8 bytes threshold
fd0: <1440-KB 3.5” drive> on fdc0 drive 0
atkbdc0: <Keyboard controller (i8042)> at port 0x60,0x64 on isa0
atkbd0: <AT Keyboard> flags 0x1 irq1 on atkbdc0
kbd0 at atkbd0
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: model Generic PS/@ mouse, device ID 0
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff ↺
on isa0
sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
sio0 at port 0x3f8-0x3ff irq 4 flags 0x10 on isa0
sio0: type 16550A
sio1 at port 0x2f8-0x2ff irq 3 on isa0
sio1: type 16550A
ppc0: <Parallel port> at port 0x378-0x37f irq 7 on isa0
pppc0: SMC-like chipset (ECP/EPP/PS2/NIBBLE) in COMPATIBLE mode
```

Figura 2.1: Ejemplo de resultado de prueba de dispositivos

Compruebe cuidadosamente que FreeBSD haya encontrado todos los dispositivos que debe encontrar. Si no se lista algún dispositivo significa que FreeBSD no lo ha encontrado. Un kernel personalizado le permitirá agregar al sistema el soporte de dispositivos que no aparecen en el kernel`GENERIC`, que suele ser el caso de algunas tarjetas de sonido.

En FreeBSD 6.2 y versiones siguientes verá tras la prueba de dispositivos verá lo que muestra la Figura 2.2, “Menú de selección de país”. Utilice las flechas del teclado para elegir país, región o grupo. Cuando acabe pulse Intro, con lo que habrá elegido país y esquema de teclado. En cualquier momento puede salir de easily. It is also easy to exit the sysinstall y volver a empezar.

Figura 2.2. Menú de selección de país

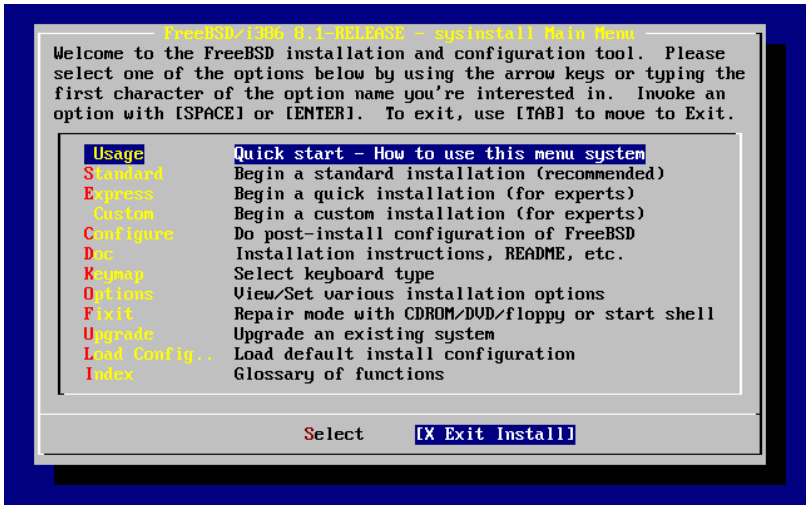


Figura 2.3. Salir de Sysinstall

Para salir de sysinstall utilice las flechas del teclado para seleccionar Exit Install en el menú de la pantalla principal de instalación. Aparecerá el siguiente mensaje:

```
                  User Confirmation Requested
       Are you sure you wish to exit? The system will reboot
         (be sure to remove any floppies/CDs/DVDs from the drives).

                           [ Yes -]    No
```

Si pulsa [Yes] y no retira el CDROM durante el reinicio el programa de instalación comenzará de nuevo.

Si está arrancando desde disquetes tendrá que retirar el floppy `boot.flp` antes de reiniciar.

2.5. ?Qué es sysinstall?

sysinstall es la aplicación que el Proyecto FreeBSD creó para la instalación del sistema. Está orientada a consola y consta de diversos menús y pantallas que podrá usar para configurar y ejecutar el proceso de instalación.

El sistema de menús de sysinstall se controla mediante las flechas del teclado, Intro,el tabulador), Espacio y otras teclas. Tiene una descripción detallada de todas esas teclas y qué es lo que hacen en la información de uso de sysinstall.

Si quiere consultarla seleccione Usage y asegúrese de que el botón [Select] esté seleccionado (como se ven en la Figura 2.4, “Selección de «Usage» en el menú principal de sysinstall”) y pulse Intro.

Aparecerán las instrucciones de uso del sistema de menús. Una vez revisadas pulse Enter y volverá al menú principal.

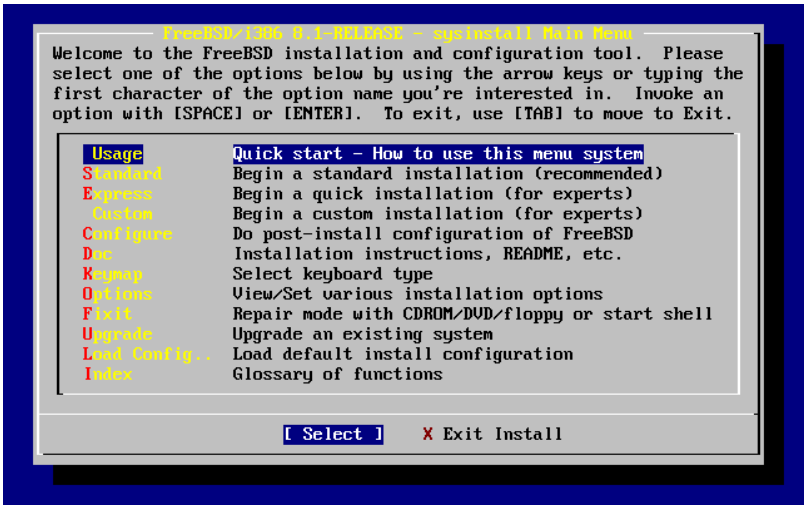


Figura 2.4. Selección de «Usage» en el menú principal de sysinstall

2.5.1. Selección del menú de documentación

Una vez en el menú principal seleccione Doc con las flechas del teclado y pulse Enter.

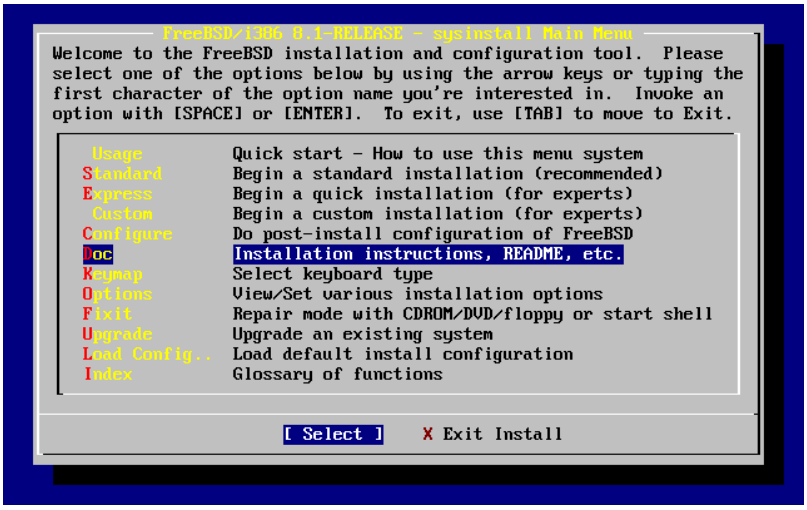


Figura 2.5. Selección del menú de documentación

Esto mostrará el menú de documentación.

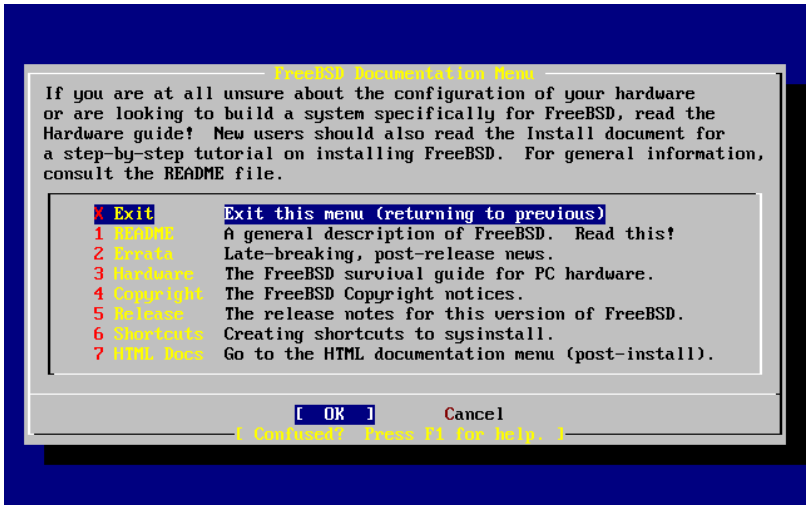


Figura 2.6. El menú de documentación de sysinstall

Es muy importante qeu lea la documentación de sysinstall.

Si quiere consultar un documento seleccionelo con las flechas y pulse Intro. Cuando acabe pulse de nuevo Intro y volverá al menú de documentación.

Si desea regresar al menú principal de la instalación seleccione Exit con las flechas y pulse Intro.

2.5.2. Selección del menú de esquemas de teclado

Si quiere cambiar el esquema de teclado seleccione el que Keymap y pulse Enter. Tendrá que hacer esto si su teclado no es el estándar en los EEUU.

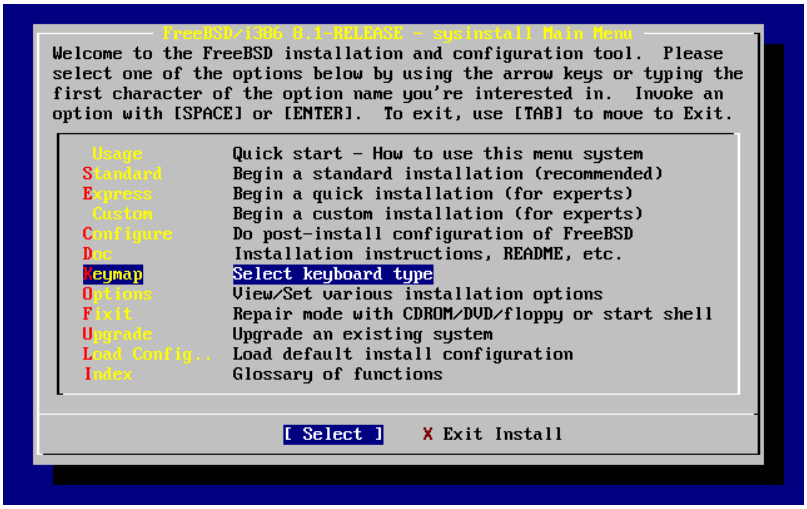


Figura 2.7. Menú principal de sysinstall

La selección de un esquema de teclado se hace del siguiente modo: seleccione uno de la lista mediante las flechas arriba/abajo y pulse Espacio. Si pulsa Espacio otra vez anulará la selección. Una vez que tenga clara su elección use las flechas para seleccionar [OK] y pulse Intro.

En esta pantalla solamente se muestra una parte de la lista. Seleccione [Cancel] usando el tabulador si ha decido seguir usando el esquema de teclado por omisión y regresar al menú principal de instalación.

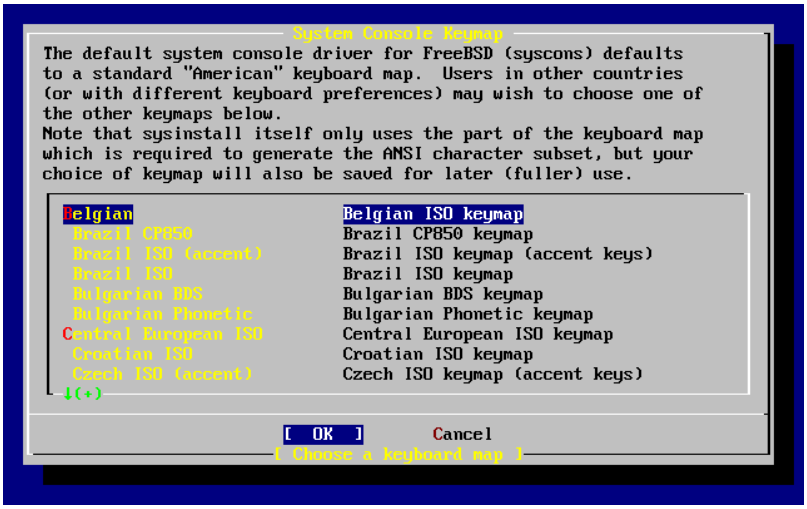


Figura 2.8. Menú de esquemas de teclado de sysinstall

2.5.3. Pantalla de opciones de instalación

Seleccione Options y pulse Intro.

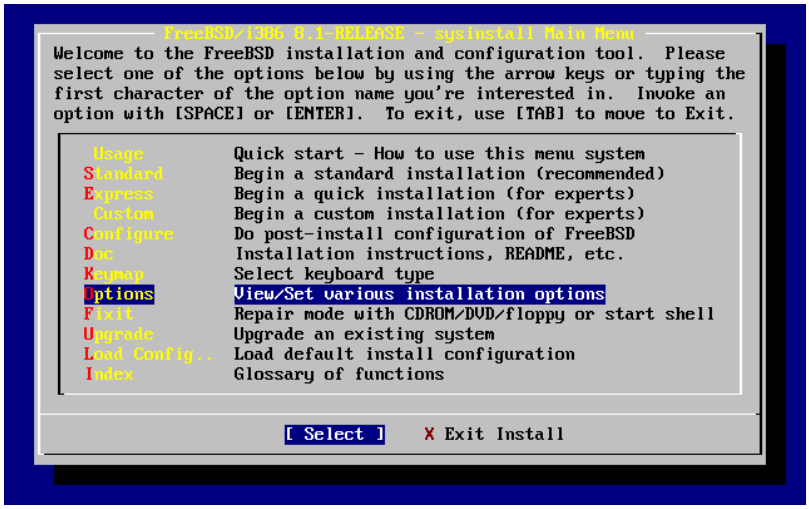


Figura 2.9. Menú principal de sysinstall

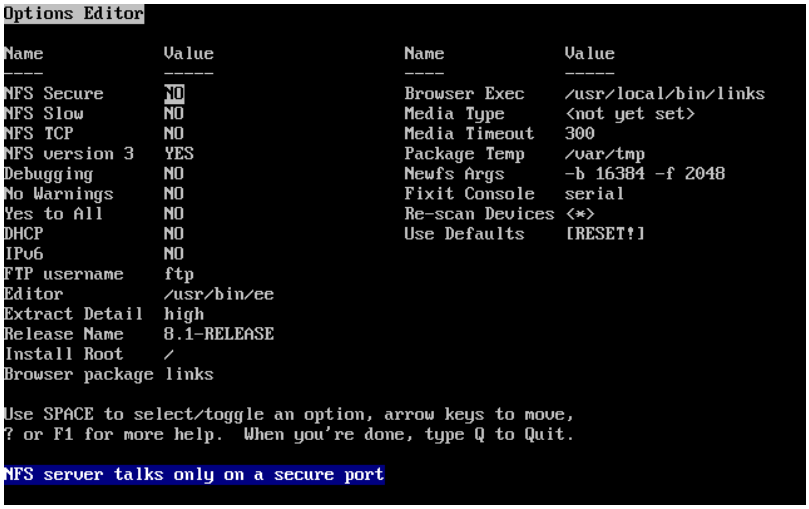


Figura 2.10. Opciones de sysinstall

Los valores por omisión deberían bastar para la mayoría de usuarios. El número de «release» variará según sea la versión que se está instalando.

La descripción del elemento seleccionado aparecerá resaltada en azul en la parte baja de la pantalla. Observe que una de las opciones es Use Defaults, que devuelve todos los valores a los asignados por omisión al principio del proceso de instalación.

Si pulsa F1 podrá leer la pantalla de ayuda de las diversas opciones.

Si pulsa Q volverá al menú principal de la instalación.

2.5.4. Comenzar una instalación estándar

La instalación estándar (Standard) es la opción más adecuada para aquellas personas con poca o ninguna experiencia con UNIX® o FreeBSD. Utilice las flechas para seleccionar Standard y pulse Intro; a partir de ahí comenzará la instalación.

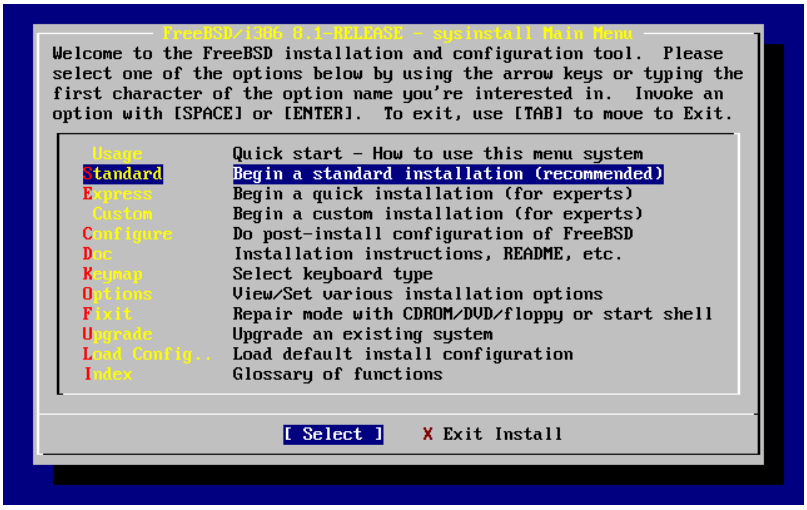


Figura 2.11. Comenzar una instalación estándar

2.6. Asignación de espacio en disco

Lo primero que tiene que hacer es asignar espacio en disco a FreeBSD para que sysinstall lo pueda dejar listo para su uso. Para ello debe saber cómo espera FreeBSD encontrar la información en el disco.

2.6.1. Numeración de unidades desde el punto de vista de la BIOS

Antes de instalar y configurar FreeBSD en su sistema hay una cosa más de la que ocuparse, especialmente si tiene más de un disco duro.

La BIOS es capz de abstraer el orden normal de los discos si hablamos de un PC en el que se está ejecutando un sistema operativo "BIOS-dependiente" como MS-DOS® o Microsoft® Windows®, que admiten esos cambios sin problemas. Tal cosa permite al usuario arrancar desde un disco duro distinto del llamado «maestro principal». Esto viene muy bien a los usuarios que tienen el «backup» del sistema más barato que existe, comprar un disco duro idéntico al primero y copiar periódicamente la primera unidad en la segunda mediante Ghost o XCOPY. De este modo si la primera unidad falla, sufre el ataque de un virus o sufre las consecuencias de un fallo del sistema operativo sólo hay que decirle a la BIOS que interpole lógicamente las unidades. Es como intercambiar las conexiones de los discos sin tener que abrir la caja del sistema.

en sistemas más caros a veces pueden encontrarse controladoras SCSI que incorporan extensiones BIOS, que permiten organizar hasta siete unidades SCSI de un modo muy similar.

Cualquiera que esté acostumbrado a usar las técnicas descritas se llevará una sorpresa al intentar usarlas en FreeBSD. FreeBSD no usa la BIOS y no tiene en cuenta en absoluto la «notación de unidades lógicas desde el punto de vista de la BIOS». Esto puede dar lugar a situaciones bastante chocantes, especialmente cuando las unidades son físicamente idénticas en geometría y han sido creadas como clones de datos la una de la otra.

Si va a usar FreeBSD recuerde siempre devolver a la BIOS a los valores de numeración «natural» antes de la instalación y dejarlos así. Si tiene que intercambiar unidades hágalo, pero a la vieja usanza: abra la caja, toque los «jumpers» y los cables todo lo que sea menester.

Veamos un instructivo ejemplo extraído de Las excepcionales aventuras de Pedro y Pablo:

Pedro decide dedicar una vieja máquina «Wintel» para prepararle un sistema FreeBSD a Pablo. Pedro le instala una sola unidad SCSI como «unidad SCSI 0» e instala FreeBSD en ella.

Pablo comienza a usar el sistema pero a los pocos dias advierte que esa vieja unidad SCSI está mostrando numerosos errores lógicos, así que informa del hecho a Pedro.

Dias después Pedro decide solucionar el problema; consigue un disco idéntico al que instaló a Pablo. Le pasa un chequeo de superficie que da resultados satisfactorios, así que instala este disco como «unidad SCSI 4» y hace una copia binaria de la unidad 0 (el primer disco) a la recién instalada unidad 4. Una vez que el nuevo disco está instalado Pedro decide que hay que empezar a usarlo, así que configura la BIOS SCSI para reorganizar las unidades de disco para que el sistema arranque desde la unidad SCSI 4. FreeBSD arranca desde dicha unidad y todo funciona perfectamente.

Pablo sigue trabajando varios dias y no pasa mucho tiempo antes de que Pedro y Pablo decidan que ya es hora de meterse en una nueva aventura: actualizarse a la nueva versín de FreeBSD. Bill retira la unidad SCSI 0 de la máquina porque le parece poco de fiar y la reemplaza por un disco que obtiene de su en apariencia inagotable trastero. Pedro instala la última versión de FreeBSD en la nueva unidad SCSI mediante los disquetes FTP mágicos que Pablo ha bajado de Internet. La instalación se ejecuta sin problemas.

Pablo utiliza la nueva versión de FreeBSD durante unos dias y comprueba que en efecto es lo bastante buena como para usarla en el departamento de ingeniería. Ha llegado, por tanto, el momento de copiar los datos que tiene en la versión anterior de FreeBSD a su nuevo alojamiento. Pablo monta la unidad SCSI 4 (la última que instaló con la versión anterior de FreeBSD). Pablo queda consternado al comprobar que en la unidad SCSI 4 no hay ni rastro de sus preciados datos.

?Dónde están los datos?

Cuando Pedro hizo la copia binaria de la unidad SCSI 0 en la unidad SCSI 4, la unidad se convirtió en el «el nuevo clon». Cuando Pedro reconfiguró la BIOS SCSI para poder arrancar desde la unidad SCSI 4 tan sólo estaba engañándose a sí mismo puesto que FreeBSD seguía arrancando desde la unidad SCSI 0. Al hacer tal cambio en la BIOS causamos que parte del código de arranque y del cargador del sistema se copie de una a otra unidad pero en cuanto los controladores del kernel de FreeBSD toman el control de la situación el esquema de nomenclatura de unidades de la BIOS es desestimado y FreeBSD recupera el esquema normal de nomenclatura. En nuestro ejemplo el sistema sigue funcionando desde la unidad SCSI 0 y ahí están todos los datos de Pablo, no en la unidad SCSI 4. El hecho de que pareciera que el sistema estaba funcionando en la unidad SCSI 4 era un producto de la imaginación humana.

Estamos encantados de decir que ni un solo byte fue herido o maltratado durante el descubrimiento de este fenómeno. La vieja unidad SCSI 0 fué recuperada del montón de discos caídos en acto de servicio y fue posible devolver todo el trabajo de Pablo a su legítimo dueño. Por si fuera poco, ahora Pedro sabe con certeza que es capaz de contar hasta cero.

En el ejemplo se han usado unidades SCSI pero los conceptos pueden aplicarse perfectamente a unidades IDE.

2.6.2. Creación de «slices» con FDisk

Nota

Aún no ha hecho modificaciones en su disco duro. Si cree que ha cometido algún error y quiere comenzar de nuevo puede hacerlo, salga de los menús de sysinstall e inténtelo de nuevo o pulse U para ejecutar la opción Undo (deshacer). Si se pierde o no sabe cómo salir siempre puede apagar su sistema.

Si ha elegido iniciar una instalación estándar sysinstall le mostrará el siguiente mensaje:

```
                                 Message
 In the next menu, you will need to set up a DOS-style ("fdisk")
 partitioning scheme for your hard disk. If you simply wish to devote
 all disk space to FreeBSD (overwriting anything else that might be ↺
on
 the disk(s) selected) then use the (A)ll command to select the ↺
default
 partitioning scheme followed by a (Q)uit. If you wish to allocate ↺
only
 free space to FreeBSD, move to a partition marked "unused" and use ↺
the
 (C)reate command.
                                [  OK  -]

                     [ Press enter or space -]
```

Es decir:

```
                                 Mensaje
 En el siguiente menú tendrá que configurar un esquema de ↺
particionado
 estilo DOS ("fdisk") en su disco duro.  Si quiere dedicar todo el
 espacio de disco a FreeBSD (cosa que sobreescribirá cualquier cosa
 que contuviera el/los disco/s) use el comando (A)ll (Todo) para
 seleccionar el esquema de particiones por defecto y luego pulse
 (Q)uit (Salir).  Si quiere asignar a FreeBSD sólamente el espacio
 libre en la unidad elija una partición que figure marcada como
 "unused" (sin usar) y ejecute el comando (C)reate (Crear).
                                [  OK  -]

                     [ Pulse Intro o Espacio -]
```

Pulse Intro tal y como se le dice. Se le mostrará una lista con todas las unidades de disco duro que el kernel ha econtrado al hacer el chequeo del sistema. La Figura 2.12, “Elija en qué unidad usar FDisk” muestra un ejemplo de un sistema con dos discos IDE. Reciben los nombres de `ad0` y `ad2`.

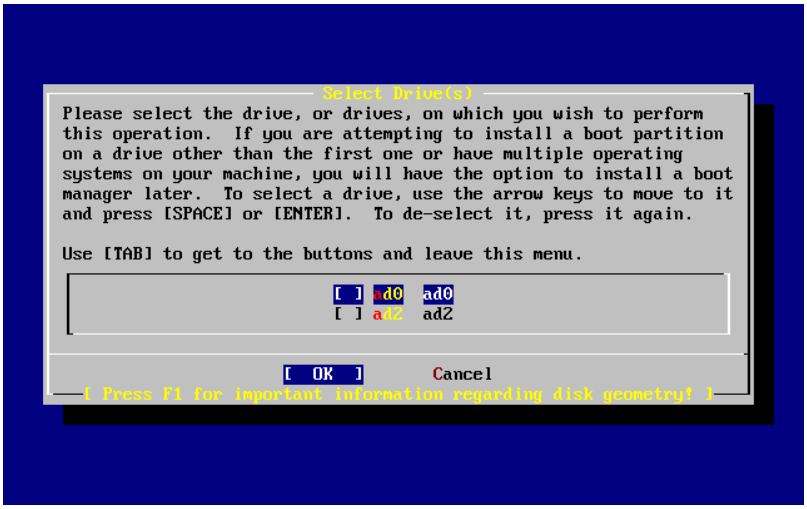


Figura 2.12. Elija en qué unidad usar FDisk

Quizás esté preguntandose por qué `ad1` no aparece por ningún lado. ?Acaso hemos pasado algo por alto?

Veamos qué ocurriría si tuviera usted dos discos duros IDE, uno como maestro de la primera controladora IDE y el otro como maestro en la segunda controladora IDE. Si FreeBSD asignara números de unidad en el orden en el que las encuentra en el ejemplo habríamos visto `ad0` y `ad1` y todo funcionaría sin mayor problema.

Pero si tuviera usted que añadir un tercer disco, digamos como esclavo de la primera controladora IDE, tendría que llamarse `ad1` y el disco que antes era `ad1` pasaría a llamarse `ad2`. Los nombres de dispositivo (como por ejemplo `ad1s1a`) se usan para ubicar sistemas de ficheros, así que podría encontrarse con que alguno de sus sistemas de ficheros no está donde debiera y tendría que modificar la configuración de su FreeBSD.

Para evitar este problema el kernel puede configurarse para asignar nombres a discos IDE basándose en dónde están en lugar de hacerlo por el orden en el que los encuentra. Según dicho esquema el disco maestro de la segunda contoladora IDE *siempre* será `ad2`, incluso si no existen dispositivos `ad0` o `ad1`.

Ésta es la configuración por omisión que incorpora el kernel de FreeBSD, por lo que muestra las unidades `ad0` y `ad2`. La máquina en la que se tomaron las capturas de pantalla tiene sendos discos IDE en ambos canales maestros de las dos controladoras y carece de discos en los canales esclavos.

Seleccione el disco en el que desea instalar FreeBSD y pulse [OK]. FDisk arrancará y le mostrará una pantalla similar a la que muestra la Figura 2.13, “Un ejemplo de particionamiento típico con FDisk”.

La pantalla de FDisk se divide en tres partes.

La primera parte, que ocupa las dos primeras líneas de la pantalla, muestra los detalles del disco que seleccionemos, el nombre que FreeBSD le da, la geometría del disco y el tamaño total del disco.

La segunda parte muestra las «slices» que haya en el disco, dónde comienzan y acaban, cu´l es su tamaño, qué nombre les da FreeBSD, su descripción y subtipo. Este ejemplo muestra dos pequeñas «slices» sin usar; las «slices» (del inglés «rebanadas») son un tipo de esquema estructural de los discos de PC. También muestra una gran «slice» FAT, que casi con total seguridad aparecerá como `C:` en MS-DOS® / Windows®, y una «slice» extendida, que podría contener otras letras de unidad de MS-DOS® / Windows®.

La tercera parte muestra las órdenes que pueden usarse en FDisk.

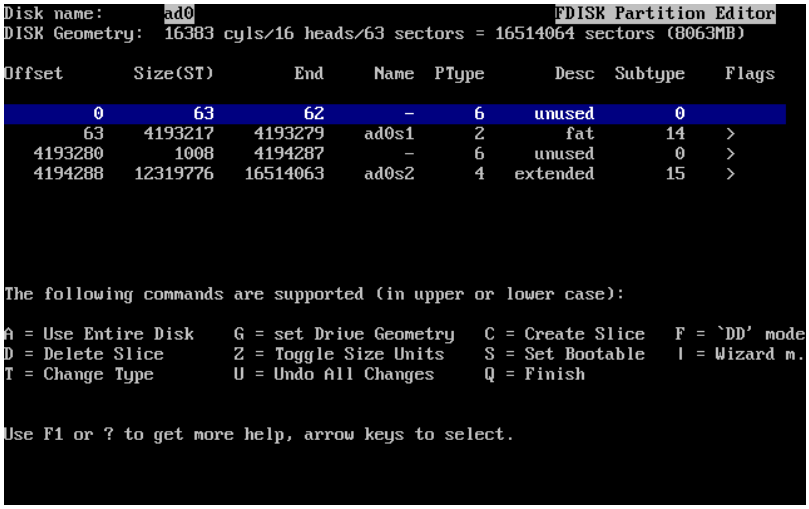


Figura 2.13. Un ejemplo de particionamiento típico con FDisk

Lo que deba hacer a partir de ahora dependerá mucho de la forma en la que quiera distribuir su disco.

Si desea usar FreeBSD en el resto del disco (lo que implica borrar todos los datos que pudiera haber en el disco una vez que le confirme a sysinstall que desea seguir adelante con la instalación) pulse A, que equivale a la opción Use Entire Disk (Utilizar el disco íntegro). Las «slices» que existieran se borrarán y serán reemplazadas por un pequeño área de disco marcado como `sin usar` (otro recurso de la estructura de disco de los PC) y tras él una gran «slice» destinada a FreeBSD. Ahora marque la partición FreeBSD que acaba de crear como arrancable: pulse S. La pantalla que verá ha de ser muy similar a la Figura 2.14, “Partición con FDisk usando el disco completo”. Observe la `A` en la columna `Flags`: indica que la «slice» es *activa* y se arrancará desde ella.

Si desea borrar una «slice» para hacer más sitio a FreeBSD selecciónela mediante las flechas y pulse D. Después pulse C y verá un mensaje en el que se le pedirá el tamaño de la «slice» que va a crear. Introduzca los datos apropiados y pulse Intro. El valor por defecto en ésta pantalla es el de la «slice» más grande que pueda crear en el disco, que debería ser la mayor parte del disco que queda sin usar o el tamaño del disco duro completo.

Si ha hecho sitio para FreeBSD (quizás con una herramienta como PartitionMagic®) puede pulsar press C y crear una nueva «slice». Verá de nuevo un mensaje en el que se le pedirá que escriba el tamaño de la «slice» que va a crear.

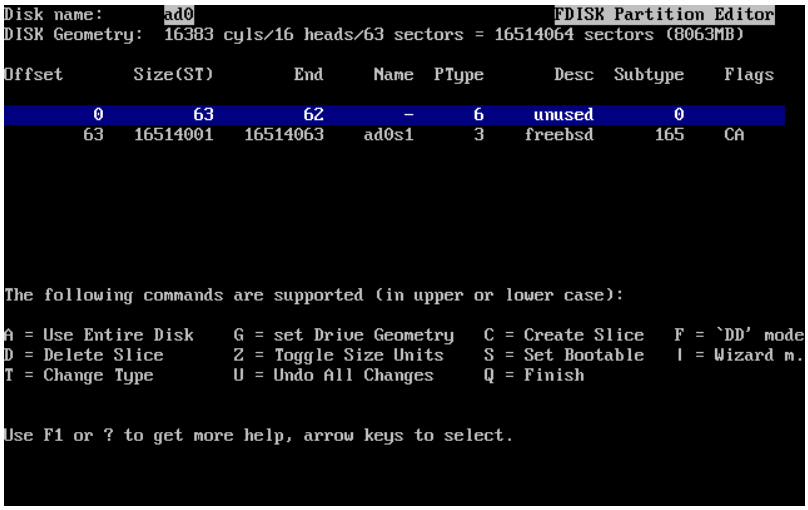


Figura 2.14. Partición con FDisk usando el disco completo

Cuando acabe pulse Q. Sus cambios se guardarán en sysinstall, pero de momento no se guardarán en disco.

2.6.3. Instalación de un gestor de arranque

Ha llegado el momento de instalar un gestor de arranque. Elija el gestor de arranque de FreeBSD si:

- Tiene más de un disco y ha instalado FreeBSD en cualquiera que no sea el primero.
- Ha instalado FreeBSD codo con codo con otro sistema operativo y quiere poder elegir si arrancar FreeBSD o ese otro sistema operativo cuando arranque su sistema.

Si FreeBSD va a ser el único sistema operativo en el sistema y va a instalarlo en el primer disco duro elija el gestor estándar (que, obviamente, está en la opción Standard). Elija None (ninguno) si va a usar algún otro gestor de arranque que sea capaz de arrancar FreeBSD.

Elija y pulse Intro.

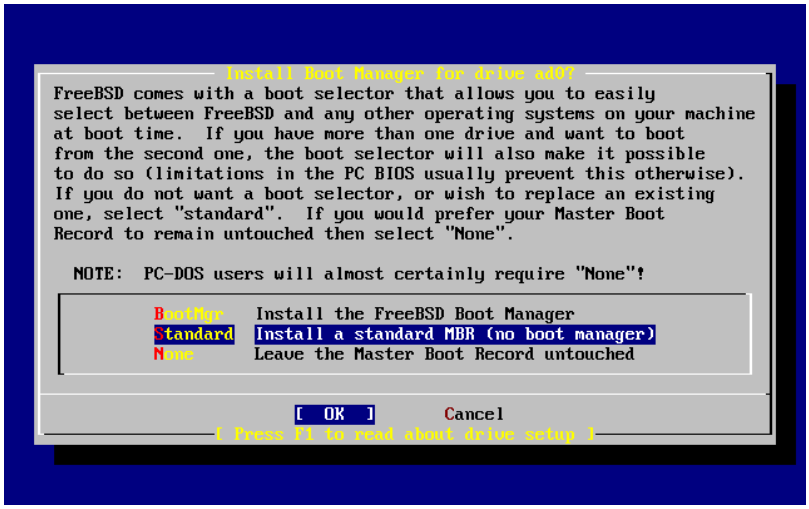


Figura 2.15. Menú de gestores de arranque de sysinstall

La pantalla de ayuda, que puede consultar en cualquier momento pulsando F1, puede serle de mucha ayuda con los problemas que puede encontrarse al intentar compartir un disco duro entre varios sistemas operativos.

2.6.4. Creación de «slices» en otra unidad.

Si hay más de una unidad de disco el programa de instalación volverá a la pantalla «Select Drives» («selección de unidades») una vez elegido el gestor de arranque. Si quiere instalar FreeBSD en más de un disco seleccione aquí ese otro disco y repita el proceso con las «slices» mediante FDisk.

Importante

Si va a instalar FreeBSD en una unidad que no sea la primera tendrá que instalar el gestor de arranque de FreeBSD en ambas unidades.

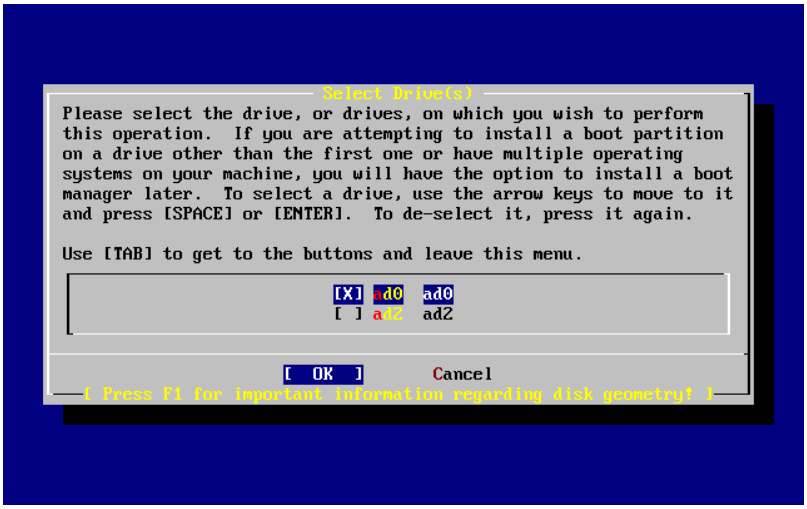


Figura 2.16. Salir de la selección de unidad

El tabulador se usa para hacer pasar el cursor entre [OK], [Cancel].

Pulse tabulador para pasar el cursor a [OK] y pulse Enter para proseguir con la instalación.

2.6.5. Creación de particiones con Disklabel

Tendrá que crear particiones dentro de las «slices» que haya creado. Recuerde que cada partición está asociada a una letra entre la `a` y la `h`y que las particiones `b`, `c` y `d` tienen significados heredados que tendrá que respetar.

Ciertas aplicaciones pueden optimizar su rendimiento de un esquema de particiones concreto, sobre todo si hace particiones repartidas en más de un disco. Si embargo si esta es su primera instalación de FreeBSD no necesita dedicarle demasiado tiempo a decidir cómo partir el disco duro. Es mucho más importante que instale FreeBSD y empiece a aprender a usarlo. Siempre estáa tiempo de reinstalar FreeBSD y cambiar su esquema de particiones cuando esté más familiarizado con el sistema operativo.

El siguiente esquema consta de cuatro particiones: una para la swap y tres para los sistemas de ficheros.

Tabla 2.2. Estructura de particiones del primer disco

Partición	Sistema de ficheros	Tamaño	Descripción
`a`	`/`	512 MB	Este es el sistema de ficheros raíz. el resto de sistemas de ficheros se montarán en algún punto de este sistema raíz. 100 MB es un tamaño razonable para él. No es el mejor sitio para almacenar muchos

Partición	Sistema de ficheros	Tamaño	Descripción
			datos y la instalación de FreeBSD escribirá cerca de 40 MB de datos en ella. El resto del espacio es para datos temporales y por si futuras versiones de FreeBSD necesitan más espacio en `/`.
`b`	N/A	2-3 x RAM	Esta partición es el espacio de memoria de intercambio (o «swap») del sistema. La elección de la cantidad correcta de swap es casi un arte en sí mismo. Hay una regla básica que es asignar a la swap el doble o el triple de MB de los que haya en la memoria física (RAM) del sistema. Debería tener al menos 64 MB de swap así que si tiene menos de 32 MB de RAM asígnele a su swap 64 MB. Si tiene más de un disco puede poner espacio swap en cada disco. FreeBSD usará ambas swap con el resultado de acelerar notablemente el intercambio de páginas de memoria. Calcule el total de swap que necesita (por ejemplo 128 MB) y divida esa cantidad por el número de discos que tenga (por ejemplo dos) y ya tiene la cantidad de espacio que debe destinar a swap en cada uno de sus discos. En nuestro ejemplo 64 MB de swap por disco.
`e`	`/var`	256 MB to 1024 MB	El directorio `/var` contiene ficheros que están en continuo cambio, como «logs» y otros ficheros administrativos. Muchos de esos ficheros son una consecuencia o son de gran ayuda para el correcto funcionamiento diario de FreeBSD. FreeBSD ubica dichos ficheros en ese sistema de ficheros para optimizar el acceso a los mismos sin afectar a otros ficheros ni directorios que tienen similar patrón de accesos.
`f`	`/usr`	Resto del disco(al menos 2 GB)	El resto de sus ficheros pueden almacenarse en `/usr` y sus subdirectorios.

Aviso

Los valores que se han mostrado son simples ejemplos y se recomienda su uso exclusivamente a usuarios experimentados. Reco-

> mendamos a los usuarios que utilicen la opción `Auto Defaults`, que elegirá unas proporciones adecuadas para el espacio que encuentre disponible.

Si va a instalar FreeBSD en más de un disco puede crear particiones el las demás «slices» que haya creado. La forma más fácil de hacerlo es creando dos particiones en cada disco, una para la swap y la otra para los sistemas de ficheros.

Tabla 2.3. Esquema de particiones para varios discos

Partición	Sistema de ficheros	Tamaño	Descripción
`b`	N/A	Ver descripción	Tal y como se ha explicado puede repartir su espacio swap entre varios discos. Aunque piense que la partición `a` está libre la costumbre (y se dice que «las costumbres son leyes» dicta que el espacio swap reside en la partición `b`.
`e`	/disco*n*	Resto del disco	El resto del disco puede dejarse en una sola partición. Esto podría ubicarse simplemente en la partición `a` en lugar de en la `e`. Sin embargo la costumbre dice que la partición `a` de una «slice» está reservada para el sistema de ficheros que ha de albergar el sistema de ficheros raíz (`/`). No tiene por qué seguir la costumbre pero tenga en cuenta que sysinstall sí que lo hace, así que si la sigue sabe que está haciendo una instalación más limpia. Puede montar los sistemas de ficheros donde prefiera; este ejemplo le sugiere que los monte como directorios `/discon`, donde *n* es un número que cambia en cada disco. Por supuesto que puede usar el esquema que prefiera.

Una vez que haya elegido el esquema de particiones creelo en sysinstall. Verá este mensaje:

```
                                Message
Now, you need to create BSD partitions inside of the fdisk
```

```
 partition(s) just created. If you have a reasonable amount of disk
 space (200MB or more) and don't have any special requirements, ↺
simply
 use the (A)uto command to allocate space automatically. If you have
 more specific needs or just don't care for the layout chosen by
 (A)uto, press F1 for more information on manual layout.

                                [  OK  -]
                           [ Press enter or space -]
```

Es decir:

```
                                  Message
 Debe crear particiones BSD dentro de las 'particiones
 fdisk' que acaba de crear.  Si tiene una cantidad de espacio en
 disco razonablemente grande (200MB o más) y no tiene necesidades
 especiales puede simplemente usar el comando (A)uto para asignar
 el espacio automáticamente. Si tiene necesidades más concretas o
 simplemente no le gusta la estructura que le da (A)uto pulse
 F1 y obtendrá más información sobre la creación manual de
 la estructura.

                                [  OK  -]
                           [ Pulse Intro o Espacio -]
```

Al pulsar Intro arrancará el editor de particiones de FreeBSD, Disklabel.

La Figura 2.17, “El editor Disklabel” muestra lo que verá cuando arranque Disklabel. La pantalla se divide en tres secciones.

Las primeras líneas muestran el nombre del disco en el que estamos haciendo cambios y el de la «slice» que contiene las particiones que estamos creando (Disklabel las muestra bajo `Partition name` («nombre de partición») en lugar de hacerlo como «slices»). Vemos también ahí la cantidad de espacio libre en la «slice», esto es, el espacio que hay asignado a la «slice» pero que aún no ha sido asignado a ninguna partición.

En la mitad de la pantalla se muestran las particiones que se han creado, el nombre de los sistemas de ficheros que contiene cada partición, sus tamaños y algunas opciones relacionadas con la creación de sistemas de ficheros.

La tercera parte de la pantalla, la de más abajo, muestra los atajos de teclado que pueden usarse en Disklabel.

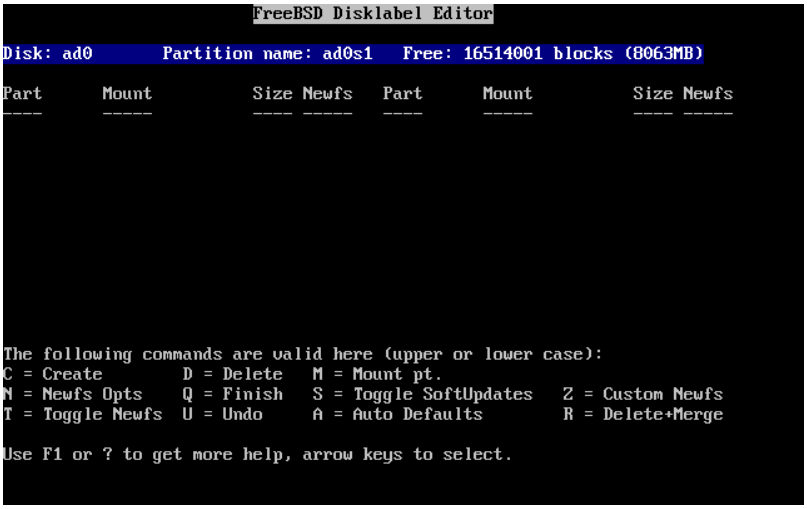


Figura 2.17. El editor Disklabel

Disklabel puede crear automáticamente particiones y asignarles tamaños por omisión. Estos tamaños se calculan con la ayuda de un algoritmo interno de dimensionamiento de particiones que analiza el tamaño del disco. Puede probarlo pulsando A. Verá una pantalla similar a la que aparece en la Figura 2.18, “Editor Disklabel con valores por omisión”. Dependiendo del tama˜o del disco que esté usando los valores por omisión pueden o no ser los apropiados. Esto no es algo de lo que deba preocuparse dado que no está obligado a aceptar esos valores por omisión.

Nota

En el esquema de particiones por omisión el directorio `/tmp` tiene su propia partición en lugar de formar parte de `/`. Esto ayuda a evitar el desbordamiento de `/` con ficheros temporales.

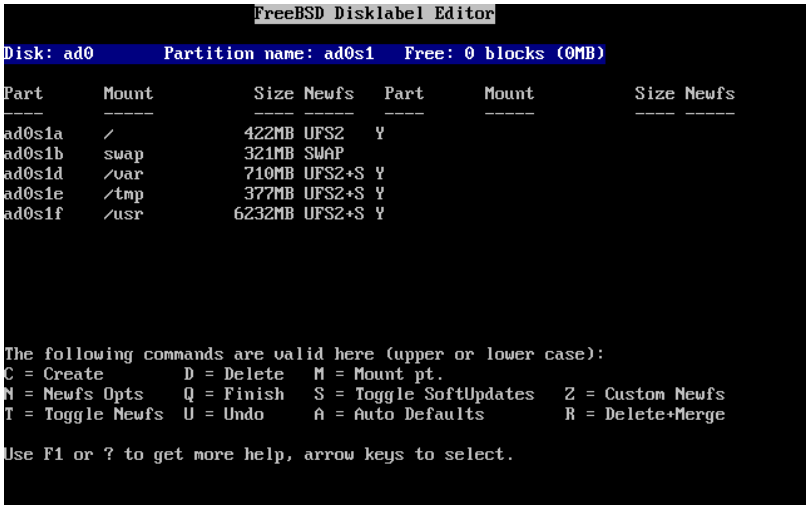


Figura 2.18. Editor Disklabel con valores por omisión

Si decide no usar los valores por defecto para las particiones y quiere reemplazar tales valores por los suyos use las flechas: elija la primera partición y pulse `D` para borrarla. Repita el proceso para borrar todas las particiones que desee.

Vamos a crear la primera partición (`a`, montada como `/` o raíz). Seleccione la «slice» adecuada y pulse `C`. Aparecerá un diálogo pidiéndole que escriba el tamaño de la nueva partición (como se muestra en la in Figura 2.19, “Liberar espacio para la partición raíz”). Puede introducir el tamaño expresado en bloques de disco o como un número seguido de una `M` para expresarlo en megabytes, una `G` para usar gygabytes o una `C` si quiere expresarlo en cilindros.

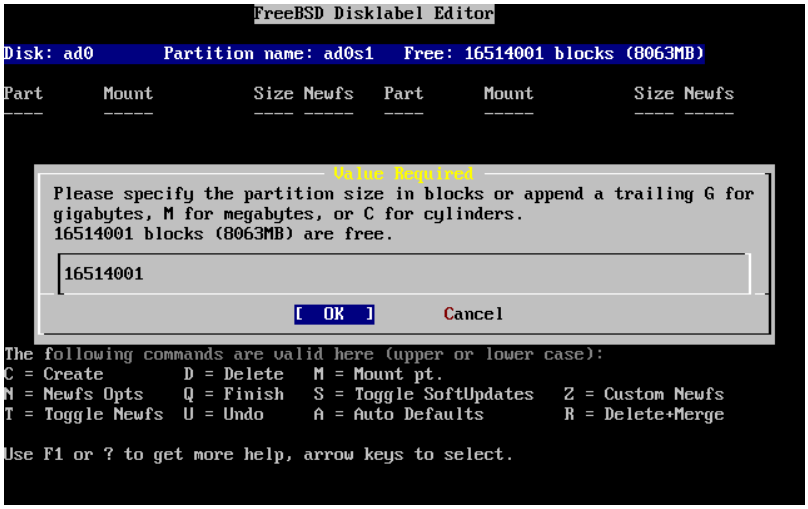


Figura 2.19. Liberar espacio para la partición raíz

El tamaño por omisión que se muestra creará una partición que ocupe el resto de la «slice». Si va a usar los tamaños de partición que se daban en el ejemplo anterior borre el texto existente pulsando Retroceso; escriba **64M**, como se puede ver en la Figura 2.20, "Edición del tamaño de la partición raíz". Después pulse [OK].

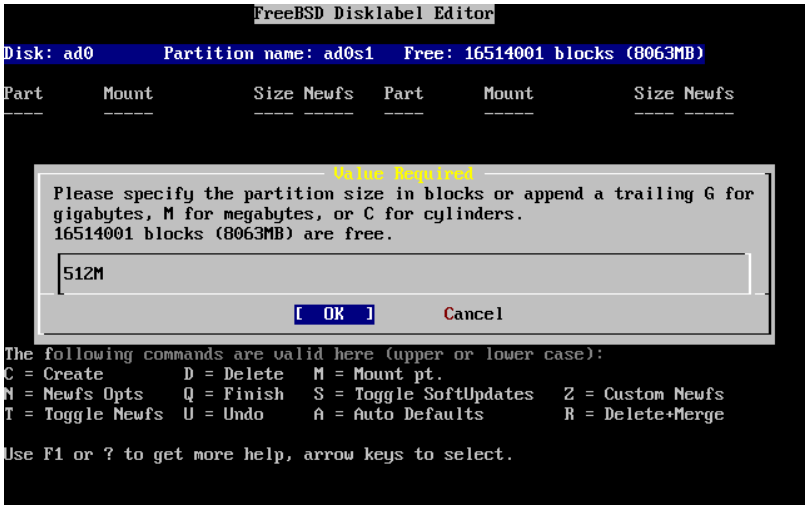


Figura 2.20. Edición del tamaño de la partición raíz

Una vez elegido el tamaño de la partición tendrá que elegir si esta partición ha de contener un sistema de ficheros o espacio swap y escribir lo que corresponda en una pantalla como

la que se muestra en la Figura 2.21, “Elegir el tipo de partición raíz”. Esta primera partición contendrá un sistema de ficheros, así que seleccionamos FS y pulsamos Intro.

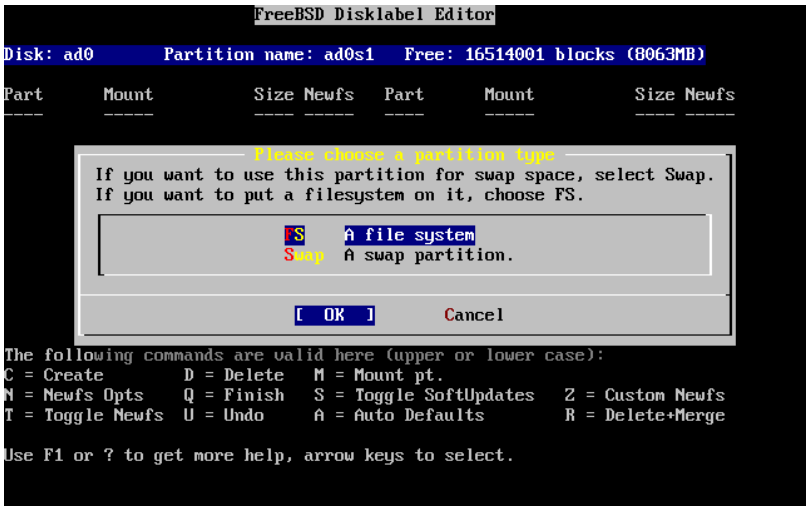


Figura 2.21. Elegir el tipo de partición raíz

Para acabar, dado que está creando un sistema de ficheros, tendrá que decirle a Disklabel dónde hay que montarlo. La pantalla se muestra en la Figura 2.22, “Elegir el punto de montaje del sistema de ficheros raíz”. El punto de montaje del sistema de ficheros raíz es /, así que tendrá que teclear /y luego pulsar Intro.

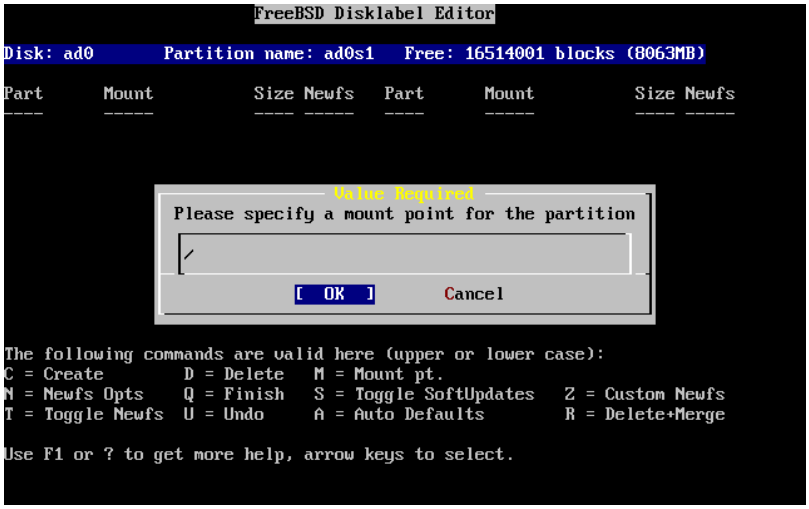


Figura 2.22. Elegir el punto de montaje del sistema de ficheros raíz

La pantalla le mostrará ahora la partición que acaba de crear. Repita el procedimiento todas las veces que necesite con las demás particiones. Cuando cree la partición swap no se le pedirá el punto de montaje puesto que las particiones swap nunca se montan como tales. Cuando cree la última partición `/usr`, puede dejar el tamaño que se le sugiere, esto es, usar el resto del espacio en la «slice».

La última pantalla del editor Disklabel será muy parecida a la Figura 2.23, “Editor Disklabel”, aunque los valores pueden ser diferentes. Pulse Q para salir del editor.

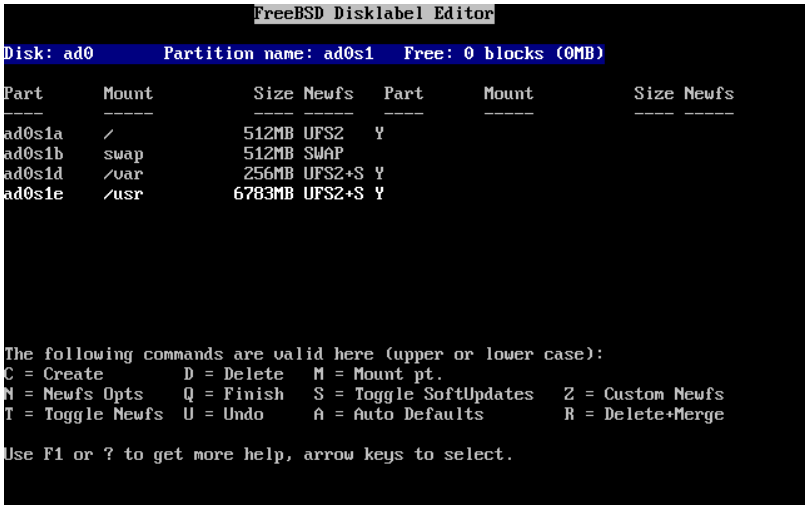


Figura 2.23. Editor Disklabel

2.7. Elección de qué instalar

2.7.1. Elección del tipo de instalación

La elección de qué tipo de instalación debe hacer depende enormemente del uso que se va a dar al sistema y del espacio de disco disponible. El rango de opciones predefinidas está entre hacer la instalación más pequeña posible o instalarlo todo. Las personas con poco o ninguna experiencia en UNIX® o FreeBSD deberán elegir alguna de las opciones predefinidas que se les ofrecen, a las que llamaremos distribuciones (de «distribution set») tal y como aparecen en el menú de sysinstall. Una instalación ·a medida es algo más adecuado para para usuarios con más experiencia.

Si pulsa F1 podrá acceder a más información sobre las opciones de tipo de instalación y qué contiene cada distribución. Cuando acabe de consultar la ayuda pulse Intro y volverá al al menú de selección de instalación de distribuciones.

Si tiene intención de instalar un interfaz gráfico de usuario tendrá que instalar una de las distribuciones cuyo nombre comienza por X. La configuración del servidor X y la selec-

ción de un entorno de escritorio son algunas de las tareas que tendrá una vez instalado FreeBSD. Tiene más información sobre la configuración de un servidor X en Capítulo 5, *El sistema X Window*.

La versión de X11 por omisión en FreeBSD es Xorg.

Si preve compilar un kernel a medida selecciones la opción que incluye el código fuente. Para más información sobre las razones por las que debe compilarse un kernel a medida y sobre cómo compilarlo consulte el Capítulo 8, *Configuración del kernel de FreeBSD*.

Evidentemente el sistema más versátil es aquél que lo tiene todo. Si dispone de espacio de disco suficiente seleccione All, como se muestra en la Figura 2.24, "Elección de distribuciones", usando las flechas y pulsando Intro. Si el espacio en disco es limitado piense en usar alguna de las otras opciones. No pierda con ello demasiado tiempo puesto que el resto de distribuciones pueden añadirse en cualquier momento tras la intalación.

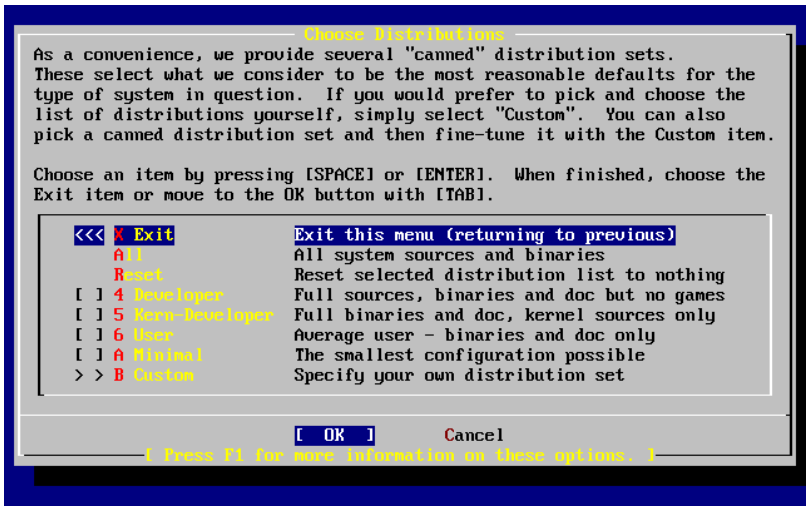


Figura 2.24. Elección de distribuciones

2.7.2. Instalación de la colección de ports

Tras seleccionar la distribución elegida se le presentará la opción de instalar el árbol de ports, o Colección de Ports, de FreeBSD. Los ports son una forma fácil y cómoda de instalar software. El árbol de ports no contiene el código fuente necesario para compilar software; es una gran colección de ficheros que automatiza la descarga, la compilación y la instalación de paquetes de software de todo tipo. El Capítulo 4, *Instalación de aplicaciones: «packages» y ports* explica con detalle cómo utilizar los ports.

El programa de instalación no comprueba si tiene espacio suficiente. Seleccione esta opción si dispone de sitio en el disco. En FreeBSD 10.0 los ports ocupan cerca de 500 MB en

disco. Puede asumir tranquilamente que en las nuevas versiones del sistema irán ocupando más y más.

```
                          User Confirmation Requested
 Would you like to install the FreeBSD ports collection?

 This will give you ready access to over 24,000 ported software ↺
packages,
 at a cost of around 500 MB of disk space when "clean" and possibly ↺
much
 more than that if a lot of the distribution tarballs are loaded
 (unless you have the extra CDs from a FreeBSD CD/DVD distribution
 available and can mount it on /cdrom, in which case this is far less
 of a problem).

 The Ports Collection is a very valuable resource and well worth ↺
having
 on your /usr partition, so it is advisable to say Yes to this ↺
option.

 For more information on the Ports Collection & the latest ports,
 visit:
     http://www.FreeBSD.org/ports

                               [ Yes -]     No
```

Es decir:

```
 Petición de confirmación del usuario
 ?Quiere instalar la colección de ports de FreeBSD?

 Tendrá acceso inmediato a más de 24,000 paquetes de software listos
 para usarse en FreeBSD, aunque necesitará cerca de 500 MB de
 espacio en disco como mínimo y posiblemente mucho más
 si descarga los 'tarballs' de código fuente (aunque si tiene los CD
 extra de una versión de FreeBSD en CD/DVD puede montarlos en /
cdrom, de forma
 que esto dejaría de ser un problema).

 La colección de ports es un recurso extremadamente valioso y es muy
 recomendable que la instale en su partición /usr, así que debería
 responder Sí a la siguiente pregunta.

 Si quiere estar al tanto de las últimas novedades y las últimas
 entradas en la colección de ports visite:
     http://www.FreeBSD.org/ports

                               [ Yes -]     No
```

Seleccione [Yes] con las flechas e instale los ports o [No] para obviar la pregunta. Pulse Intro para seguir con la instalación. Volvemos al menú de elección de distribuciones.

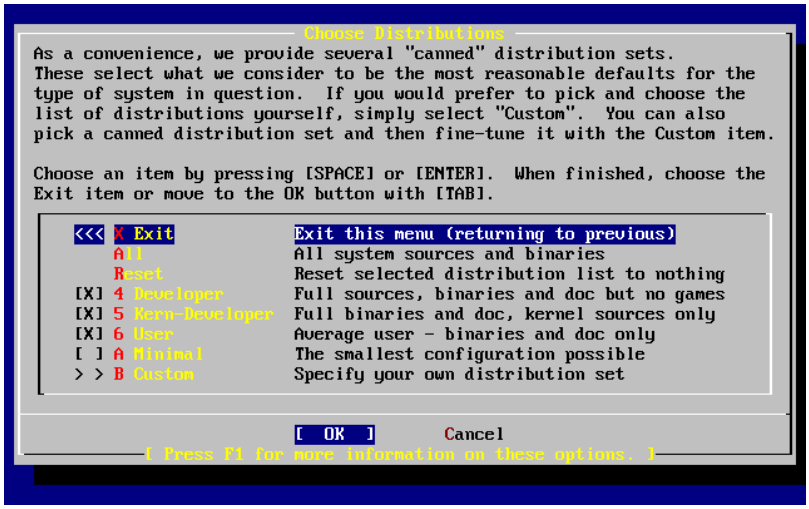


Figura 2.25. Confirmar la elección de distribuciones

Si ya ha elegido todo lo que necesita seleccione Exit con las flechas, asegúrese de que [OK] está seleccionado también y pulse Intro.

2.8. Elección del medio de instalación

Si va a instalar FreeBSD desde CDROM o DVD seleccione Install from a FreeBSD CD/DVD con las flechas. Una vez [OK] está seleccionado pulse Intro y siga adelante con la instalación.

Si quiere usar otro método de instalación seleccione la opción correspondiente y siga las instrucciones.

Pulse F1 si necesita acceder a la ayuda del medio de instalación elegido. Pulse Intro para regresar al menú de selección de medios.

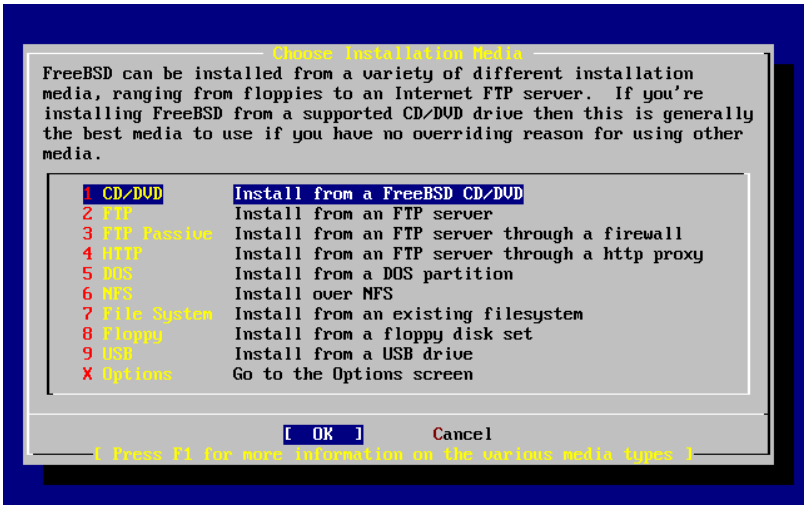


Figura 2.26. Choose Installation Media

Tipos de instalación desde FTP

Hay tres tipos de instalación por FTP entre las que puede elegir: FTP activo, FTP pasivo o a través de un proxy HTTP.

FTP Activo: Install from an FTP server, es decir: «Instalar desde un servidor FTP»

> Esta opción hará que todas las transferencias FTP usen el modo «Activo». No funcionará a través de cortafuegos pero seguramente funcionará con viejos servidores FTP que no soportan el modo pasivo. Si su conexión se cuelga cuando usa el modo pasivo (el modo por omisión) pruebe el modo activo.

FTP Pasivo: Install from an FTP server through a firewall, es decir «Instalar desde un servidor FTP tras un cortafuegos».

> Esta opción configura sysinstall para que use el modo «Pasivo» para todas las operaciones FTP. Esto permite al usuario pasar a través de cortafuegos que no permiten conexiones entrantes a puertos TCP aleatorios.

FTP via un proxy HTTP: Install from an FTP server through a http proxy, es decir «Instalar desde un un servidor FTP via un proxy HTTP».

Esta opción configura sysinstall para que use el protocolo HTTP (como si fuera un navegador web) para conectarse a un proxy en todas las operaciones FTP. El proxy traducirá las peticiones del usuario de forma que pasarán a través de un cortafuegos que no permita ningún tipo de conexiones FTP pero ofrezca un proxy HTTP. Tendrá que configurar los datos del proxy además de los del servidor FTP.

Si quiere usar un servidor proxy FTP tendrá que usar el nombre del servidor con el que realmente quiere conectar como nombre de usuario seguido de un signo «@». Veamos un ejemplo. Quiere usted instalar desde `ftp.FreeBSD.org` a través del servidor proxy FTP `talycual.ejemplo.com`, que escucha en el puerto 1024.

Vaya al menú opciones, ponga `ftp@ftp.FreeBSD.org` como nombre de usuario FTP, y su dirección de correo electrónico como contraseña. El medio de instalación será FTP (o FTP pasivo si el proxy lo permite) y la URL `ftp://talycual.ejemplo.com:1234/pub/FreeBSD`.

Dado accede al directorio `/pub/FreeBSD` de `ftp.FreeBSD.org` a través del proxy `talycual.ejemplo.com` puede usted instalar desde *ésta* máquina, que irá descargando los ficheros que necesite desde `ftp.FreeBSD.org` a medida que el proceso de instalación los vaya requiriendo.

2.9. El punto sin retorno

A partir de aquí entramos en la instalación propiamente dicha. Esta es la última oportunidad antes de empezar a escribir datos en el disco duro.

```
                      User Confirmation Requested
Last Chance! Are you SURE you want to continue the installation?

If you're running this on a disk with data you wish to save then WE
STRONGLY ENCOURAGE YOU TO MAKE PROPER BACKUPS before proceeding!

We can take no responsibility for lost disk contents!

                             [ Yes -]    No
```

Es decir:

```
   Petición de confirmación del usuario
¡Última oportunidad!.  ?SEGURO que quiere seguir con la instalación?
```

```
 Si está ejecutando este proceso en un disco que contenga datos que
 quiera coservar LE RECOMENDAMOS ENCARECIDAMENTE QUE HAGA COPIAS DE
 SEGURIDAD FIABLES antes de instalar.

 No podemos asumir nigún tipo de responsabilidad por datos que ↺
pierda.

                              [ Yes -]    No
```

Seleccione [Yes] y pulse Intro.

La instalación tardará más o menos tiempo según la distribución que haya elegido, el medio de instalación y la velocidad del sistema. Se le irán mostrando mensajes durante el proceso para irle informando de cómo van las cosas.

Cuando acabe la instalación verá un mensaje como este:

```
                              Message

Congratulations! You now have FreeBSD installed on your system.

We will now move on to the final configuration questions.
For any option you do not wish to configure, simply select No.

If you wish to re-enter this utility after the system is up, you may
do so by typing: sysinstall .

                                [ OK -]

                      [  Press enter to continue  -]
```

Es decir:

```
                              Mensaje

         ¡Enhorabuena! Ha instalado FreeBSD en su sistema.

Ahora terminaremos la configuración del sistema.  Si hay alguna
opción que no quiere configurar bastará con que elija No.

Si en algún momento quiere regresar a este programa escriba
sysinstall .

                                [ OK -]

                      [  Pulse Intro  -]
```

Pulse Intro; pasaremos a acometer ciertas tareas posteriores a la instalación.

Si selecciona [No] y pulsa Intro la instalación se detendrá para evitar hacer más modificaciones en su sistema. Verá el el siguiente mensaje.

```
                                Message
Installation complete with some errors.  You may wish to scroll
through the debugging messages on VTY1 with the scroll-lock feature.
You can also choose "No" at the next prompt and go back into the
installation menus to retry whichever operations have failed.

                                 [ OK -]
```

Es decir:

```
                                Mensaje
La instalación ha finalizado con errores.  Puede moverse por la
pantalla de mensajes de depuración de VTY1 tras pulsar la tecla
de Bloqueo de Pantalla.  También puede elegir "No" en el próximo
menú y volver a los menús de instalación y repetir cualquier
operación que haya fallado.

                                 [ OK -]
```

Este mensaje se ha generado porque no se ha instalado nada. Pulse Intro para volver al menún principal y salir de la instalación.

2.10. Después de la instalación

Para poder terminar una instalación de FreeBSD que merezca tal nombre debemos responder todavía a unas cuantas preguntas. Para ello debemos entrar en la configuración antes de entrar en el nuevo sistema FreeBSD o una vez en dentro del sistema desde `sysinstall`, seleccionando Configure.

2.10.1. Network Device Configuration

Si ha tenido que configurar PPP para poder instalar desde FTP esta pantalla no aparecerá. Puede modificar la configuración tal y como se ha explicado más arriba.

Si necesita información sobre redes de área local y la configuración de FreeBSD necesaria para que haga de «gateway/router» consulte el Advanced Networking chapter.

```
                      User Confirmation Requested
   Would you like to configure any Ethernet or SLIP/PPP network ↺
devices?

                              [ Yes -]   No
```

Si quiere configurar dispositivos de red seleccione [Yes] y pulse Intro. Si no quiere seleccione [No].

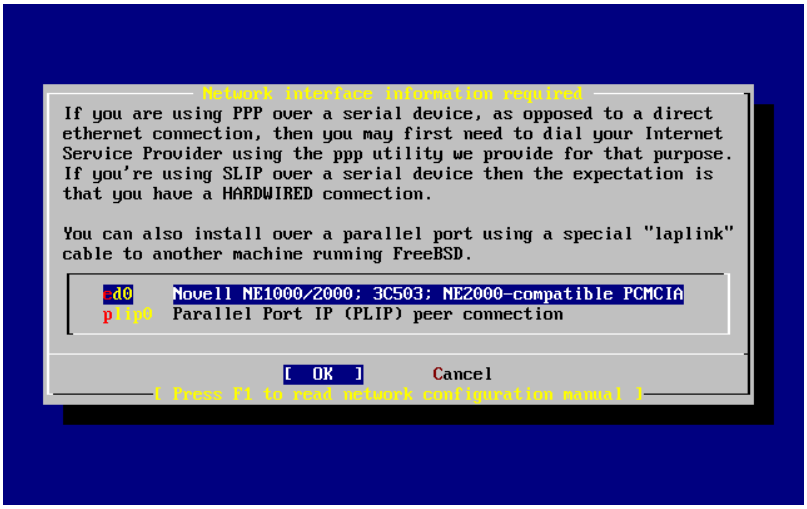


Figura 2.27. Selección de un dispositivo Ethernet

Seleccione con las flechas el interfaz de red que desea configurar y pulse Enter.

```
                    User Confirmation Requested
       Do you want to try IPv6 configuration of the interface?

                                 Yes    [ No -]
```

En la red local del ejemplo el tipo de protocolo de Internet que había (IPv4) parecía más que suficiente, así que se eligió [No].

Si puede conectar con alguna red IPv6 mediante un servidor RA elija [Yes] y pulse Intro. La búsqueda de servidores RA se hará en unos instantes.

```
                           User Confirmation Requested
       Do you want to try DHCP configuration of the interface?

                                 Yes    [ No -]
```

Si no necesita DHCP seleccione [No] con las flechas y pulse Intro.

Si selecciona [Yes] ejecutará dhclient y, si ha funcionado, completará los datos de configuración automáticamente. Para más información consulte la Sección 29.10, “DHCP”.

La siguiente ilustración muestra la configuración de un dispositivo Ethernet en un sistema que cumplirá las funciones de «gateway» en una red local.

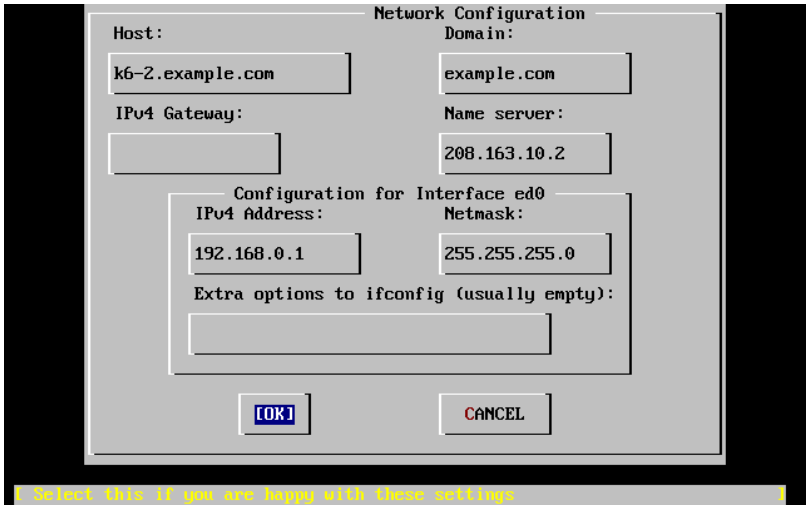


Figura 2.28. Configuración de ed0

Use el tabulador para ir pasando de un campo al siguiente una vez que los vaya rellenando:

Host
: El nombre de la máquina; por ejemplo, `k6-2.ejemplo.com` .

Dominio
: El nombre del dominio al que pertenece la máquina, en este caso `ejemplo.com` .

«Gateway»IPv4
: La dirección IP del sistema que reenvia paquetes a destinos fuera de la red local. Debe rellenar este campo si esta función la realiza una máquina que forme parte de la red. *Déjelo en blanco* si el sistema es el enlace de su red con Internet. El «gateway» recibe también los nombres de puerta de enlace o ruta por omisión.

Servidor de nombres
: Dirección IP de su servidor local de DNS. En la red del ejemplo no hay servidor DNS local así que se ha introducido la dirección IP del servidor DNS del proveedor de Internet: `208.163.10.2` .

Dirección IPv4
: En este interfaz se usará la dirección IP `192.168.0.1`

Máscara de red
: En esta red local se usa un bloque de redes de Clase C `192.168.0.0` - `192.168.0.255` . La máscara de red es, por tanto, `255.255.255.0` .

Opciones adicionales de ifconfig
: Cualquiera de las opciones que quiera agregar a su interfaz mediante `ifconfig`. En nuestro caso no había ninguna.

Utilice el tabulador para seleccionar [OK] cuando haya acabado y pulseIntro.

```
                    User Confirmation Requested
      Would you like to Bring Up the ed0 interface right now?

                              [ Yes -]   No
```

Seleccione [Yes] y pulse Enter si quiere conectar inmediatamente su sistema a la red mediante el o los interfaces que acaba de configurar, pero recuerde que aún tendrá que reiniciar la máquina.

2.10.2. Configuración del «gateway»

```
                     User Confirmation Requested
      Do you want this machine to function as a network gateway?

                               [ Yes -]    No
```

Si el sistema hará de enlace de la red local y reenviará paquetes entre otras máquinas elija [Yes] y pulse Intro. Si la máquina es un nodo de una red elija [No] y pulse Intro.

2.10.3. Configuración de servicios de internet

```
                     User Confirmation Requested
Do you want to configure inetd and the network services that it ↺
provides?

                                Yes   [ No -]
```

Si selecciona [No] varios servicios de la máquina, como telnetd y otros, no se activarán. Eso significa que los usuarios remotos no podrán acceder al sistema mediante telnet. Los usuarios locales, en cambio, podrán acceder a sistemas remotos mediante telnet.

Dichos servicios pueden activarse en cualquier momento editando `/etc/inetd.conf` con el editor de texto que prefiera. Para más información consulte la Sección 29.14.1, “Resumen”.

Seleccione [Yes] si desea configurar estos servicios durante la instalación. Se le mostrará el siguiente mensaje:

```
                     User Confirmation Requested
The Internet Super Server (inetd) allows a number of simple Internet
services to be enabled, including finger, ftp and telnetd.  Enabling
these services may increase risk of security problems by increasing
the exposure of your system.

With this in mind, do you wish to enable inetd?

                               [ Yes -]   No
```

Select [Yes] to continue.

```
                        User Confirmation Requested
inetd(8) relies on its configuration file, /etc/inetd.conf, to ↺
determine
which of its Internet services will be available.  The default ↺
FreeBSD
inetd.conf(5) leaves all services disabled by default, so they must ↺
be
specifically enabled in the configuration file before they will
function, even once inetd(8) is enabled.  Note that services for
IPv6 must be separately enabled from IPv4 services.

Select [Yes] now to invoke an editor on /etc/inetd.conf, or [No] to
use the current settings.

                                [ Yes -]   No
```

Es decir:

```
      Petición de confirmación del usuario
El Super Servidor de Internet (inetd) le permite activar diversos
servicios sencillos de Internet, como finger, ftp y telnetd.  Si
activa tales servicios puede sobreexponer su sistema al exterior, lo
que puede incrementar el riesgo de tener problemas de seguridad.

Dicho esto ?desea activar inetd?

                                [ Yes -]   No
```

Si selecciona [Yes] podrá añadir servicios borrando caracteres # al comienzo de las líneas correspondientes.

```
^[ (escape) menu  ^y search prompt  ^k delete line   ^p prev li   ^g prev page
^o ascii code     ^x search         ^l undelete line ^n next li   ^v next page
^u end of file    ^a begin of line  ^w delete word   ^b back 1 char
^t top of text    ^e end of line    ^r restore word  ^f forward 1 char
^c command        ^d delete char    ^j undelete char ^z next word
=====line 1 col 0 lines from top 1 ========================================
# $FreeBSD: src/etc/inetd.conf,v 1.73.10.2.4.1 2010/06/14 02:09:06 kensmith Exp
#
# Internet server configuration database
#
# Define *both* IPv4 and IPv6 entries for dual-stack support.
# To disable a service, comment it out by prefixing the line with '#'.
# To enable a service, remove the '#' at the beginning of the line.
#
#ftp    stream  tcp     nowait  root    /usr/libexec/ftpd       ftpd -l
#ftp    stream  tcp6    nowait  root    /usr/libexec/ftpd       ftpd -l
#ssh    stream  tcp     nowait  root    /usr/sbin/sshd          sshd -i -4
#ssh    stream  tcp6    nowait  root    /usr/sbin/sshd          sshd -i -6
#telnet stream  tcp     nowait  root    /usr/libexec/telnetd    telnetd
#telnet stream  tcp6    nowait  root    /usr/libexec/telnetd    telnetd
#shell  stream  tcp     nowait  root    /usr/libexec/rshd       rshd
#shell  stream  tcp6    nowait  root    /usr/libexec/rshd       rshd
#login  stream  tcp     nowait  root    /usr/libexec/rlogind    rlogind
#login  stream  tcp6    nowait  root    /usr/libexec/rlogind    rlogind
file "/etc/inetd.conf", 118 lines
```

Figura 2.29. Edición de `inetd.conf`

Una vez que haya añadido los servicios que quiera en el sistema pulse Esc y verá un menú en el que se le preguntará si quiere guardar los cambios.

2.10.4. Activar el acceso al sistema mediante SSH

```
                    User Confirmation Requested
                 Would you like to enable SSH login?
                          Yes        [  No  -]
```

Si selecciona [Yes] activará sshd(8), el dæmon de OpenSSH. De este modo será posible el acceso remoto y seguro al sistema. Para más información sobre OpenSSH visite Sección 14.11, “OpenSSH”.

2.10.5. FTP anónimo

```
                      User Confirmation Requested
 Do you want to have anonymous FTP access to this machine?

                                Yes    [ No -]
```

2.10.5.1. Rechazar el acceso al sistema mediante FTP anónimo

si acepta la respuesta por omisión ([No]) y pulsa Intro tenga en cuenta que los usuarios que tengan cuenta en la máquina podrán seguir accediendo al sistema.

2.10.5.2. Aceptar el acceso al sistema mediante FTP anónimo

Si decide aceptar conexiones FTP anónimas cualquiera podrá conectarse al sistema. Antes de activar esta opción debería meditar sobre las implicaciones de seguridad de esa decisión. Para más información sobre la seguridad y FreeBSD consulte el Capítulo 14, *Seguridad*.

Si ha decidido activar el FTP anónimo seleccione con las flechas [Yes] y pulse Intro. Verá la siguiente pantalla (o una muy similar):

```
This screen allows you to configure the anonymous FTP user.

The following configuration values are editable:

UID:     The user ID you wish to assign to the anonymous FTP user.
         All files uploaded will be owned by this ID.

Group:   Which group you wish the anonymous FTP user to be in.

Comment: String describing this user in /etc/passwd


FTP Root Directory:
```

```
        Where files available for anonymous FTP will be kept.

Upload subdirectory:

        Where files uploaded by anonymous FTP users will go.
```

Es decir:

```
En esta pantalla puede configurar el usuario de FTP
        anónimo.

Puede editar los siguientes valores de la configuración:

UID:     El ID del usuario que quiere asignar al usuario anónimo de
         de FTP.  Todos los ficheros que se suban le  pertenecerán.

Group:   El grupo al que pertenecerá el usuario FTP anónimo.

Comment: La descripción del usuario en /etc/passwd


FTP Root Directory:

        Dónde se guardarán contenidos para los usuarios anónimos.

Upload subdirectory:

        Dónde se guardarán los ficheros que suban los usuarios
 de FTP anónimo.
```

Por omisión el directorio raíz del ftp será `/var`. Si no hay sitio suficiente para lo que prevea que va a necesitar puede usar `/usr`; puede poner el Directorio Raíz de FTP en `/usr/ftp`.

Cuando haya terminado con la configuración pulse Intro.

```
                          User Confirmation Requested
        Create a welcome message file for anonymous FTP users?

                               [ Yes -]    No
```

Si selecciona [Yes] y pulsa Intro arrancará automáticamente un editor y podrá crear un mensaje que verá los usuarios de FTP anónimo al conectarse a la máquina.

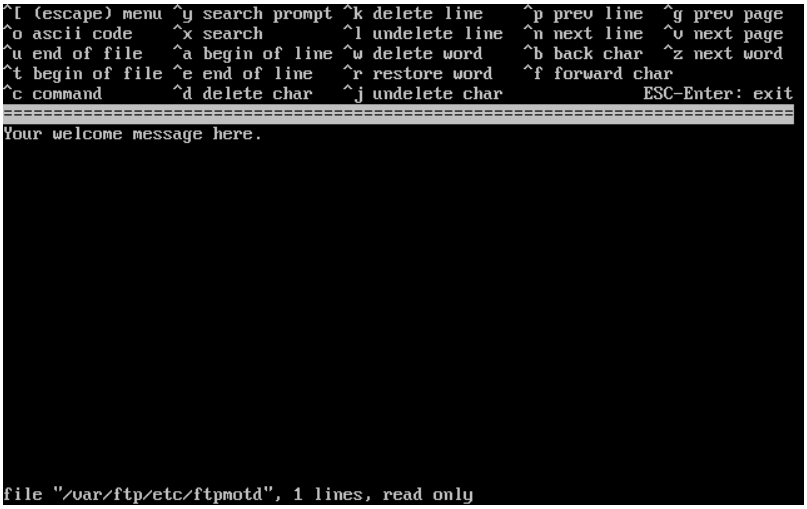


Figura 2.30. Edición del mensaje de bienvenida de FTP

Vemos en acción un editor de texto llamado ee. Puede modificar el mensaje ahora mismo o hacerlo en cualquier otro momento con el editor de texto que prefiera. Observe el fichero y su ubicación en la parte baja de la pantalla.

Pulse Esc, aparecerá una ventana flotante con la opción por omisión de to a) leave editor (salir del editor). Pulse Intro si quiere salir y seguir con lo que estaba haciendo. Pulse Intro de nuevo para guardar los cambios que hubiera hecho.

2.10.6. Configuración de NFS

NFS («Network File System» (de Network File System, o Sistema de Ficheros en Red) le permitirá compartir ficheros a través de una red. Una máquina puede configurarse como servidor, como cliente o ambos. Para má información consulte la Sección 29.6, “NFS”.

2.10.6.1. Servidor NFS

```
                    User Confirmation Requested
 Do you want to configure this machine as an NFS server?

                              Yes    [ No -]
```

Si no tiene necesidad de usar un servidor NFS seleccione [No] y pulse Intro.

Si ha elegido [Yes] verá un mensaje emergente indicando que hay que crear el fichero `exports`.

```
                              Message
 Operating as an NFS server means that you must first configure an
```

```
/etc/exports file to indicate which hosts are allowed certain kinds ↺
of
access to your local filesystems.
Press [Enter] now to invoke an editor on /etc/exports
                                [ OK -]
```

Es decir:

```
                                Mensaje
Hacer funcionar un servidor NFS implica que tendrá que configurar
un fichero /etc/exports para indicar qué hosts estarán autorizados a
acceder de qué manera a sus sistemas de ficheros locales.
Pulse [Intro] para abrir /etc/exports en un editor
                                [ OK -]
```

Pulse Intro para seguir adelante. Se abrirá un editor de texto gracias al cual se podrá editar y crear el fichero `exports`.

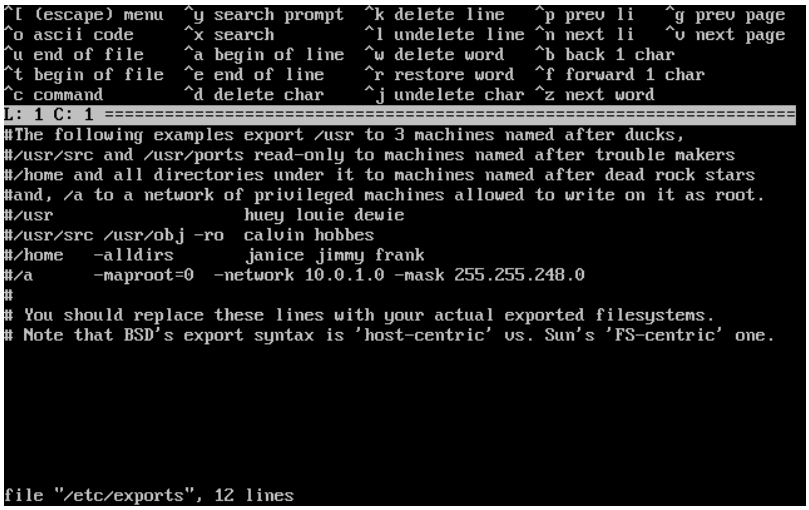


Figura 2.31. Edición de `exports`

Puede editar el fichero ahora mismo o más tarde con el editor de texto que prefiera. Observe que el nombre del fichero y su ubicación aparecen en la parte de abajo de de la pantalla.

Pulse Escape; aparecerá un mensaje emergente que por omisión le ofrecerá a) leave editor (es decir, salir del editor). Pulse Intro para salir del editor y seguir adelante.

2.10.6.2. Cliente NFS

El cliente NFS permite que su máquina pueda acceder a servidores NFS.

```
                      User Confirmation Requested
 Do you want to configure this machine as an NFS client?

                                Yes   [ No -]
```

Utilice las flechas para elegir si quiere o no disponer de cliente NFS (respectivamente con [Yes] y [No] y pulse Intro.

2.10.7. Configuración de la consola del sistema

La consola del sistema tiene diversas opciones que puede usted adaptar a sus gustos o necesidades.

```
                    User Confirmation Requested
       Would you like to customize your system console settings?

                                [ Yes -]  No
```

Si quiere ver y modificar las opciones seleccione [Yes] y pulse Intro.

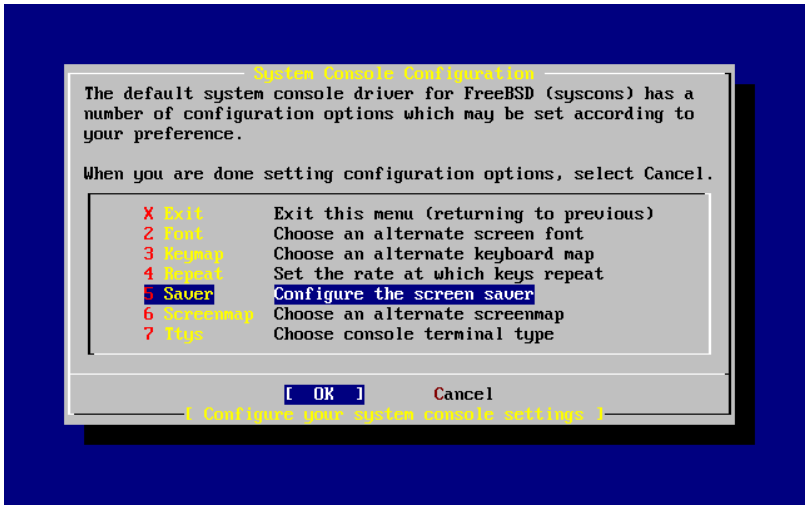


Figura 2.32. Opciones de configuración de la consola del sistema

Una opción que suele elegirse es el uso del salvapantallas. Elija Saver y luego pulse Intro.

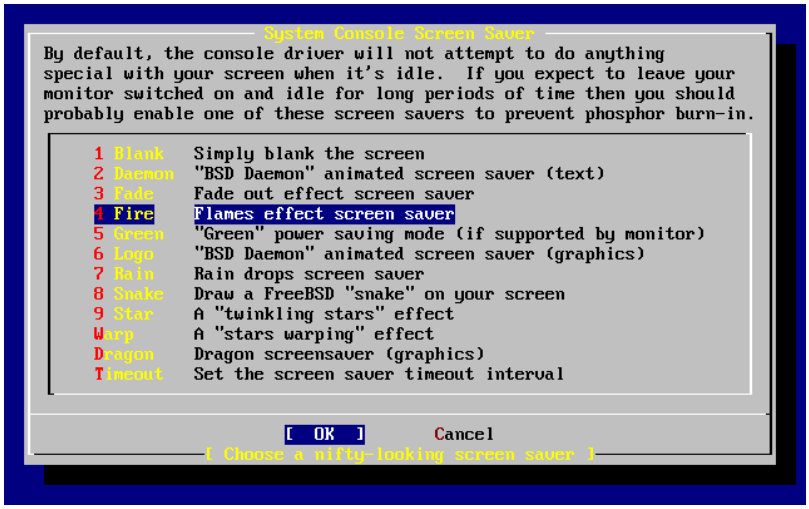


Figura 2.33. Opciones del salvapantallas

Desplácese arriba y abajo con las flechas por la lista de salvapantallas y elija el que prefiera pulsando Intro sobre él. Se le mostrará de nuevo el menú de configuración de la consola.

El intervalo por defecto es de 300 segundos. Si quiere cambiarlo seleccione Saver otra vez. Busque Timeout en el menú de opciones del salvapantallas y pulse Intro. Aparecerá un mensaje en pantalla:

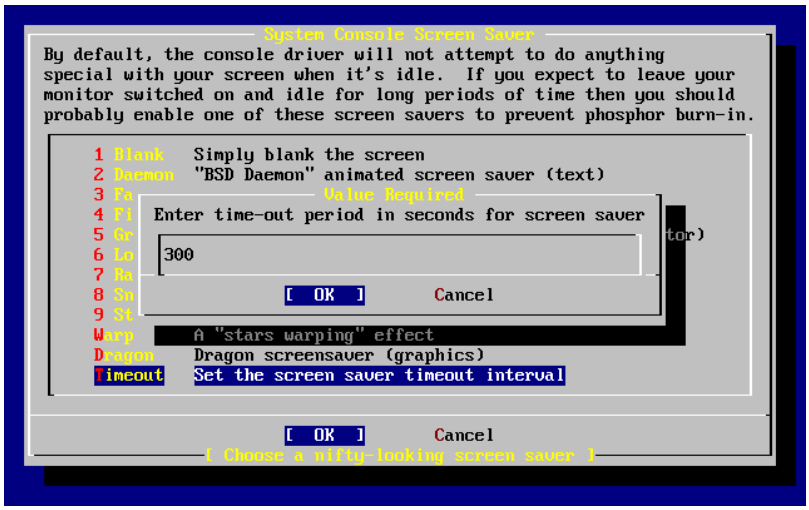


Figura 2.34. Retardo del salvapantallas

Asigne el retardo del salvapantallas a su gusto, seleccione [OK] y pulse Intro para volver al menú de configuración de la consola del sistema.

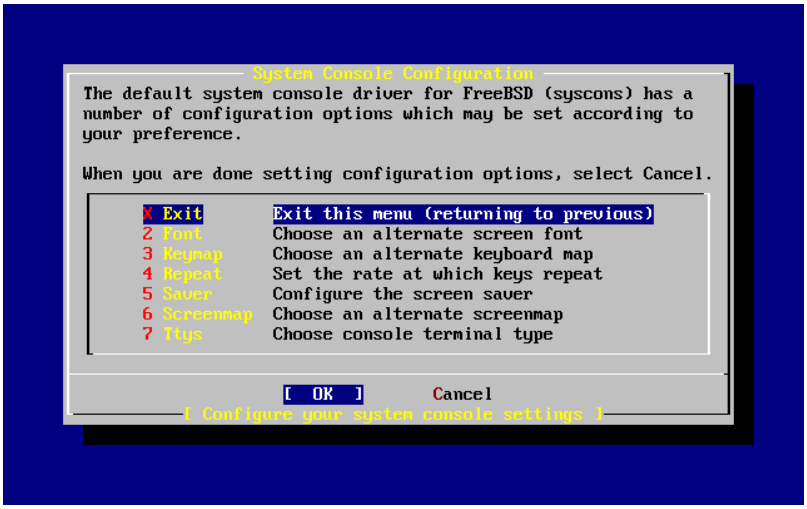


Figura 2.35. Salida del menú de configuración de la consola del sistema

Seleccione Exit y pulse Intro para seguir adelante con la configuración necesaria tras la instalación.

2.10.8. Configuración de la zona horaria

Si su sistema tiene correctamente configurada la zona horaria podrá corregir cualquier automáticamente cualquier cambio horario regional, así como cumplir adecuadamente con otras funciones relacionadas con zonas horarias.

El ejemplo que se muestra en las capturas de pantalla es de una máquina ubicada en la zona horaria del Este de los EEUU.

```
                    User Confirmation Requested
         Would you like to set this machine's time zone now?

                              [ Yes -]   No
```

Seleccione [Yes] y pulse Intro. Vamos a configurar la zona horaria del sistema.

Seleccione [Yes] y pulse Intro.

```
                        User Confirmation Requested
 Is this machine's CMOS clock set to UTC? If it is set to local time
 or you don't know, please choose NO here!

                                 Yes   [ No -]
```

Seleccione [Yes] o [No] según esté configurado el reloj del sistema y pulse Intro.

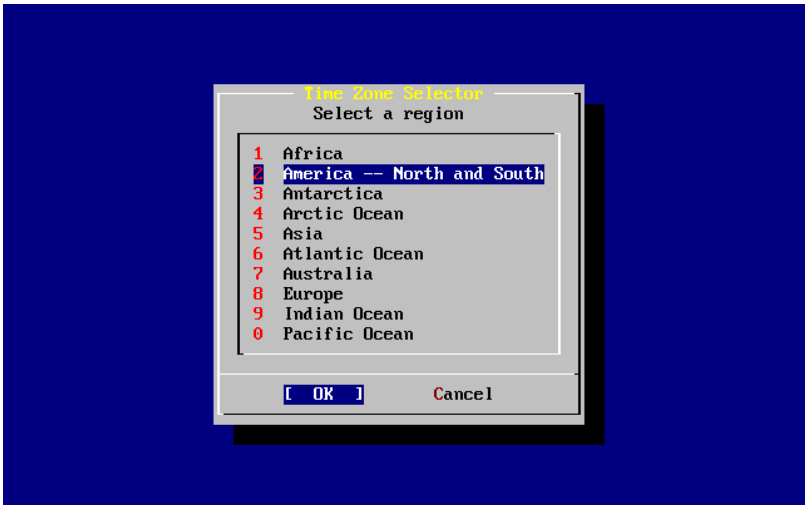


Figura 2.36. Elección de región

Elija la zona adecuada mediante las flechas y pulse Intro.

Figura 2.37. Elección de país

Elija el país adecuado con las flechas y pulse Intro.

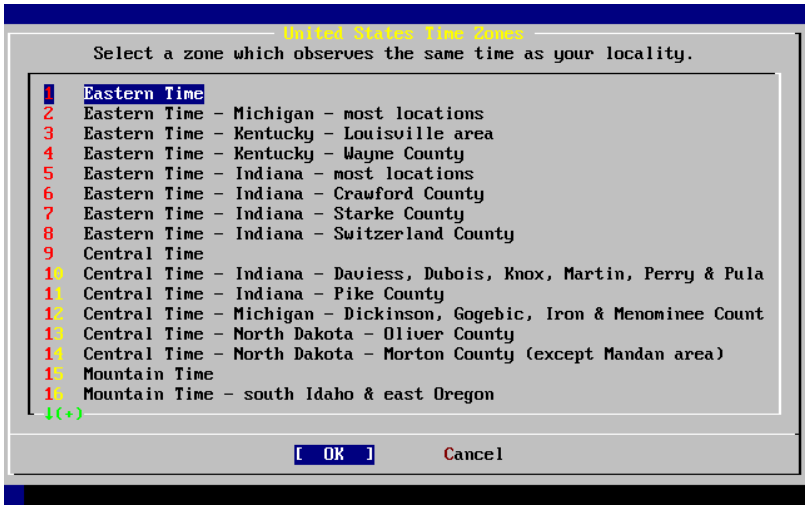
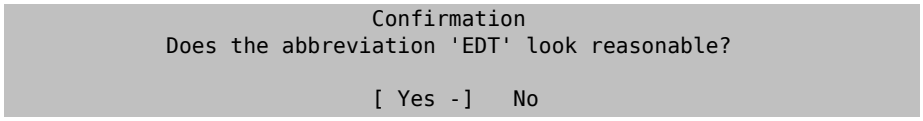


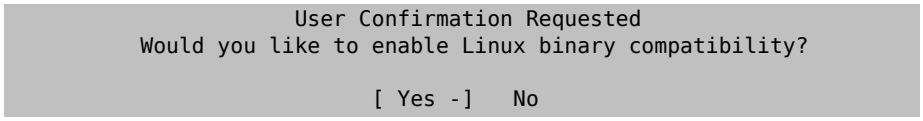
Figura 2.38. Elección de zona horaria

Elija la zona horaria adecuada con las flechas y pulse Intro.



Confirme si la abreviatura de la zona horaria es la correcta. Cuando todo esté correcto pulse Intro y siga adelante.

2.10.9. Compatibilidad con Linux®



Si selecciona [Yes] y pulsa Intro podrá ejecutar software Linux® en FreeBSD. La instalación añadirá los paquetes necesarios para poder tener compatibilidad binaria con Linux®.

Si realiza la instalación por FTP la máquina necesitará conectarse a Internet. A veces los servidores ftp no tienen todas las distribuciones, de forma que si no puede instalar la distribución de compatibilidad con Linux® no se preocupe, puede probar con otro servidor o instalarla más tarde.

2.10.10. Configuración del ratón

Esta opción le permitirá cortar y pegar texto en consola y en otros programas mediante un ratón de tres botones. Consulte moused(8) si usa uno de 2 botones, es posible emular ese

tercer botón. En el siguiente ejemplo veremos la configuración de un ratón «no USB» (es decir, PS/2 o de puerto COM):

```
                    User Confirmation Requested
          Does this system have a PS/2, serial, or bus mouse?

                                [ Yes -]    No
```

Seleccione [Yes] si tiene un ratón que no sea USB o por el contrario seleccione [No] si tiene un ratón USB. Después pulse Intro.

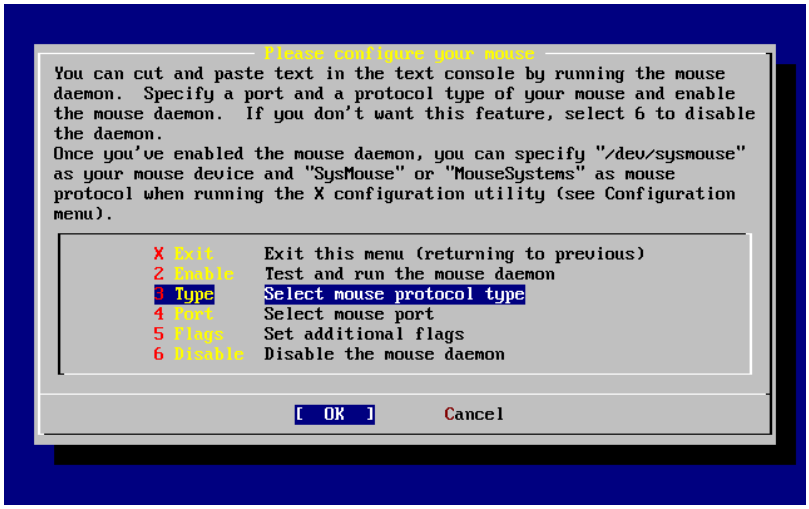


Figura 2.39. Elija el protocolo que usa el ratón

Seleccione Type usando las flechas y pulse Intro.

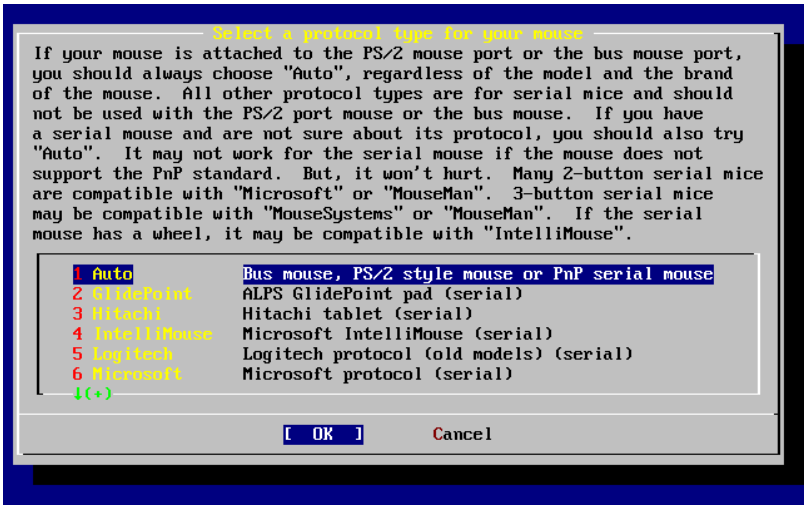


Figura 2.40. Ajuste del protocolo del ratón

En el ejemplo se ha usado un ratón PS/2, así que Auto era correcto. Para cambiar el protocolo use las flechas para moverse por el menú y elegir otra opción. Para salir seleccione [OK] y pulse Intro.

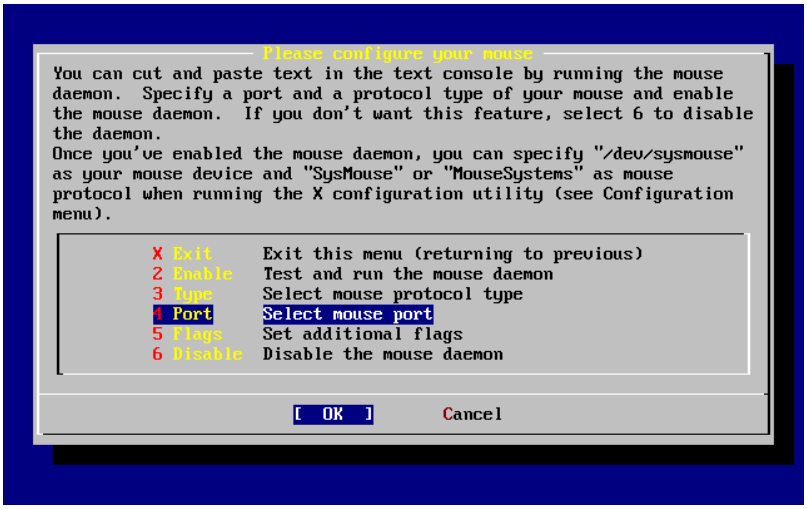


Figura 2.41. Configuración del puerto del ratón

Seleccione Port y pulse Intro.

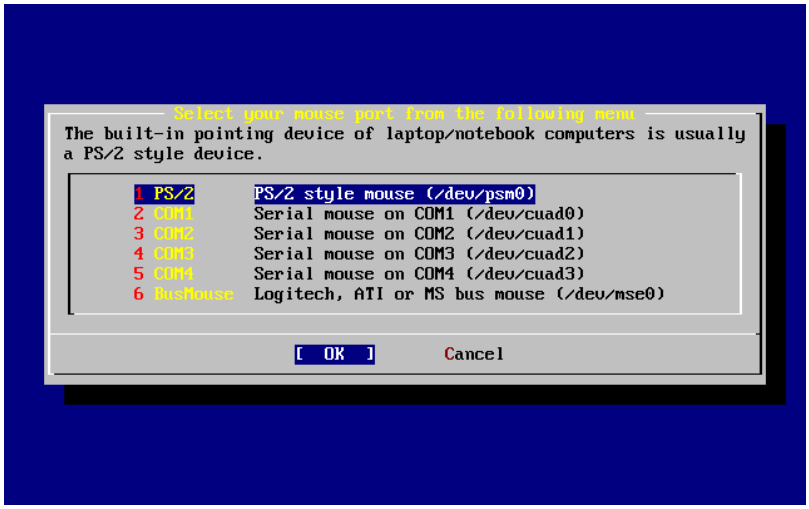


Figura 2.42. Configuración del puerto del ratón

El sistema de ejemplo tiene un ratóna PS/2, de forma que la configuración por omisión basta. Si quiere modificarla utilice las flechas y después pulseIntro.

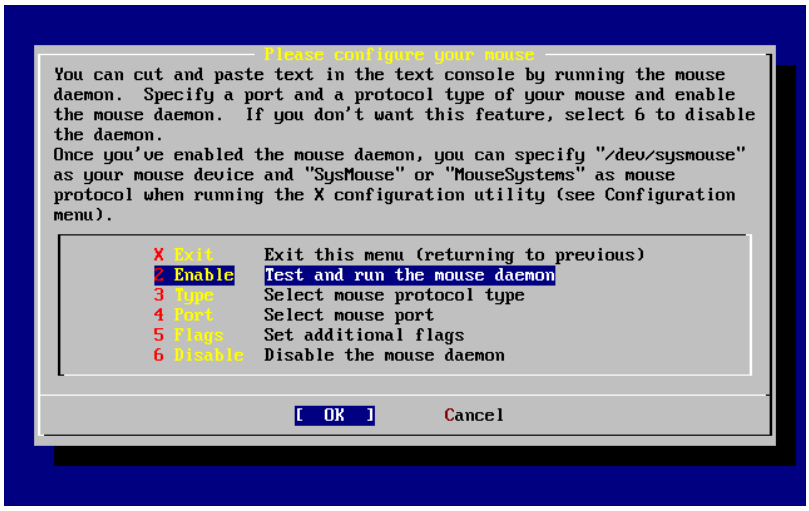


Figura 2.43. Arranque del dæmon del ratón

Por último, utilice las flechas para elegir Enabley pulse Intro; así se activa y prueba el dæmon del ratón.

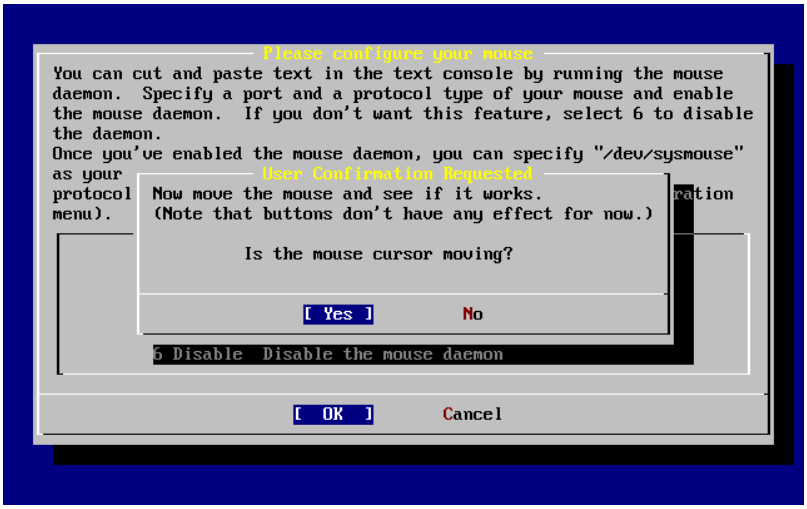


Figura 2.44. Prueba del dæmon del ratón

Mueva el ratón un poco por la pantalla hasta asegurarse de que el cursor responde adecuadamente. Si todo es correcto seleccione [Yes] y pulse Intro. Si hay algo que no funcione correctamente seleccione [No] e inténtelo con otras opciones de configuración.

Seleccione Exit y pulse Intro para volver a la configuración del sistema.

2.10.11. Instalación de «packages»

Los «packages» son binarios precompilados; son una forma muy cómoda de instalar software.

Veamos la instalación de un «package» Este es un buen momento para instalar «packages» si así lo desea. De todos modos una vez concluida la instalación puede entrar cuando quiera a `sysinstall` e instalar lo que necesite.

```
                    User Confirmation Requested
The FreeBSD package collection is a collection of hundreds of
ready-to-run applications, from text editors to games to WEB servers
and more. Would you like to browse the collection now?

                             [ Yes -]   No
```

Elija [Yes] y pulse Intro: verá la pantalla de selección de paquetes:

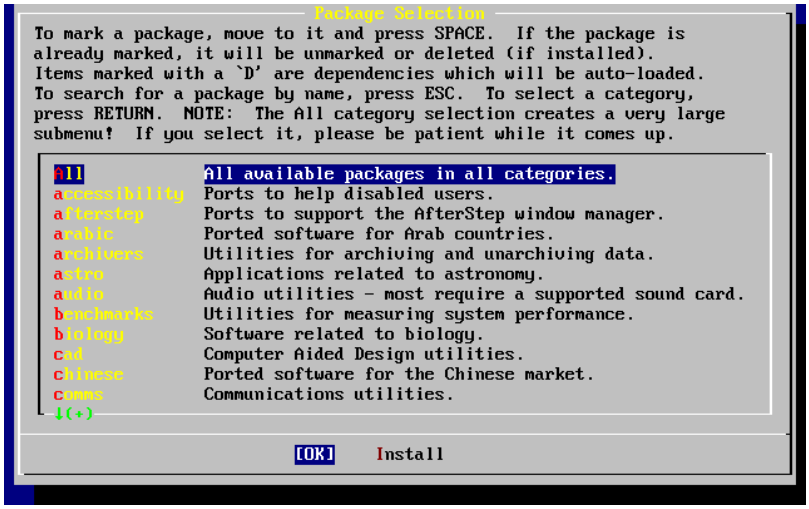


Figura 2.45. Categorías de «packages»

Los «packages» presentes en el medio de instalación que esté accesible en cada momento son los únicos que se pueden instalar.

Todos los «packages» disponibles en el medio se mostrarán al seleccionar la categoría All. Elija una categoría y pulse Intro.

Al seleccionar una categoría aparecerá un menú con los paquetes disponibles existentes en la misma:

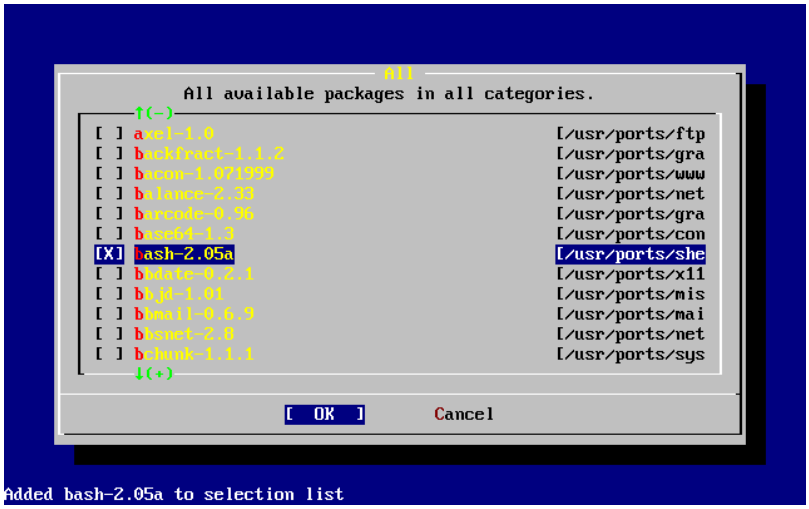


Figura 2.46. Selección de «packages»

Hemos elegido la shell bash. Seleccione tantos «packages» como quiera instalar y pulse Espacio. Se mostrará una breve descripción de cada «package» en la esquina inferior izquierda de la pantalla.

Pulsando el Tabulador desplazará el cursor entre el último «package» que haya seleccionado, [OK] y [Cancel].

Cuando haya seleccionado todos los «packages» que quiera instalar pulse Tabulador una sola vez para que el cursor pase a [OK] y pulse Intro, lo que le llevará al menú de selección de «packages».

Las flechas «izquierda» y «derecha» pueden usarse para mover el cursor entre [OK] y [Cancel]. Use esto para seleccionar [OK] y pulsar Intro para volver al menú de selección de «packages».

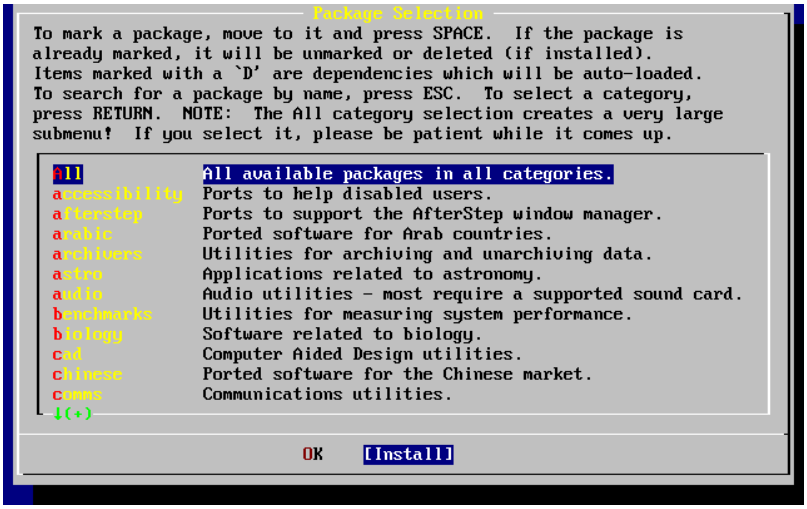


Figura 2.47. Instalación de «packages»

Utilice el tabulador y las flechas para seleccionar [Install] y pulse Intro. Se le pedirá que confirme que quiere instalar «packages»:

Figura 2.48. Confirmación previa a la instalación de «packages»

Si selecciona [OK] y pulsa Intro comenzará la instalación de «packages». Irán apareciendo mensajes relacionados con los diversos procesos de instalación hasta que se cumplan todos. Esté atento por si aparecieran mensajes de error.

Tras la instalación de «packages» nos quedan unos toques finales a la configuración. Si no ha elegido ningún «package» y quiere regresar al menú de configuración seleccione Install de todos modos.

2.10.12. Añadir usuarios y grupos

Deberí al menos añadir un usuario al sistema para poder usarlo para acceder al sistema una vez reiniciado sin tener que recurrir a `root`. La partición raíz suele ser pequeña y ejecutar aplicaciones como `root` tiene a llenarla rápidamente. Pero hay un peligro mucho mayor:

```
                    User Confirmation Requested
 Would you like to add any initial user accounts to the system? ↺
Adding
 at least one account for yourself at this stage is suggested since
 working as the "root" user is dangerous (it is easy to do things ↺
which
 adversely affect the entire system).

                            [ Yes -]   No
```

Es decir:

```
     Petición de confirmación del usuario
 ?Quiere añadir algún usuario al sistema? Le sugerimos que añada al ↺
menos
```

```
 uno para usted puesto que trabajar como "root" es peligroso (es ↺
fácil
 hacer algo con consecuencias en todo el sistema).

                              [ Yes -]   No
```

Seleccione [Yes] y pulse Intro para añadir un usuario.

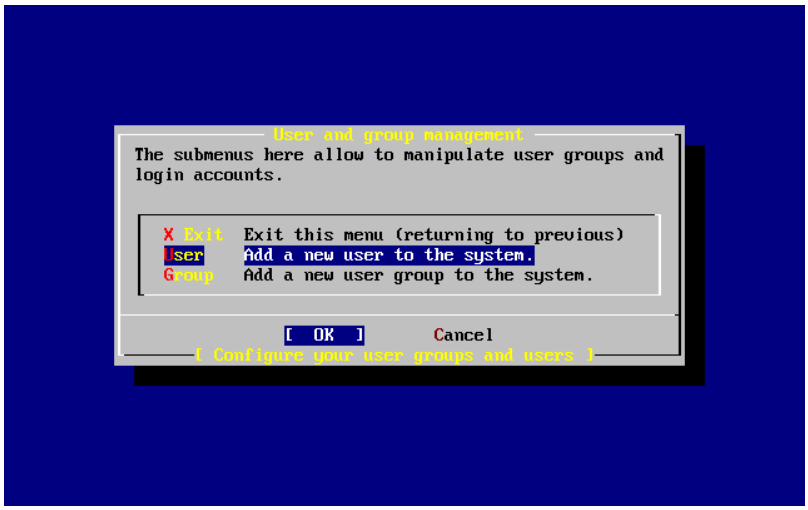


Figura 2.49. Selección de «Añadir un usuario»

Seleccione User con las flechas y y pulseIntro.

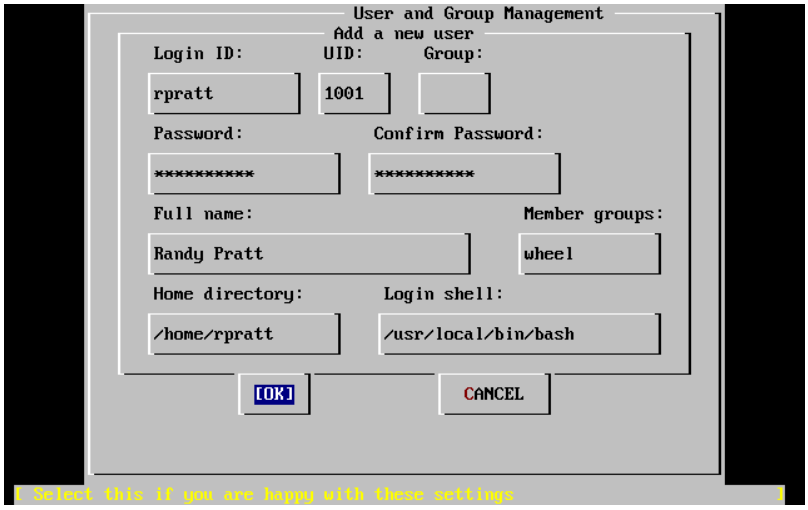


Figura 2.50. Añadir la información del usuario

A medida que vaya seleccionando los campos ayudándose del Tabulador se le irán mostrando las siguientes descripciones en la parte baja de la pantalla:

Login ID
: El nombre del usuario (obligatorio).

UID
: El ID numérico del usuario Déjelo en blanco si quiere el que sistema lo asigne automáticamente.

Group
: El grupo al que pertenecerá el usuario. Déjelo en blanco si quiere que el sistema lo asigne automáticamente.

Password
: La contraseña del usuario. *Rellene este campo con mucho cuidado.*

Full name
: El nombre completo del usuario (o un comentario descriptivo.

Member groups
: Los grupos a los que este usuario pertenece, es decir, de los que hereda sus derechos de acceso.

Home directory
: El directorio «home» del usuario; déjelo en blanco si quiere que el sistema lo asigne automáticamente.

Login shell
: La shell con la que el usuario accederá al sistema. Déjela en blanco si le sirve `/bin/sh`, la shell por omisión.

En nuestro ejemplo no se eligió `/bin/sh` sino `/usr/local/bin/bash` , para lo cual hubo que instalar previamente la shell bash como «package». No intente usar una shell que no existe o no podrá acceder al sistema con ese usuario. La shell más habitual en el mundo BSD es `/bin/tcsh` , la «C shell».

El usuario se añadió también al grupo `wheel` para que pueda convertirse en superusuario con privilegios de `root`.

Cuando haya terminado pulse `[ OK ]` y volverá al menú de gestión de grupos y usuarios.

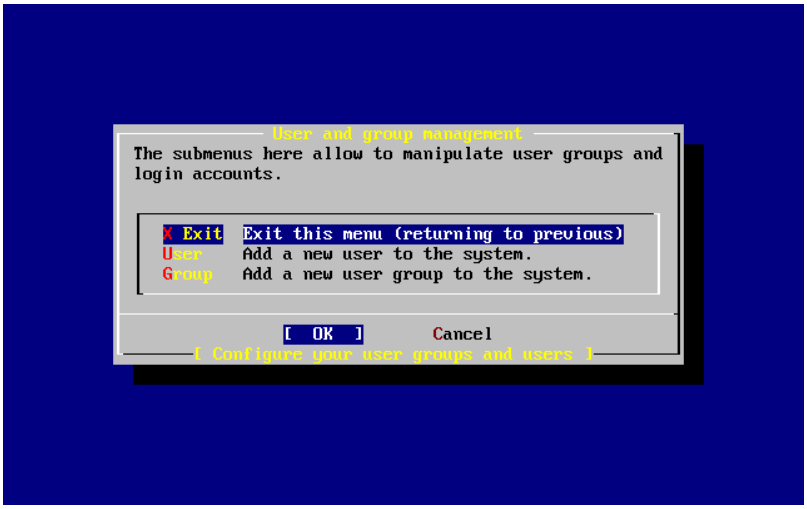


Figura 2.51. Salir de la gestión de usuarios y grupos

Los grupos también pueden añadirse en este momento si fuera necesario. También puede hacerse desde `sysinstall` una vez culminada la instalación.

Cuando haya acabado de añadir usuarios seleccione Exit con las flechas y pulse Intro y siga con la instalación.

2.10.13. Asignar contraseña a root Password

```
                        Message
Now you must set the system manager's password.
This is the password you'll use to log in as "root".

                         [ OK -]

               [ Press enter or space -]
```

Es decir:

```
                        Mensaje
Debe asignar la contraseña del administrador del sistema.
Esta es la contraseña que usará cuando acceda al sistema como
"root".

                         [ OK -]

                    [ Pulse Intro -]
```

Pulse Intro y asígnele una contraseña a `root`.

Tendrá que escribir la contraseña correctamente dos veces. No hay necesidad de decirle que es importante que recuerde esa contraseña.

```
New password:
Retype new password :
```

La instalación proseguirá una vez que introduzca la contraseña correctamente dos veces.

2.10.14. Salir de la instalación

Si tiene que configurar servicios de red o cualquier otra cosa, puede hacerlo ahora mismo o tras terminar la instalación ejecutando `sysinstall`.

```
                    User Confirmation Requested
Visit the general configuration menu for a chance to set any last
options?

                                Yes   [ No -]
```

Seleccione [No] con las flechas y pulse Intro para volver al menú principal de la instalación.

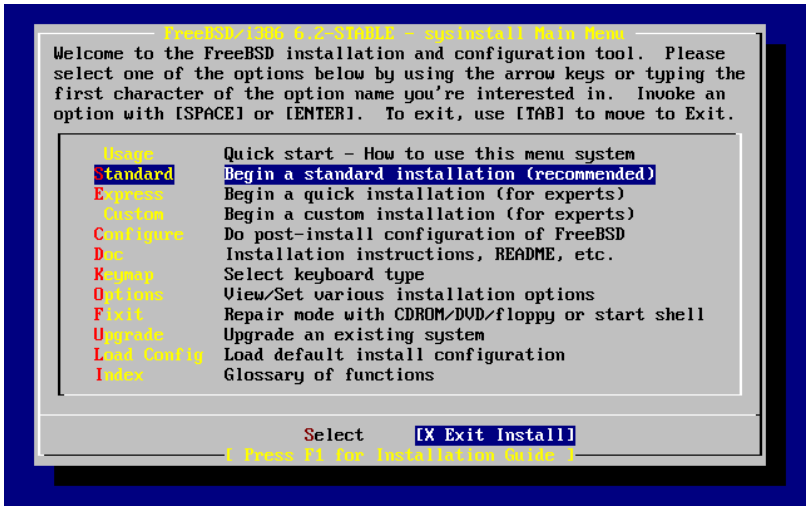


Figura 2.52. Salir de la instalación

Seleccione [X Exit Install] con las flechas y pulse Intro. Se le pedirá que confirme que quiere salir de la instalación:

```
                    User Confirmation Requested
Are you sure you wish to exit? The system will reboot (be sure to
remove any floppies/CDs/DVDs from the drives).

                              [ Yes -]   No
```

Es decir:

```
            Petición del Confirmación al usuario
 ?Seguro que quiere salir?  El sistema reiniciará (compruebe que
 ha retirado los disquetes de la unidad).

                              [ Yes -]   No
```

Seleccione [Yes] y extraiga el disquete si ha arrancado desde floppy. La unidad CDROM está bloqueada hasta que la máquina comience a reiniciarse. La unidad CDROM se desbloquea y (actuando con agilidad felina) puede extraerse el CDROM.

El sistema reiniciará. Esté atento por si aparece algún mensaje de error.

Seleccione [Yes] y extraiga el disquete si ha arrancado desde floppy. La unidad lectora de CDROM está bloqueada hasta que la máquina comience a reiniciarse. La unidad CDROM se desbloquea y (actuando con agilidad felina) puede extraerse el CDROM.

Si apareciera algún error durante el arranque consulte la Sección 2.10.16, “El arranque de FreeBSD”.

2.10.15. Configuración de servicios de red

Escrito por Tom Rhodes.

La configuración de servicios de red puede ser una tarea peliaguda para usuarios inexpertos si no tienen demasiados conocimientos en la materia. Todo lo relacionado con las redes (y ahí entra Internet) tiene una importancia crítica en cualquier sistema operativo moderno y FreeBSD no es una excepción. Es por esta razón que le será muy útil saber un poco sobre la conectividad en general de FreeBSD desde el momento mismo de la instalación y los servicios de que puede disponer.

Los servicios de red son programas que aceptan entradas de datos desde cualquier punto de la red. Se pone mucho empeño en evitar que estos programas puedan ser «dañinos» pero por desgracia los programadores no son perfectos y de tanto en cuanto aparecen errores en los servicios de red que algunos logran aprovechar para hacer maldades. Es crucial que solamente active los servicios que sabe que va a necesitar. Si tiene dudas con alguno lo mejor es que no lo active hasta que no sea evidente que lo necesita. Puede activarlo cuando sea necesario ejecutando later by re-running sysinstall o utilizando el fichero `/etc/rc.conf` file.

Si selecciona la opción Networking verá un menú muy parecido a este:

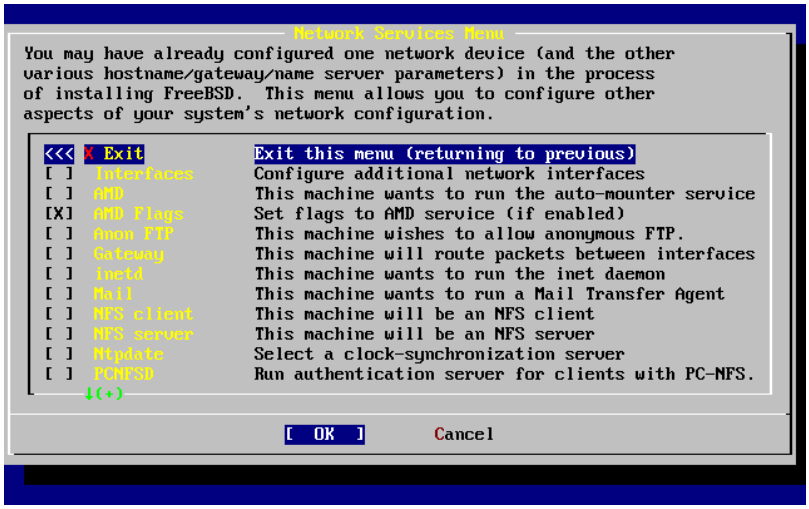


Figura 2.53. Primer nivel de servicios de red

La primera opción, Interfaces, la hemos visto en la Sección 2.10.1, “Network Device Configuration”, así que podemos ignorarla.

Si selecciona la opción AMD añadirá al sistema la aplicación montaje automático de dispositivos de BSD. Suele usarse junto con el protocolo NFS protocol (ver más adelante) para automatizar el montaje de sistemas de ficheros remotos. No es necesario que configure nada.

La siguiente opción es AMD Flags. Si la selecciona verá un menú emergente que le requerirá parámetros específicos de AMD. El menú dispone ya de un conjunto de opciones por omisión:

```
-a /.amd_mnt -l syslog /host /etc/amd.map /net /etc/amd.map
```

La opción `-a` fija el punto de montaje por omisión, en este caso `/.amd_mnt` . La opción `-l` indica el `log` por omisió, aunque si usa `syslogd` todos los datos de log se enviarán al dæmon de logs del sistema. El directorio `/host` se usa para montar sistemas de ficheros exportados desde una máquina remota, mientras que el directorio `/net` se usa para montar sistemas de ficheros exportados desde una dirección IP. El fichero `/etc/amd.map` define las opciones de exportación que AMD exporta por omisión.

La opción Anon FTP permite conexiones FTP anónimas. Seleccione esta opción si quiere utilizar la máquina como servidor servidor FTP anónimo. Tenga muy en cuenta los riesgos de seguridad que conlleva esta opción. Se le ofrecerá otro menú en el que se explican en profundidad los riesgos de seguridad y la configuración.

El menú de configuración Gateway preparará la máquina para que cumpla las funciones de pasarela, tal y como se ha explicado previamente. Puede utilizarlo también para desactivar la opción Gateway si la seleccionón sin querer durante la instalación.

La opción Inetd permite desactivar el dæmon inetd(8).

La opción Mail se usa para configurar el MTA (de Mail Transfer Agent, agente de transferencia de correo) del sistema. Si selecciona esta opción llegará al siguiente menú:

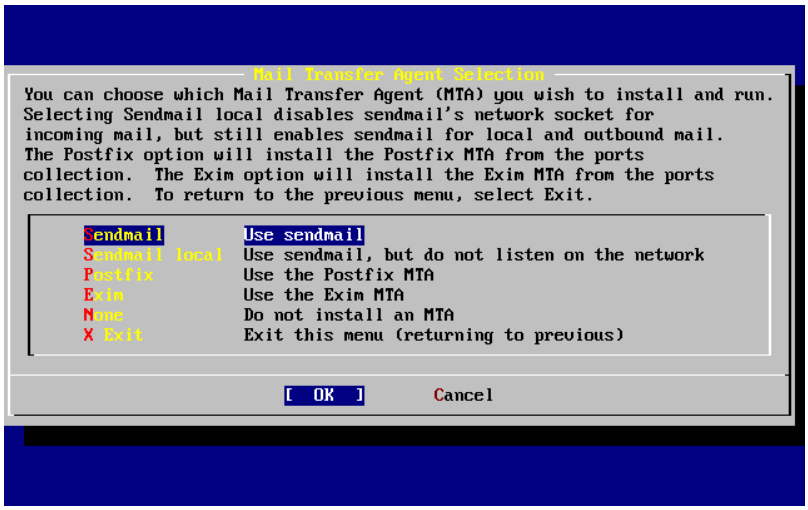


Figura 2.54. Elegir el MTA del sistema

Aquí se le pide que elija qué MTA quiere instalar en su sistema. Un MTA no es otra cosa que un servidor de correo que entrega correo electrónico a los usuarios del sistema o los que acceden a través de Internet.

Si selecciona Sendmail instalará el célebre sendmail, el MTA de FreeBSD por omisión. La opción Sendmail local hará que sendmail sea el MTA del sistema pero desactivará la capacidad de recibir correo entrante desde Internet. Las demás opciones, Postfix y Exim son en el fonodo similares a Sendmail. Ambas aplicaciones gestionan correo aunque hay usuarios que los elegirían como su MTA antes que sendmail.

Tras elegir un MTA (o si ha elegido no usar ninguno) el menú de configuración de red nos muestra la siguiente opción:NFS client.

La opción NFS client configura el sistema para comunicarse con servidores NFS. Un servidor NFS permite que, mediante el uso del protocolo NFS, otras máquinas de la red puedan acceder a sus sistemas de ficheros. Si no hay más máquinas en su red puede dejar la opción sin seleccionar. El sistema puede necesitar más configuración; consulte la Sección 29.6, “NFS” para más información sobre cómo configurar el cliente y el servidor.

La siguiente opción es NFS server, que le permitirá configurar su sistema como servidor NFS. Con esta opción añadirá la información necesaria para el arranque de RPC, los servicios de llamada de procedimientos remotos. RPC se usa para coordinar conexiones entre sistemas y programas.

La siguiente opción es Ntpdate, la sincronización de la hora. Al seleccionarla verá un menú como este:

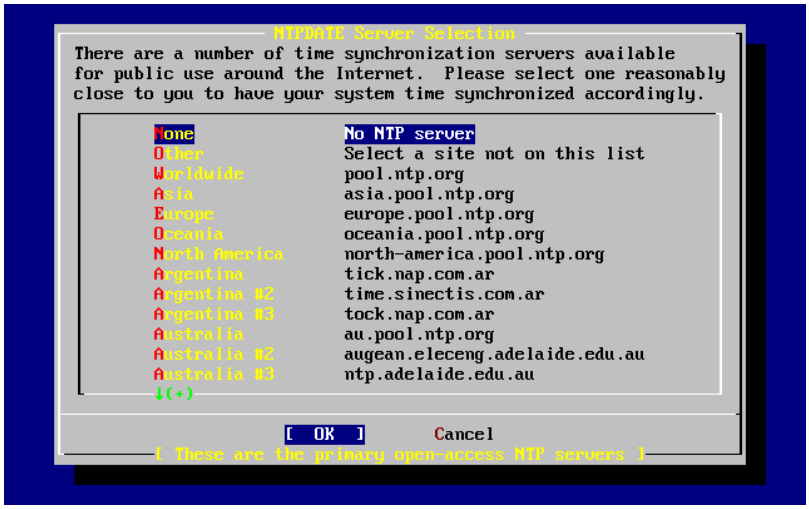


Figura 2.55. Configuración de Ntpdate

Elija en el menú el servidor más cercano al lugar donde está la máquina. De este modo la sincronozación será más precisa, puesto que un servidor más lejano puede padecer de conexiones con más latencia.

La siguiente opción es PCNFSD. Esta opción instalará el «package» net/pcnfsd desde la colección de Ports. Es una aplicación muy útil que da servicios de autentificación NFS a sistemas que son incapaces de tenerlos por sí mismos, como por ejemplo el sistema operativo MS-DOS® de Microsoft.

Si usa las flechas podrá ver más opciones:

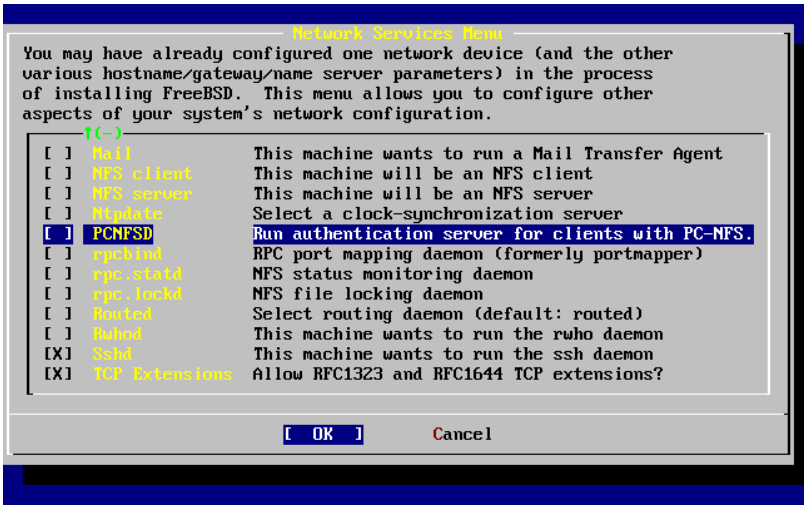


Figura 2.56. Segundo nivel de configuración de red

Las aplicaciones rpcbind(8), rpc.statd(8) y rpc.lockd(8) utilities se utilizan en las llamadas de procesos remotos (RPC). `rpcbind` gestiona la comunicación entre servidores y clientes NFS; los servidores NFS lo necesitan para poder funcionar correctamente. El dæmon rpc.statd se comunica con el dæmon rpc.statd de otras máquinas para facilitar seguimiento de estado. La información de estado se deposita por omisión en `/var/db/statd.status` . La siguiente opción que vemos es rpc.lockd; si se selecciona facilita servicios de bloqueo de ficheros. Se suele usar conjuntamente con rpc.statd para monitorizar qué máquina pide bloqueos y con qué frecuencia. Si bien es cierto que estas dos útimas opciones son fantásticas para depuración, también lo es que no son necesarias para que clientes y servidores NFS funcionen correctamente.

El siguiente elemento de la lista es Routed, un dæmon de encaminamiento. routed(8) gestiona tablas de rutas, encuentra «routers» multicast y, bajo petición, facilita una copia de la tabla de rutas a cualquier máquina conectada físicamente a la red. Suele usarse principalmente en máquinas que hacen funciones de pasarela de una red local. Si la selecciona verá un menú en el que se le requerirá que indique la ubicación de la aplicación. Por omisión ya está configurada, así que basta que la confirme pulsando Intro key. Se le presentará un menú más, pidiendole esta vez parámetros que quiera proporcionarle a routed. Por omisión `-q` estará ya dispuesto y debería aparecer así en pantalla.

La siguiente opción que aparece es Rwhod; si la selecciona arrancará el dæmon rwhod(8) durante el arranque del sistema. La aplicación `rwhod` envía mensajes del sistema (en broadcast) periódicamente a través de la red, o si está en modo «consumer» los va recogiendo. Tiene más información en las páginas de manual de ruptime(1) y rwho(1).

La siguiente opción del menú está relacionada con el dæmon sshd(8). Se trata del servidor de shell segura de OpenSSH y le recomendamos encarecidamente su uso en lugar de los

servidores telnet y FTP. El servidor sshd se usa para crear una conexión segura de una máquina a otra mediante conexiones cifradas.

La última opción es TCP Extensions, que activa las extensiones TCP definidas en las RFC 1323 y RFC 1644. En algunas máquinas puede acelerar conexiones, pero también puede haber conexiones que se pierdan. No se recomienda su uso en servidores, pero puede ser de utilidad en máquinas aisladas.

Una vez configurados los servicios de red, vaya al principio del menú (X Exit)a o vuelva a sysinstall seleccionando dos veces X Exit y después [X Exit Install].

2.10.16. El arranque de FreeBSD

2.10.16.1. El arranque de FreeBSD/i386

Si todo ha ido bien debería ver una pantalla de mensajes pasando frente a usted hasta que llegue a lo que llamamos un «login prompt». Puede volver a ver los mensajes del arranque pulsando Bloq Deslp y usando RePág y AvPág. Pulsando Bloq Despl otra vez volverá al prompt.

El mensaje del arranque no puede mostrarse completo debido a las limitaciones del búfer, pero puede consultarse desde la shell mediante `dmesg`.

Entre al sistema utilizando un nombre de usuario y contraseña que haya creado durante la instalación (en nuestro ejemplo `rpratt`). Evite entrar al sistema como `root` salvo en los casos en los que sea estrictamente necesario.

Este es un típico mensaje de arranque (se ha eliminado la información de la versión):

```
Copyright (c) 1992-2002 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, ↺
1994
        The Regents of the University of California. All rights ↺
reserved.

Timecounter "i8254"  frequency 1193182 Hz
CPU: AMD-K6(tm) 3D processor (300.68-MHz 586-class CPU)
  Origin = "AuthenticAMD"  Id = 0x580  Stepping = 0
  Features=0x8001bf<FPU,VME,DE,PSE,TSC,MSR,MCE,CX8,MMX>
  AMD Features=0x80000800<SYSCALL,3DNow!>
real memory  = 268435456 (262144K bytes)
config> di sn0
config> di lnc0
config> di le0
config> di ie0
config> di fe0
config> di cs0
config> di bt0
config> di aic0
config> di aha0
config> di adv0
```

```
config> q
avail memory = 256311296 (250304K bytes)
Preloaded elf kernel "kernel" at 0xc0491000.
Preloaded userconfig_script "/boot/kernel.conf" at 0xc049109c.
md0: Malloc disk
Using $PIR table, 4 entries at 0xc00fde60
npx0: <math processor> on motherboard
npx0: INT 16 interface
pcib0: <Host to PCI bridge> on motherboard
pci0: <PCI bus> on pcib0
pcib1: <VIA 82C598MVP (Apollo MVP3) PCI-PCI (AGP) bridge> at device ↺
1.0 on pci0
pci1: <PCI bus> on pcib1
pci1: <Matrox MGA G200 AGP graphics accelerator> at 0.0 irq 11
isab0: <VIA 82C586 PCI-ISA bridge> at device 7.0 on pci0
isa0: <ISA bus> on isab0
atapci0: <VIA 82C586 ATA33 controller> port 0xe000-0xe00f at device ↺
7.1 on pci0
ata0: at 0x1f0 irq 14 on atapci0
ata1: at 0x170 irq 15 on atapci0
uhci0: <VIA 83C572 USB controller> port 0xe400-0xe41f irq 10 at ↺
device 7.2 on pci0
usb0: <VIA 83C572 USB controller> on uhci0
usb0: USB revision 1.0
uhub0: VIA UHCI root hub, class 9/0, rev 1.00/1.00, addr 1
uhub0: 2 ports with 2 removable, self powered
chip1: <VIA 82C586B ACPI interface> at device 7.3 on pci0
ed0: <NE2000 PCI Ethernet (RealTek 8029)> port 0xe800-0xe81f irq 9 at
device 10.0 on pci0
ed0: address 52:54:05:de:73:1b, type NE2000 (16 bit)
isa0: too many dependant configs (8)
isa0: unexpected small tag 14
fdc0: <NEC 72065B or clone> at port 0x3f0-0x3f5,0x3f7 irq 6 drq 2 ↺
on isa0
fdc0: FIFO enabled, 8 bytes threshold
fd0: <1440-KB 3.5" drive> on fdc0 drive 0
atkbdc0: <keyboard controller (i8042)> at port 0x60-0x64 on isa0
atkbd0: <AT Keyboard> flags 0x1 irq 1 on atkbdc0
kbd0 at atkbd0
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: model Generic PS/2 mouse, device ID 0
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff ↺
on isa0
sc0: <System console> at flags 0x1 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
sio0 at port 0x3f8-0x3ff irq 4 flags 0x10 on isa0
sio0: type 16550A
sio1 at port 0x2f8-0x2ff irq 3 on isa0
sio1: type 16550A
ppc0: <Parallel port> at port 0x378-0x37f irq 7 on isa0
ppc0: SMC-like chipset (ECP/EPP/PS2/NIBBLE) in COMPATIBLE mode
ppc0: FIFO with 16/16/15 bytes threshold
ppbus0: IEEE1284 device found /NIBBLE
```

```
Probing for PnP devices on ppbus0:
plip0: <PLIP network interface> on ppbus0
lpt0: <Printer> on ppbus0
lpt0: Interrupt-driven port
ppi0: <Parallel I/O> on ppbus0
ad0: 8063MB <IBM-DHEA-38451> [16383/16/63] at ata0-master using ↺
UDMA33
ad2: 8063MB <IBM-DHEA-38451> [16383/16/63] at ata1-master using ↺
UDMA33
acd0: CDROM <DELTA OTC-H101/ST3 F/W by OIPD> at ata0-slave using PIO4
Mounting root from ufs:/dev/ad0s1a
swapon: adding /dev/ad0s1b as swap device
Automatic boot in progress...
/dev/ad0s1a: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1a: clean, 48752 free (552 frags, 6025 blocks, 0.9% ↺
fragmentation)
/dev/ad0s1f: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1f: clean, 128997 free (21 frags, 16122 blocks, 0.0% ↺
fragmentation)
/dev/ad0s1g: FILESYSTEM CLEAN; SKIPPING CHECKS
/dev/ad0s1g: clean, 3036299 free (43175 frags, 374073 blocks, 1.3% ↺
fragmentation)
/dev/ad0s1e: filesystem CLEAN; SKIPPING CHECKS
/dev/ad0s1e: clean, 128193 free (17 frags, 16022 blocks, 0.0% ↺
fragmentation)
Doing initial network setup: hostname.
ed0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet 192.168.0.1 netmask 0xffffff00 broadcast 192.168.0.255
        inet6 fe80::5054::5ff::fede:731b%ed0 prefixlen 64 tentative ↺
scopeid 0x1
        ether 52:54:05:de:73:1b
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x8
        inet6 ::1 prefixlen 128
        inet 127.0.0.1 netmask 0xff000000
Additional routing options: IP gateway=YES TCP keepalive=YES
routing daemons:.
additional daemons: syslogd.
Doing additional network setup:.
Starting final network daemons: creating ssh RSA host key
Generating public/private rsa1 key pair.
Your identification has been saved in /etc/ssh/ssh_host_key.
Your public key has been saved in /etc/ssh/ssh_host_key.pub.
The key fingerprint is:
cd:76:89:16:69:0e:d0:6e:f8:66:d0:07:26:3c:7e:2d root@k6-2.example.com
 creating ssh DSA host key
Generating public/private dsa key pair.
Your identification has been saved in /etc/ssh/ssh_host_dsa_key.
Your public key has been saved in /etc/ssh/ssh_host_dsa_key.pub.
The key fingerprint is:
f9:a1:a9:47:c4:ad:f9:8d:52:b8:b8:ff:8c:ad:2d:e6 root@k6-2.example.↺
com.
setting ELF ldconfig path: /usr/lib /usr/lib/compat /usr/X11R6/lib
```

```
/usr/local/lib
a.out ldconfig path: /usr/lib/aout /usr/lib/compat/aout /usr/X11R6/
lib/aout
starting standard daemons: inetd cron sshd usbd sendmail.
Initial rc.i386 initialization:.
rc.i386 configuring syscons: blank_time screensaver moused.
Additional ABI support: linux.
Local package initialization:.
Additional TCP options:.

FreeBSD/i386 (k6-2.example.com) (ttyv0)

login: rpratt
Password:
```

La generación de claves RSA y DSA puede llevar su tiempo en máquinas lentas. Esto solamente sucede en el primer arranque después de la instalación. Los demás arranques serán más rápidos.

Si ya hay un servidor X configurado y ha elegido un escritorio por omisión puede arrancarlo tecleando `startx` en la shell.

2.10.16.2. El arranque de FreeBSD/alpha

Una vez acabado el proceso de instalación podrá arrancar FreeBSD tecleando algo muy parecido a esto en el prompt de SRM:

Once the install procedure has finished, you will be able to start FreeBSD by typing something like this to the SRM prompt:

```
>>>BOOT DKC0
```

Esto hace que el firmware arranque desde el disco especificado. Para que FreeBSD arranque automáticamente de ahí en adelante utilice lo siguiente:

```
>>> SET BOOT_OSFLAGS A
>>> SET BOOT_FILE ''
>>> SET BOOTDEF_DEV DKC0
>>> SET AUTO_ACTION BOOT
```

Los mensajes de arranque serán muy similares (aunque no iguales) a los que aparecen durante el arranque de FreeBSD en i386™.

2.10.17. El apagado de FreeBSD

Es importante apagar correctamente el sistema operativo. No basta con darle al interruptor. Lo primero que debe hacer es convertirse en superusuario tecleando en la shell `su` e introduciendo la contraseña de `root`. Tenga en cuenta que esto funcionará solamente

si el usuario es miembro del grupo `wheel`. Si no lo es siempre puede acceder al sistema como `root` y utilizar `shutdown -h now`.

```
The operating system has halted.
Please press any key to reboot.
```

Una vez que vea el mensaje de «*Please press any key to reboot* » puede apagar la máquina. Si pulsa cualquier tecla en lugar de apagar la máquina el sistema reiniciará.

También puede usar la combinación de teclas Ctrl+Alt+Supr para reiniciar el sistema, aunque no le recomendamos que lo haga durante el funcionamiento normal del sistema.

2.11. Solución de problemas

Esta sección trata sobre la solución de problemas habituales durante la instalación que la comunidad de usuarios ha documentado. Hay también unas cuantas preguntas y respuestas para quienes quieren poder elegir en el arranque entre FreeBSD y MS-DOS®.

2.11.1. Qué hacer si algo va mal

A causa de las diversas limitaciones de la arquitectura PC es imposible hacer que nada funcione con un 100% de probabilidades, pero al menos hay unas cuantas cosas que pueden hacerse si algo va mal.

Consulte la lista de hardware soportado de su versión de FreeBSD y asegúrese de que su hardware debería funcionar con ella.

Si su hardware está soportado y sigue experimentando «cuelgues» u otro tipo de problemas tendrá que compilar un kernel personalizado. Esto le permitirá disponer de soporte que no está incluido en el kernel `GENERIC`. El kernel que se incluye en los discos de arranque está configurado asumiendo que la mayoría de los dispositivos van a funcionar con la configuración por omisión de fábrica (en términos de IRQ, direcciones IO y canales DMA). Si su hardware ha sido reconfigurado quizás tendrá que usar el editor de configuración para decirle a FreeBSD dónde encontrar cada cosa.

Es posible también que la prueba de un dispositivo que no está en el sistema cause problemas más tarde al probar un dispositivo que sí está en la máquina. Si ese puede ser su caso desactive las pruebas de controladores conflictivos.

Nota

Algunos problemas de instalación pueden evitarse o al menos aliviarse actualizando el firmware de ciertos tipos de hardware; esto

es especialmente cierto en el caso de las placas El firmware de las placas base se llama también BIOS y la mayoría de las placas base y de los fabricantes de computadoras tienen un sitio web en el que suelen encontrarse actualizaciones e información sobre su uso.

La mayoría de los fabricantes recomienda que no se actualice la BIOS de la placa base a menos que tenga una buena razón para hacerlo, por ejemplo una actualización crítica. El proceso de actualización *podría* fallar, lo que puede acarrrear daños en el chip de la BIOS.

2.11.2. Uso de sistemas de ficheros de MS-DOS® y Windows®

No puede usar sistemas de ficheros comprimidos con Double Space™. Deberá por lo tanto descomprimir el sistema de ficheros antes de acceder a los datos desde FreeBSD. Ejecute el Agente de compresión desde Inicio> Programas > Herramientas del sistema.

FreeBSD puede usar sistemas de ficheros MS-DOS®. mount_msdos(8) inserta estos sistemas de ficheros en la jerarquía de directorios del sistema, haciendo posible el acceso a los datos. mount_msdos(8) no suele usarse directamente; el sistema lo utiliza debido a alguna entrada en `/etc/fstab` o porque mount(8) lo invoque debido a que se le hayan dado los parámetros adecuados para ello.

Veamos una de esas entradas en el fichero `/etc/fstab`:

```
/dev/ad0sN  /dos  msdosfs rw  0 0
```

El directorio `/dos` debe existir previamente o no funcionará. Tiene todos los detalles del formato correcto de las entradas en `/etc/fstab` en fstab(5).

Veamos ahora un ejemplo de llamada de mount(8) a un sistema de ficheros MS-DOS®:

```
# mount_msdos /dev/ad0s1 /mnt
```

En el ejemplo el sistema de ficheros MS-DOS® está en la primera partición del primer disco duro. La situación en su sistema puede ser diferente: compruebe la salida de `dmesg` y `mount`. Deberían darle suficiente información como para darle una idea completa de la estructura de particiones en la que está trabajando.

Nota

FreeBSD suele numerar las las «slices» (que es como decir los sistemas de ficheros MS-DOS®) de un modo distinto al de otros sistemas operativos. Más concretamente, las particiones extendidas de MS-DOS® suelen tener un número mayor de «slice» que las particiones

> primarias de MS-DOS®. fdisk(8) le será de ayuda a la hora de determinar qué «slices» pertenecen a FreeBSD y cuáles a otros sistemas operativos.

Las particiones NTFS se montan de una forma muy similar gracias a mount_ntfs(8).

2.11.3. Preguntas y respuestas de la resolución de problemas

P: Mi sistema se «cuelga» durante el arranque, o se comporta de modo extraño durante la instalación, o no llega a comprobar el funcionamiento del lector de disquetes.

R: FreeBSD utiliza profusamente el servicio ACPI del sistema en las plataformas i386, amd64 e ia64 con el fin de ayudar en la configuración del sistema durante la detección de hardware durante el arranque. Por desgracia todavía hay errores tanto en el controlador ACPI como en algunas BIOS y placas base. Puede desactivar ACPI en la tercera fase del cargador de arranque del sistema haciendo que `hint.acpi.0.disabled` tenga el siguiente valor:

```
set hint.acpi.0.disabled="1"
```

Tenga en cuenta que tendría que volver a hacerlo cada vez que el sistema arranque, así que añada `hint.acpi.0.disabled="1"` al fichero `/boot/loader.conf` . Tiene más información sobre el cargador de arranque en Sección 12.1, “Sinopsis”.

P: Cuando voy a arrancar FreeBSD por primera vez después de la instalación el kernel carga y prueba mi hardware, pero se detiene y muestra mensajes como este:

```
changing root device to ad1s1a panic: cannot mount root
```

?Cuál es el problema?

?Qué es lo que significa este mensaje de `bios_drive:interface(unit,partition)kernel_name` que aparece en la ayuda del arranque?

R: Hay un problema que ya lleva tiempo con nosotros cuando el disco desde el que se arranca no es el primero en el sistema. La BIOS utiliza un esquema de numeración distinto al que usa FreeBSD y a veces saber qué número es el que corresponde a qué resulta un poco complicado.

Cuando el disco de arranque no sea el primer disco del sistema FreeBSD necesitará un poco de ayuda para poderlo encontrar. Puede haber dos escenarios donde suceda esto y en ambos hay que decirle a FreeBSD, dónde encontrar el sistema de ficheros raíz. Esto se hace especificando el número de disco según la BIOS, el tipo de disco y el número de disco que FreeBSD le dará a ese disco.

El primer caso es cuando hay dos discos IDE, ambos configurados como maestros de sus respectivos buses IDE y quiere usted arrancar desde el segundo disco. La BIOS dice que son el disco 0 y el disco 1, mientras que FreeBSD los ve como `ad0` y `ad2`.

FreeBSD está en el disco BIOS 1, de tipo `ad` y el número de disco FreeBSD es el 2, así que habrá que decir lo siguiente:

```
1:ad(2,a)kernel
```

Tenga en cuenta que si tiene un disco esclavo en el bus primario esto no es necesario (de hecho es un error hacerlo).

El segundo escenario es el arranque desde un disco SCSI teniendo uno o más discos IDE en el sistema. En este caso el número de disco FreeBSD es más bajo que el número de disco BIOS. Si tiene dos discos IDE además del disco SCSI, el disco SCSI es el disco 2 de BIOS, del tipo `da` y para FreeBSD es el disco número disk number 0, so 0, así que tendrá que usar

```
2:da(0,a)kernel
```

para decirle a FreeBSD que quiere arrancar desde el disco 2 de BIOS, que es el primer disco SCSI del sistema. Si solamente tiene un disco IDE tendrá que poner `1:`.

Una vez que sabe con exactitud los valores que debe usar póngalos exactamente como los ha tecleado en `/boot.config` utilizando el editor de texto que prefiera. Salvo que se le diga lo contrario FreeBSD utilizará el contenido de este fichero decidiráel comportamiento por omisión del prompt de `boot:` prompt.

P: Cuando voy a arrancar desde disco duro por primera vez tras la instalación de FreeBSD el prompt del gestor de arranque muestra `F?` en el menú de arranque y no pasa de ahí.

R: La geometría del disco duro se configuró mal cuando se utilizó el editor de particiones durante la instalación. Ejecute el editor de particiones e introduzca la geometría del disco correctamente. Tendrá que reinstalar FreeBSD completamente dado que habrá cambiado la geometría del disco.

Si no encuentra la manera de saber cuál es la geometría correcta pruebe con esto: Cree una pequeña partición DOS al principio del disco e instale FreeBSD en el espacio de disco inmediatamente contiguo. El programa de instalación verá la partición DOS e intentará deducir la geometría correcta a partir de ahí; esto suele funcionar.

Le mostramos otra sugerencia que, aunque no es recomendable de seguir, se muestra para que sirva de referencia:

> Si lo que está instalando es un servidor o máquina de escritorio dedicado a FreeBSD y no le preocupa que en el futuro cercano tenga que mantener la convivencia con DOS, Linux u otro sistema operativo siempre tiene la opción de utilizar el disco entero (la opción A en el editor de particiones). Es una opción no estándar gracias a la cual FreeBSD ocupa todo el disco, desde el primer sector al último. De este modo nos olvidamos de todos los problemas relacionados con la geometría de disco, pero le obliga a no poder instalar ningún otro sistema operativo aparte de FreeBSD en ese disco.

P: El sistema encuentra mi tarjeta de red ed(4) pero me dan errores de «timeout».

R: Lo más probable es que su tarjeta esté utilizando una IRQ distinta de la que debería estar usando según lo que aparece en `/boot/device.hints` . El controlador ed(4) no utiliza configuración does not use the «soft» por omisión (es decir, valores que se le suministren mediante EZSETUP bajo DOS) pero como casi todo en esta vida hay solución. En este caso, de hecho, hay dos.

Ponga el «jumper» de la tarjeta en una configuración «dura» (es posible que tenga que modificar la configuración del kernel debido a esto) o modifique el valor de la IRQ con el valor `-1` del siguiente modo: by setting the hint `hint.ed.0.irq="-1"` . Así le dirá al kernel que utilice la configuración «suave».

Hay otra posibilidad, que es cuando su tarjeta esté utilizando la IRQ 9, que suele compartir funciones con la IRQ 2, circunstancia que es con frecuencia causa de problemas (sobre todo cuando entra en escena una tarjeta VGA que utiliza justamente la IRQ 2). Le recomendamos encarecidamente que haga todo lo posible para no utilizar ni la IRQ 2 ni la IRQ 9.

2.12. Guía avanzada de instalación

Escrito por Valentino Vaschetto.

This section describes how to install FreeBSD in exceptional cases.

2.12.1. Instalación de FreeBSD en un sistema sin monitor o teclado

Es posible instalar FreeBSD en máquinas que no tengan teclado ni siquiera monitor conectado. De hecho no tienen por qué tener ni salida VGA. ?Cómo es posible?, se preguntará: Haciendolo todo a través de una consola serie. Una consola serie consiste básicamente en usar otro sistema como la pantalla y el teclado de una máquina. Siga las instrucciones

de creación de los disquetes que se detallan en la Sección 2.3.7, "Preparación del medio de arranque".

Tendrá que modificar esos disquetes para que arranquen en la consola serie; siga estas instrucciones:

1. Modificación de los disquetes de arranque para que permitan el arranque en la consola serie

   Si arrancara con los disquetes que acaba de crear FreeBSD arrancaría en el modo normal de instalación. Lo que queremos es que FreeBSD arranque en una consola serie, así que hemos de montar el disquete `kern.flp` en un sistema FreeBSD mediante mount(8).

   ```
   # mount /dev/fd0 /mnt
   ```

   Una vez montado el disquete entre en el directorio `/mnt` :

   ```
   # cd /mnt
   ```

   Aquí es donde enseñaremos al disquete a arrancar en la consola serie. Debe crearse un fichero llamado `boot.config` que contenga lo siguiente: `/boot/loader -h` . Con esto le pasamos cargador de arranque (el «bootloader») el parámetro necesario para que intente arrancar en una consola serie.

   ```
   # echo "/boot/loader -h" > boot.config
   ```

   Una vez configurado correctamente nuestro disquete lo desmontamos con umount(8):

   ```
   # cd /
   # umount /mnt
   ```

   Y podemos extraer el disquete de la unidad.

2. Conexión del cable modem

   Tiene que conectar un cable módem nulo entre dos máquinas. *Un cable serie normal no funcionará*, necesitará un cable módem nulo porque tiene cruzado alguno de los hilos.

3. Arranque de la instalación

   Podemos iniciar la instalación. Introduzca el disquete `kern.flp` en la unidad de disquetes de la máquina en la que quiere hacer la instalación y encienda la máquina.

4. La conexión a su máquina sin teclado ni monitor

   Conéctese usando cu(1):

```
# cu -l /dev/cuaa0
```

?Ya está! Desde este momento ya puede controlar su máquina sin monitor ni teclado desde la sesión cu que acaba de abrir. El sistema le pedirá que introduzca el disco `kern1.flp` y después tendrá que elegir qué tipo de terminal va a utilizar. Elija la consola en color de FreeBSD y buena suerte con la instalación.

2.13. Cómo preparar su propio medio de instalación

Nota

Para evitar la repetición «disco FreeBSD» será en adelante un CDROM o un DVD de FreeBSD que haya comprado o hecho por sus propios medios.

Es posible que tengaque crear su propio medio de instalación de FreeBSD. Puede tratarse de un medio físico como una cinta o una fuente para que sysinstall pueda obtener los ficheros que vaya necesitando, como un sitio FTP local, o bien puede tratarse de una partición MS-DOS®.

Por ejemplo:

- Tiene muchas máquinas conectadas a una red local y un sólo disco FreeBSD. Quiere crear un sitio FTP local con el contenido del disco FreeBSD y que desde él sus máquinas obtengan los ficheros en lugar de tener que conectarse a Internet.
- Tiene un disco FreeBSD y FreeBSD no reconoce su unidad CD/DVD pero MS-DOS®/Windows® sí. Quiere copiar los ficheros de instalación de FreeBSD a una partición DOS en mismo sistema y luego instalar FreeBSD usando esos ficheros.
- El sistema en el que quiere instalar FreeBSD no tiene unidad CD/DVD ni tarjeta de red pero puede conectar un cable paralelo o un cable serie «Laplink» a una computadora que sí que tiene.
- Quiere crear una cinta desde la que se pueda instalar FreeBSD.

2.13.1. Creación de un CDROM de instalación

Con cada «release» el Proyecto FreeBSD libera dos imágenes de CDROM («imágenes ISO»). Dichas imágenes pueden copiarse a CD si dispone de una grabadora y usarse posteriormente para instalar FreeBSD. Si tiene una grabadora de CD y el ancho de banda no le supone un problema la forma más fácil de instalar FreeBSD es esta.

1. Descargar la imagen ISO correcta

   Puede descargar las imágenes ISO de cada «release» desde `ftp://ftp.FreeBSD.org/pub/FreeBSD/ISO-IMAGES-arquitectura/versión` o desde la réplica más próxima. Sustituya *`arquitectura`* y *`versión`* por lo que corresponda.

   Ese directorio contendrá normalmente las siguientes imágenes:

   Tabla 2.4. Nombres y funciones de las imágenes de FreeBSD 6.*X* and 7.*X*

Nombre	Contenido
`versión-RELEASE-arch-bootonly.iso`	Todo lo necesario para arrancar un kernel de FreeBSD y ejecutar la interfaz de instalación. Los ficheros que vayan haciendo falta tendrán que irse descargando por FTP o por algún otro medio.
`versión-RELEASE-arch-disc1.iso`	Todo lo necesario para instalar AFreeBSD y un «live filesystem», que le será muy útil (conjuntamente con la aplicación «Repair» de sysinstall.
`versión-RELEASE-arch-disc2.iso`	Todas las aplicaciones que puedan caber en un disco.
`versión-RELEASE-arch-docs.iso`	La documentaciÓN de FreeBSD.

   Descargue la imagen ISO de arranque (si existe para su plataforma) o la imagen del disco uno. *No* descargue ambas, puesto que el disco uno contiene todo lo que hay en la imagen ISO de arranque.

   Utilice la imagen ISO de arranque si su acceso a Internet es barato. Con ella podrá instalar FreeBSD e instalar todo tipo de software descargándolo como paquetes/ports (consulte el Capítulo 4, *Instalación de aplicaciones: «packages» y ports*) cuando lo precise.

   Utilice la imagen del disco uno si quiere disponer en el propio disco de una selección bastante completa de software.

   Si tiene acceso de alta velocidad a Internet las demás imágenes de disco son útiles pero no esenciales.

2. Escribir los CD

   Si va a grabar los CD en un sistema FreeBSD consulte la Sección 18.6, “Creación y uso de medios ópticos (CD)” (en particular la Sección 18.6.3, “burncd” y la Sección 18.6.4, “cdrecord”).

Las imágenes se hacen en un formato ISO estándar; si va a grabarlas en cualquier otra plataforma no debería tener problemas para hacerlo cualquiera que sea la aplicación que use para grabar CD.

Nota

Si lo que quiere hacer es crear una «release» personalizada de FreeBSD consulte el artículo Release Engineering.

2.13.2. Creación de un sitio FTP local a partir de un disco de instalación de FreeBSD

Los discos de FreeBSD tienen la misma estructura que los sitios FTP. Esa circunstancia le facilitará mucho la tarea de crear un sitio FTP local para uso de otras máquinas de su red durante la instalación de FreeBSD.

1. Monte el CDROM en el directorio `/cdrom` del sistema que va a albergar el sitio FTP.

```
# mount /cdrom
```

2. Cree una cuenta apta para FTP anónimo en `/etc/passwd` editando `/etc/passwd` con vipw(8). Añádale lo siguiente:

```
ftp:*:99:99::0:0:FTP:/cdrom:/nonexistent
```

3. Compruebe que el servicio FTP está activado en `/etc/inetd.conf` .

Ahora cualquier máquina capaz de conectarse a su sistema a través de una red puede elegir como medio de instalación «FTP» y escribir **`ftp://su máquina`** tras seleccionar «Other» en el menú de sitios FTP.

Aviso

Acabamos de exponer una buena solución para usar en un sistema de su propia red y que además está protegido por un cortafuegos. Si ofrece servicios FTP a máquinas de Internet (y no de su red local) expone su sistema a caer bajo la atención de «crackers» y otras variedades de indeseable. Le recomendamos encarecidamente que siga a rajatabla políticas sensatas de seguridad.

2.13.3. Creación de disquetes de instalación

Si quiere instalar FreeBSD desde disquetes (cosa que, de antemano, le sugerimos que *no* haga) bien sea porque FreeBSD no soporta cierto componente necesario de su sistema o sencillamente porque insiste en querer hacer las cosas de la manera más difícil, tendrá que tener a mano unos cuantos disquetes.

Como mínimo necesitará tantos disquetes de 1.44 MB o 1.2 MB como hagan falta para dar cabida a todos los ficheros que hay en `bin`. Recuerde la entrada «binary distribution» (distribución binaria) durante la instalación. Si está generando los disquetes en DOS *debe formatearlos* con `FORMAT` de MS-DOS®. Si está usando Windows® puede usar el Explorador para formatear disquetes: botón derecho del ratón sobre la unidad `A:` y elija «Formatear».

No confíe en el preformateo de fábrica de los disquetes. Formatéelos usted de nuevo, sólo para asegurarse. La mayoría de los problemas de los que se ha informado se han debido a formateos defectuosos, téngalo en cuenta.

Si puede crear los disquetes en una máquina FreeBSD un formateo sigue sin ser una mala idea, pero no necesitará crear un sistema de ficheros DOS en cada floppy. Use `disklabel` y `newfs` para crear un sistema de ficheros UFS en los disquetes tal y como se muestra aquí (en un floppy de 1.44 MB y 3.5") :

```
# fdformat -f 1440 fd0.1440
# disklabel -w -r fd0.1440 floppy3
# newfs -t 2 -u 18 -l 1 -i 65536 /dev/fd0
```

Nota

Tendrá que usar `fd0.1200` y `floppy5` si usa discos de 5.25" y 1.2 MB.

Ahora puede montarlos y escribir en ellos igual que en cualquier otro sistema de ficheros del sistema.

Tras el formateo de los disquetes pasamos a llenarlos. Los ficheros necesarios para crear la distribución se dividen en partes de tamaño regular de modo que cinco de ellas entren en otros tantos disquetes de 1.44 MB. Empaquete tantos ficheros como pueda en cada uno hasta que tenga todas las distribuciones que necesita listas para pasar a los disquetes. Cada distribución debería ir en un subdirectorio del floppy, por ejemplo `a:\bin\bin.aa` , `a:\bin\bin.ab` , etc.

Una vez en el menú de selección de medio de instalación elija Floppy y siga las instrucciones.

2.13.4. Instalación desde una partición MS-DOS®

Copie los ficheros de la distribución en un directorio llamado `freebsd` en el directorio raíz de la partición, por ejemplo `c:\freebsd`. La estructura de directorios del sitio FTP o el CDROM debe reproducirse parcialmente en este directorio, así que le sugerimos que use `xcopy` si está copiando los datos desde un CD. En el siguiente ejemplo vamos a preparar una instalación mínima de FreeBSD:

```
C:\> md c:\freebsd
C:\> xcopy e:\bin c:\freebsd\bin\ /s
C:\> xcopy e:\manpages c:\freebsd\manpages\ /s
```

Se ha asumido que `C:` es donde tiene espacio libre y `E:` es donde tiene montada su unidad CDROM.

Si no tiene unidad CDROM puede descargar la distribución desde ftp.FreeBSD.org. Cada distribución dispone de su propio directorio; por ejemplo, la distribución *base* está en el directorio10.0/base/.

La única distribución *imprescindible* es `BIN`. Guarde esta y todas las distribuciones que quiera instalar desde una partición MS-DOS® bajo `c:\freebsd`.

2.13.5. Creación de una cinta de instalación

La instalación desde cinta es probablemente el método más fácil, casi tanto como una instalación desde una réplica FTP o desde un CDROM. Lo único que el programa de instalación necesita es que los ficheros estén empaquetados con tar y en una cinta. Veamos un ejemplo:

```
# cd /freebsd/distdir
# tar cvf /dev/rwt0 dist1 ... dist2
```

Cuando esté preparando la instalación tenga la precaución de dejar sitio suficiente en algún directorio temporal (accesible en ese momento) para dejar *todos los ficheros* que hubiera en la cinta de instalación. A causa del acceso secuencial propio del diseño de las cintas este método de instalación requiere un poco de almacenamiento temporal.

Nota

Recuerde que antes de arrancar desde el disquete de arranque la cinta *debe estar en la unidad*. La prueba de hardware de la instalación no encontraría la cinta.

2.13.6. Before Installing over a Network

Hay tres tipos de instalación a través de una red disponibles en FreeBSD: Puerto serie (SLIP o PPP), puerto paralelo (PLIP (cable «laplink»)) y Ethernet.

El adaptador Ethernet es, debido a su velocidad de transferencia, la mejor elección. FreeBSD soporta la mayoría de los adaptadores Ethernet. Tiene una lista de dispositivos soportados en las notas de hardware de cada «release» de FreeBSD. Si disponde alguno de los adaptadores Ethernet PCMCIA soportados por FreeBSD recuerde que debe estar presente en el sistema *antes* de que el sistema arranque. Por desgracia FreeBSD no soporta aún la inserción en caliente de tarjetas PCMCIA durante la instalación.

Tendrá que saber también la dirección IP de la red, la máscara de red correcta para la clase de su red y el nombre de su máquina. Si va a instalar mediante una conexión PPP y no tiene IP estática no se preocupe, su ISP puede asignarle una IP temporal dinámicamente. Su administrador de sistemas seguramente puede ayudarle con la configuración de su red. Si va a intentar conectarse con otras máquinas mediante sus nombres en lugar de sus direcciones IP necesitará los datos de un servidor de nombres y es posible que necesite también la dirección de un «gateway» (si usa PPP vale con la dirección IP de su proveedor). Si va a instalar por FTP a través de un proxy HTTP necesitará la dirección del proxy. Si no sabe qué pueden significar todas (o la mayoría) de estas preguntas hable con su administrador de sistemas o con su ISP *antes* de intentar instalar FreeBSD de esta forma.

2.13.6.1. Antes de instalar via NFS

La instalación por NFS es bastante sencilla. Sólo tendrá que copiar los ficheros de la distribución a un servidor NFS, seleccionar NFS como medio de instalación y apuntar a los contenidos a donde los haya copiado.

si el servidor sólo soporta «puertos privilegiados» (que suele ser la configuración por omisión en estaciones de trabajo Sun) tendrá que seleccionar la opción `NFS Secure` en el menú «Options» antes de comenzar con la instalación.

Si usa tarjeta Ethernet de mala calidad y está experimentando velocidades de transferencia muy bajas puede probar con el parámetro `NFS Slow`.

Para que la instalación via NFS funcione el servidor debe permitir el montaje de subdirectorios. Por ejemplo, si su distribución FreeBSD 10.0 está en `ziggy:/usr/archive/stuff/FreeBSD` el servidor `ziggy` tendrá que permitir que pueda montarse `/usr/archive/stuff/FreeBSD` , no simplemente como `/usr` o `/usr/archive/stuff` .

En el fichero `/etc/exports` de FreeBSD eso puede controlarse mediante la opción `-alldirs`. Puede que otros servidores NFS requieran de parámetros diferentes. Si ve mensajes de permission denied seguramente no ha activado esto correctamente.

Capítulo 3. Conceptos básicos de Unix

Reescrito por Chris Shumway.

3.1. Sinopsis

El siguiente capítulo comprende la funcionalidad y órdenes básicas del sistema operativo FreeBSD. Gran parte de este material es relevante para cualquier sistema operativo tipo UNIX®. Puede saltarse este capítulo si considera que ya conoce el funcionamiento de UNIX®. Si no tiene experiencia previa con FreeBSD debería leer este capítulo con mucha atención.

Después de leer este capítulo, usted sabrá:

- Cómo usar las «consolas virtuales» de FreeBSD.
- Cómo funcionan los permisos de fichero UNIX® en relación con las banderas de fichero en FreeBSD.
- La disposición de sistemas de ficheros por omisión en FreeBSD.
- La organización de disco de FreeBSD.
- Cómo montar y desmontar sistemas de ficheros.
- Qué son los procesos, dæmons y señales.
- Qué es una shell, cómo modificar sus variables de entorno por omisión.
- Cómo utilizar editores de texto básicos.
- Qué son los dispositivos y nodos de dispositivos.
- Qué formato binario se usa en FreeBSD.
- Cómo buscar información en las páginas de manual.

3.2. Consolas virtuales y terminales

FreeBSD puede utilizarse de muchas maneras. Una de ellas es tecleando órdenes en una terminal de texto. De este modo, mucha de la flexibilidad y poder de un sistema operativo UNIX® está inmediatamente en sus manos cuando usa FreeBSD. Esta sección describe qué son «terminales» y «consolas» y cómo puede usarlas en FreeBSD.

3.2.1. La consola

Si no ha configurado FreeBSD para ejecutar automáticamente un entorno gráfico en el arranque, el sistema le presentará un «prompt» de entrada después del arranque, inmediatamente después de que los «scripts» de inicio terminen de ejecutarse. Verá algo similar a esto:

```
Additional ABI support:.
Local package initialization:.
Additional TCP options:.

Fri Sep 20 13:01:06 EEST 2002

FreeBSD/i386 (pc3.ejemplo.org) (ttyv0)

login:
```

Los mensajes pueden ser un poco diferentes en su sistema, pero verá algo similar. Las últimas dos líneas son las que nos interesan por el momento. La penúltima línea dice:

```
FreeBSD/i386 (pc3.ejemplo.org) (ttyv0)
```

Esta línea contiene información acerca del sistema que acaba de arrancar. Esta usted ante una consola «FreeBSD» que se ejecuta en un procesador Intel o compatible de la arquitectura x86[1]. El nombre de esta máquina (todas las máquinas UNIX® tiene un nombre) es `pc3.ejemplo.org` , y usted está ahora ante su consola de sistema (la terminal `ttyv0`).

Para acabar, la última línea siempre es:

```
login:
```

Este es el lugar donde se usted tecleará su «nombre de usuario» para entrar en FreeBSD. La siguiente sección describe cómo hacerlo.

3.2.2. La entrada a FreeBSD

FreeBSD es un sistema multiusuario multiprocesador. Esta es la descripción formal que se suele dar a un sistema que puede ser utilizado por muchas personas diferentes, que simultáneamente ejecutan muchos programas en un sola máquina.

Cada sistema multiusuario necesita algún modo de distinguir a un «usuario» del resto. En FreeBSD (y en todos los sistemas operativos de tipo UNIX®) esto se logra requiriendo que cada usuario «acceda» al sistema antes de poder ejecutar programas. Cada usuario tiene un nombre único (el «nombre de usuario») y una clave secreta, personal (la «contraseña»). FreeBSD preguntará por ambos antes de permitirle a un usuario ejecutar cualquier programa.

[1] Esto es lo que `i386` significa. Tenga en cuenta que incluso si no está ejecutando FreeBSD en una CPU Intel 386, éste va a ser `i386`. No es el tipo de su procesador, sino la «arquitectura» la que se muestra aquí.

Justo después de que FreeBSD arranque y termine de ejecutar sus «scripts» de inicio [2], le presentará un «prompt» y solicitará un nombre válido de usuario:

```
login:
```

En este ejemplo vamos a asumir que su nombre de usuario es `john`. Teclée `john` en el «prompt» y pulse Intro. Debería presentársele un «prompt» donde introducir una «contraseña»:

```
login: john
Password:
```

Teclée ahora la contraseña de `john` y pulse Enter. La contraseña *no se muestra en pantalla*, pero no debe preocuparse por ello. Esto se hace así por motivos de seguridad.

Si ha tecleado su contraseña correctamente ya está usted en un sistema FreeBSD, listo para probar todas las órdenes disponibles.

Verá el MOTD (mensaje del día) seguido por un «prompt» (un caracter `#`, `$` o `%`). Esto confirma que ha validado con éxito su usuario en FreeBSD.

3.2.3. Consolas múltiples

Ejecutar órdenes UNIX® en una consola está bien, pero FreeBSD puede ejecutar muchos programas a la vez. Tener una consola donde se pueden teclear órdenes puede ser un desperdicio cuando un sistema operativo como FreeBSD puede ejecutar docenas de programas al mismo tiempo. Aquí es donde las «consolas virtuales» muestran su potencial.

FreeBSD puede configurarse para presentarle diferentes consolas virtuales. Puede cambiar de una de ellas a cualquier otra consola virtual pulsando un par de teclas en su teclado. Cada consola tiene su propio canal de salida, y FreeBSD se ocupa de redirigir correctamente la entrada del teclado y la salida al monitor cuando cambia de una consola virtual a la siguiente.

Se han reservado ciertas combinaciones especiales de teclas para pasar de unas consolas virtuales a otras en FreeBSD [3]. Puede utilizar Alt+F1, Alt+F2 y así sucesivamente hasta Alt+F8 para cambiar a una consola virtual diferente en FreeBSD.

[2] Los «scripts» de inicio son programas que FreeBSD ejecuta automáticamente cuando arranca. Su función principal es preparar todo lo necesario para que todo lo demás se ejecute, e iniciar cualquier servicio que tenga configurado para ejecutarse en segundo plano algo útiles.

[3] Tiene una descripción bastante técnica y correcta de todos los detalles de la consola FreeBSD y los controladores de teclado pueden encontrarse en las páginas de manual de syscons(4), atkbd(4), vidcontrol(1) y kbdcontrol(1). No nos extenderemos en los detalles aquí, pero el lector interesado siempre puede consultar dichas páginas de manual si quiere una explicación más detallada y profunda de cómo funcionan estas cosas.

Mientras está cambiando de una consola a la siguiente, FreeBSD se ocupa de guardar y restaurar la salida de pantalla. El resultado es la «ilusión» de tener varias pantallas y teclados «virtuales» que puede utilizar para teclear órdenes para que FreeBSD los ejecute. El programa que usted lanza en una consola virtual no deja de ejecutarse cuando la consola no está visible. Continúan ejecutándose cuando se cambia a una consola virtual diferente.

3.2.4. El fichero /etc/ttys

La configuración por omisión de FreeBSD iniciará con ocho consolas virtuales. No es una configuración estática por hardware, así que puede personalizar fácilmente su sistema para arrancar con más o menos consolas virtuales. El número y propiedades de las consolas virtuales están detallados en /etc/ttys .

En /etc/ttys es donde se configuran las consolas virtuales de FreeBSD. Cada línea no comentada de este fichero (líneas que no comienzan con un caracter #) contiene propiedades para una sola terminal o consola virtual. La versión por omisión de este fichero en FreeBSD configura nueve consolas virtuales y habilita ocho de ellas. Son las líneas que comienzan con ttyv:

```
# name  getty                           type    status          ↺
comments
#
ttyv0   "/usr/libexec/getty Pc"         cons25  on  secure
# Virtual terminals
ttyv1   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv2   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv3   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv4   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv5   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv6   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv7   "/usr/libexec/getty Pc"         cons25  on  secure
ttyv8   "/usr/X11R6/bin/xdm -nodaemon"  xterm   off secure
```

Consulte ttys(5) si quiere una descripción detallada de cada columna en este fichero y todas las opciones que puede usar para configurar las consolas virtuales.

3.2.5. Consola en modo monousuario

En la Sección 12.6.2, “Modo monousuario” encontrará una descripción detallada de lo que es «modo monousuario». No importa que sólo exista una consola cuando ejecuta FreeBSD en modo monousuario. No hay otra consola virtual disponible. Las configuraciones de la consola en modo monousuario se pueden encontrar también en /etc/ttys . Busque la línea que comienza por console:

```
# name  getty                           type    status          ↺
comments
#
# Si la consola está marcada como "insecure", entonces init
```

```
# le pedirá la contraseña de root al entrar a modo monousuario.
console none                            unknown off secure
```

Nota

Tal y como indican los comentarios por encima de la línea `console`, puede editar esta línea y cambiar `secure` por `insecure`. Si lo hace, cuando FreeBSD arranque en modo monousuario el sistema le pedirá la contraseña de `root`.

Tenga cuidado si cambia esto a `insecure`. Si olvida la contraseña de `root` arrancar en modo monousuario será con seguridad más complicado. Sigue siendo posible, pero será un poco difícil para alguien que no esté familiarizado con le proceso de arranque de FreeBSD y los programas involucrados en dicho proceso.

3.2.6. Cambio del modo de video de la consola

La consola FreeBSD por omisión tiene un modo de video que puede ajustarse a 1024x768, 1280x1024 o cualquier otra resolución que admita su chip gráfico y su monitor. Si quiere utilizar uno diferente tendrá que recompilar su kernel con estas dos opciones añadidas:

```
options VESA
options SC_PIXEL_MODE
```

Una vez recompilado el kernel con esas dos opciones en él determine qué modos de video admite su hardware; para ello use vidcontrol(1). Con lo siguiente le mostrará una lista de modos de video soportados:

```
# vidcontrol -i mode
```

La salida de esta orden es una lista de los modos de que admite su tarjeta. Para elegir uno de ellos tendrá que ejecutar vidcontrol(1) en una consola como `root`:

```
# vidcontrol MODE_279
```

Si el modo de video que ha elegido le parece adecuado puede configurarlo de forma permanente haciendo que funcione desde el momento del arranque; para ello debe editar `/etc/rc.conf` file:

```
allscreens_flags="MODE_279"
```

3.3. Permisos

FreeBSD, cuya raíz histórica es el UNIX® BSD, se fundamenta en varios conceptos clave de UNIX. El primero y más importante es que FreeBSD es un sistema operativo multi-usuario. El sistema puede gestionar múltiples usuarios trabajando simultáneamente y en tareas que no guarden relación entre sí. El sistema se encarga de compartir y administrar peticiones de dispositivos de hardware, periféricos, memoria y tiempo de CPU de manera equitativa para cada usuario.

Debido a que el sistema es capaz de soportar múltiples usuarios, todo lo que el sistema administra tiene un conjunto de permisos que usa para decidir quién puede leer, escribir y ejecutar un recurso. Estos permisos se guardan como octetos divididos en tres partes: una para el propietario del fichero, otra para el grupo al que el fichero pertenece, y otra para todos los demás grupos y usuarios. Veamos una representación numérica de esto:

Valor	Permiso	Listado de directorio
0	No leer, no escribir, no ejecutar	`---`
1	No leer, no escribir, ejecutar	`--x`
2	No leer, escribir, no ejecutar	`-w-`
3	No leer, escribir, ejecutar	`-wx`
4	Leer, no escribir, no ejecutar	`r--`
5	Leer, no escribir, ejecutar	`r-x`
6	Leer, escribir, no ejecutar	`rw-`
7	Leer, escribir, ejecutar	`rwx`

Puede utilizar el parámetro de línea de órdenes `-l` de ls(1) para ver un listado largo que incluya una columna con información acerca de los permisos de fichero para el propietario, grupo y los demás. Por ejemplo, un `ls -l` en un directorio puede mostrar algo como esto:

```
% ls -l
total 530
-rw-r--r--  1 root  wheel     512 Sep  5 12:31 mifichero
-rw-r--r--  1 root  wheel     512 Sep  5 12:31 otrofichero
-rw-r--r--  1 root  wheel    7680 Sep  5 12:31 email.txt
...
```

Aquí está como se divide la primera columna de `ls -l`:

```
-rw-r--r--
```

El primer caracter (más a la izquierda) indica si este fichero es un fichero regular, un directorio, un dispositivo especial de caracter, un socket o cualquier otro dispositivo especial pseudo-ficheroa . En este caso, el `-` un fichero regular. Los siguientes tres caracteres, `rw-` en este ejemplo, dan los permisos para el propietario del fichero. Los siguientes tres

caracteres, `r--`, dan los permisos para el grupo al que el fichero pertenece. Los últimos tres caracteres, `r--`, dan los permisos para el resto del mundo. Un guión indica que el permiso está desactivado. En el caso de este fichero, los permisos están asignados de tal manera que el propietario puede leer y escribir en el fichero, el grupo puede leer el fichero, y el resto del mundo sólo puede leer el fichero. De acuerdo con la tabla de arriba, los permisos para este fichero serían `644`, donde cada dígito representa las tres partes de los permisos del fichero.

Todo ésto está muy bien, pero ?cómo controla el sistema los permisos de los dispositivos? FreeBSD en realidad trata la mayoría de los dispositivos hardware como un fichero que los programas pueden abrir, leer y en los que pueden escribir datos como si de cualquier otro fichero se tratara. Estos ficheros especiales de dispositivo se encuentran en el directorio `/dev`.

Los directorios también son tratados como ficheros. Tienen permisos de lectura, escritura y ejecución. El bit de ejecución en un directorio tiene un significado ligeramente distinto que para los ficheros. Cuando un directorio está marcado como ejecutable significa que se puede mirar dentro, se puede hacer un «cd» (cambiar directorio) a él. Esto también significa que dentro del directorio es posible acceder a ficheros cuyos nombres son conocidos (sujeto, claro está, a los permisos de los ficheros mismos).

En particular, para poder realizar un listado de directorio, el permiso de lectura debe ser activado en el directorio, mientras que para borrar un fichero del que se conoce el nombre es necesario tener permisos de escritura *y* ejecución en el directorio que contiene el fichero.

Existen más permisos, pero se usan principalmente en circunstancias especiales como los binarios ejecutables de tipo setuid y los los directorios de tipo «sticky». Si desea más información acerca de los permisos de ficheros y cómo establecerlos, consulte chmod(1).

3.3.1. Permisos simbólicos

Escrito por Tom Rhodes.

Los permisos simbólicos, también conocidos como expresiones simbólicas, utilizan caracteres en lugar de valores octales para asignar permisos a ficheros o directorios. Las expresiones simbólicas utilizan la sintaxis de (quién) (acción) (permisos) mediante los siguientes valores:

Opción	Letra	Representa
(quién)	u	Usuario
(quién)	g	Grupo propietario
(quién)	o	Otro
(quién)	a	Todos («todo el mundo»)

Opción	Letra	Representa
(acción)	+	Añadir permisos
(acción)	-	Quitar permisos
(acción)	=	Activar permisos explícitamente
(permisos)	r	Lectura
(permisos)	w	Escritura
(permisos)	x	Ejecución
(permisos)	t	Bit Sticky («pegajoso»)
(permisos)	s	Activar UID o GID

Estos valores se aplican con chmod(1) de la misma manera que los anteriores, pero con letras. Por ejemplo, podría usar la siguiente orden para bloquear a otros usuarios el acceso a *FICHERO*:

```
% chmod go= FICHERO
```

Puede usarse una lista separada por comas cuando se quiera aplicar más de un conjunto de cambios a un fichero. Por ejemplo la siguiente orden eliminará los permisos de escritura de grupo y «mundo» a *FICHERO*, y añade permisos de ejecución para todos:

```
% chmod go-w,a+x FILE
```

3.3.2. Banderas de fichero en FreeBSD

Escrito por Tom Rhodes.

Además de los permisos de fichero previamente expuestos, FreeBSD permite el uso de «banderas de fichero». Estas banderas añaden un nivel de seguridad y control adicional a los ficheros, pero no a los directorios.

Las banderas de fichero añaden un nivel adicional de control sobre los ficheros ayudando a asegurar que en algunos casos ni siquiera `root` pueda eliminar o alterar ficheros.

Las banderas de fichero se modifican mediante chflags(1), gracias a una interfaz muy sencilla. Por ejemplo, para habilitar la bandera imborrable de sistema en `fichero1`, escriba lo siguiente:

```
# chflags sunlink fichero1
```

Y para dehabilitar la bandera imborrable de sistema, simplemente escriba la orden previa con «no» antes de `sunlink`. Observe:

```
# chflags nosunlink fichero1
```

Para ver las banderas de este fichero, utilice ls(1) con las opciones `-lo`:

```
# ls -lo fichero1
```

La salida debería ser como esta:

```
-rw-r--r--  1 trhodes  trhodes  sunlnk 0 Mar  1 05:54 fichero1
```

Varias banderas solo pueden ser añadidas o retiradas de ficheros por el usuario `root`. En otros casos, el propietario del fichero puede activar estas banderas. Se recomienda que para más información la persona encargada de la administración del sistema consulte las páginas de manual chflags(1) y chflags(2).

3.3.3. Los permisos `setuid`, `setgid` y `sticky`

Escrito por Tom Rhodes.

Además de los permisos que se han explicado hay más, hay tres tipos más que todos los administradores deberían conocer. Son los permisos `setuid`, `setgid` y `sticky`.

Estos permisos juegan un papel clave en ciertas operaciones UNIX® puesto que facilitan funcionalidades que no se suelen permitir a los usuarios normales. Para comprenderlas totalmente hay que comprender la diferencia entre el ID real del usuario y el ID efectivo.

El ID del usuario real es el UID que arranca (y el propietario) del proceso. El UID efectivo es el ID bajo el que se ejecuta el proceso. Veamos un ejemplo; el programa passwd(1) se ejecuta con el ID real del usuario puesto que el usuario está cambiando su contraseña. Pero para poder manipular la base de datos de contraseñas debe ejecutarse con el ID efectivo del usuario `root`. De este modo es posible que los usuarios cambien su contraseña sin llegar a ver un error de Permission Denied (permiso denegado).

Nota

La opción `nosuid` de mount(8) hace que estos binarios den errores silenciosos, es decir, fallarán pero el usuario no recibirá ningún mensaje de error por ello. Esta opción no funciona siempre, pues, según la página man de mount(8), un «wrapper» `nosuid` puede sortear esta limitación.

El permiso setuid puede asignarse colocando un número cuatro (4) antes de los permisos. Se ve mejor con un ejemplo:

```
# chmod 4755 ejemplosuid.sh
```

Los permisos de `ejemplosuid.sh` deberían ser así:

```
-rwsr-xr-x   1 trhodes  trhodes    63 Aug 29 06:36 ejemplosuid.sh
```

Fíjese atentamente en la `s` que ha aparecido en los permisos del fichero, en la parte de los permisos del propietario; esa `s` está en donde estaría el bit de ejecución. Gracias a esto el funcionamiento de aplicaciones que necesitan permisos elevados, como `passwd`, pueden funcionar.

Si quiere ver un ejemplo con sus propios ojos abra dos terminales. En una arranque un proceso (ejecute) `passwd` con un usuario normal. Mientras la aplicación espera a que le de una nueva contraseña busque la información de usuario del proceso `passwd` en la tabla de procesos.

En la terminal A:

```
Changing local password for trhodes
Old Password:
```

En la terminal B:

```
# ps aux | grep passwd
```

```
trhodes  5232  0.0  0.2  3420  1608   0  R+    2:10AM   0:00.00 ↺
grep passwd
root     5211  0.0  0.2  3620  1724   2  I+    2:09AM   0:00.01 ↺
passwd
```

Tal y como se ha dicho, un usuario normal puede ejecutar `passwd`, pero en realidad está utilizando el UID efectivo de `root`.

El permiso `setgid` actúa del mismo modo que el `setuid`, pero afecta a los permisos del grupo. Cuando una aplicación funciona con esta configuración lo hace con los permisos del grupo al que pertenece el fichero, no los del usuario que ha arrancado el proceso.

Si quiere utilizar el permiso `setgid` debe situar un núnmero dos (2) al principio de los permisos que vaya a asignar mediante `chmod`.

```
# chmod 2755 ejemplosuid.sh
```

La nueva configuración tiene un aspecto muy similar a la que tenía antes, pero observe que la `s` de antes está ahora en el campo de los permisos de grupo:

```
-rwxr-sr-x   1 trhodes  trhodes    44 Aug 31 01:49 ejemplosuid.sh
```

Nota

En ambos ejemplos, incluso si el «script» en cuestión es ejecutable, no se va a ejecutar con un EUID distinto o un ID efectivo de usuario

porque los «scripts» de shell no pueden acceder a la llama del sistema setuid(2).

Los dos permisos que acabamos de mostrar los bits de permisos (`setuid` y `setgid`) pueden reducir el nivel de seguridad haciendo que se escalen los permisos. Pero hay un tercer bit especial de permisos que puede ser de mucha ayuda para reforzar la seguridad del sistema: el `sticky bit`.

El `sticky bit` (que podríamos traducir como «bit pegajoso») aplicado a un directorio hace que solamente el propietario de un fichero pueda borrarlo. Esto evita el borrado de ficheros ajenos en directorios públicos como `/tmp`. Si quiere usarlo coloque un uno (1) antes de los permisos. Veamos un ejemplo:

```
# chmod 1777 /tmp
```

Para ver el ;`sticky bit` en acción usamos `ls`:

```
# ls -al / | grep tmp
```

```
drwxrwxrwt  10 root  wheel         512 Aug 31 01:49 tmp
```

El `sticky bit` es la letra `t` al final de los permisos.

3.4. Estructura de directorios

La jerarquía del sistema de ficheros de FreeBSD es fundamental para obtener una compresión completa del sistema. El concepto más importante a entender es el del directorio raíz, «/». Este directorio es el primero en ser montado en el arranque y contiene el sistema básico necesario para preparar el sistema operativo para su funcionamiento en modo multiusuario. El directorio raíz también contiene puntos de montaje para cualquier otro sistema de ficheros que se pretenda montar.

Un punto de montaje es un directorio del que se pueden colgar sistemas de ficheros adicionales en un sistema padre (que suele ser el directorio raíz). Esto se explica con detalle en la Sección 3.5, “Organización de disco”. Los puntos de montaje estándar son, por ejemplo, `/usr`, `/var`, `/tmp`, `/mnt` y `/cdrom`. Estos directorios suelen corresponderse con entradas en `/etc/fstab`. `/etc/fstab` es una tabla que sirve de referencia al sistema y contiene los diferentes sistemas de ficheros y sus respectivos puntos de montaje. La mayoría de los sistemas de ficheros en `/etc/fstab` se montan automáticamente en el arranque gracias al «script» rc(8), a menos que contengan la opción `noauto`. Si quiere más información consulte la Sección 3.6.1, “El fichero `fstab`”.

Veremos ahora una descripción de los directorios más comunes. Si desea información más completa consulte hier(7).

Directorio	Descripción
`/`	Directorio raíz del sistema de ficheros.
`/bin/`	Utilidades de usuario fundamentales tanto para el ambiente monousuario como para el multiusuario.
`/boot/`	Programas y ficheros de configuración necesarios durante el arranque del sistema operativo.
`/boot/defaults/`	Ficheros de configuración por omisión del arranque; ver loader.conf(5).
`/dev/`	Nodos de dispositivo; ver intro(4).
`/etc/`	Ficheros de configuración y «scripts» del sistema.
`/etc/defaults/`	Ficheros de configuración por omisión del sistema; ver rc(8).
`/etc/mail/`	Ficheros de configuración para agentes de transporte de correo como sendmail(8).
`/etc/namedb/`	Ficheros de configuración de `named`; ver named(8).
`/etc/periodic/`	«Scripts» que se ejecutan diariamente, semanalmente y mensualmente mediante cron(8); ver periodic(8).
`/etc/ppp/`	Ficheros de configuración de `ppp`; ver ppp(8).
`/mnt/`	Directorio vacío utilizado de forma habitual por administradores de sistemas como punto de montaje temporal.
`/proc/`	Sistema de ficheros de procesos; ver procfs(5), mount_procfs(8).
`/rescue/`	Programas enlazados estáticamente para restauraciones de emergencia; ver rescue(8).
`/root/`	Directorio local para la cuenta `root`.
`/sbin/`	Programas del sistema y utilidades fundamentales de administración para ambientes monousuario y multiusuario.
`/tmp/`	Ficheros temporales. El contenido de `/tmp` *NO* suelen conservarse después de un reini-

Directorio	Descripción
	cio del sistema. Los sistemas de ficheros basados en memoria suelen montarse en `/tmp` Puede automatizarse mediante variables de tmpmfs en rc.conf(5) (o con una entrada en `/etc/fstab` ; ver mdmfs(8), o para FreeBSD 4.X, mfs(8)).
`/usr/`	La mayoría de utilidades y aplicaciones de usuario.
`/usr/bin/`	Aplicaciones comunes, herramientas de programación y otras aplicaciones.
`/usr/include/`	Ficheros «include» estándar de C.
`/usr/lib/`	Bibliotecas.
`/usr/libdata/`	Ficheros de datos con diversas funciones.
`/usr/libexec/`	Dæmons del sistema y utilidades del sistema (ejecutados por otros programas).
`/usr/local/`	Ejecutables locales, bibliotecas, etc. también se usa como destino por omisión de la infraestructura de ports de FreeBSD. Dentro de `/usr/local` debe seguirse el esquema general definido en hier(7) para `/usr`. Las excepciones son el directorio man, que está directamente bajo `/usr/local` en lugar de debajo de `/usr/local/share` , y la documentación de los ports está en `share/doc/port`.
`/usr/obj/`	Arbol destino dependiente de arquitectura fruto de la compilación del árbol `/usr/src` .
`/usr/ports`	La colección de Ports de FreeBSD (opcional).
`/usr/sbin/`	Dæmons del sistema y utilidades del sistema (ejecutados por usuarios del sistema).
`/usr/share/`	Ficheros independientes de arquitectura.
`/usr/src/`	Ficheros fuente BSD y/o local.
`/usr/X11R6/`	Ejecutables de la distribución X11R6, bibliotecas, etc (opcional).
`/var/`	Ficheros multipropósito de log, temporales, en tránsito y de «spool». En ocasiones se

Directorio	Descripción
	monta en `/var` un sistema de ficheros basado en memoria.
`/var/log/`	Diversos ficheros de log del sistema.
`/var/mail/`	Ficheros de buzones de correo de usuarios.
`/var/spool/`	Directorios diversos del sistema de spool de impresora y correo.
`/var/tmp/`	Ficheros temporales. Suelen conservarse tras el Estos ficheros suelen conservarse tras el reinicio del sistema, a menos que `/var` sea un sistema de ficheros basado en memoria.
`/var/yp`	Mapas NIS.

3.5. Organización de disco

La unidad más pequeña que FreeBSD utiliza para ubicar ficheros es el nombre de fichero. Los nombres de fichero son sensibles a las mayúsculas, lo que significa que `readme.txt` y `README.TXT` son dos ficheros distintos. FreeBSD no utiliza la extensión (`.txt`) de un fichero para determinar si es un programa, o un documento o alguna otra forma de datos.

Los ficheros se almacenan en directorios. Un directorio puede estar vacío, o puede contener cientos de ficheros. Un directorio también puede contener otros directorios, permitiéndole contruir una jerarquía de directorios dentro de otro. Esto hace mucho más fácil la organización de sus datos.

Para referirse a ficheros o directorios se usa el nombre de archivo o de directorio, seguido por una barra, `/`, seguido por cualquier otro nombre de directorio que sea necesario. Si tiene un directorio `tal`, el cual contiene el directorio `cual`, el cual contiene el fichero `readme.txt` , entonces el nombre completo o *ruta* al fichero es `tal/cual/readme.txt` .

Los directorios y ficheros se almacenan en un sistema de ficheros. Cada sistema de ficheros contiene un sólo directorio en el nivel más elevado, que es el *directorio raíz* de ese sistema de ficheros. Este directorio raíz puede contener otros directorios.

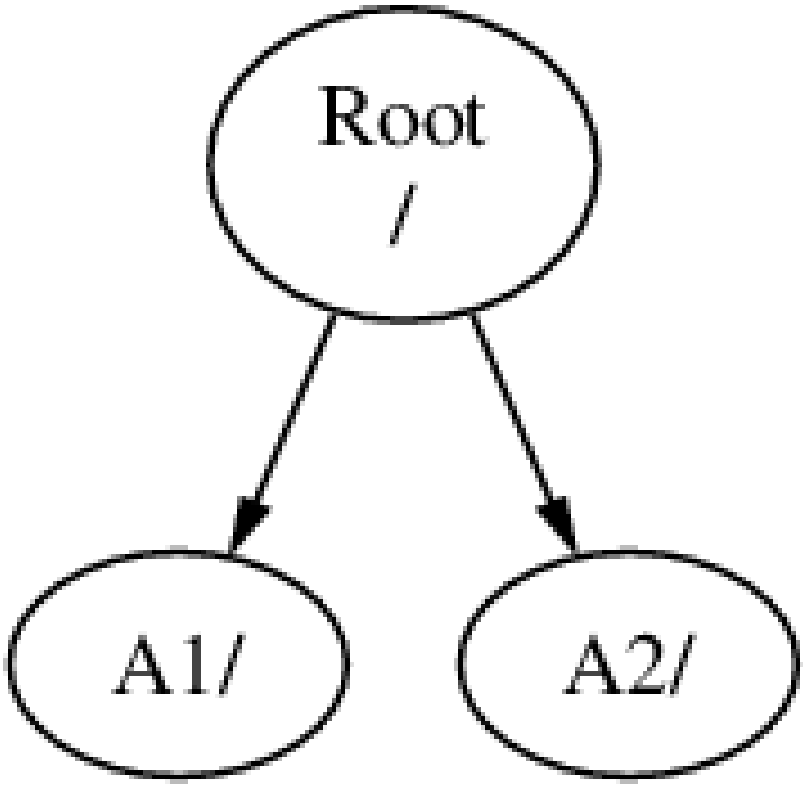
Lo visto hasta ahora probablemente sea similar a cualquier otro sistema operativo que pueda haber utilizado, pero hay unas cuantas diferencias; por ejemplo, MS-DOS® utiliza `\` para separar nombres de fichero y directorio, mientras que Mac OS® usa `:`.

FreeBSD no utiliza letras de unidades, u otro nombre de unidad en la ruta. Por tanto, no podrá escribir `c:/tal/cual/readme.txt` en FreeBSD.

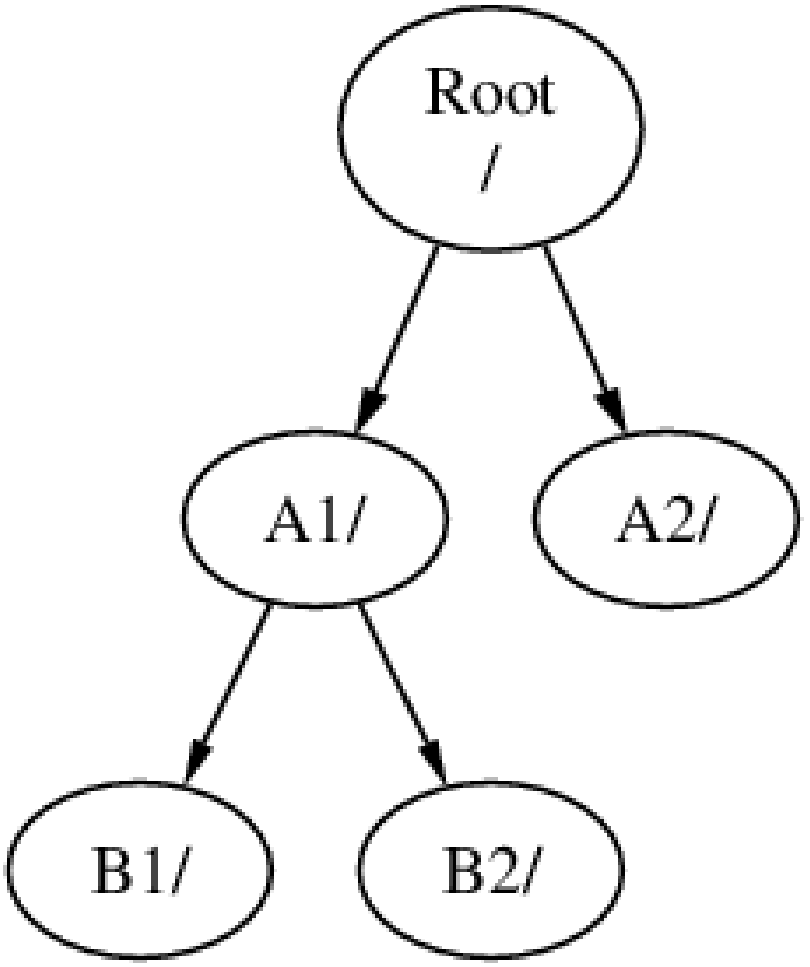
En FreeBSD, en cambio, un sistema de ficheros recibe el nombre de *sistema de ficheros raíz*. El directorio raíz del sistema de ficheros raíz se representa como `/`. Cualquier otro sistema de ficheros, por tanto, se *monta* bajo el sistema de ficheros raíz. No importa cuántos discos tenga en su sistema FreeBSD, cada directorio parecerá ser parte del mismo disco.

Suponga que tiene tres sistemas de ficheros, denominados `A`, `B` y `C`. Cada sistema de ficheros tiene un directorio raíz, el cual contiene otros dos directorios, llamados `A1`, `A2` (y de la misma manera `B1`, `B2` y `C1`, `C2`).

Usaremos `A` como sistema de ficheros raíz. Si usara `ls` para ver el contenido de este directorio vería dos subdirectorios, `A1` y `A2`. El árbol de directorios sería como este:

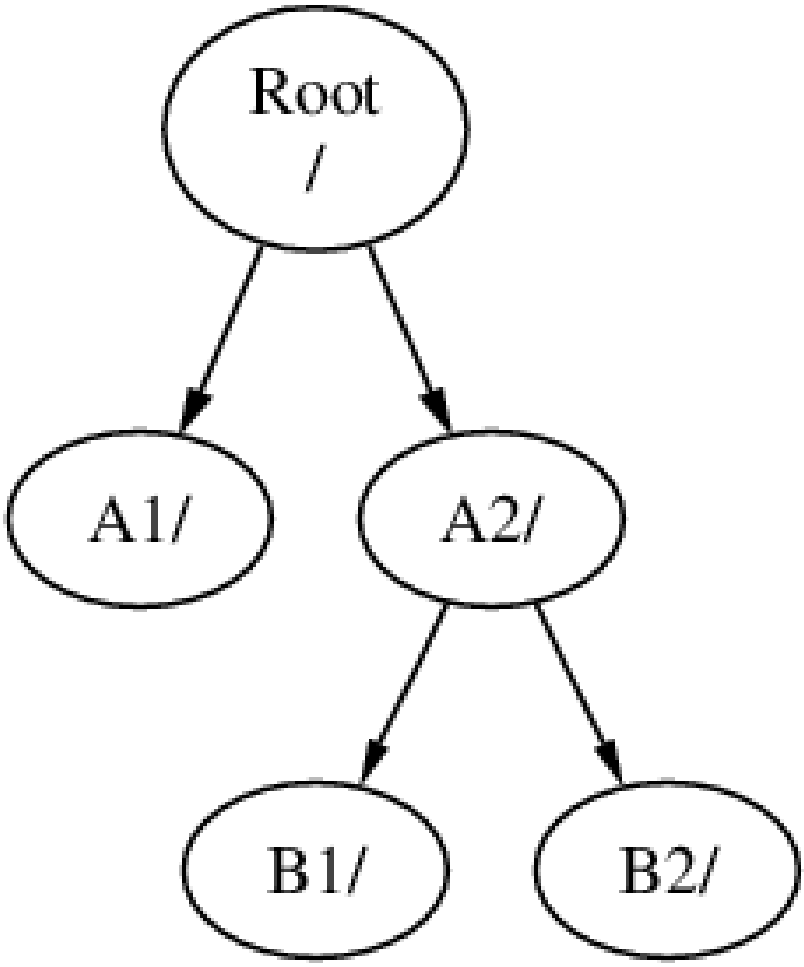


Un sistema de ficheros debe montarse en un directorio de otro sistema de ficheros. Ahora suponga que monta el sistema de ficheros `B` en el directorio `A1`. El directorio raíz de `B` reemplaza a `A1`, y los directorios en `B` aparecen de esta manera:



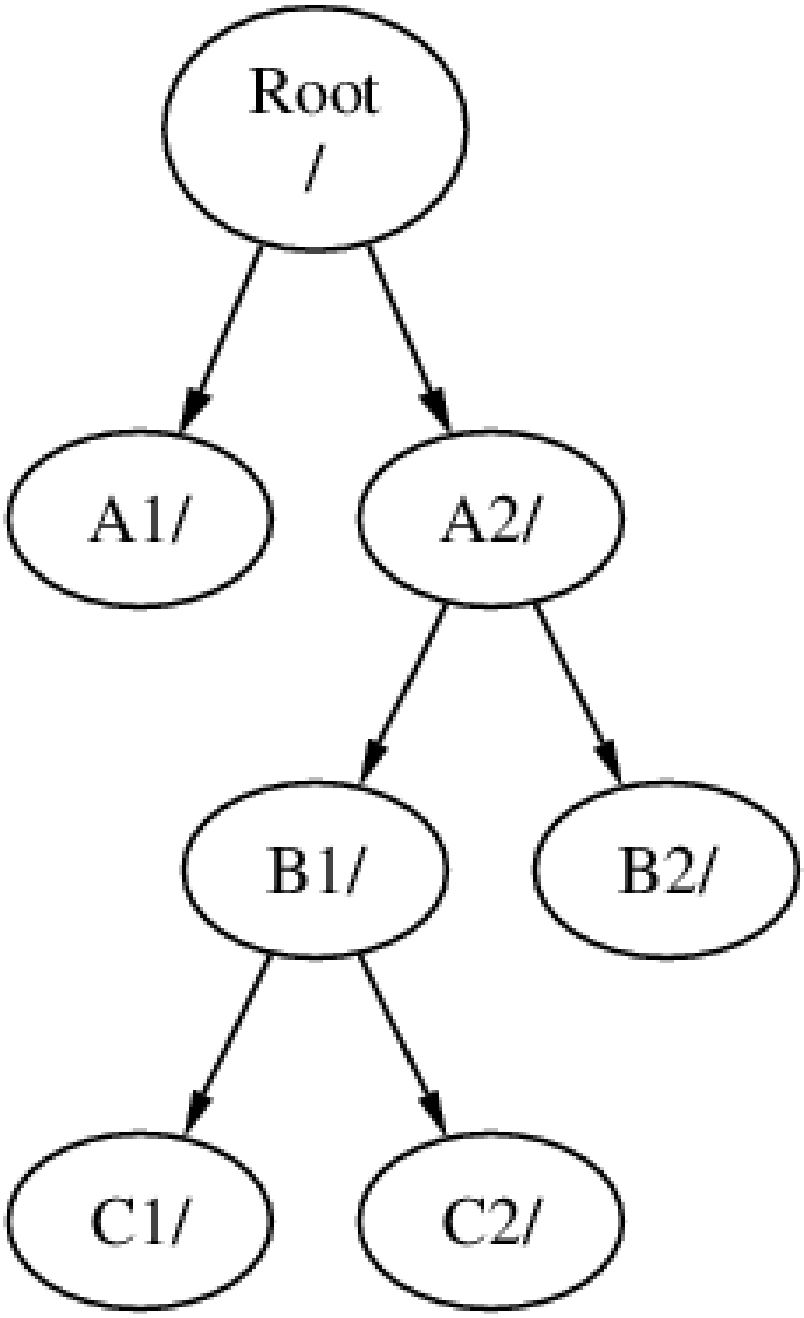
Cualquier fichero que esté en el directorio `B1` o `B2` puede encontrarse con la ruta `/A1/B1` o `/A1/B2` según sea necesario. Cualquier fichero que esté en `/A1` ha desaparecido temporalmente. Aparecerán de nuevo si `B` se *desmonta* de `A`.

Si `B` se monta en `A2` entonces el diagrama se vería así:

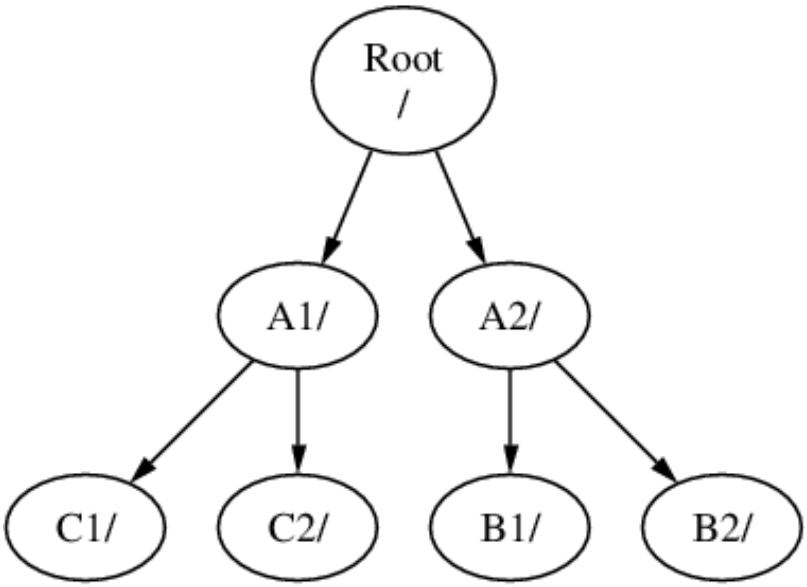


y las rutas serían `/A2/B1` y `/A2/B2` respectivamente.

Pueden montarse sistemas de ficheros uno sobre otro. Continuando con el ejemplo anterior, el sistema de ficheros `C` podría montarse en el directorio `B1` en el sistema de ficheros `B`, lo que nos llevaría a esto:


Root
/
A1/
A2/
B1/
B2/
C1/
C2/

O `C` podría montarse directamente en el sistema de ficheros `A`, bajo el directorio `A1`:



Si está familiarizado con MS-DOS® esto es similar, aunque no idéntico, a utilizar `join`.

Esto no es algo a lo deba usted dedicar tiempo de forma habitual. Normalmente creará sistemas de ficheros al instalar FreeBSD y decidirá dónde montarlos; lo más habitual es que no los cambie de sitio a menos que agregue un disco nuevo.

Es perfectamente posible tener un sistema de ficheros raíz extenso y no necesitar crear otros. Este esquema tiene unos cuantos inconvenientes y una ventaja:

- Si dispone de varios sistemas de ficheros puede optar por usar distintas *opciones de montaje*. Por ejemplo, gracias a una planificación cuidadosa, el sistema de ficheros raíz puede montarse como sólo-lectura, haciendo imposible borrar sin querer o editar un fichero crítico. Al mantener separados sistemas de ficheros en los que los usuarios pueden escribir, como `/home`, de otros sistemas de ficheros también le permite montar con la opción *nosuid*; dicha opción previene que los bits *suid*/*guid* en los ejecutables almacenados en el sistema de ficheros tengan efecto, mejorando en cierto modo la seguridad.

- FreeBSD optimiza automáticamente el esquema de ficheros en un sistema de ficheros, dependiendo de cómo el sistema de ficheros esté siendo utilizado. Uno que contenga muchos ficheros pequeños tendrá una optimización distinta de uno que contenga menos ficheros y más grandes. Si sólo tiene un gran sistema de ficheros no hay manera de aplicar esta optimización.

- Los sistemas de ficheros de FreeBSD son muy robustos en caso de sufrir un caída eléctrica. De todas maneras, un fallo eléctrico en un momento crítico puede dañar la estructura del sistema de ficheros. Si reparte sus datos en múltiples sistemas de ficheros hará que sea más probable que el sistema arranque después de uno de esos fallos, haciéndole además más fácil la tarea de restaurarlo desde un respaldo si fuera necesario.

- Los sistemas de ficheros son de un tamaño fijo. Si crea un sistema de ficheros cuando instala FreeBSD y le da un tamaño específico, tal vez descubra más tarde que necesita hacer la partición más grande. Esto no es fácil de realizar sin hacer una copia de seguridad, crear de nuevo el sistema de ficheros con el nuevo tamaño y entonces restaurar los datos respaldados.

Importante

FreeBSD dispone de growfs(8), que permite incrementar el tamañño de un sistema de ficheros «al vuelo», eliminando esta limitación.

Los sistemas de ficheros están alojados en particiones. Este es un detalle muy importante, puesto que el término *partición* no significa aquí lo mismo que en otros entornos (por ejemplo, en MS-DOS®) debido a la herencia UNIX® de FreeBSD. Cada partición se identifica con una letra desde `a` hasta `h`. Cada partición puede contener solamente un sistema de ficheros, lo que significa que los sistemas de ficheros suelen definirse mediante su punto de montaje en la jerarquía del sistema de ficheros o por la letra de la partición en la que están alojados.

FreeBSD también utiliza espacio de disco como *espacio de intercambio (swap)*. El espacio de intercambio le brinda a FreeBSD *memoria virtual*. Esto permite al su sistema comportarse como si tuviera más memoria de la que realmente tiene. Cuando a FreeBSD se le agota la memoria mueve algunos de los datos que no está utilizando en ese momento al espacio de intercambio, y los vuelve a poner donde estaban (desplazando alguna otra cosa) cuando los necesita.

Algunas particiones tienen ciertas convenciones heredadas.

Partición	Representación
`a`	Normalmente contiene el sistema de ficheros raíz
`b`	Normalmente contiene el espacio de intercambio (swap)
`c`	Suele tener el mismo tamaño de la «slice» que la encierra. Esto permite a las utilidades que necesitan trabajar en toda la «slice» entera (por ejemplo durante una búsqueda de bloques dañados) trabajar en la partición `c`.

Partición	Representación
	Normalmente no debería usted crear un sistema de ficheros en esta partición.
`d`	La partición `d` solía tuvo un significado especial asociado pero ya no lo tiene.

Cada partición que contiene un sistema de ficheros se almacena en lo que FreeBSD llama una «slice». «slice» es en FreeBSD lo que en otros ámbitos se denomina partición; es un hecho que deriva de los orígenes de FreeBSD como ya sabemos basado en UNIX®.

Los números de «slice» muestran el nombre de dispositivo, al que precede una `s` y un número que puede ser un 1 u otro número mayor. Por lo tanto «da0*s1*» es la primera slice en la primera unidad SCSI. Sólo puede haber cuatro «slice» físicas en un disco, pero puede haber «slice» lógicas dentro «slice» físicas del tipo apropiado. Estas «slice»extendidas se numeran a partir de 5, así que «ad0*s5*» es la primera «slice» extendida en el primer disco IDE. Estos dispositivos se usan en sistemas de ficheros que se preve que ocupen una slice.

Tanto las «slice» y las unidades físicas «peligrosamente dedicadas», como otras unidades contienen *particiones*, que se designan mediante letras desde la `a` hasta `h`. Esta letra se añade al nombre del dispositivo. Se verá mucho mejor mediante ejemplos: «da0*a*» es la partición a en la primera unidad da y es una de esas a las que llamamos «peligrosamente dedicada». «ad1s3*e*» es la quinta partición en la tercera slice de la segunda unidad de disco IDE.

Para terminar, cada disco en el sistema tiene también su designación. El nombre de disco comienza por un código que indica el tipo de disco, luego un número, que indica qué disco es. A diferencia de las «slice», la numeración de discos comienza desde `0`. Puede las numeraciones más comunes en el Tabla 3.1, “Códigos de dispositivos de disco”.

Cuando se hace referencia a una partición, FreeBSD necesita que también se nombre la «slice» y el disco que contiene la partició. Esto se hace con el nombre de disco, `s`, el número «slice» y por último la letra de la partición. Tiene varios casos en el Ejemplo 3.1, “Ejemplo de nombres de disco, «slice» y partición”.

En el Ejemplo 3.2, “Modelo conceptual de un disco” muestra un modelo conceptual del esquema de un disco que debería ayudarle a aclarar las cosas.

Antes de instalar FreeBSD tendrá que configurar las «slice» de disco, después crear particiones dentro de las «slice» que vaya a usar en FreeBSD y luego crear un sistema de ficheros (o swap) en cada partición y luego decidir cuál va a ser el punto de montaje del sistema de ficheros.

Tabla 3.1. Códigos de dispositivos de disco

Código	Significado
`ad`	Disco ATAPI (IDE)

Código	Significado
`da`	Disco de acceso directo SCSI
`acd`	CDROM ATAPI (IDE)
`cd`	CDROM SCSI
`fd`	Disquete (floppy)

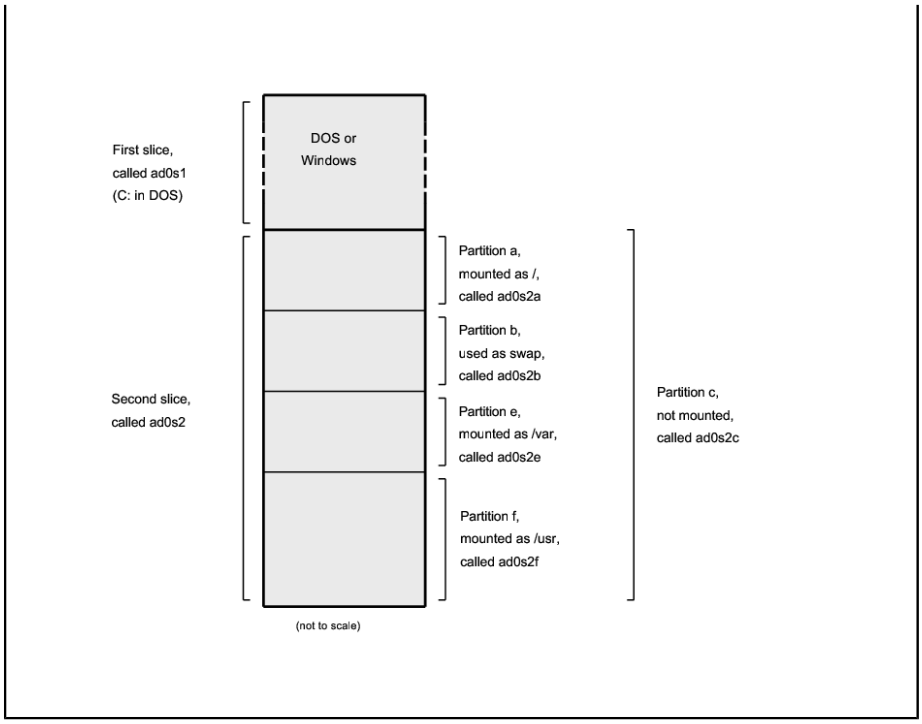
Ejemplo 3.1. Ejemplo de nombres de disco, «slice» y partición

Nombre	Significado
`ad0s1a`	La primera partición (`a`) en la primera slice (`s1`) en el primer disco IDE (`ad0`).
`da1s2e`	La quinta partición (`e`) en la segunda slice (`s2`) en el segundo disco SCSI (`da1`).

Ejemplo 3.2. Modelo conceptual de un disco

Este diagrama muestra cómo ve FreeBSD el primer disco IDE en el sistema. Se asume que el disco es de 4 GB, y contiene dos «slices» de 2 GB (particiones MS-DOS®). La primera partición contiene un disco MS-DOS®, `C:`, y la segunda partición contiene una instalación de FreeBSD. Esta instalación de ejemplo tiene tres particiones, y una partición swap.

Cada una de las tres particiones tiene un sistema de ficheros. La partición `a` se utilizará para el sistema de ficheros raíz, `e` para la jerarquía del directorio `/var`, y `f` para la jerarquía del directorio `/usr`.



3.6. Montaje y desmontaje de sistemas de ficheros

El sistema de ficheros se visualiza mejor como un árbol enraizado, tal y como esá, en `/`. `/dev`, `/usr` y todos los demás directorios en el directorio raíz son raamas, las cuales pueden tener sus propias ramas, como `/usr/local` y así sucesivamente.

Existen varias razones para albergar algunos de estos directorios en sistemas de ficheros separados. `/var` contiene los directorios `log/`, `spool/` y varios tipos de ficheros temporales y pueden llegar a desbordarse. Agotar el espacio del sistema de ficheros raíz no es nada bueno desde cualquier punto de vista, así que separar `/var` de de `/` es algo que debería hacerse siempre que sea posible.

Otra razón para meter ciertos árboles de directorios en otros sistemas de ficheros es si van a estar albergados en discos físicos separados, o si son discos virtuales separados, como un montaje por NFS en el caso de unidades de CDROM.

3.6.1. El fichero `fstab`

Durante el proceso de arranque los sistemas de ficheros listados en `/etc/fstab` se montan automáticamente (a menos que estén listados con la opción `noauto`).

`/etc/fstab` contiene una lista de líneas con el siguiente formato:

```
dispositivo        /punto-de-montaje  punto de montaje      opciones
  dumpfreq      passno
```

`dispositivo`
: Un nombre de dispositivo (*debe existir*).

`punto-de-montaje`
: Un directorio (que debe existir) en el que montar el sistema de ficheros.

`tipo de sistema ficheros`
: El tipo de sistema de ficheros es un parámetro que interpretará mount(8). El sistema de ficheros por omisión de FreeBSD es `ufs`.

`opciones`
: Ya sea `rw` para sistemas de ficheros de lectura-escritura, o `ro` para sistemas de ficheros de sólo lectura, seguido de cualquier otra opción que sea necesaria. Una opción muy habitual es `noauto`, que se suele usar en sistemas de ficheros que no se deben montar durante la secuencia de arranque. Tiene otras opciones en la página de manual de mount(8).

`dumpfreq`
: dump(8) la usa para determinar qué sistema de ficheros requieren volcado. Si el campo no está declarado se asume un valor de cero.

`passno`
: Determina el orden en el cual los sistemas de ficheros deben revisarse. Los sistemas de ficheros que hayan de saltarse deben tener su `passno` a cero. El sistema de ficheros raíz (que obviamente debe ser revisado antes que cualquier otro) debe tener su `passno` puesto a uno, y los demás sistemas de ficheros deben tener valores mayores que uno. Si más de un sistema de ficheros tiene el mismo `passno` fsck(8) tratará de revisarlos en paralelo en caso de ser posible.

Consulte la página de manual de fstab(5) para mayor información sobre el formato del fichero `/etc/fstab` y las opciones que contiene.

3.6.2. La orden `mount`

mount(8) es al fin y al cabo quien monta los sistemas de ficheros.

En su forma más básica se usa lo siguiente:

```
# mount dispositivo punto-de-montaje
```

Existe una gran cantidad de opciones (las encontrará todas en mount(8)) pero las más comunes son:

`-a`
: Montar todos los sistemas de ficheros que aparezcan en `/etc/fstab`, excepto aquellos marcados como «noauto», excluidos por el parámetro `-t` o aquellos que ya estén montados.

`-d`
: Realizar todo excepto la llamada real de montaje del sistema. Esta opción es muy útil en caso de problemas si se combina con la opción `-v` para determinar qué es lo que mount(8) está haciendo realmente.

`-f`
: Forzar el montaje de un sistema de ficheros inestable (por ejemplo uno que da errores tras un reinicio súbito, algo que es bastante *peligroso*), o forzar la revocación de accesos de escritura cuando se cambia el estado de un sistema de ficheros de lectura-escritura a solo lectura.

`-r`
: Montar el sistema de ficheros como sólo lectura. Esto es idéntico a utilizar el argumento `ro` (`rdonly` para versiones anteriores a FreeBSD 5.2) en la opción `-o`.

`-t` *`tipo de sistema de ficheros`*
: Montar un sistema de ficheros dado con el tipo de sistema de ficheros, o montar solamente sistemas de ficheros del tipo dado si se proporciona la opción `-a`.

  «ufs» es el sistema de ficheros por omisión.

`-u`
: Actualizar puntos de montaje en el sistema de ficheros.

`-v`
: Mostrar mayor información.

`-w`
: Montar el sistema de ficheros como lectura-escritura.

La opción `-o` toma una lista las siguientes opciones separada por comas:

nodev
: No interpretar dispositivos especiales en el sistema ficheros. Es una opción de seguridad que puede ser muy útil.

noexec
: No permitir la ejecución de binarios en este sistema de ficheros. Esta también es una opción de seguridad útil.

nosuid
: No interpretar bits setuid o setgid en el sistema de ficheros. Esta también es una opción de seguridad útil.

3.6.3. La orden umount

umount(8) toma como parámetro un punto de montaje, un nombre de dispositivo, o la opción -a o -A.

Todas las formas toman -f para forzar el desmontaje y -v para mostrar más información. Tenga muy en cuenta que usar -f no suele ser una forma recomendable de proceder. Desmontar a la fuerza los sistemas de ficheros puede acarrear el congelar la máquina o dañar los datos en el sistema de ficheros.

-a y -A se usan para desmontar todos los sistemas de ficheros montados, con la ventaja de poder elegir el tipo de sistema de ficheros que se use tras -t. De todas maneras -A no trata de desmontar el sistema de ficheros raíz.

3.7. Procesos

FreeBSD es un sistema operativo multitarea. Esto significa que parece como si más de un programa se estuviera ejecutando al mismo tiempo. Cada programa uno de esos programas que se está ejecutando en un momento dado se denomina *proceso*. Cada orden que ejecuta iniciará al menos un proceso nuevo, y hay muchos procesos que se están que se están ejecutando en todo momento, manteniendo el sistema en funcionamiento.

Cada proceso tiene un identificador individual consistente en un número llamado *ID del proceso*, o *PID*, y al igual que sucede con los ficheros, cada proceso tiene un propietario y un grupo. La información de propietario y grupo se usa para determinar qué ficheros y dispositivos puede abrir el proceso mediante los permisos de fichero explicados anteriormente. La mayoría de los procesos también tiene un proceso padre. El proceso padre es el proceso que los inició. Por ejemplo, si está tecleando órdenes en la shell, la shell es un proceso, y cualquier orden que usted ejecute también lo será. De este modo cada proceso que ejecute tendrá como proceso padre a su shell. La excepción es un proceso especial llamado init(8). init es siempre el primer proceso, así que su PID siempre es 1. El kernel arranca automáticamente init en el arranque de FreeBSD.

Hay dos órdenes particularmente útiles para ver los procesos en el sistema, ps(1) y top(1). ps se usa para mostrar una lista estática de los procesos que se ejecutan en el sistema en es momento, y puede mostrar sus PID, cuánta memoria está usando, la línea de órdenes con la que fueron iniciados y muchas más cosas. top despliega todos los procesos que se están ejecutando y actualiza la pantalla cada pocos segundos para que pueda ver lo que está haciendo su sistema.

Por omisión ps solo le muestra las órdenes que están ejecutando *y* que sean propiedad de su usuario. Por ejemplo:

```
% ps
  PID  TT  STAT      TIME COMMAND
```

```
  298  p0  Ss     0:01.10 tcsh
 7078  p0  S      2:40.88 xemacs mdoc.xsl (xemacs-21.1.14)
37393  p0  I      0:03.11 xemacs freebsd.dsl (xemacs-21.1.14)
48630  p0  S      2:50.89 /usr/local/lib/netscape-linux/navigator-
linux-4.77.bi
48730  p0  IW     0:00.00 (dns helper) (navigator-linux-)
72210  p0  R+     0:00.00 ps
  390  p1  Is     0:01.14 tcsh
 7059  p2  Is+    1:36.18 /usr/local/bin/mutt -y
 6688  p3  IWs    0:00.00 tcsh
10735  p4  IWs    0:00.00 tcsh
20256  p5  IWs    0:00.00 tcsh
  262  v0  IWs    0:00.00 -tcsh (tcsh)
  270  v0  IW+    0:00.00 /bin/sh /usr/X11R6/bin/startx -- -bpp 16
  280  v0  IW+    0:00.00 xinit /home/nik/.xinitrc -- -bpp 16
  284  v0  IW     0:00.00 /bin/sh /home/nik/.xinitrc
  285  v0  S      0:38.45 /usr/X11R6/bin/sawfish
```

Como puede ver en este ejemplo la salida de ps(1) está organizada en columnas. `PID` es el ID de proceso anteriormente expuesto. Los PIDs se asignan a partir del 1 y hasta 99999, y vuelven a comenzar desde el 1 otra cuando se terminan los números. La columna `TT` muestra la tty en la que el programa se está ejecutando; podemos ignorarla tranquilamente por el momento. `STAT` muestra el estado del programa; de momento también podemos ignorarlo. `TIME` es la cantidad de tiempo que el programa ha se ha estado ejecutando en la CPU (generalmente no es el tiempo transcurrido desde que se inició el programa, ya que la mayoría de los programas pasan mucho tiempo esperando antes de que necesiten gastar tiempo en la CPU. Finalmente, `COMMAND` es la línea de órdenes que se empleó para ejecutar el programa.

ps(1) admite muchas opciones sobre la información que se desea ver. Uno de los conjuntos más útiles es `auxww`. `a` muestra información acerca de todos los procesos ejecutándose, no solamente los suyos. `u` despliega el nombre de usuario del propietario del proceso, así como el uso de memoria. `x` despliega información acerca de los procesos dæmon y `ww` hace que ps(1) despliegue la línea de órdenes completa, en lugar de truncarla cuando es demasiado larga para caber en la pantalla.

La salida de top(1) es similar. Veamos un ejemplo:

```
% top
last pid: 72257;  load averages:  0.13,  0.09,  0.03    up 0+13:38:33
  22:39:10
47 processes:  1 running, 46 sleeping
CPU states: 12.6% user,  0.0% nice,  7.8% system,  0.0% interrupt, ↺
79.7% idle
Mem: 36M Active, 5256K Inact, 13M Wired, 6312K Cache, 15M Buf, 408K ↺
Free
Swap: 256M Total, 38M Used, 217M Free, 15% Inuse

  PID USERNAME PRI NICE  SIZE    RES STATE    TIME   WCPU    CPU ↺
COMMAND
```

```
72257 nik       28   0  1960K  1044K RUN      0:00 14.86%  1.42% top
 7078 nik        2   0 15280K 10960K select   2:54  0.88%  0.88% ↺
xemacs-21.1.14
  281 nik        2   0 18636K  7112K select   5:36  0.73%  0.73% ↺
XF86_SVGA
  296 nik        2   0  3240K  1644K select   0:12  0.05%  0.05% ↺
xterm
48630 nik        2   0 29816K  9148K select   3:18  0.00%  0.00% ↺
navigator-linu
  175 root       2   0   924K   252K select   1:41  0.00%  0.00% ↺
syslogd
 7059 nik        2   0  7260K  4644K poll     1:38  0.00%  0.00% mutt
...
```

La salida está dividida en dos secciones. La cabecera (las primeras cinco líneas) muestra el PID del último proceso en ejecutarse, la carga promedio del sistema (una medida de la carga del sistema), el «uptime» del sistema (tiempo desde el último reinicio) y la hora actual. Las otras cifras en la cabecera se relacionan con cuántos procesos hay en ejecución en el sistema (47 en este caso), cuánta memoria y espacio de intercambio (swap) está en uso, y cuánto tiempo está el sistema en diferentes estados de CPU.

Más abajo hay una serie de columnas con información similar a la salida de ps(1). Igual que antes, puede usted ver el PID, el nombre de usuario, la cantidad de tiempo de CPU en uso y la orden que se ejecutó. top(1) también mostrará por omisión la cantidad de espacio de memoria que emplea cada proceso. Está dividido en dos columnas, una para el tamaño total y otra para el tamaño residente (el tamaño total es cuánta memoria ha necesitado la aplicación y el tamaño residente es cuánta se está usando en ese momento concreto). En este ejemplo puede verse que Netscape® requerido casi 30 MB de RAM, pero actualmente solo está usando 9 MB.

top(1) actualiza automáticamente el listado cada dos segundos, pero este lapso puede cambiarse mediante la opción `s`.

3.8. Dæmons, señales y cómo matar procesos

cuando ejecuta un editor es fácil controlarlo, decirle que cargue ficheros y demás. Puede hacerse porque el editor permite ese control y porque el editor depende de una *terminal*. Algunos programas no están diseñados para ejecutarse entradas continuas por parte del usuario, así que se desconectan de la terminal a la primera oportunidad. Por ejemplo, un servidor web pasa todo el dia respondiendo peticiones web y normalmente no necesita que usted le haga caso. Los programas que transportan correo electrónico de un sitio a otro son otro ejemplo de esta clase de aplicación.

Llamamos a estos programas *dæmons*. Los Dæmons eran personajes de la mitología griega; no eran ni buenos ni malos, eran pequeños espíritus sirvientes que, en gran medida, hacían cosas útiles por la humanidad. Algo muy parecido a cómo los servidores web y los

servidores de correo hacen hoy día tantas cosas útiles para nosotros. Por eso, desde hace mucho tiempo la mascota de BSD es ese dæmon de aspecto tan ufano con su tridente y su cómodo calzado deportivo.

Hay una norma según la cual se da nombre a los programas que suelen ejecutarse como dæmons con una «d» final. BIND es el dæmon de nombres de Berkeley (y el programa que en realidad se ejecuta se llama `named`); el servidor web Apache se llama `httpd`, el dæmon de «spool» de impresora de línea es `lpd` y así sucesivamente. Se trata de un acuerdo, no una ley férrea; por ejemplo el dæmon principal de correo de Sendmail se llama `sendmail`, y no `maild`, como podría suponerse visto lo precedente.

Algunas veces necesitará comunicarse con un dæmon. Estas comunicaciones se denominan *señales*; es posible comunicarse con un dæmon (o con cualquier otro proceso ejecutándose) mandándole una señal. Existen diversos tipos de señales diferentes que puede mandar: algunas tienen un significado especial, otras son interpretadas por la aplicación y la documentación de la aplicación le dirá cómo interpreta la señal esa aplicación). Solamente puede enviar una señal a un del que sea usted propietario. Si manda una señal a un proceso de otro usuario con kill(1) o kill(2) verá un mensaje del sistema en el que se le comunica que no tiene permiso para hacer tal cosa. La excepción a esto es el usuario `root`, que puede mandar señales a los procesos de cualquier usuario del sistema.

FreeBSD también enviará señales de aplicación en determinados casos. Si una aplicación está mal escrita y trata de acceder a memoria a la que no está previsto que acceda FreeBSD manda al proceso la señal *Violación de segmento* (`SIGSEGV`). Si una aplicación ha utilizado la llamada de sistema alarm(3) para ser avisada después de que un periodo de tiempo haya transcurrido se le mandará la señal de alarma (`SIGALRM`), y así sucesivamente.

Hay dos señales que pueden usarse para detener un proceso, `SIGTERM` y `SIGKILL`. `SIGTERM` es la manera amable de matar un proceso; el proceso puede *recibir* la señal, darse cuenta que usted quiere que se apague, cerrar cualquier fichero de log que pueda tener abierto y generalmente terminar cualquier tarea que esté realizando en ese momento antes de apagarse. En algunos casos un proceso puede incluso ignorar `SIGTERM` si se encuentra inmerso en una tarea que no puede ser interrumpida.

Por el contrario, un proceso no puede ignorar una señal `SIGKILL`. Esta es la señal «No me importa lo que estás haciendo, detente ahora mismo». Si manda un `SIGKILL` a un proceso FreeBSD detendrá ese proceso inmediatamente.[4].

Otras señales que puede usar: `SIGHUP`, `SIGUSR1` y `SIGUSR2`. Son señales de propósito general, y aplicaciones diferentes pueden hacer cosas diferentes cuando se utilicen.

[4]Esto no es del todo cierto (ciertas cosas no pueden ser interrumpidas. Por ejemplo, si el proceso está tratando de leer desde un fichero que está en otro sistema de la red, y el otro sistema no está disponible por alguna razón (por estar apagada, o que la red tenga un fallo), tenemos un caso de lo que llamamos «proceso ininterrumpible». Más tarde, al proceso se le acabará el tiempo de espera, generalmente pasados dos minutos. Tan pronto como esto ocurra el proceso será liquidado.

Suponga que ha cambiado el fichero de configuración de su servidor web; es un buen momento para decirle al servidor web que lea y aplique la configuración. Podría detener y reiniciar `httpd`, pero esto implicaría un período breve de suspensión del servicio de su servidor web, lo cual puede no ser asumible. La mayoría de los dæmons fueron creados pensando en que fueran capaces de responder a la señal `SIGHUP` releyendo su fichero de configuración. En lugar de matar y reiniciar `httpd` le podría mandar la señal `SIGHUP`. Dado que no hay una manera estándar para responder a estas señales, diferentes dæmons tendrán diferente comportamiento, así que asegúrese de leer la documentación del dæmon en cuestión.

Las señales se envian mediante kill(1), como puede verse en el siguiente ejemplo.

Procedimiento 3.1. Envío de una señal a un proceso

Este ejemplo muestra como enviar una señal a inetd(8). El fichero de configuración de `inetd` es `/etc/inetd.conf` e `inetd` releerá dicho fichero de configuración cuando se le envíe un `SIGHUP`.

1. Identifique el ID de proceso del proceso al que quiere enviarle la señal mediante ps(1) y grep(1). grep(1) se usa para buscar cadenas de texto de su elección en la salida estándar. Puede ejecutarse como un usuario normal, mientras que inetd(8) se ejecuta como `root`, así que debe pasarle los parámetros `ax` a ps(1).

   ```
   % ps -ax | grep inetd
     198  ??  IWs    0:00.00 inetd -wW
   ```

   Vemos que el PID de inetd(8) es 198. En algunos casos `grep inetd` también puede aparecer en esta salida. Esto se debe a la manera en que ps(1) tiene que encontrar la lista de procesos ejecutándose.

2. Utilice kill(1) para enviar la señal. Debido a que inetd(8) está siendo ejecutado po `root` tendrá que usar primero su(1) para volverse `root`.

   ```
   % su
   Password:
   # /bin/kill -s HUP 198
   ```

   Del mismo modo que la mayoría de órdenes UNIX® kill(1) no imprimirá ninguna salida si ha funcionado bien. Si envía una señal a un proceso del que no es el propietario verá lo siguiente: kill: *`PID`*: Operation not permitted. Si no teclea bien el PID puede enviar la señal a un proceso equivocado, lo cual puede ser malo, o si tiene suerte, habrá enviado la señal a un proceso que no está en uso y verá el sistema le dir´ kill: *`PID`*: No such process.

¿Por qué utilizar `/bin/kill`?

Muchas shells incorporan su propio `kill`; esto es, el shell mandará la señal directamente, en lugar de ejecutar `/bin/kill`. Esto puede ser útil pero las diferentes shells tienen diferentes sintaxis para especificar el nombre de la señal que envían. En lugar de tratar de aprederse todas ellas, es más fácil simplemente usar `/bin/kill ...` sea la que sea la shell que prefiera usar.

El envío de otras señales es muy similar; sustituya `TERM` o `KILL` en la línea de órdenes según sea necesario.

Importante

Matar procesos aleatorios en el sistema es una mala idea. En particular, init(8), ID de proceso 1, es muy especial. Ejecutar `/bin/kill -s KILL 1` es una manera rápida de apagar su sistema. *Siempre* revise dos veces los argumentos con los que ejecuta kill(1) *antes* de pulsar Intro.

3.9. Shells

En FreeBSD gran parte del trabajo diario se hace en una interfaz de línea de órdenes llamada shell. El trabajo principal de la shell es ir recibiendo órdenes mediante un canal de entrada y ejecutarlos. Muchas shells también tienen funciones integradas para ayudar ayudar con las tareas diarias como manipulación de ficheros, gestión de archivos con expresiones regulares, edición en la propia línea de órdenes, macros de órdenes y variables de entorno. FreeBSD incluye diversas shells, como `sh`, el shell Bourne y `tcsh`, el shell C mejorado. Hay muchas otras shells disponibles en la colección de ports de FreeBSD, como `zsh` y `bash`.

?Qué shell usar? es una cuestión de gustos. Si va a programar en C puede preferir usar una shell tipo C, como `tcsh`. Si viene del mundo Linux o si es nuevo en la interfaz de línea de órdenes de UNIX® puede probar con `bash`. Tenga en cuenta que cada shell posee unas propiedades únicas que pueden o no funcionar con su entorno de trabajo preferido y que puede usted elegir qué shell usar.

Una de las propiedades comunes de las shell es que completan los nombres de fichero. Una vez que ha introducido las primeras letras de una orden o del nombre de un fichero, se puede hacer que la shell complete automáticamente el resto de la orden o nombre de fichero pulsando la tecla Tab. Veamos un ejemplo. Suponga que tiene dos ficheros llamados `talcual` y `tal.cual` . Usted quiere borrar `tal.cual` . Lo que habría que teclear es: `rm ta[tabulador].[tabulador]` .

La shell mostraría en pantalla `rm ta[BIP].cual` .

El [BIP] es la campana de la consola, es decir, la shell está diciéndome que no pudo completar totalmente el nombre de fichero porque hay más de una coincidencia. Tanto `talcual` como `tal.cual` comienzan por `ta`, pero solo se pudo completar hasta `tal.` Si se teclea `.`, y de nuevo tabulador la shell es capaz de añadir el resto del nombre de fichero.

Otra función de la shell son las variables de entorno. Las variables de entorno son parejas de valores clave almacenados en el espacio de entorno del shell. Este espacio puede ser leído por cualquier programa invocado por la shell y por lo tanto contiene mucha configuración de programas. Esta es una lista de las variables de entorno más comunes y su significado:

Variable	Descripción
`USER`	Nombre de usuario en el sistema.
`PATH`	Lista de directorios, separados por punto y coma, en los que se buscan ejecutables.
`DISPLAY`	Nombre de red de la pantalla X11 a la que conectarse, si está disponible.
`SHELL`	La shell actual.
`TERM`	El nombre de la terminal del usuario. Se usa para determinar las características de la terminal.
`TERMCAP`	Base de datos donde encontrar los códigos de escape necesarios para realizar diferentes funciones en la terminal.
`OSTYPE`	Tipo de sistema operativo. Por ejemplo, FreeBSD.
`MACHTYPE`	Arquitectura de CPU en la que se está ejecutando el sistema.
`EDITOR`	El editor de texto preferido por el usuario.
`PAGER`	El paginador de texto preferido por el usuario.

Variable	Descripción
MANPATH	Lista de directorios separados por punto y coma en los que se buscan páginas de manual.

Establecer una variable de entorno difiere ligeramente de shell a shell. Por ejemplo, en las shells al estilo C como `tcsh` y `csh`, se usa `setenv` para establecer las variables de entorno. Bajo shells Bourne como `sh` y `bash`, se usa `export` para establecer las variables de entorno actuales. Por ejemplo, para establecer o modificar la variable de entorno `EDITOR` (bajo `csh` o `tcsh`) la siguiente orden establece `EDITOR` como `/usr/local/bin/emacs` :

```
% setenv EDITOR /usr/local/bin/emacs
```

Bajo shells Bourne:

```
% export EDITOR="/usr/local/bin/emacs"
```

También se puede hacer que la mayoría de las shells muestren el contenido de una variable de entorno situando el carácter `$` delante del nombre de la variable en la línea de órdenes. Por ejemplo, `echo $TERM` mostrará cualquiera que sea el valor que haya establecido para `$TERM`, porque la shell expande el valor de `$TERM` y se lo pasa al programa `echo`.

Las shells manejan muchos caracteres especiales, llamados metacaracteres, como representaciones especiales de datos. El más común es el caracter *, que representa cualquier número de caracteres en un nombre de fichero. Estos metacaracteres especiales se pueden usar para la expansión de nombres de fichero. Por ejemplo, teclear `echo *` es casi lo mismo que introducir `ls` porque la shell toma todos los ficheros que coinciden con * y se los pone en la línea de órdenes para que `echo` los vea.

Para evitar que la shell interprete estos caracteres especiales pueden escamotearse anteponiéndoles una contrabarra (\). `echo $TERM` imprime el nombre de terminal que esté usando. `echo \$TERM` imprime `$TERM`, literalmente.

3.9.1. Cómo cambiar su shell

La manera más fácil de cambiar de shell es mediante `chsh`. `chsh` le colocará dentro del editor que esté configurado en la variable de entorno `EDITOR`; si no la ha modificado, el sistema ejecutará `vi`, el editor por omisión. Cambie la línea «Shell:» según sus gustos.

También se le puede suministrar a `chsh` la opción `-s`; ésto establecerá la shell sin necesidad de entrar en el editor de texto. Si por ejemplo quiere que `bash` sea su shell por omisión puede configurarlo del siguiente modo:

```
% chsh -s /usr/local/bin/bash
```

Ejecutar `chsh` sin parámetros y editar la shell desde ahí también funciona.

Nota

La shell que se desee usar *debe* estar incluida en `/etc/shells` . Si se ha instalado una shell desde la colección de ports esto deberí estar hecho automáticamente. Si ha instalado la shell manualmente, tendrá usted que realizar el cambio oportuno en `/etc/shells` .

Por ejemplo, si instaló manualmente `bash` y lo ubicó en `/usr/local/bin` debería hacer lo siguiente:

```
# echo "/usr/local/bin/bash" >> /etc/shells
```

Hecho esto vuelva a ejecutar `chsh`.

3.10. Editores de texto

Gran parte de la configuración de FreeBSD se realiza modificando ficheros de texto, así que le conviene familiarizarse con alguno de ellos. FreeBSD viene con unos cuantos como parte del sistema base y encontrará muchos más en la colección de ports.

El editor de textos más sencillo y fácil de aprender es uno llamado ee, cuyo nombre proviene del inglés «easy editor» (editor fácil). Para iniciar ee se debe teclear en la línea de órdenes `ee nombre-de-fichero` , donde *`nombre-de-fichero`* es el nombre del fichero que se quiere editar. Por ejemplo, para editar `/etc/rc.conf` teclée `ee /etc/rc.conf` . Una vez dentro de `ee` todas las órdenes para manipular las funciones del editor están listadas en la parte superior de la pantalla. El caracter `^` representa la tecla Ctrl del teclado, por lo tanto `^e` significa la combinación de teclas Ctrl+e. Para salir de ee pulse la tecla Esc y elija abandonar («leave») el editor. El editor preguntará entonces si quiere conservar los cambios si el fichero hubiera sido modificado.

FreeBSD viene también con editores de texto mucho más potentes, como vi, como parte del sistema base, mientras que otros editores, como Emacs y vim son parte de la colección de ports de FreeBSD (editors/emacs y editors/vim). Estos editores son muchísimo más poderosos, pero tienen la desventaja de ser un poco más complicados de aprender a manejar. De cualquier manera si planea hacer mucho trabajo de edición de texto, aprender a usar un editor de texto más poderoso como vim o Emacs le ahorrará muchísimo más tiempo a la larga.

3.11. Dispositivos y nodos de dispositivos

Dispositivo es un término utilizado la mayoría de las veces para actividades relacionadas con hardware del sistema, como discos, impresoras, tarjetas gráficas y teclados. Cuando FreeBSD arranca, la mayoría de lo que FreeBSD despliega son dispositivos en el momento de ser detectados. Si lo desea, puede volver a ver todos los mensajes que el sistema emite durante el arranque consultando `/var/run/dmesg.boot` .

Por ejemplo, `acd0` es la primera unidad CDROM IDE, mientras que `kbd0` representa el teclado.

En un sistema operativo UNIX® debe accederse a la mayoría de estos dispositivos a través de ficheros especiales llamados nodos de dispositivo, que se encuentran en el directorio `/dev`.

3.11.1. Creación de nodos de dispositivo

Cuando agregue un nuevo dispositivo a su sistema, o compile soporte para dispositivos adicionales, puede que necesite crear uno o más nodos de dispositivo.

3.11.1.1. `DEVFS` Dispositivo de sistema de ficheros (DEVice File System)

El dispositivo de sistema de ficheros, o `DEVFS`, ofrece acceso a dispositivos del espacio de nombres del kernel en el espacio de nombres del sistema de ficheros global. En lugar de tener que crear y modificar nodos de dispositivo, `DEVFS` se encarga del mantenimiento dinámico de este sistema de fichero.

Consulte devfs(5) si quiere más información.

3.12. Formatos binarios

Para poder entender por qué FreeBSD utiliza el formato elf(5) primero debe saber ciertas cosas sobre los tres formatos de ejecutables «dominantes» en UNIX®:

- a.out(5)

  El formato objeto de UNIX® más antiguo y «clásico». Utiliza una cabecera corta y compacta con un número mágico al inicio que se usa frecuentemente para identificar el formato (vea a.out(5) para más información). Contiene tres segmentos cargados: .text, .data, y .bss además de una tabla de símbolos y una tabla de cadena («strings»).

- COFF

  El formato objeto de SVR3. La cabecera consiste en una tabla de sección, para que pueda tener más contenido además de las secciones .text, .data, y .bss.

- elf(5)

  Es el sucesor de COFF; dispone de secciones múltiples y valores posibles de 32-bits o 64-bits. Una gran desventaja: ELF fué también diseñado asumiendo que solamente existiría una ABI por cada arquitectura de sistema. Esa suposición es en realidad bastante incorrecta y ni siquiera en el mundo comercial SYSV (el cual tiene al menos tres ABIs: SVR4, Solaris y SCO) se puede dar por buena.

  FreeBSD trata de solucionar este problema de alguna manera ofreciendo una herramienta para *marcar* un ejecutable ELF conocido con información acerca de la ABI con la que funciona. Si quiere más información consulte la página de manual de brandelf(1).

FreeBSD viene del campo «clásico» y ha utilizado el formato a.out(5), una tecnología usada y probada en muchas de muchas generaciones de versiones de BSD hasta el inicio de la rama 3.X. Aunque era posible compilar y ejecutar binarios nativos ELF (y kernels) en un sistema FreeBSD desde algún tiempo antes de esto, FreeBSD al principio se mantuvo «contra corriente» y no cambió a ELF como formato por defecto. ?Por qué? Bueno, cuando el mundo Linux efectuó su dolorosa transición a ELF no fué tanto por huir del formato `a.out` como por su inflexible mecanismo de bibliotecas compartidas basado en tablas de saltos, que hacía igual de difícil la construcción de bibliotecas compartidas tanto para los desarrolladores como para los proveedores. Ya que las herramientas ELF disponibles ofrecían una solución al problema de las bibliotecas compartidas y eran vistas por mucha gente como «la manera de avanzar», el costo de migración fué aceptado como necesario y se realizó la transición. El mecanismo de bibliotecas compartidas de FreeBSD está diseñado de manera más cercana al estilo del sistema de bibliotecas compartidas de SunOS™ de Sun y, como tal, es muy sencillo de utilizar.

Entonces, ?por qué existen tantos formatos diferentes?

En un tiempo muy, muy lejano, existía hardware simple. Este hardware tan simple soportaba un sistema pequeño, simple. `a.out` era idóneo para el trabajo de representar binarios en este sistema tan simple (un PDP-11). A medida que la gente portaba UNIX® desde este sistema simple, retuvieron el formato `a.out` debido a que era suficiente para los primeros portes de UNIX® a arquitecturas como 68k de Motorola, VAXen, etc.

Entonces algún brillante ingeniero de hardware decidió que si podía forzar al software a hacer algunos trucos sucios podría sortear ciertos obstáculos del diseño y permitir al núcleo de su CPU correr más rápidamente. Aunque estaba hecho para trabajar con este nuevo tipo de hardware (conocido entonces como RISC), `a.out` no estaba bien adaptado para este hardware, así que varios formatos fueron desarrollados para obtener un rendimiento mayor de este hardware que el podía extraerse del limitado y simple formato `a.out`. Así fué cómo COFF, ECOFF y algunos otros formatos más extraños fueron inventados y sus limitaciones exploradas hasta que se fué llegando a la elección de ELF.

Además, el tamaño de los programas estaba volviendose gigante y los discos (y la memoria física) eran relativamente pequeños, así que el concepto de una biblioteca compartida

nació. El sistema VM también se volvió más sofisticado. A pesar de que todos estos avances se hicieron utilizando el formato `a.out`, su utilidad se iba reduciendo paulatinamente con cada nueva opción. Además, la gente quería cargar cosas dinámicamente en el momento de ejecución, o descartar partes de su programa después de que el código de inicio se ejecutara para ahorrar memoria principal y espacio de swap. Al volverse más sofisticados los lenguajes, la gente empezó a ver la necesidad de introducir código antes del inicio del programa de forma automática. Se hicieron muchos hacks al formato `a.out` para permitir que todas estas cosas sucedieran y lo cierto es que por un tiempo funcionaron. Pero `a.out` no estaba para solucionar todos estos problemas sin incrementar la carga y complejidad del código. Aunque ELF resolvía muchos de estos problemas, en ese momento hubiera sido terrible dejar de lado un sistema que funcionaba, así que ELF tuvo que esperar hasta que fué más doloroso permanecer con `a.out` que migrar a ELF.

De todas maneras, con el paso del tiempo las herramientas de compilación de las que FreeBSD derivó las suyas (en especial el ensamblador y el cargador) evolucionaron en dos árboles paralelos. El árbol FreeBSD FreeBSD añadió bibliotecas compartidas y corrigió algunos errores. La gente de GNU (que fueron quienes escribieron estos programas) los reescribió y añadieron una forma más simple de disponer de compiladores cruzados («cross compilers»), el uso de diferentes formatos, etc. Aunque mucha gente quería compiladores cruzados con FreeBSD como «blanco» no hubo suerte, porque los fuentes que que FreeBSD tenía para as y ld no estaban listos para cumplir esa tarea. La nueva cadena de herramientas GNU (binutils) soporta compilación cruzada, ELF, bibliotecas compartidas, extensiones C++, etc. Además, muchos proveedores están liberando binarios ELF y es algo muy bueno que FreeBSD los pueda ejecutar.

ELF es más expresivo que `a.out` y permite un sistema base más extensible. Las herramientas ELF están mejor mantenidas y ofrecen soporte de compilación cruzada, muy importante para mucha gente. ELF puede ser un poco más lento que `a.out`, pero tratar de medirlo puede ser difícil. También existen numerosos detalles que son diferentes entre los dos en cómo gestionan páginas, cómo gestionan código de inicio, etc. Ninguna es muy importante, pero las diferencias existen. Con el tiempo, el soporte para `a.out` será eliminado del kernel `GENERIC` y es muy posible que se elimine del kernel la posibilidad de ejecutar tales binarios una vez que la necesidad de usar programas `a.out` haya pasado.

3.13. Más información

3.13.1. Páginas de manual

La documentación más exhaustiva de FreeBSD está en las páginas de manual. Casi todos los programas del sistema vienen con un breve manual de referencia explicando el funcionamiento básico y sus diferentes argumentos. Estos manuales pueden revisarse mediante `man`. El uso de `man` es simple:

```
% man orden
```

`orden` es el nombre de la orden sobre la que que desea saber más. Por ejemplo, para más información sobre `ls` escriba:

```
% man ls
```

El manual en línea está dividido en secciones numeradas:

1. Comandos de usuario.
2. Llamadas al sistema y números de error.
3. Funciones en las bibliotecas de C.
4. Controladores de dispositivo.
5. Formatos de fichero.
6. Juegos y demás pasatiempos.
7. Información sobre temas diversos.
8. Comandos relacionados con el mantenimiento del sistema y su funcionamiento.
9. Desarrolladores del Kernel.

En algunos casos, el mismo tema puede aparecer en más de una sección del manual en línea. Por ejemplo, existe una orden de usuario `chmod` y una llamada del sistema `chmod()`. En este caso se le puede decir a `man` cuál desea consultar especificando la sección:

```
% man 1 chmod
```

Esto desplegará la página de manual de la orden de usuario `chmod`. Las referencias a una sección concreta del manual en línea tradicionalmente suelen colocarse entre paréntesis en la documentación escrita, por lo tanto chmod(1) se refiere a la orden de usuario `chmod` y chmod(2) se refiere a la llamada de sistema.

Esto está muy bien si se conoce el nombre del programa y simplemente se quiere saber como usarlo. Pero ?y si no puede recordar el nombre de la orden? Se puede usar man para que realice una búsqueda mediante palabras clave en las descripciones de programas utilizando el argumento `-k`:

```
% man -k mail
```

Dicha orden mostrará una lista de órdenes que contengan la palabra clave «mail» en sus descripciones. Esto es funcionalmente equivalente a usar `apropos`.

Así que, ¿está viendo todos esos programas tan atractivos en `/usr/bin` pero no tiene ni la menor idea de lo que la mayoría de ellos hace? Haga lo siguiente:

```
% cd /usr/bin
% man -f *
```

o

```
% cd /usr/bin
% whatis *
```

que hace exactamente lo mismo.

3.13.2. Ficheros de información GNU: info

FreeBSD incluye muchas aplicaciones y utilidades producidas por la FSF (Free Software Foundation). Además de con las correspondientes páginas de manual, estos programas vienen con documentos de hipertexto más detallados, llamados ficheros `info`, los cuales pueden ser visualizados con `info`, o si tiene instalado emacs, con el modo info de emacs.

Si quiere utilizar la orden info(1) teclée:

```
% info
```

Para una breve introducción teclée `h`. Cuando necesite una referencia rápida, teclée `?`.

Capítulo 4. Instalación de aplicaciones: «packages» y ports

4.1. Sinopsis

FreeBSD viene con una excelente colección de herramientas de sistema como parte del sistema base. A pesar de esto, existe gran cantidad de cosas que uno requiere hacer para poner las cosas realmente en marcha, para lo cual se necesita instalar software adicional de terceros. FreeBSD ofrece dos tecnologías complementarias para instalar software de terceros en nuestro sistema: la Colección de Ports de FreeBSD y «packages»(paquetes) binarios. Cualquiera de los dos sistemas puede usarse para instalar las versiones más recientes, de forma local o directamente desde la red.

Después de leer este capítulo usted sabrá:

- Cómo instalar packages binarios de software de terceros.
- Cómo compilar software de terceros desde la Colección de Ports.
- Cómo eliminar packages o ports instalados previamente.
- Cómo sobreescribir los valores por omisión que utiliza la colección de ports.
- Cómo encontrar la aplicación adecuada.
- Cómo actualizar aplicaciones.

4.2. Aproximación a la instalación de software

Si ha usado un sistema UNIX® con anterioridad sabrá que el procedimiento típico para instalar software de terceros es algo similar a esto:

1. Descargar el software, que puede distribuirse en formato de código fuente o binario.
2. Desempaquetar el software de su formato de distribución (normalmente en un «tarball» comprimido con compress(1), gzip(1) o bzip2(1)).
3. Consultar la documentación (quizás un fichero `INSTALL` o `README`, o los ficheros del subdirectorio `doc/`) para ver como instalar el software.

4. Si el software se distribuye como fuente, compilarlo. Esto puede requerir que editemos el fichero `Makefile` o que ejecutemos el «script» `configure`, entre otras cosas.

5. Instalar y probar el software.

Y esto solamente si todo marcha bien. Si está instalando un software que no ha sido portado específicamente para FreeBSD, puede que sea necesario editar el código para que funcione correctamente.

Si lo desea puede continuar instalando software de la forma «tradicional» en FreeBSD, aunque FreeBSD dispone de dos tecnologías que le pueden ahorrar gran cantidad de tiempo y esfuerzo: los packages y los ports. En el momento de escribir esto existen más de 24,000 aplicaciones de terceros.

En cualquier aplicación el paquete de FreeBSD es un solo fichero que tiene que descargar. Los paquetes contienen copia de los programas binarios precompilados de la aplicación, así como cualquier fichero de configuración o documentación. Los paquetes descargados se pueden manipular con las herramientas de gestión de paquetes de FreeBSD: pkg_add(1), pkg_delete(1), pkg_info(1), etc. Instalar una aplicación nueva puede realizarse con una sola orden.

Un port de FreeBSD es una colección de ficheros diseñada para automatizar el proceso de compilación desde el código fuente.

Recuerde que existen ciertos pasos que deberá llevar a cabo para compilar un programa por usted mismo (descargar, desempaquetar, parchear, compilar e instalar). Los ficheros que conforman un port permiten que el sistema se encargue de todo esto. Usted ejecuta un conjunto simple de órdenes y el código fuente se descarga, desempaqueta, parchea, compila e instala.

De hecho el sistema de ports también se puede usar para crear paquetes que posteriormente se pueden manipular con `pkg_add` y las demás utilidades de gestión de packages que veremos en breve.

Tanto el sistema de ports como el de paquetes entienden las *dependencias*. Suponga que desea instalar una aplicación que depende de que una biblioteca específica esté instalada. Tanto la biblioteca como la aplicación existen en FreeBSD como paquete o port. Si utiliza `pkg_add` o el sistema de ports para instalar la aplicación ambos notarán que la biblioteca no está instalada y procederán a instalarla antes que nada.

Dado que ambas tecnologís son similares quizás se pregunte por qué FreeBSD se toma la molestía de contar con ambas. Los Packages y los Ports tienen sus propias ventajas, así que cuál utilizar dependerá de su elección en cada momento.

- El fichero tarball de un package es normalmente más pequeño que el tarball con el código fuente de la aplicación.

- Los paquetes no requieren compilación. En el caso de aplicaciones grandes, como Mozilla, KDE, o GNOME este detalle puede ser importante, especialmente si usa un equipo lento.

- Los paquetes no requieren que comprenda el proceso que envuelve el compilar software bajo FreeBSD.

- Normalmente los paquetes se compilan con opciones conservadoras, en virtud del gran número de equipos donde se instalarán. Al instalar como port puede usted editar las preferencias para (por ejemplo) generar código específico para un procesador Pentium IV o Athlon.

- Algunas aplicaciones tienen opciones sobre lo que pueden y lo que no pueden hacer. Por ejemplo Apache se puede configurar con una gran variedad de opciones. Al compilarlo desde los ports usted no tiene por que aceptar las opciones predefinidas, puede seleccionar que opciones desea.

  En algunos casos existen varios paquetes para la misma aplicación con diferentes opciones. Por ejemplo, Ghostscript está disponible como paquete `ghostscript` y como paquete `ghostscript-nox11`, dependiendo de si usted cuenta o no con servidor X11. En este caso decidir puede ser sencillo e incluso pueden facilitarse packages con ambas opciones, pero elegir se puede volver un problema si hay más de una o dos opciones de compilación.

- Las condiciones de la licencia de algunas aplicaciones prohiben la distribución de binarios. Solo permiten la distribución del código fuente.

- Existe gente que no confía en los binarios. Al menos puede usted (en teoría) revisar el código fuente y localizar problemas potenciales.

- Si tiene parches locales necesitará el código fuente para poder aplicarlos.

- Hay gente a la que le gusta tener el código fuente para poder leerlo en ratos de ocio, modificarlo, tomar partes prestadas (cuando la licencia lo permite, claro está), etc.

Para tener constancia de los ports actualizados, suscríbase a la lista de correo sobre los ports de FreeBSD y a la lista de correo sobre errores en los ports de FreeBSD.

Aviso

Antes de instalar cualquier aplicación consulte `http://vuxml.freebsd.org` para comprobar los posibles problemas de seguridad relacionados con ella.

También puede instalar security/portaudit que automáticamente comprobará las vulnerabilidades conocidas de todas las aplicacio-

nes instaladas; también lo comprobará antes de compilar cualquier port. Mientras tanto, puede usar `portaudit -F -a` después de instalar algunos packages.

El resto de este capítulo le explicará cómo usar los ports y los packages para instalar software de terceros en FreeBSD.

4.3. Cómo encontrar aplicaciones

Antes de poder instalar cualquier software debe saber qué es lo que quiere instalar y cómo se llama la aplicación.

La lista de software disponible para FreeBSD crece constantemente. Afortunadamente existen varias formas de localizar lo que busca:

- El sitio web de FreeBSD mantiene una lista (actualizada y que admite búsquedas) de todas las aplicaciones disponibles para FreeBSD en la dirección http://www.FreeBSD.org/ports/ . Esta base esta dividida en categorías, y puede buscar por nombre (si lo sabe) o bien listar las aplicaciones disponibles en cierta categoría.

- Dan Langille mantiene FreshPorts en `http://www.FreshPorts.org/` . FreshPorts verifica los cambios en las aplicaciones del árbol de ports, y le permite «seguir» uno o más ports, sobre los que le enviarán un correo cada vez que se actualicen.

- Si no conoce el nombre de la aplicación que desea pruebe a usar un sitio como FreshMeat (http://www.freshmeat.net/) para encontrar una aplicación; después consulte el sitio web de FreeBSD para ver si la aplicación ha sido portada.

- Si sabe el nombre exacto del port y solo necesita saber en qué categoria está puede usar whereis(1). Simplemente escriba `whereis fichero`, donde *`fichero`* es el programa que quiera instalar. Si está en su sistema, le dirá dónde está:

  ```
  # whereis lsof
  lsof: /usr/ports/sysutils/lsof
  ```

  Esto nos dice que `lsof` (una utilidad de sistema) está en el directorio `/usr/ports/sysutils/lsof`.

- Otra forma de encontrar un port en particular es usando el mecanismo de búsqueda integrado en la colección de ports. Para poder usar esta opción de búsqueda debe estar en el directorio `/usr/ports` . Una vez en ese directorio ejecute `make search name=nombre-del-programa` , donde *`nombre-del-programa`* es el nombre del programa que desea encontrar. Por ejemplo, si busca `lsof`:

```
# cd /usr/ports
# make search name=lsof
Port:   lsof-4.56.4
Path:   /usr/ports/sysutils/lsof
Info:   Lists information about open files (similar to fstat(1))
Maint:  obrien@FreeBSD.org
Index:  sysutils
B-deps:
R-deps:
```

Debe prestar especial atención a la línea «Path:»de la salida, porque es la que indica dónde puede encontrar el port. El resto de información no se necesita para instalar el port.

Si quiere hacer una búsqueda más a fondo utilice `make search key=cadena` donde *cadena* es el término que busca. Podrá buscar nombres de ports, comentarios, descripciones y dependencias; también se puede usar para encontrar ports que tengan relación con algún tema en particular si no conoce el nombre del programa que busca.

En ambos casos la cadena de búsqueda no distingue entre mayúsculas y minúsculas. Buscar «LSOF» dará los mismos resultados que buscar «lsof».

4.4. Uso del sistema de packages

Texto de Chern Lee.

4.4.1. Instalar un package

Puede usar pkg_add(1) para instalar un package de FreeBSD desde un fichero local o desde un servidor remoto vía red.

Ejemplo 4.1. Descarga de un paquete manualmente e instalación en nuestro sistema

```
# ftp -a ftp2.FreeBSD.org
Connected to ftp2.FreeBSD.org.
220 ftp2.FreeBSD.org FTP server (Version 6.00LS) ready.
331 Guest login ok, send your email address as password.
230-
230-     This machine is in Vienna, VA, USA, hosted by Verio.
230-         Questions? E-mail freebsd@vienna.verio.net.
230-
230-
230 Guest login ok, access restrictions apply.
```

```
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /pub/FreeBSD/ports/packages/sysutils/
250 CWD command successful.
ftp> get lsof-4.56.4.tgz
local: lsof-4.56.4.tgz remote: lsof-4.56.4.tgz
200 PORT command successful.
150 Opening BINARY mode data connection for 'lsof-4.56.4.tgz' ↺
(92375 bytes).
100% |***********************************************| 92375
       00:00 ETA
226 Transfer complete.
92375 bytes received in 5.60 seconds (16.11 KB/s)
ftp> exit
# pkg_add lsof-4.56.4.tgz
```

Si no tiene una fuente local de packages (por ejemplo un CDROM de FreeBSD) probablemente la mejor opción sea utilizar el argumento `-r` de pkg_add(1), que hará que determine automáticamente el formato del objeto correcto, para posteriormente descargarlo e instalarlo desde un FTP.

```
# pkg_add -r lsof
```

En el ejemplo anterior el package correspondiente se descargará e instalará sin intervención del usuario. Si desea utilizar una réplica («mirror») de packages que no sea el sitio principal de FreeBSD solo tiene configurar la variable de entorno `PACKAGESITE` correctamente para sobreescribir el sitio predefinido. pkg_add(1) utiliza fetch(1) para descargar los ficheros, tomando como referencia varias variables de entorno, como `FTP_PASSIVE_MODE` , `FTP_PROXY` y `FTP_PASSWORD` . Quizás deba modificar alguna de estas si se encuentra detrás de un cortafuegos, o usa un proxy FTP/HTTP. Consulte fetch(3) si quiere ver una lista completa. Observe también que en el ejemplo anterior usamos `lsof` en vez de `lsof-4.56.4` . Cuando usamos la opción de descarga remota no es necesario especificar la versión del paquete. pkg_add(1) descargará automáticamente la última versión de la aplicación.

Nota

pkg_add(1) descargará la última versión de su aplicación si usa FreeBSD-CURRENT o FreeBSD-STABLE, pero si usa una versión -RELEASE, descargará la versión del package que se construyó con dicha versión. Es posible configurar esto sobreescribiendo la variable de entorno `PACKAGESITE`.

Los packages se destribuyen en formatos `.tgz` y `.tbz`. Puede encontrarlos en `ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/packages/` , o en la distribución de FreeBSD en CD-ROM. Todos los CDs en el set de 4 CDs de FreeBSD (y en el PowerPak, etc.) contienen packages en el directorio `/packages`. La estructura de los packages es similar a la del arbol de `/usr/ports` . Cada categoría tiene su propio directorio, y todos los paquetes se pueden encontrar dentro del directorio `All`.

La estructura del directorio de packageckages es idéntica a la de los ports, y funcionan como un todo para conformar el sistema de paquetes/ports.

4.4.2. Eliminar un package

Para eliminar un package instalado en el sistema utilice pkg_delete(1).

```
# pkg_delete xchat-1.7.1
```

4.4.3. Miscelánea

Toda la información de los packages se guarda en `/var/db/pkg` . Ahíl encontrará la lista completa de paquetes instalados y las descripciones de los mismos.

4.5. Uso de la colección de ports

La siguiente sección incluye las instrucciones básicas para instalar o eliminar programas mediante la Colección de Ports de su sistema.

4.5.1. Cómo obtener la Colección de Ports

Para poder instalar ports primero debe hacerse, obviamente, con la Colección de Ports —; en esencia está compuesta por `Makefiles`, parches y ficheros con la descripción de los ports y está en `/usr/ports` .

Cuando instaló su sistema FreeBSD el programa sysinstall le preguntó si querí instalar la Colección de Ports. Si contestó que no siga estas instrucciones:

Procedimiento 4.1. Mediante CVSup

Este es un método rápido de conseguir y mantener una copia de la Colección de Ports al dia mediante el protocolo CVSup. Si quiere saber más sobre CVSup consulte Uso de CVSup.

Nota

La implementación del protocolo CVSup que se incluye en FreeBSD se llama csup. Apareció en FreeBSD 6.2. Los usuarios de releases anteriores de FreeBSD puden instalar net/csup como port o package.

Asegúrese de que `/usr/ports` está vacío antes de ejecutar csup por primera vez. Si ya tiene la Colección de Ports porque la ha instalado por otros medios csup no purgará los parches de ports eliminados.

1. Ejecute `csup`:

```
# csup -L 2 -h cvsup.FreeBSD.org /usr/share/examples/cvsup/ports-supfile
```

Cambie *`cvsup.FreeBSD.org`* por algún otro servidor CVSup que tenga cerca. Consulte Réplicas CVSup (Sección 30.2.2, "Servidores"), donde encontrará una lista completa de las réplicas CVSup.

Nota

Puede usar una versión de `ports-supfile` confeccionada a su gusto, por ejemplo para evitar tener que indicarle el nombre del servidor CVSup a mano.

1. Haga lo siguiente: como `root` copie `/usr/share/examples/cvsup/ports-supfile` en otro sitio, por ejemplo `/root` o su directorio /home.

2. Edite `ports-supfile`.

3. Reemplace *`CHANGE_THIS.FreeBSD.org`* por un servidor CVSup que esté cerca de donde esté usted. Consulte Réplicas CVSup (Sección 30.2.2, "Servidores") si necesita ver una lista completa de las mismas.

4. Ejecute `csup`del siguiente modo:

```
# csup -L 2 /root/ports-supfile
```

2. Al ejecutar csup(1) descargará y aplicará todos los cambios recientes que haya sufrido la Colección de Ports, pero tenga en cuenta que no actualizará ninguno de los ports que ya tenga instalados en su sistema.

Procedimiento 4.2. Mediante portsnap

Portsnap es un método alternativo de distribuir la Colección de Ports. Se incluyó por primera vez en FreeBSD 6.0. Puede instalar portsnap(8) en versiones anteriores de FreeBSD como port (ports-mgmt/portsnap) o como package:

```
# pkg_add -r portsnap
```

1. Puede saltarse esta paso a partir de FreeBSD 6.1-RELEASE y en versiones recientes de Portsnap (port o package). `/usr/ports` se creará automáticamente la primera vez que ejecute portsnap(8). En versiones anteriores de portsnap había que crear un `/usr/ports` vacío si no existía previamente:

   ```
   # mkdir /usr/ports
   ```

2. Descargue una instantánea comprimida de la Colección de Ports en `/var/db/portsnap`. Hecho esto puede desconectar de Internet si quiere.

   ```
   # portsnap fetch
   ```

3. Si está ejecutando Portsnap por vez primera debe extraer la instantánea en `/usr/ports`:

   ```
   # portsnap extract
   ```

   Si ya tiene un `/usr/ports` y solamente está actualizando su árbol de ports ejecute lo siguiente:

   ```
   # portsnap update
   ```

Procedimiento 4.3. Mediante Sysinstall

Este método implica el uso de sysinstall para instalar la Colección de Ports.

1. Como usuario `root` ejecute `sysinstall` (`/stand/sysinstall` en cualquier versión previa a FreeBSD 5.2); del siguiente modo:

   ```
   # sysinstall
   ```

2. Posiciónese en la opción Configure y pulse Intro.

3. Seleccione la opción Distributions y pulse Intro.

4. Seleccione la opción ports y pulse la barra espaciadora.

5. Seleccione el medio de instalación deseado (CDROM, FTP, etc.)

6. Diríjase a la opción Exit y pulse Intro.

7. Pulse X para salir de sysinstall.

4.5.2. Instalación de ports

Al hablar de la Colección de Ports lo primero que hay que explicar es a qué nos referimos cuando hablamos de un «esqueleto» (skeleton). El esqueleto de un port es un conjunto mínimo de ficheros que indican a FreeBSD cómo compilar e instalar un programa. Cada esqueleto incluye:

- Un `Makefile`. Este `Makefile` contiene diversas sentencias que le indican al sistema cómo compilarlo y dónde instalarlo en su sistema.

- Un fichero `distinfo`. Este fichero contiene información sobre los ficheros que se debe descargar para poder compilar el programa, así como el identificador «checksum», que se usa para comprobar mediante md5(1) que la descarga ha sido correcta y que el la integridad del fichero está garantizada.

- Un directorio `files`. Este directorio contiene los parches necesarios para compilar e instalar el programa en su sistema FreeBSD. Básicamente los parches son pequeños ficheros que especifican cambios en ficheros concretos. Su formato es en texto plano y suelen decir cosas como «borra la línea 10» o «Cambia la línea 26 por esto». Estos parches también se conocen como «diffs» ya que se generan con diff(1).

  Es posible que este directorio tambié contenga algún otro fichero necesario para compilar e instalar el port.

- Un fichero `pkg-descr` . Es una descripción más detallada del programa. En algunas ocasiones ocupa varias líneas.

- Un fichero `pkg-plist` . Es una lista de todos los ficheros que instalará el port. También le indica al sistema de ports qué ficheros eliminar durante la desinstalación del programa.

Algunos ports tienen otros ficheros, como `pkg-message`. El sistema de ports los usa para gestionar situaciones especiales. Si desea conocer los detalles, incluso sobre los ports en general, consulte el libro FreeBSD Porter's Handbook.

El port incluye las instrucciones necesarias para obtener software a partir del código fuente, pero no incluye el código. Puede obtener el código desde un CDROM o desde Internet. El código se distribuye del modo que el autor estime oportuno. Normalmente es un fichero tar comprimido con gzip, pero puede comprimirse con otra herramienta o incluso no estar comprimido. El código del programa, venga como venga, se llama «distfile». A continuación veremos los dos métodos de instalación de un port.

Nota

Debe ser el usuario `root` para instalar ports.

Aviso

Antes de instalar cualquier port asegúrese de tener la Colección de Ports actualizada y de comprobar en `http://vuxml.freebsd.org/` la existencia de posibles problemas de seguridad que pudiera tener el port.

Puede realizar la comprobación de seguridad con portaudit antes de instalar cualquier aplicación. Esta herramienta está en la Colección de Ports (security/portaudit). Le rogamos que al menos considere ejecutar `portaudit -F` antes de instalar un port nuevo para que descargue la nueva base de datos de vulnerabilidades. Durante la comprobación diaria de seguridad actualizará la base de datos y hará una auditoría del sistema. Para más información lea las páginas de manúal de portaudit(1) y periodic(8).

La Colección de Ports asume que tiene usted conexión con Internet. Si no es así tendrá que disponer de una copia del distfile en `/usr/ports/distfiles` .

El primer paso es ubicarse en el directorio del port que desea instalar:

```
# cd /usr/ports/sysutils/lsof
```

Una vez en el directorio `lsof` puede ver el esqueleto del port. El siguiente paso es compilar el port. Solamente tiene que teclear `make` en el prompt. Una vez hecho verá algo como esto:

```
# make
>> lsof_4.57D.freebsd.tar.gz doesn't seem to exist in /usr/ports/
distfiles/.
>> Attempting to fetch from ftp://lsof.itap.purdue.edu/pub/tools/
unix/lsof/.
===>  Extracting for lsof-4.57
...
[La salida de la descompresión se ha eliminado]
...
>> Checksum OK for lsof_4.57D.freebsd.tar.gz.
```

```
===>  Patching for lsof-4.57
===>  Applying FreeBSD patches for lsof-4.57
===>  Configuring for lsof-4.57
...
[La salida de la configuración se ha eliminado]
...
===>  Building for lsof-4.57
...
[La salida de la compilación se ha eliminado]
...
#
```

Una vez que acabe la compilación se le devolverá el control del prompt. El siguiente paso es instalar el port. Para ello bastará con que añada una palabra a la orden `make`: esa palabra es `install`:

```
# make install
===>  Installing for lsof-4.57
...
[La salida de la instalación se ha eliminado]
...
===>   Generating temporary packing list
===>   Compressing manual pages for lsof-4.57
===>   Registering installation for lsof-4.57
===>  SECURITY NOTE:
      This port has installed the following binaries which execute ↺
with
      increased privileges.
#
```

Una vez que vuelva usted al prompt podrá ejecutar la aplicación que acaba de instalar. Dado que `lsof` es un programa que se ejecuta con privilegios altos se le ha mostrado una advertencia de seguridad. Durante la compilación e instalación es posible que hayan aparecido otros.

Le recomendamos que borre el directorio que contiene todos los ficheros temporales necesarios durante la compilación. No solo consume valioso espacio en disco sino que puede dar problemas cuando vaya a actualizar el port a una versión más reciente.

```
# make clean
===>  Cleaning for lsof-4.57
#
```

Nota

Puede ahorrarse teclear dos pasos si para instalar un port teclea `make install clean` en lugar de `make`, `make install` y `make clean` como tres pasos separados.

Nota

Algunas shells mantienen una caché de órdenes disponibles en los directorios que aparecen en la variable de entorno `PATH` con el fin de acelerar las operaciones de búsqueda de ejecutables de esas órdenes. Si usa una de esas shells tendrá que utilizar la orden `rehash` tras instalar un port o no podrá ejecutar aplicaciones recién instaladas. Esta orden funciona en shells como `tcsh`. Utilice `hash -r` en shells `sh`. Para más información consulte la documentación de su shell.

Algunos DVD-ROM de terceros, como el FreeBSD Toolkit de FreeBSD Mall contienen distfiles. Puede usarlos con la Colección de Ports. Monte el DVD-ROM en `/cdrom`. Si utiliza un punto de montaje diferente asigne a `CD_MOUNTPTS` el valor adecuado. Los distfiles se irán copiando automáticamente a medida que vayan siendo necesarios.

Nota

Por favor, tenga en cuenta que la licencia de unos cuantos ports muy concretos no permite su distribución en CD-ROM. Puede deberse a que es necesario rellenar un formulario de registro antes de descargarlo, porque la redistribución no esté permitida o por otra razón. Si quiere instalar un port que no está en el CD-ROM tendrá que tener salida a Internet.

El sistema de ports utiliza fetch(1) para descargar ficheros; en este proceso intervienen varias variables de entorno, como `FTP_PASSIVE_MODE` , `FTP_PROXY` , y `FTP_PASSWORD` . Si está detrás de un cortafuegos tendrá que asignar valores a una o más de estas variables, así como si necesita utilizar un proxy FTP/HTTP. Consulte fetch(3), donde ecntrará una lista detallada.

La opción `make fetch` se creó para los usuarios que no disponen de conexión contínua. Ejecute esta orden en el directorio raíz (`/usr/ports`) y se descargarán todos los ficheros necesarios. Esta orden también funciona en directorios situados más abajo, por ejemplo `/usr/ports/net` . Tenga en cuenta que si un port depende de librerías u otros ports éstos distfiles *no* se descargarán, a menos que reemplace `fetch` por `fetch-recursive` , que se encargará de descargar todas las dependencias de cada port.

Nota

Si quiere compilar todos los ports de una categoría y de una sola vez ejecutando la orden `make` en el directorio raíz, de muy similar manera que lo que se acaba de ver con `makefetch`. Tenga en cuenta que esto es bastante peligroso porque algunos ports no pueden coexistir. También se dan casos de ports que instalan instalan dos ficheros diferentes con el mismo nombre.

En algunos casos (raros) el usuario tendrá que descargar los tarball de un sitio que no es el que se guarda en la variable de entorno `MASTER_SITES` (el sitio desde el que se descargan todos los demás normalmente). Puede sobreescribir la opción `MASTER_SITES` con la siguiente orden:

```
# cd /usr/ports/directory
# make MASTER_SITE_OVERRIDE= \
ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/distfiles/ fetch
```

En este ejemplo vamos a darle a la opción `MASTER_SITES` el valor `ftp.FreeBSD.org/pub/FreeBSD/ports/distfiles/` .

Nota

Algunos ports permiten (o incluso exigen) que se le faciliten opciones de compilación para activar o desactivar partes de la aplicación que no se necesiten, determinadas opciones de seguridad, etc. Casos típicos de esto son www/mozilla, security/gpgme, and mail/sylpheed-claws. Cuando debe usted tomar este tipo de decisiones se le muestra un mensaje con las opciones disponibles.

4.5.2.1. Sobreescribir directorios por omisión de ports

Algunas veces es útil (u obligatorio) utilizar un directorio de trabajo o un «target» distinto al que tenemos por omisión. Las variables de entorno `WRKDIRPREFIX` y `PREFIX` pueden sobreescribirse según nuestra conveniencia. Veamos un ejemplo:

```
# make WRKDIRPREFIX=/usr/home/ejemplo/ports install
```

compilará el port en `/usr/home/ejemplo/ports` y lo instalará en `/usr/local`.

```
# make PREFIX=/usr/home/ejemplo/local install
```

lo compilará en `/usr/ports` y lo instalará en `/usr/home/ejemplo/local`.

Y por supuesto,

```
# make WRKDIRPREFIX=../ports PREFIX=../local install
```

combinará ambas; (ocupa demasiado para mostrarlo en una página, pero la idea general queda clara).

Puede asignar valores a estas variables del mismo modo que a cualquier otra de su entorno. Consulte la documentación de su shell para más información.

4.5.2.2. Uso de `imake`

Algunos ports que usan `imake` (un componente del Sistema X Window) no funcionan correctamente con `PREFIX` e insistirán en instalarse en `/usr/X11R6`. Del mismo modo algunos ports de Perl ignoran `PREFIX` y se instalan en el árbol de Perl. Hacer que estos ports respeten `PREFIX` es difícil y a veces imposible.

4.5.2.3. Reconfigurar ports

Al compilar ciertos ports se le presentará un menú basado en ncurses en el cual podrá elegir entre más o menos opciones de compilación. No es raro que los usuarios quieran volver a usar ese menú para añadir, quitar o cambiar opciones una vez que el port ya está compilado. Hay bastantes formas de hacerlo. Una manera es entrar al directorio que contiene el port y teclear `make config`, que hará que se le presente de nuevo el menú con las opciones que estuvieran seleccionadas previamente. Otra opción es usar la orden `make showconfig`, que le mostrará todas las opciones de configuración del port. Hay otra opción más, `make rmconfig`, que borrará todas las opciones que estuvieran seleccionadas y le permitirá por tanto empezar desde cero. Todas estas opciones y muchas más las encontrará descritas con gran detalle en ports(7).

4.5.3. Cómo desinstalar ports

Ahora que sabe instalar ports probablemente quiera saber cómo eliminarlos; puede que haya instalado alguno y posteriormente se haya dado cuenta de que ha instalado el port

incorrecto. Vamos a desinstalar el port del ejemplo anterior (que, para todos aquellos que no estaban atentos, era `lsof`). Igual que al instalar ports, lo primero que debemos hacer es ubicarnos en el directorio del port que deseamos eliminar del sistema, en nuestro caso `/usr/ports/sysutils/lsof` . Los ports se desinstalan exactamente igual que los packages; esto se explica en la sección de packages) utilizando la orden pkg_delete(1):

```
# pkg_delete lsof-4.57
```

4.5.4. Actualización de ports

Antes de nada necesita ver una lista de ports instalados de los cuales exista una nueva versión en la Colección de Ports. Utilice pkg_version(1):

```
# pkg_version -v
```

4.5.4.1. `/usr/ports/UPDATING`

Una vez actualizada la Colección de Ports (y *antes* de intentar actualizar ningún port) debe consultar `/usr/ports/UPDATING` . Este fichero describe todas las novedades, problemas que pueden encontrarse y pasos que deben seguir los usuarios al actualizar un port; hay cambios de formato, cambios de ubicación de ficheros de configuración o incompatibilidades con versiones anteriores.

Si `UPDATING` se contradice de cualquier modo con lo que lea aquí tenga muy encuenta que lo que aparezca en `UPDATING` tiene prioridad absoluta.

4.5.4.2. Actualización de ports con portupgrade

La aplicación portupgrade se diseñó para actualizar fácilmente los ports instalados en un sistema. Puede instalarla desde el port ports-mgmt/portupgrade. La instalación es como al de cualquier otro port, use la orden `make install clean` command:

```
# cd /usr/ports/ports-mgmt/portupgrade
# make install clean
```

Compruebe la lista de ports instalados con `pkgdb -F` y arregle todas las inconsistencias que aparezcan. Le recomendamos que haga esta comprobación de forma regular y siempre antes de una actualización.

Si ejecuta `portupgrade -a` portupgrade intentará actualizar todos y cada uno de los ports instalados en su sistema. Utilice el parámetro `-i` si quiere que le pida confirmación antes de actualizar cada uno de los ports.

```
# portupgrade -ai
```

Si solamente quiere actualizar una sola aplicación (y no absolutamente todos los ports) utilice la orden `portupgrade nombre-de-aplicación`. Añada el modificador `-R` si quiere

que portupgrade actualice antes todos los ports de los que depende la aplicación en cuestión.

```
# portupgrade -R firefox
```

Si quiere usar packages en lugar de ports use el modificador `-P` flag. Con esta opción portupgrade busca en los directorios locales que aparezcan en `PKG_PATH` o descarga los packages desde un sitio remoto si es que no los encuentra en local. Si es imposible encontrar los packages ni en local ni en remoto portupgrade utilizará ports. Si no quiere usar ports pase lo que pase utilice el modificador `-PP`.

```
# portupgrade -PP gnome2
```

Si quiere solamente descargar los distfiles (o los packages, usando `-P`) sin compilar ni instalar nada, use `-F`. Para más información consulte portupgrade(1).

4.5.4.3. Actualización de ports con portmanager

Portmanager es otra aplicación pensada para la actualización sencilla de ports instalados en el sistema. Puede encontrarla en ports-mgmt/portmanager:

```
# cd /usr/ports/ports-mgmt/portmanager
# make install clean
```

Puede actualizar todos los ports que tenga instalados con una sola orden:

```
# portmanager -u
```

Si usa el modificador `-ui` se le pedirá confirmación a cada paso que Portmanager vaya a dar. Portmanager también puede usarse para instalar nuevos ports. A diferencia del habitual `make install clean` actualizará todos los ports que dependan antes de compilar e instalar ese port que queramos instalar.

```
# portmanager x11/gnome2
```

Si aparecen problemas con las dependencias del port que quiere instalar puede usar Portmanager para recompilar todos ello en el orden correcto. Una vez que acabe el port que estaba dando problemas será también recompilado.

```
# portmanager graphics/gimp -f
```

Para más información consulte portmanager(1).

4.5.4.4. Actualización de ports con portmaster

Portmaster es otra aplicación para actualizar ports. Portmaster se diseñó para que utilizara las herramientas del sistema «base» (es decir, no depende de otros ports) y utiliza la

información que se almacena en el directorio `/var/db/pkg/` para determinar qué port es el que hay que actualizar. Puede encontrarlo en ports-mgmt/portmaster:

```
# cd /usr/ports/ports-mgmt/portmaster
# make install clean
```

Portmaster agrupa los ports en cuatro categorías:

- «Root ports» (no tienen dependencias, ningún port depende de ellos)
- «Trunk ports» (no tienen dependencias, otros ports dependen de ellos)
- «Branch ports» (tienen dependencias, otros ports dependen de ellos)
- «Leaf ports» (tienen dependencias, ningún port depende de ellos)

Puede ver una lista de los ports instalados y buscar actualizaciones para ellos usando el modificador `-L` :

```
# portmaster -L
===>>> Root ports (No dependencies, not depended on)
===>>> ispell-3.2.06_18
===>>> screen-4.0.3
        ===>>> New version available: screen-4.0.3_1
===>>> tcpflow-0.21_1
===>>> 7 root ports
...
===>>> Branch ports (Have dependencies, are depended on)
===>>> apache-2.2.3
        ===>>> New version available: apache-2.2.8
...
===>>> Leaf ports (Have dependencies, not depended on)
===>>> automake-1.9.6_2
===>>> bash-3.1.17
        ===>>> New version available: bash-3.2.33
...
===>>> 32 leaf ports

===>>> 137 total installed ports
        ===>>> 83 have new versions available
```

Con la siguiente orden puede actualizar todos los ports del sistema:

```
# portmaster -a
```

Nota

Por omisión Portmaster guardará una copia de seguridad (un package) de cada port antes de borrarlo. Si la instalación de la nueva

versión funciona Portmaster borrará el package. Si utiliza `-b` le dirá a Portmaster que no borre automáticamente el package. Si usa el modificador `-i` arrancará Portmaster en modo interactivo, lo que significa que le pedirá confirmación antes de actualizar cada port.

Si se encuentra con errores durante el proceso de actualización puede utilizar el modificador `-f` para actualizar o recompilar todos los ports:

```
# portmaster -af
```

También puede usar Portmaster para instalar nuevos ports en el sistema, actualizando todas sus dependencias antes de compilar e instalar el nuevo port:

```
# portmaster shells/bash
```

Por favor, consulte portmaster(8) para más información.

4.5.5. Los ports y el espacio en disco

Usar la Colección de Ports consume mucho espacio de disco según pasa el tiempo. Por culpa de la tendencia del árbol de ports a crecer sin parar le recomendamos que después de compilar e instalar sofware desde los ports recuerde limpiar los directorios temporales `work` mediante la orden `make clean`. Puede limpiar de un plumazo los directorios temporales de toda la Colección de Ports con la siguiente orden:

```
# portsclean -C
```

Rápidamente acumulará gran cantidad de viejas distribuciones de código en `distfiles`. Puede borrarlos a mano, pero también puede usar la siguiente orden para borrar todos los distfiles que no tengan relación con ningún port:

```
# portsclean -D
```

También puede borrar todos los distfiles sin relación con ningún port instalado en el sistema:

```
# portsclean -DD
```

Nota

`portsclean` forma parte de la «suite» portupgrade.

No olvide borrar los ports instalados una vez que deja de necesitarlos. Hay una herramienta muy útil para ayudar a automatizar esta tarea: ports-mgmt/pkg_cutleaves.

4.6. Después de instalar un port

Una vez que ha instalado una nueva aplicación normalmente deberáa leer toda la documentación que incluya, editar los ficheros de configuración necesarios, asegurarse de que la aplicación se ejecute al arrancar el sistema (si es un dæmon)etc.

Obviamente los pasos exactos para configurar cada aplicación son distintos de una a otra. De cualquier forma, si acaba de instalar aplicación y se pregunta «?Y ahora qué?» estos consejos puede que le ayuden:

- Use pkg_info(1) para ver qué ficheros se instalaron y dónde. Por ejemplo, si instalón el paquete Un-Package 1.0.0, lo siguiente:

  ```
  # pkg_info -L un-package-1.0.0 | less
  ```

  mostrará los ficheros instalados por el paquete. Preste especial atención a los ficheros en el directorio `man/`, que son las páginas de ayuda y los del directorio `etc/`, que contiene los ficheros de configuración; bajo el directorio `doc/` podrá encontrar información más detallada.

  Si no está seguro de qué versión del paquete ha instalado, teclée:

  ```
  # pkg_info | grep -i un-package
  ```

  y verá todos los paquetes que se llaman *un-package*. Reemplace *un-package* en la línea de órdenes que tenga que usar en su caso.

- Una vez que ha identificado las páginas de manual de la aplicación revíselas mediante man(1). Revise también los ficheros de configuración, así como cualquier otro tipo de documentación que venga con el software.

- Si la aplicación tiene sitio web visítelo. Es muy posible que encuentre más documentación, listas de preguntas frecuentes (FAQ), etc. Si no sabe cuál puede ser la dirección del sitio web examine la salida de la orden

  ```
  # pkg_info un-package-1.0.0
  ```

  a veces incluye una línea `WWW:`, que contiene la dirección del sitio web de la aplicación.

- Los ports que deben arrancar con el sistema (como los servidores de Internet) normalmente instalaran un «script» de ejemplo en `/usr/local/etc/rc.d`. Debe revisar este «script» para comprobar que todo sea correcto, editarlo o renombrarlo si fuera necesario. Consulte Cómo arrancar servicios para más información.

4.7. Ports que no funcionan

Si encuentra un port que no funciona hay varias cosas que puede hacer:

1. Consulte en la Base de datos de informes de error si ya se ha publicado un parche que lo solucione. Si es el caso puede aplicar aplicar el parche en su sistema.

2. Pida ayuda al responsable del mantenimiento (conocido como «mantainer») del port. Teclée `make maintainer` o consulte el `Makefile` del port si no sabe cuál es la dirección de correo del «mantainer». Recuerde que debe incluir el nombre y la versión del port. Envíe la línea que contiene `$FreeBSD:` al principio del `Makefile` del port y salida con el error que obtiene al intentar usar la aplicación.

   ### Nota

   Algunos ports no son responsabilidad de un individuo sino de una lista de correo. Muchas, aunque no todas, tienen un aspecto parecido a <`freebsd-nombredelalista@FreeBSD.org`>. Por favor, téngalo en cuenta cuando envie ese correo sobre el port que no funciona.

   Hay un caso especial: Los ports que tienen como «mantainer» <`freebsd-ports@FreeBSD.org`> en realidad no son responsabilidad de nadie. Los parches y el soporte, en caso de existir, vienen de la comunidad de usuarios y desarrolladores que están suscritos a la lista. Por cierto, siempre hacen falta más voluntarios.

   Si no recibe respuesta utilice send-pr(1) para enviar un informe de errores. Consulte Cómo escribir informes de error) [1]

3. *Arreglarlo.* El «Porter's Handbook» incluye información detallada sobre la infraestructura de los ports, así que puede arreglar un port roto o incluso enviar su propio port.

4. Descargue el paquete de algún sitio FTP. El servidor «principal» de la colección de paquetes está en el directorio de paquetes de `ftp.FreeBSD.org` *pero asegúrese de mirar antes en la réplica más próxima.* Esta es una forma de instalar aplicaciones más fácil y más rápida. Instale el paquete mediante pkg_add(1).

[1] Hay traducción al castellano de este artículo. Está aquí: `http://www.freebsd.org/doc/es_ES.ISO8859-1/articles/problem-reports/article`

Capítulo 5. El sistema X Window

Actualizado al servidor X11 de X.Org por Ken Tom y Marc Fonvieille.

5.1. Sinopsis

FreeBSD usa X11 para proporcionar una potente interfaz gráfica. X11 es una implementación de código abierto del sistema X Window que incluye Xorg y XFree86™. En las versiones de FreeBSD hasta FreeBSD 4.10-RELEASE y FreeBSD 5.3-RELEASE el sistema X window que se instalará por defecto es XFree86™, el servidor X11 distribuido por el proyecto XFree86™. Después de FreeBSD 5.3-RELEASE el sistema X Window pasó a ser Xorg, el servidor X11 distribuido por la Fundación X.Org.

Este capítulo cubre la instalación y configuración de X11 poniendo énfasis en Xorg. Si quiere información sobre la configuración de XFree86™ o versiones anteriores de Xorg consulte las versiones archivadas del «Handbook» de FreeBSD en `http://docs.FreeBSD.org/doc/` .

Para mas información del hardware de vídeo que X11 soporta consulte la página web de Xorg.

Después de leer este capítulo usted sabrá:

- Cuáles son los diferentes componentes del sistema X Window y como interoperan.
- Cómo instalar y configurar X11.
- Cómo instalar y usar diferentes gestores de ventanas.
- Cómo usar tipos de letra TrueType® en X11.
- Como preparar su sistema para iniciar la sesión de forma gráfica (XDM).

Antes de leer este capítulo debería:

- Saber cómo instalar software de terceros (Capítulo 4, *Instalación de aplicaciones: «packages» y ports*).

Nota

Este capítulo cubre la instalación y configuración de los servidores X11 Xorg y XFree86™. La mayoría de los ficheros de configuración,

órdenes y sintaxis son idénticos. En los casos en que haya diferencias se mostrará la sintaxis de Xorg y XFree86™.

5.2. Entender X

Usar X por primera vez puede resultar chocante para alguien familiarizado con otros entornos gráficos, como Microsoft® Windows® o Mac OS®.

No es necesario entender todos los detalles de los diferentes componentes de X y como interactúan, pero un conocimiento básico hace posible el sacarle más provecho a la potencia de X.

5.2.1. ?Por qué X?

X no es el primer sistema de ventanas escrito para UNIX® pero es el más popular de todos ellos. El equipo original que desarrolló X trabajó en otro sistema de ventanas antes de escribir X. Ese sistema se llamó «W» (de «Window» que significa ventana en inglés). X era solo la siguiente letra en el alfabeto Romano.

X se puede llamar «X» «X Window System», «X11» y de otras formas. También puede usar el término «X Windows» para describir X11 si lo que quiere es ofender a algunas personas; si quiere saber más sobre esto consulte X(7).

5.2.2. El modelo cliente/servidor de X

X fue diseñado desde el principio pensando en redes y adoptó un modelo «cliente-servidor».

En el modelo de X el «servidor X» se ejecuta en la máquina que tiene el teclado, monitor y ratón. El servidor es responsable de manejar la pantalla, la entrada de datos a través del teclado, el ratón, etc. Cada aplicación X (como una XTerm o Netscape®) es un «cliente». Un cliente manda mensajes al servidor como «Por favor dibuja una ventana en estas coordenadas» y el servidor devuelve mensajes como «El usuario ha pulsado el botón OK».

Sí solo hay un ordenador involucrado, como en una casa o una pequeña oficina, el servidor X y los clientes X seguramente se ejecutarán en el mismo sistema. Sin embargo, es perfectamente posible ejecutar el servidor X en un sistema de escritorio menos potente y ejecutar las aplicaciones X (los clientes) en, por ejemplo, la potente y cara máquina que sirve la oficina. En este panorama la comunicación entre el cliente X y el servidor tiene lugar a través de la red.

Esto confunde a algunas personas, porque la terminología es exactamente opuesta a lo que ellos esperan. Ellos esperan que el «servidor X» sea la máquina grande que hay en el sótano y que el «cliente X» sea la máquina de su mesa.

Es importante que recuerde que el servidor X es la máquina con el monitor y el teclado y que los clientes X son los programas que muestran las ventanas.

No hay nada en el protocolo que obligue a que el cliente y el servidor tengan que usar el mismo sistema operativo o incluso que estén en el mismo sistema. Es posible ejecutar un servidor X en Microsoft® Windows® o en el Mac OS® de Apple; de hecho hay varias aplicaciones libres y comerciales que hacen exactamente eso.

A partir de FreeBSD 5.3-RELEASE, el servidor X que se instala con FreeBSD es Xorg, que es libre y se distribuye bajo una licencia muy similar a la de FreeBSD. También hay servidores X comerciales para FreeBSD.

5.2.3. El gestor de ventanas

La filosofía de diseño de X es más parecida a la de UNIX®: «herramientas, no normas». Esto significa que X no impone cómo debe hacerse una tarea. En lugar de eso proporciona herramientas al usuario y es responsabilidad suya decidir qué hacer con ellas.

Esta filosofía se extiende a X; no impone como deben verse las ventanas en la pantalla, cómo moverlas con el ratón, qué teclas deberían usarse para moverse entre ventanas (por ejemplo Alt+Tab, en Microsoft® Windows®), cómo deben ser las barras de título en cada ventana, si tienen o no botón de cierre, etc.

En lugar de eso X delega esta responsabilidad en una aplicación llamada «gestor de ventanas». Hay docenas de gestores de ventanas disponibles para X: AfterStep, Blackbox, ctwm, Enlightenment, fvwm, Sawfish, twm, Window Maker, etc. Cada uno de estos gestores de ventanas tiene un aspecto diferente; algunos soportan «escritorios virtuales»; otros permiten personalizar las teclas para manejar el escritorio; otros tienen un botón de «Inicio» o algo similar; otros tienen «diseños modificables», permitiendo un cambio completo del aspecto mediante un nuevo diseño. Estos gestores de ventanas, y muchos otros, están en la categoría `x11-wm` de los ports.

Además, los entornos de escritorio KDE y GNOME tienen sus propios gestores de ventanas integrados con el escritorio.

Cada gestor de ventanas tiene diferentes mecanismos de configuración; en algunos debe escribir el fichero de configuración a mano, otros tienen herramientas gráficas para la mayor parte de la configuración; y al menos uno (Sawfish) tiene un fichero de configuración escrito en un dialecto del lenguaje Lisp.

Política de foco

Otra responsabilidad del gestor de ventanas es la «política de foco» del ratón. Todos los sistemas de ventanas necesitan algo para elegir una ventana activa que recibirá los eventos y también debieran indicar claramente qué ventana está activa.

Una política muy popular es la de «click to focus». Este es el modelo que utiliza Microsoft® Windows®, en el que una ventana se convierte en activa al recibir un clic del ratón.

X no tiene ninguna política de foco. En lugar de eso el gestor de ventanas controla qué ventana tiene el foco en todo momento. Los diferentes gestores de ventanas tienen diferentes políticas de foco. Todos soportan la política «clic-para-foco» y la mayoría de ellos también soportan otros métodos.

Las políticas de foco más populares son:

el-foco-sigue-al-ratón (`focus-follows-mouse`)
: La ventana que esta bajo el puntero del ratón es la ventana que tiene el foco. No es necesario que la ventana esté encima de las demás ventanas. El foco se cambia al apuntar a otra ventana, aunque no es necesario hacer clic en ella.

foco-descuidado `sloppy-focus`
: Esta política es una pequeña extensión de «el-foco-sigue-al-ratón». En «el-foco-sigue-al-ratón» si el ratón se mueve sobre la ventana raíz (o fondo) ninguna ventana tiene el foco y las pulsaciones del teclado se perderán. Con la política «foco-descuidado» el foco sólo cambia cuando el cursor entra en una nueva ventana y no cuando sale de la ventana actual.

click-para-foco `click-to-focus`
: La ventana activa se selecciona con un clic del ratón. La ventana se puede «levantar» y aparecer delante de las demás ventanas. Todas las pulsaciones de teclado serán redirigidas a esa ventana aunque el cursor se mueva a otra ventana.

Muchos gestores de ventanas soportan otras políticas o variaciones de estas. Consulte la documentación del propio gestor de ventanas.

5.2.4. Widgets

La idea de X de hacer herramientas en vez de normas y se extiende a los «widgets» que se ven en la pantalla de cada aplicación.

«Widget» es un término que abarca todos los objetos en el interfaz de usuario que pueden pulsarse con el ratón manipularse de alguna manera: botones, «check boxes», listas, etc. Microsoft® Windows® los llama «controles».

Microsoft® Windows® y Mac OS® de Apple tienen una política de «widgets» muy rígida. Se supone que los desarrolladores de aplicaciones comparten el aspecto. En X no se consideró imponer un estilo gráfico o un conjunto de «widgets» al que adherirse.

Como resultado no se puede esperar que las aplicaciones de X tengan un aspecto homogéneo. Hay varios conjuntos de «widgets» y variaciones muy populares como el «widget» Athena creado en el MIT, Motif® (donde los «widgets» de Microsoft® Windows® fueron modelados, con todos los bordes biselados y tres tonos de gris), OpenLook, y otros.

Las aplicaciones más modernas de X usan un conjunto de «widgets» con mejor aspecto, o Qt, utilizado por KDE, o GTK+, que usa el proyecto GNOME. Se intenta que el aspecto del escritorio de UNIX® sea parecido, lo que hace la vida más fácil al usuario novato.

5.3. Instalar X11

Xorg es la implementación de X11 por omisión en FreeBSD. Xorg está basado en código de XFree86™ 4.4RC2 y X11R6.6. La versión de Xorg que se distribuye actualmente en la Colección de Ports de FreeBSD es la 7.7.

Para compilar e instalar Xorg desde la colección de ports:

```
# cd /usr/ports/x11/xorg
# make install clean
```

Nota

Si quiere compilar e instalar Xorg entero, asegúrese de tener como mínimo 2GB de espacio disponible.

Si lo prefiere puede instalar Xorg con packages. Cuando utilice la capacidad de pkg_add(1) para instalar packages que descargue remotamente recuerde que no debe escribir el número de versión. pkg_add(1) descargará automáticamente la última versión de la aplicación..

Veamos qué es lo que hay que hacer para descargar e instalar el package de Xorg:

```
# pkg_add -r xorg
```

Nota

El anterior ejemplo instala la distribución X11 completa, es decir, servidores, clientes, tipos de letra, etc. Hay packages y ports con cada uno de esos elementos por separado.

En el resto del capítulo se explica cómo configurar X11 y cómo configurar un entorno de escritorio que le resulte productivo.

5.4. Configuración de X11

Escrito por Christopher Shumway.

5.4.1. Antes de empezar

Antes de configurar X11 necesitará la siguiente información sobre el sistema:

- Especificaciones del monitor
- Chipset de la tarjeta de vídeo
- Memoria de la tarjeta de vídeo

X11 usa las especificaciones del monitor para determinar la resolución y la frecuencia de refresco a la que funcionará. Estas especificaciones se pueden encontrar normalmente en la documentación que viene con el monitor o en la web del fabricante. Se necesitan dos rangos de números: la frecuencia de refresco horizontal y la vertical.

El chipset de la tarjeta de vídeo define qué controlador utilizará X11 para comunicarse con el hardware gráfico. Con la mayoría de chipsets es posible detectarlo automáticamente pero es útil saberlo en caso de que la detección automática no funcione.

La memoria de la tarjeta gráfica determina la resolución y la profundidad de color a la que el sistema funcionará. Es importante saberlo para que el usuario conozca las limitaciones del sistema.

5.4.2. Configurar X11

A partir de la versión 7.3 Xorg puede funcionar sin utilizar fichero de configuración; basta con teclear en el prompt:

```
% startx
```

Si no funciona, o si la configuración por omisión no es aceptable, hay que configurar X11 a mano.

Nota

Ciertos entornos de escritorio como GNOME, KDE o XFce tienen herramientas que permiten al usuario configurar fácilmente parámetros de la pantalla como la resolución. Si la configuración por omisión no sirve y tiene idea de instalar un entorno de escritorio puede seguir adelante tranquilamente con la instalación del entorno de escritorio y utilice la herramienta que el software que instale le facilite.

La configuración de X11 es un proceso que consta de múltiples pasos. El primero es generar un fichero inicial de configuración. Como superusuario teclee:

```
# Xorg -configure
```

Esto generará un fichero de configuración bastante esquemático pero completo en `/root` llamado `xorg.conf.new` (si ha utilizado su(1) o ha entrado directamente como superusuario afectará al valor que tenga la variable de directorio `$HOME`). X11 intentará probar el hardware gráfico del sistema y escribir un fichero de configuración en el que registrar los controladores de dispositivo adecuados para hacer funcionar el harware que haya detectado.

El siguiente paso es probar el fichero generado para verificar si Xorg puede funcionar con el hardware gráfico del sistema. Para ello teclee:

```
# Xorg -config xorg.conf.new
```

Si aparece una rejilla negra y gris junto con el cursor del ratón con forma de X es que la configuración ha tenido éxito. Para salir de la prueba pulse simultáneamente Ctrl+Alt+Retroceso.

Nota

Si el ratón no funciona tendrá que configurarlo antes de continuar. Consulte la Sección 2.10.10, “Configuración del ratón” en el capítulo de instalación de FreeBSD.

Después modifique el fichero `xorg.conf.new` a su gusto. Abra el fichero con un editor de texto (como emacs(1) o ee(1)). Primero añada las frecuencias del monitor. Estas suelen estar expresadas en frecuencia de refresco horizontal y vertical. Estos valores se añaden al fichero `xorg.conf.new` en la sección `"Monitor"`:

```
Section "Monitor"
        Identifier   "Monitor0"
        VendorName   "Monitor Vendor"
        ModelName    "Monitor Model"
        HorizSync    30-107
        VertRefresh  48-120
EndSection
```

Las palabras `HorizSync` y `VertRefresh` podrían no existir en el fichero de configuración. Si no existen es necesario añadirlas con la frecuencia horizontal correcta puesta después de la palabra `HorizSync` y la frecuencia vertical correcta después de la palabra `VertRefresh`. En el ejemplo anterior se pusieron las frecuencias del monitor.

X permite usar las características DPMS (Energy Star) con los monitores que lo soportan. El programa xset(1) controla el «timeout», y puede forzar los modos «standby», suspendido o apagado. Si desea habilitar las características DPMS para su monitor tendrá que añadir la siguiente línea en la sección `monitor`:

```
        Option       "DPMS"
```

Agregue a su `xorg.conf.new` la resolución por omisión y la profundidad de color deseada. Esto se define en la sección `"Screen"` :

```
Section "Screen"
        Identifier "Screen0"
        Device     "Card0"
        Monitor    "Monitor0"
        DefaultDepth 24
        SubSection "Display"
                Viewport  0 0
                Depth     24
                Modes     "1024x768"
        EndSubSection
EndSection
```

La palabra `DefaultDepth` describe la profundidad de color a la que arrancará por omisión. Esto puede sobrescribirse con el argumento `-bpp` de Xorg(1) (o XFree86(1)). La palabra `Modes` describe la resolución a la que funcionará para esa profundidad de color. Observe que sólo los modos estándar VESA están soportados por el hardware gráfico. En el ejemplo anterior la profundidad de color es 24 bits por pixel. En esta profundidad de color la resolución aceptada es 1024 por 768 pixels.

Para acabar guarde el fichero de configuración y pruébelo usando el modo de prueba descrito anteriormente

Nota

Una de las herramientas disponibles para ayudarle con el proceso de solución de problemas es el fichero de «logs», que contiene información de cada dispositivo al que el servidor X11 se conecta. Los «logs» (o bitácoras) de Xorg se pueden encontrar en `/var/log/Xorg.0.log` El nombre exacto del fichero de «log» puede variar entre `Xorg.0.log` y `Xorg.8.log` y así sucesivamente.

Si todo ha ido bien ha llegado el momento de ubicar el fichero de configuración donde Xorg(1) pueda encontrarlo. Puede dejarlo en `/etc/X11/xorg.conf` o `/usr/local/etc/X11/xorg.conf` .

```
# cp xorg.conf.new /etc/X11/xorg.conf
```

El proceso de configuración de X11 ha concluido. Arranque Xorg mediante startx(1). Puede también arrancar el servidor X11 utilizando xdm(1).

Nota

Existe también una herramienta gráfica de configuración en X11, xorgcfg(1). Permite definir de forma interactiva su configuració mediante la elección de los controladores y las opciones adecuadas. Este programa también puede usarse desde consola mediante la orden `xorgcfg -textmode` . Para más información consulte la página de manual de xorgcfg(1).

Puede usar también xorgconfig(1). Este programa es una herramienta orientada a consola que es bastante menos atractiva para el usuario, pero que puede funcionar en situaciones en las que otras herramientas no sirven.

5.4.3. Configuración avanzada

5.4.3.1. Configuración de chipsets gráficos Intel® i810

Para configurar los chipsets integrados Intel® i810 es necesario el interfaz de programación AGP `agpart` para que X11 pueda usarla. Consulte la página de manual de agp(4) para más información.

Hecho esto podrá configurar el hardware como cualquier otra tarjeta gráfica. Tenga en cuenta que si en un sistema que no tenga el controlador agp(4) intenta cargar el módulo del kernel con kldload(8) no va a funcionar. El controlador tiene necesariamente que estar en el kernel desde el mismo momento del arranque, cosa que puede obtenerse bien compilando un nuevo kernel, bien utilizando `/boot/loader.conf` .

5.4.3.2. Añadir una pantalla panorámica a nuestro sistema

A lo largo de esta sección se asumen unos mínimos de soltura con ciertos aspectos de configuración avanzada. Si Si las herramientas estándar de configuración que se han descrito previamente no le ayudan a tener una configuración funcional busque en los ficheros de log, ahí encontrará información suficiente para lograrlo. Tendrá que utilizar un editor de texto a lo largo de estas instrucciones.

Los formatos actuales de pantalla panorámica, (también conocidas como «Widescreen», WSXGA, WSXGA+, WUXGA, WXGA, WXGA+, etc, soportan formatos de 16:10 y 10:9 o aspectos-tasa que pueden ser problemáticos. Veamos algunos ejemplos de resoluciones de pantalla en formatos 16:10:

- 2560x1600
- 1920x1200
- 1680x1050
- 1440x900
- 1280x800

At some point, it will be as easy as adding one of these resolutions as a possible `Mode` in the `Section "Screen"` as such:

```
Section "Screen"
Identifier "Screen0"
Device     "Card0"
Monitor    "Monitor0"
DefaultDepth 24
SubSection "Display"
 Viewport  0 0
 Depth     24
 Modes     "1680x1050"
EndSubSection
EndSection
```

Xorg se basta y se sobra para recabar la información de resolución de su pantalla panorámica mediante la información I2C/DDC, así que sabe qué frecuencias y resoluciones puede manejar el monitor.

Si esas `ModeLines` no existen en los controladores tendrá usted que ayudar un poco a Xorg. Lea atentamente el contenido de `/var/log/Xorg.0.log` ; de él podrá extraer suficiente información como para crear un `ModeLine` que funcione. Todo lo que tiene que hacer es buscar información que se parezca a esta:

```
(II) MGA(0): Supported additional Video Mode:
(II) MGA(0): clock: 146.2 MHz   Image Size:  433 x 271 mm
(II) MGA(0): h_active: 1680  h_sync: 1784  h_sync_end 1960 ↺
h_blank_end 2240 h_border: 0
(II) MGA(0): v_active: 1050  v_sync: 1053  v_sync_end 1059 ↺
v_blanking: 1089 v_border: 0
(II) MGA(0): Ranges: V min: 48  V max: 85 Hz, H min: 30  H max: 94 ↺
kHz, PixClock max 170 MHz
```

Esta información se llama información EDID. Para crear una `ModeLine` a partir de estos datos solamente tiene que poner los números en el orden correcto:

```
ModeLine <name> <clock> <4 horiz. timings> <4 vert. timings>
```

Así que la `ModeLine` en la `Section "Monitor"` del ejemplo tendría este aspecto:

```
Section "Monitor"
Identifier      "Monitor1"
VendorName      "Bigname"
ModelName       "BestModel"
ModeLine        "1680x1050" 146.2 1680 1784 1960 2240 1050 1053 ↺
1059 1089
Option          "DPMS"
EndSection
```

Como ve, con un poco de edición hemos culminado la tarea y X debería ya poder arrancar sin problemas en su nuevo monitor panorámico.

5.5. Uso de tipos en X11

Escrito por Murray Stokely.

5.5.1. Tipos Type1

Los tipos que X11 incluye por omisión no son precisamente ideales para las aplicaciones típicas. Los tipos grandes para presentaciones aparecen dentados y con un aspecto poco profesional, mientras que los tipos pequeños en Netscape® son prácticamente ilegibles. Sin embargo hay varios tipos de letra libres Type1 de alta calidad (PostScript®) que se pueden usar con X11. Por ejemplo los tipos URW (x11-fonts/urwfonts) incluyen una versión de alta calidad de los tipos estándar Type 1 (Times Roman®, (Helvetica®, (Palatino®, y muchas otras). Las Freefonts (x11-fonts/freefonts) incluyen muchas más pero la mayoría están pensadas para usarse con software gráfico como el Gimp y no son lo bastante

completos para usarse como tipos de pantalla. Además X11 se puede configurar para usar tipos TrueType® con un esfuerzo mínimo. Para más detalles sobre esto consulte la página del manual X(7) o la sección de tipos TrueType®.

Para instalar los tipos Type1 desde los ports ejecute las siguientes órdenes:

```
# cd /usr/ports/x11-fonts/urwfonts
# make install clean
```

Proceda del mismo modo con los FreeFonts o cualesquiera otras. Para que el servidor X las detecte añada la siguiente línea al fichero de configuración de X (`/etc/X11/xorg.conf` :

```
FontPath "/usr/X11R6/lib/X11/fonts/URW/"
```

o bien desde una emulación de terminal durante una sesión X ejecute

```
% xset fp+ /usr/X11R6/lib/X11/fonts/URW
% xset fp rehash
```

Esto funcionará hasta que la sesión X se cierre a menos que lo añada a su fichero de inicio (`~/.xinitrc` en el caso de una sesión `startx` normal o `~/.xsession` si hace login desde un gestor gráfico como XDM). Una tercera forma sería usando el nuevo fichero `/usr/X11R6/etc/fonts/local.conf` : consulte la sección de anti-aliasing.

5.5.2. Tipos TrueType®

Xorg se diseñó pensando en que utilizara tipos TrueType®. Hay dos módulos diferentes que pueden habilitar esta opción. El módulo freetype se usa en en este ejemplo porque es más consistente con el resto de «back-ends» de representación. Para habilitar el módulo freetype debe añadir la siguiente línea a la sección `"Module"` del fichero `/etc/X11/xorg.conf` .

```
Load "freetype"
```

Ahora haga un directorio para los tipos TrueType® (por ejemplo `/usr/X11R6/lib/X11/fonts/TrueType`) y copie todos los tipos TrueType® en este directorio. Recuerde que los tipos TrueType® no se pueden coger directamente de un Macintosh®; tienen que estar en formato UNIX®/MS-DOS®/Windows® para usarlas con X11. Una vez que los ficheros se hayan copiado en este directorio use ttmkfdir para crear un fichero `fonts.dir` , para que el representador de tipos de X sepa que se han instalado estos ficheros nuevos. `ttmkfdir` forma parte de la Colección de Ports de FreeBSD: x11-fonts/ttmkfdir.

```
# cd /usr/local/lib/X11/fonts/TrueType
# ttmkfdir -o fonts.dir
```

Ahora añada el directorio TrueType® a la ruta de tipos. Esto es igual que lo anteriormente descrito sobre los tipos Type1. Ejecute:

```
% xset fp+ /usr/local/lib/X11/fonts/TrueType
% xset fp rehash
```

o añada una línea FontPath en su /etc/X11/xorg.conf .

Esto es todo. Ahora Netscape®, Gimp, StarOffice™ y el resto de aplicaciones X debieran reconocer los tipos TrueType® instalados. Los tipos muy pequeños (como el texto en una pantalla con alta resolución viendo una página web) y los tipos muy grandes (Dentro de StarOffice™) se verán muchísimo mejor ahora.

5.5.3. Tipos Anti-Aliasing

Actualizado por Joe Marcus Clarke.

Los «tipos anti-aliasing» han estado en X11 desde XFree86™ 4.0.2. Aunque la configuración de los tipos era incómoda antes de XFree86™ 4.3.0. A partir de XFree86™ 4.3.0 todos los tipos de X11 que se encuentren en /usr/local/lib/X11/fonts/ y ~/.fonts/ están disponibles automáticamente para aplicaciones que soporten Xft. No todas las aplicaciones soportan Xft pero muchas ya tienen soporte Xft. Ejemplos de aplicaciones que lo soportan son Qt 2.3 o superior (el «toolkit» del escritorio KDE), GTK+ 2.0 o superior (el «toolkit» del escritorio GNOME) y Mozilla 1.2 o superior.

Para controlar qué tipos «anti-aliasing» o configurar las propiedades «anti-aliasing» crée (o edite, si ya existe) el fichero /usr/local/etc/fonts/local.conf . Se pueden modificar diversas opciones avanzadas del sistema de tipos Xft usando este fichero; esta sección describe solo algunas posibilidades simples. Para más detalles consulte fonts-conf(5).

Este fichero tiene que estar en formato XML. Preste especial atención a las mayúsculas y minúsculas y asegúrese de que las etiquetas están cerradas correctamente. El fichero empieza con la típica cabecera XML seguida de una definición DOCTYPE y después la etiqueta <fontconfig>:

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
```

Como se ha dicho antes todos los tipos en /usr/local/lib/X11/fonts/ y en ~/.fonts/ están ya disponibles para las aplicaciones que soporten Xft. Si usted desea añadir otro directorio además de estos dos árboles de directorios añada una línea similar a esta en el fichero /usr/local/etc/fonts/local.conf :

```
<dir>/ruta/de/mis/tipos</dir>
```

Después de añadir los nuevos tipos, y en especial los nuevos directorios de tipos, debe ejecutar la siguiente orden para reconstruir la caché de tipos:

```
# fc-cache -f
```

Los «tipos anti-aliasing» hacen los bordes ligeramente borrosos, lo que hace que el texto muy pequeño sea más legible y elimina los escalones del texto grande pero puede causar fatiga visual aplicada al texto normal. Para excluir los tipos de menos de 14 puntos del «anti-aliasing» incluya las siguientes líneas:

```
        <match target="font">
            <test name="size" compare="less">
                <double>14</double>
            </test>
            <edit name="antialias" mode="assign">
                <bool>false</bool>
            </edit>
        </match>
        <match target="font">
            <test name="pixelsize" compare="less" qual="any">
                <double>14</double>
            </test>
            <edit mode="assign" name="antialias">
                <bool>false</bool>
            </edit>
        </match>
```

El espacio entre algunos tipos también podría ser inapropiado con «anti-aliasing», por ejemplo en el caso de KDE. Una posible solución para esto es forzar que el espacio entre los tipos sea 100. Añada las siguientes líneas:

```
       <match target="pattern" name="family">
             <test qual="any" name="family">
                 <string>fixed</string>
             </test>
             <edit name="family" mode="assign">
                 <string>mono</string>
             </edit>
         </match>
         <match target="pattern" name="family">
             <test qual="any" name="family">
                  <string>console</string>
             </test>
             <edit name="family" mode="assign">
                  <string>mono</string>
             </edit>
         </match>
```

(Esto selecciona los otros nombres comunes para «tipos fixed» como `"mono"`); añada también lo siguiente:

```
         <match target="pattern" name="family">
             <test qual="any" name="family">
                 <string>mono</string>
```

```
        </test>
        <edit name="spacing" mode="assign">
            <int>100</int>
        </edit>
    </match>
```

Algunos tipos, como Helvetica, podrían tener problemas con «anti-aliasing». Normalmente esto se manifiesta en tipos que parecen cortados verticalmente por la mitad. En el peor de los casos podría causar que el funcionamiento de aplicaciones como Mozilla se interrumpiese. Para evitar esto puede añadir lo siguiente a `local.conf` :

```
    <match target="pattern" name="family">
        <test qual="any" name="family">
            <string>Helvetica</string>
        </test>
        <edit name="family" mode="assign">
            <string>sans-serif</string>
        </edit>
    </match>
```

Una vez que haya terminado de editar `local.conf` asegúrese de que el fichero termina con la etiqueta `</fontconfig>`. Si no lo hace los cambios que haya hecho serán ignorados.

Los tipos por omisión que vienen con X11 no son los ideales para «anti-aliasing». Puede encontrar otros mucho mejores en el port x11-fonts/bitstream-vera. Este port instalará el fichero `/usr/local/etc/fonts/local.conf` si aun no se ha creado. Si el fichero existe el port creara el fichero `/usr/local/etc/fonts/local.conf-vera` . Combine el contenido de este fichero con `/usr/local/etc/fonts/local.conf` y los tipos Bitstream reemplazarán automáticamente los tipos por defecto por X11 Serif, Sans Serif y Monospaced.

Por si esto fuera poco los usuarios pueden añadir sus propios ajustes mediante sus ficheros personales `.fonts.conf` . Los usuarios que deseen hacerlo deberán crear un fichero `~/.fonts.conf` . Este fichero tiene que estar también en formato XML.

Un último punto: con pantallas LCD sería deseable un muestreo de subpixel, que consiste en configurar los componentes (separados horizontalmente) rojo, verde y azul para mejorar la resolución horizontal; los resultados pueden ser impresionantes. Para habilitarlo añada la siguiente línea a `local.conf` :

```
    <match target="font">
        <test qual="all" name="rgba">
            <const>unknown</const>
        </test>
        <edit name="rgba" mode="assign">
            <const>rgb</const>
        </edit>
    </match>
```

Nota

Dependiendo del tipo de pantalla `rgb` podría ser `bgr` o `vrgb` o `vbgr`: experimente y elija cuál de ellos funciona mejor.

Una vez concluido el proceso «anti-aliasing» se debería activar la próxima vez que arranque X, aunque los programas necesitan saber como utilizarlo. En este momento el «toolkit» Qt es capaz de hacerlo, así que KDE entero puede usar «tipos anti-aliasing». GTK+ y GNOME también pueden usar «tipos anti-aliasing» mediante el «applet» «Font» (consulte Sección 5.7.1.3, “Tipos «Anti-aliased» en GNOME” para más detalle). Por omisión Mozilla 1.2 y versiones siguientes usarán automáticamente «tipos anti-aliasing». Para deshabilitarlos recompile Mozilla usando el parámetro `-DWITHOUT_XFT` .

5.6. El gestor de pantalla X

Escrito por Seth Kingsley.

5.6.1. Descripción

El gestor de pantalla X (XDM) es una parte opcional del sistema X Window que se usa para la gestión de sesiones. Esto es útil en diferentes situaciones, como «Terminales X» con escritorios mínimos y grandes servidores de red. Como X Window System es independiente del protocolo y de la red hay muchas configuraciones posibles para ejecutar clientes y servidores X en diferentes equipos conectados a una red. XDM dispone de un interfaz gráfico para elegir a qué pantalla se quiere conectar y pedir la información de autentificación como el nombre de usuario y le contraseña.

XDM ac6túa como proveedor de la misma funcionalidad que getty(8) (consulte Sección 24.3.2, “Configuración” para más detalles). Esto quiere decir que se encarga de las entradas al sistema en la pantalla a la que está conectado y arranca el gestor de sesiones para que lo utilice el usuario (normalmente un gestor de ventanas X). XDM espera a que el programa termine, envía la señal de que el usuario ha terminado y que se le debería desconectar de la pantalla. En este punto XDM puede mostrar las pantallas de entrada al sistema y de selección de pantalla para que el siguiente usuario intente acceder al sistema.

5.6.2. Uso de XDM

El dæmon XDM está en `/usr/local/bin/xdm` . Este programa se puede arrancar en cualquier momento como `root` y empezará a gestionar la pantalla X en la máquina local. Si se quiere que XDM arranque cada vez que la máquina se encienda puede hacerlo añadiendo

una entrada a `/etc/ttys` . Para más información sobre el formato y uso de este fichero consulte la Sección 24.3.2.1, “Agregando una entrada a `/etc/ttys`”. Por omisión hay una línea en `/etc/ttys` que permite arrancar el dæmon XDM en una terminal virtual:

```
ttyv8   "/usr/local/bin/xdm -nodaemon"  xterm   off secure
```

Por omisión esta entrada está deshabilitada; para habilitarla cambie el campo 5 de `off` a `on` y reinicie init(8) usando los pasos descritos en la Sección 24.3.2.2, “Forzar `init` a que relea `/etc/ttys`”. El primer campo es el nombre de la terminal que que gestiona el programa, `ttyv8`. Es decir, XDM empezará a ejecutarse en la terminal virtual número 9.

5.6.3. Configuración de XDM

El directorio de configuración de XDM está en `/usr/local/lib/X11/xdm` . En este directorio hay varios ficheros que se usan para intervenir en el comportamiento y la apariencia de XDM. Normalmente tendrá estos ficheros:

Fichero	Descripción
`Xaccess`	Reglas de acceso para clientes.
`Xresources`	Recursos de X por omisión.
`Xservers`	Lista de pantallas locales y remotas a gestionar.
`Xsession`	«script» por omisión para el acceso al sistema.
`Xsetup_*`	«script» para lanzar aplicaciones antes del acceso al sistema.
`xdm-config`	Configuración de todas las pantallas de este equipo.
`xdm-errors`	Errores generados por el programa servidor.
`xdm-pid`	El ID de proceso del XDM que está funcionando.

En este directorio también hay unos cuantos programas y «scripts» que se usan para arrancar el escritorio cuando XDM se está ejecutando. El objetivo de cada uno de estos ficheros será descrito brevemente. La sintaxis exacta y el uso de todos estos ficheros se describe en xdm(1).

La configuración por omisión es una simple ventana rectangular de acceso al sistema con el nombre de la máquina en la parte superior en un tipo de letra grande y unos campos para que introduzca«Login:» y «Password:» debajo. Este es un buen punto de partida para aprender a cambiar el aspecto de las pantallas de XDM.

5.6.3.1. Xaccess

El protocolo para conectar con las pantallas controladas por XDM se llama «X Display Manager Connection Protocol» (XDMCP). Este fichero es un conjunto de reglas para controlar las conexiones XDMCP desde otros equipos. Por defecto permite a cualquier cliente conectar, pero eso no importa a menos que `xdm-config` se cambie para escuchar conexiones remotas. La configuración básica no permite conexiones desde ningún cliente.

5.6.3.2. Xresources

Este es el fichero de opciones por defecto para el selector de pantalla y acceso al sistema. Aquí es donde se puede modificar la apariencia del programa de «login». El formato es idéntico al fichero de opciones por defecto descrito en la documentación de X11.

5.6.3.3. Xservers

Una lista de pantallas remotas entre las que elegir.

5.6.3.4. Xsession

Este es el «script» de sesión que por defecto XDM arrancará después de que el usuario acceda al sistema. Normalmente cada usuario tendrá una versión personalizada del «script» de sesión en `~/.xsession` que sobreescribe el «script» por defecto.

5.6.3.5. Xsetup_*

Estos se arrancarán automáticamente antes de que se muestre el interfaz de selección o de acceso al sistema. Hay un «script» por cada gestor de ventanas disponible llamados `Xsetup_` seguido por el número de pantalla local (por ejemplo `Xsetup_0`). Normalmente estos «scripts» arrancan uno o dos programas en segundo plano, por ejemplo `xconsole`.

5.6.3.6. xdm-config

Este fichero contiene las opciones por defecto aplicables a todas las pantallas que gestiona la instalación.

5.6.3.7. xdm-errors

Este fichero contiene la salida de los servidores X que XDM intente arrancar. Si alguna pantalla que XDM intente arrancar se cuelga por alguna razón este es un buen sitio para buscar los mensajes de error. Estos mensajes también se escriben en el fichero `~/.xsession-errors` del usuario en cada sesión.

5.6.4. Ejecutar un servidor de pantalla a través de la red

Para que otros clientes puedan conectar al servidor de pantalla edite las reglas de acceso y habilite la escucha en la conexión. Por omisión se aplican restricciones. Si quiere que XDM escuche conexiones comente la siguiente línea en el fichero `xdm-config`:

```
! SECURITY: do not listen for XDMCP or Chooser requests
! Comment out this line if you want to manage X terminals with xdm
DisplayManager.requestPort:     0
```

y después reinicie XDM. Recuerde que los comentarios en los ficheros de opciones por defecto comienzan con el carácter«!» en vez del típico «#». Puede aplicar controles más restrictivos. Consulte las opciones de ejemplo en `Xaccess` y la página de manual de xdm(1).

5.6.5. Alternativas a XDM

Hay algunas alternativas al programa por defecto XDM. Una de ellas, kdm (que se incluye con KDE) se describe más adelante en este capítulo. kdm ofrece muchas ventajas visuales y funcionalidad para permitir a los usuarios elegir su gestor de ventanas en el momento del acceso al sistema.

5.7. Entornos de escritorio

Escrito por Valentino Vaschetto.

Está sección describe los diferentes entornos de escritorio disponibles para X en FreeBSD. Un «entorno de escritorio» puede ser cualquier cosa que vaya desde un simple gestor de ventanas hasta una completa «suite» de aplicaciones de escritorio como KDE o GNOME.

5.7.1. GNOME

5.7.1.1. Acerca de GNOME

GNOME es un entorno de escritorio amigable que permite a los usuarios una configuración fácil de sus sistemas. GNOME incluye un panel (para iniciar las aplicaciones y mostrar su estado), un escritorio (donde pueden colocarse los datos y las aplicaciones), un conjunto de herramientas y aplicaciones de escritorio estándar y un conjunto de convenciones para facilitar la creación de aplicaciones que sean consistentes y colaboren unas con otras. Los usuarios de otros sistemas operativos o entornos pueden sentirse como en casa utilizando el poderoso entorno gráfico GNOME. Encontrará más información sobre GNOME en FreeBSD en la página del Proyecto FreeBSD GNOME.

5.7.1.2. Instación de GNOME

GNOME puede instalarse fácilmente utilizando packages precompilados o desde la Colección de Ports:

Para instalar el paquete de GNOME desde la red, simplemente teclee:

```
# pkg_add -r gnome2
```

Para compilar GNOME a partir del código fuente, es decir, como port:

```
# cd /usr/ports/x11/gnome2
# make install clean
```

Una vez GNOME esté instalado debe indicarle al servidor X como iniciar GNOME para convertirlo en el gestor de ventanas predeterminado.

La manera más fácil de arrancar GNOME es con GDM, el gestor de pantalla de GNOME. GDM, que se instala con el escritorio GNOME al ser parte del mismo (aunque venga desactivado por omisión) se activa añadiendo `gdm_enable="YES"` a `/etc/rc.conf` . Tras reiniciar, GNOME arrancará automáticamente al acceder al sistema; no es necesario configurar nada más.

Puede arrancar GNOME también desde la línea de órdenes configurando adecuadamente un fichero llamado `.xinitrc`. Si ya tiene un `.xinitrc` personalizado tendrá que reemplazar la línea que arranca el gestor de ventanas anterior por otra que contenga /usr/local/bin/gnome-session. Si no ha modificado el fichero previamente basta con que teclée lo siguiente:

```
% echo "/usr/local/bin/gnome-session" > ~/.xinitrc
```

Después teclée `startx` y debería iniciarse el entorno de escritorio GNOME.

Nota

Si utiliza un gestor de pantalla antíguo, como XDM, lo anteriormente descrito no funcionará. Cree un fichero ejecutable `.xsession` con la misma orden en él. Edite el fichero y sustituya la orden para iniciar su actual gestor de ventanas por /usr/local/bin/gnome-session:

```
% echo "#!/bin/sh" > ~/.xsession
% echo "/usr/local/bin/gnome-session" >> ~/.xsession
% chmod +x ~/.xsession
```

Otra opción es configurar el gestor de pantalla para que permita seleccionar el gestor de ventanas durante el acceso; la sección de detalles de KDE explica cómo hacerlo para kdm, el gestor de pantalla de KDE.

5.7.1.3. Tipos «Anti-aliased» en GNOME

X11 soporta «anti-aliasing» a través de la extensión «RENDER». Gtk+ 2.0 y superiores (el «toolkit» utilizado por GNOME) disponen de esta funcionalidad. La configuración del «anti-aliasing» se describe en la Sección 5.5.3, “Tipos Anti-Aliasing”. Una vez configurado el software es posible utilizar «anti-aliasing» en el entorno de escritorio GNOME. Sólamente

tiene que ir a Applications → Desktop Preferences → Font, y seleccione entre Best shapes, Best contrast, o Subpixel smoothing (LCDs). En el caso de una aplicación Gtk+ que no forme parte del escritorio GNOME asigne a la variable de entorno `GDK_USE_XFT` un valor de 1 antes de arrancar el programa.

5.7.2. KDE

5.7.2.1. Acerca de KDE

KDE es un entorno de escritorio contemporáneo y fácil de usar. Veamos alguna de las cosas que KDE brinda al usuario:

- Un bello escritorio contemporáneo.
- Un escritorio completamente transparente en red.
- Un sistema de ayuda integrado que facilita un acceso adecuado y consistente a la ayuda durante el uso del escritorio KDE y sus aplicaciones.
- Un aspecto consistente de todas las aplicaciones KDE.
- Menús y barras de herramientas estándares, teclas programables, esquemas de color, etc.
- Internacionalización: KDE está disponible en más de 40 idiomas.
- Un diálogo de configuración del escritorio centralizado y consistente.
- Un gran número de útiles aplicaciones KDE.

KDE incluye un navegador web llamado Konqueror, que es un sólido competidor del resto de navegadores web existentes en sistemas UNIX®. Puede encontrar más información sobre KDE en el sitio web de KDE. Si quiere información específica y recursos sobre KDE en FreeBSD consulte el sitio web del equipo de KDE en FreeBSD.

Hay dos versiones de KDE en FreeBSD: la versión 3 lleva ya bastante tiempo en el sistema y es muy estable en todos los sentidos. La versión 4, la nueva generación de KDE, también está en los ports. Si quiere puede instalar ambas versiones simultáneamente.

5.7.2.2. Instalación de KDE

Igual que sucede con GNOME o cualquier otro entorno de escritorio puede instalarlo como package o desde la Colección de Ports:

Si quiere instalar como package KDE3 (desde la red) escriba:

```
# pkg_add -r kde
```

Si quiere instalar como package KDE4 (desde la red) escriba:

```
# pkg_add -r kde4
```

pkg_add(1) descargará automáticamente la versión más reciente de la aplicación.

Si lo que desea es compilar KDE3 desde el código fuente utilice el árbol de ports:

```
# cd /usr/ports/x11/kde3
# make install clean
```

Si lo que desea es compilar KDE4 desde el código fuente utilice el árbol de ports:

```
# cd /usr/ports/x11/kde4
# make install clean
```

Una vez instalado KDE debe indicarle al servidor X cómo iniciar la aplicación para que se convierta en el gestor de ventanas por defecto. Edite `.xinitrc`:

Para KDE3:

```
% echo "exec startkde" > ~/.xinitrc
```

Para KDE4:

```
% echo "exec /usr/local/kde4/bin/startkde" > ~/.xinitrc
```

A partir de ahora cuando se llame al sistema X Window con `startx` el escritorio será KDE.

Si utiliza un gestor de pantalla, como por ejemplo xdm, la configuración es ligeramente diferente. En vez de lo antedicho edite `.xsession`. Las instrucciones para kdm se decribirán más adelante en este capítulo.

5.7.3. Más detalles sobre KDE

Una vez KDE está instalado en el sistema la mayoría de las cosas se pueden ir descubriendo utilizando las páginas de ayuda o sencillamente a golpe de ratón en los diversos menús. Los usuarios de Windows® y Mac® se sentiráan como en casa.

La mejor referencia para KDE es la documentación «on-line». KDE incluye su propio navegador, Konqueror, docenas de útiles aplicaciones y una extensa documentación. El resto de esta sección trata sobre cuestiones técnicas que son difíciles de aprender mediante una exploración al azar.

5.7.3.1. El gestor de pantalla de KDE

El administrador de un sistemas multiusuario puede querer ofrecer a sus usuarios una pantalla de bienvenido que se les presente al acceder al sistema. Tal y como se ha explicado previamente puede usar XDM. De todos modos KDE incluye su propia alternativa,

alternative, kdm, diseñado pensando en ser más atractivo y tener más opciones que ofrecer en el momento del acceso al sistema. Más concretamente ofrece a los usuarios la posibilidad de elegir mediante un menú qué entorno de escritorio (KDE, GNOME, o cualquier otro) quiere ejecutar el usuario una vez validado.

Si quiere activar kdm debe modificar la entrada de `ttyv8` en `/etc/ttys` . La línea debe quedar del siguiente modo:

Si va a usar KDE3:

```
ttyv8 "/usr/local/bin/kdm -nodaemon" xterm on secure
```

En el caso de KDE4:

```
ttyv8 "/usr/local/kde4/bin/kdm -nodaemon" xterm on secure
```

5.7.4. XFce

5.7.4.1. Acerca de XFce

XFce es un entorno de escritorio basado en el «toolkit» GTK utilizado por GNOME pero es mucho más ligero y está pensado para aquellos que quieran un escritorio sencillo, eficiente y fácil de utlizar y configurar. Visualmente es muy parecido a CDE, que podemos encontrar en sistemas UNIX® comerciales. Alguna de las caracterésticas de XFce son:

- Un escritorio sencillo y fácil de utilizar.
- Completamente configurable utilizando el ratón, con arrastrar y soltar, etc.
- Panel principal similar al de CDE, con menús, «applets» y accesos directos a las aplicaciones.
- Gestor de ventanas integrado, gestor de ficheros, gestor de sonido, módulo de compatibilidad con GNOME, y muchas cosas más.
- Aspecto configurable (utiliza GTK).
- Rápido, ligero y eficiente: ideal para máquinas viejas, lentas o máquinas con memoria limitada.

Puede encontrar más información sobre XFce en el sitio web de XFce.

5.7.4.2. Instalar XFce

En el momento de escribir este texto existe un paquete binario para XFce. Para instalarlo escriba:

```
# pkg_add -r xfce4
```

También puede compilarlo desde el código gracias a la Colección de Ports:

```
# cd /usr/ports/x11-wm/xfce4
# make install clean
```

Para indicar al servidor X que use XFce la próxima vez que arranque simplemente escriba lo siguiente:

```
% echo "/usr/X11R6/bin/startxfce4" > ~/.xinitrc
```

La próxima vez que arranque X XFce será el escritorio. Como antes, si se usa un gestor de pantalla como `xdm` cree un `.xsession` tal y como se ha descrito en la anterior sección en GNOME, pero con la orden `/usr/local/bin/startxfce4` ; puede configurar el gestor de pantalla para permitir elegir el escritorio en el momento de acceder al sistema tal y como se ha explicado en la sección de kdm.

Parte II. Tareas comunes

Una vez tratados los aspectos más básicos, pasaremos a detallar alguna de las características más utilizadas de FreeBSD:

- Introducción a algunas de las aplicaciones de escritorio más útiles y extendidas: navegadores, herramientas de productividad, visores de documentos, etc.
- Introducción a algunas de las herramientas multimedia de FreeBSD.
- Explicación de cómo crear un kernel personalizado en FreeBSD, lo que le permitirá añadir nuevas funcionalidades al sistema.
- Descripción en detalle del sistema de impresión, tanto en entornos de escritorio como en impresoras conectadas en red.
- Cómo ejecutar aplicaciones de Linux® en FreeBSD.

Algunos de los capítulos recomiendan lecturas de capítulos anteriores, lo que se indica al inicio de cada capítulo.

Tabla de contenidos

6. Aplicaciones de escritorio 199
6.1. Sinopsis 199
6.2. Navegadores 200
6.3. Productividad 203
6.4. Visores de documentos 206
6.5. Finanzas 208
6.6. Resumen 210
7. Multimedia 211
7.1. Sinopsis 211
7.2. Configuración de la tarjeta de sonido 212
7.3. Sonido MP3 216
7.4. Reproducción de vídeo 219
7.5. Configuración de tarjetas de TV 229
7.6. Escáneres de imágenes 230
8. Configuración del kernel de FreeBSD 237
8.1. Sinopsis 237
8.2. ?Qué razones hay para compilar un kernel personalizado? 237
8.3. Inventario de hardware del sistema 238
8.4. Controladores del kernel, subsistemas y módulos 240
8.5. Compilación e instalación de un kernel personalizado 240
8.6. El fichero de configuración 244
8.7. Qué hacer si algo va mal 260
9. Imprimir 263
9.1. Sinopsis 263
9.2. Introducción 264
9.3. Configuración básica 265
9.4. Configuración avanzada de impresoras 265
9.5. Cómo utilizar impresoras 265
9.6. Alternativas a LPD 265
9.7. Solución de problemas 265
10. Compatibilidad binaria con Linux 267
10.1. Sinopsis 267
10.2. Instalación 268
10.3. Instalación de Mathematica® 272
10.4. Instalación de Maple™ 274
10.5. Instalación de MATLAB® 276
10.6. Instalación de Oracle® 280
10.7. Instalación de SAP® R/3® 284
10.8. Temas avanzados 308

Capítulo 6. Aplicaciones de escritorio

Escrito por Christophe Juniet.

6.1. Sinopsis

FreeBSD puede ejecutar gran número de aplicaciones de escritorio, como navegadores y procesadores de texto. La mayoría de ellas están disponibles como packages o se pueden compilar automáticamente desde la colección de ports. Muchos usuarios nuevos esperan encontrar este tipo de aplicaciones en su escritorio. Este capítulo mostrará cómo instalar de forma fácil algunas aplicaciones populares, bien como packages o desde la colección de ports.

Tenga en cuenta que cuando se instalan programas desde los ports se compilan desde el código fuente. Esto puede tardar muchísimo tiempo, dependiendo de qué esté compilando y la potencia de proceso de su máquina. Si compilar una o más aplicaciones le requiere una cantidad de tiempo desmesurada para sus necesidades recuerde que puede instalar la mayoría de programas de la Colección de Ports como packages precompilados.

FreeBSD tiene compatibilidad binaria con Linux®, así que muchas aplicaciones originariamente desarrolladas para Linux existen también para su escritorio. Le recomendamos encarecidamente que lea el Capítulo 10, *Compatibilidad binaria con Linux* antes instalar aplicaciones Linux. Muchos de los ports que usan la compatibilidad binaria de Linux empiezan por «linux-». Recuérdelo cuando busque un port en particular, por ejemplo con whereis(1). En el siguiente texto se asume que ha habilitado usted la compatibilidad binaria de Linux antes de instalar aplicaciones de Linux.

Estas son las categorías que cubre este capítulo:

- Navegadores (como Mozilla, Netscape®, Opera, Firefox, Konqueror)
- Productividad (como KOffice, AbiWord, The GIMP, OpenOffice.org)
- Visores de documentos (como Acrobat Reader®, gv, Xpdf, GQview)
- Finanzas (como GnuCash, Gnumeric, Abacus)

Antes de leer este capítulo, usted debería:

- Saber como instalar software de terceros (Capítulo 4, *Instalación de aplicaciones: «packages» y ports*).
- Saber como instalar software de Linux (Capítulo 10, *Compatibilidad binaria con Linux*).

Si quiere más información sobre cómo conseguir un entorno un entorno multimedia en FreeBSD consulte el Capítulo 7, *Multimedia*. Si quiere configurar y usar el correo electrónico consulte el Capítulo 27, *Correo Electrónico*.

6.2. Navegadores

FreeBSD no trae ningún navegador preinstalado, pero en el directorio www de la colección de ports tiene muchos navegadores listos para instalar. Si no tiene tiempo para compilarlo todo (esto puede requerir muchísimo tiempo en algunos casos) muchos de ellos están disponibles como packages.

KDE y GNOME ya proporcionan navegadores HTML. Consulte la Sección 5.7, "Entornos de escritorio" para más información sobre cómo configurar estos escritorios.

Si busca navegadores ligeros tiene en la colección de ports www/dillo, www/links, o www/w3m.

Esta sección cubre estas aplicaciones:

Nombre de aplicación	Recursos necesarios	Instalación desde los Ports	Dependencias importantes
Mozilla	pesado	pesado	Gtk+
Opera	ligero	ligero	Versiones para FreeBSD y Linux disponibles. La versión de Linux depende de la compatibiliad binaria de Linux y de linux-openmotif.
Firefox	medio	pesado	Gtk+
Konqueror	medio	pesado	KDE Librerís

6.2.1. Mozilla

Mozilla es una navegador moderno y estable que dispone de soporte completo en FreeBSD: dispone de un motor de presentación de HTML muy ajustado a los estándares, tiene cliente de correo electrónico, lector de noticias e incluso ofrece un editor de páginas HTML por si quiere crear páginas web. Los usuarios de Netscape® encontrarán semejanzas con la suite Communicator, dado que ambos navegadores comparten parte de la base de desarrollo.

En máquinas lentas, con una CPU de velocidad inferior a 233MHz o con menos de 64MB de RAM, Mozilla puede consumir demasiados recursos para poder utilizarse. Puede usar el navegador Opera, mucho más ligero, que se describe más adelante en este mismo capítulo.

Si no puede o no quiere compilar Mozilla por algún motivo, el equipo de FreeBSD GNOME ya lo ha hecho por usted. Instale el package desde la red del siguiente modo:

```
# pkg_add -r mozilla
```

Si no hay package disponible y tiene suficiente tiempo y espacio en el disco puede obtener el código de Mozilla, compilarlo e instalarlo en su sistema. Esto se hace con:

```
# cd /usr/ports/www/mozilla
# make install clean
```

El port de Mozilla se asegura de que se inicialice adecuadamente ejecutando la configuración del registro «chrome» con privilegios de `root`, aunque si quiere instalar algún añadido como «mouse gestures» (gestos del ratón) tendrá que ejecutar que correr Mozilla como `root` para que se instale correctamente.

Una vez efectuada la instalación de Mozilla no es necesario ser `root` por más tiempo. Puede arrancar Mozilla como navegador escribiendo:

```
% mozilla
```

Puede arrancarlo directamente como lector de correo y noticias del siguiente modo:

```
% mozilla -mail
```

6.2.2. Firefox

Firefox es un navegador de nueva generación basado en el código fuente de Mozilla. Mozilla es una suite completa de aplicaciones en la que encontrará un navegador, un cliente de correo, un cliente de chat y muchas otras. Firefox es solamente un navegador, lo que hace que sea mucho más pequeño y ligero.

Puede instalarlo como package del siguiente modo:

```
# pkg_add -r firefox
```

Si prefiere compilar el código fuente e instalarlo desde los ports puede hacer esto otro:

```
# cd /usr/ports/www/firefox
# make install clean
```

6.2.3. Firefox, Mozilla y el «plugin» Java™

Nota

En esta sección y en la siguiente asumiremos que ya ha instalado Firefox o Mozilla.

La FreeBSD Foundation disponde una licencia de Sun Microsystems que le permite distribuir binarios FreeBSD del Java Runtime Environment (JRE™) y del Java Development Kit (JDK™). Puede descargar packages binarios para FreeBSD desde el sitio webde la FreeBSD Foundation.

Para añadir soporte Java™ a Firefox o Mozilla tiene que instalar el port the java/javavmwrapper. Después descargue el package Diablo JRE™ desde `http://www.freebsdfoundation.org/downloads/java.shtml` e instálelo con pkg_add(1).

Arranque su navegador, introduzca `about:plugins` en la barra de navegación y pulse Intro. Verá una página con una lista de los «plugins» que tenga instalados; el «plugin» Java™ debería ser uno de los que aparezcan. Si no es así ejecute (como `root`) lo siguiente

```
# ln -s /usr/local/diablo-jre1.6.0/plugin/i386/ns7/libjavaplugin_oji.↺
so \
  /usr/local/lib/browser_plugins/
```

y reinicie su navegador.

6.2.4. Opera

Opera es un navegador muy rápido, completo y que cumple con los estándares. Hay dos versiones: la versión «nativa» de FreeBSD y la versión que se ejecuta bajo la emulación de Linux. Para cada sistema operativo hay una versión gratuita que muestra publicidad y otra sin publicidad que se puede comprar en la página web de Opera.

Para navegar por la web con la versión para FreeBSD de Opera tendrá que instalar el package:

```
# pkg_add -r opera
```

Algunos sitios FTP no tienen todos los paquetes, pero puede instalarlo desde la Colección de Ports:

```
# cd /usr/ports/www/opera
# make install clean
```

Si quiere instalar la versión de Linux de Opera ponga `linux-opera` en lugar de `opera` en los atnteriores ejemplos. La versión de Linux es útil en situaciones donde se necesitan

«plugins» de los que sólo hay versiones para Linux, como Adobe Acrobat Reader®. En todos los demás aspectos las versiones de FreeBSD y Linux son iguales.

6.2.5. Konqueror

Konqueror forma parte de KDE pero también se puede instalar y usar independienmente de KDE instalando x11/kdebase3. Konqueror es mucho más que un navegador, también es gestor de ficheros y visor multimedia.

Konqueror también trae un conjunto de «plugins», disponibles en misc/konq-plugins.

Konqueror también soporta Flash™, hay un tutorial en `http://freebsd.kde.org/howto.php` .

6.3. Productividad

Cuando se trata de productividad, normalmente los nuevos usuarios piensan en una buena suite ofimática o un procesador de textos amigable. Mientras que algunos entornos de escritorio como KDE ofrecen una suite ofimática propia, no hay una aplicación por defecto. FreeBSD permite usar todo lo que se necesite sin importar qué entorno de escritorio use.

Esta sección cubre estas aplicaciones:

Nombre de la aplicación	Recursos necesarios	Instalación desde los ports	Dependencias importantes
KOffice	ligero	pesado	KDE
AbiWord	ligero	ligero	Gtk+ o GNOME
The Gimp	ligero	pesado	Gtk+
OpenOffice.org	pesado	enorme	GCC 3.1, JDK™ 1.3, Mozilla

6.3.1. KOffice

La comunidad de KDE ha dotado su entorno de escritorio con una suite ofimática totalmente independiente de KDE. Incluye los 4 componentes estándar que se pueden encontrar en otras suites ofimáticas. KWord es el procesador de textos, KSpread es la hoja de cálculo, KPresenter se encarga de las presentaciones y Kontour le permite dibujar documentos gráficos.

Antes de instalar el último KOffice asegúrese de que tiene una versión actualizada de KDE.

Para instalar KOffice como un paquete ejecute la siguiente orden:

```
# pkg_add -r koffice
```

Si el paquete no está disponible puede usar la colección de ports. Por ejemplo, para instalar KOffice para KDE3 haga lo siguiente:

```
# cd /usr/ports/editors/koffice-kde3
# make install clean
```

6.3.2. AbiWord

AbiWord es un procesador de textos libre con un aspecto parecido a Microsoft® Word. Es útil para escribir documentos, cartas, informes, notas, etc. Es muy rápido, tiene muchas opciones y es muy amigable.

AbiWord puede importar o exportar muchos formatos de fichero, incluyendo algunos propietarios como Microsoft `.doc`.

AbiWord está disponible como package. Si quiere instalarlo haga lo siguiente:

```
# pkg_add -r AbiWord2
```

Si el package no está disponible por el motivo que fuere puede instalarlo desde la colección de Ports. La colección de ports debería estar más actualizada:

```
# cd /usr/ports/editors/AbiWord2
# make install clean
```

6.3.3. The GIMP

GIMP es un programa de manipulación de imágenes muy sofisticado que le permitirá también crearlas y retocarlas. Puede usarlo como un programa de dibujo sencillo o como una sofisticada herramienta de retoque fotográfico. Admite una gran cantidad de «plug-ins» y tiene opciones tan interesantes como una interfaz para «scripts». GIMP puede leer y escribir un gran número de formatos de fichero. Soporta escáners y tabletas gráficas.

Instale el package mediante la siguiente orden:

```
# pkg_add -r gimp
```

Si su servidor FTP no tiene este package puede usar la colección de ports. El directorio graphics de la colección de ports también contiene The Gimp Manual. Si quiere instalarlo haga lo siguiente:

```
# cd /usr/ports/graphics/gimp
# make install clean
# cd /usr/ports/graphics/gimp-manual-pdf
```

```
# make install clean
```

Nota

El directorio graphics de la colección de ports tiene la versión de desarrollo de GIMP en graphics/gimp-devel. Hay una versión en HTML de The Gimp Manual en graphics/gimp-manual-html.

6.3.4. OpenOffice.org

OpenOffice.org incluye todas las aplicaciónes esenciales en una completa suite ofimática: un procesador de textos, una hoja de cálculo, un gestor de presentaciones y un programa de dibujo. Su interfaz de usuario es muy parecida a otras suites ofimáticas y puede importar y exportar diversos formatos muy extendidos. Está disponible en diversos idiomas; el proceso de internacionalización incluye interfaces, correctores ortográficos y diccionarios.

El procesador de textos de OpenOffice.org usa un fichero XML nativo para aumentar la portabilidad y la eficiencia. El programa de hojas de cálculo tiene un lenguaje de macros y puede usarse desde bases de datos externas. OpenOffice.org es estable y hay versiones nativas para Windows®, Solaris™, Linux, FreeBSD, y Mac OS® X. Tiene más información sobre OpenOffice.org en la página web de OpenOffice. Si quiere información específica para FreeBSD, o si quiere bajar directamente los packages use la página web de FreeBSD OpenOffice Porting Team.

Para instalar OpenOffice.org ejecute:

```
# pkg_add -r openoffice
```

Nota

Si usa una versión -RELEASE de FreeBSD todo lo antedicho funcionará. Si usa cualquier otra versión visite el sitio web del FreeBSD OpenOffice.org Porting Team, desde el cual podrá descargar e instalar el package más adecuado e instalarlo mediante pkg_add(1). En este sitio web encontrará tanto una versión current como la de desarrollo.

Una vez instalado el package escriba la siguiente orden para ejecutar OpenOffice.org:

```
% openoffice.org
```

Nota

En el primer arranque se le harán diversas preguntas se creará un directorio `.openoffice.org2` en su «home».

Si no hay packages de OpenOffice.org disponibles siempre tiene la opción de compilar el port. Tenga muy en cuenta que necesitará gran cantidad de espacio en disco y que este port necesita una cantidad de tiempo monstruosa para efectuar la compilación.

```
# cd /usr/ports/editors/openoffice.org-2
# make install clean
```

Nota

Si quiere compilar una de las muchas versiones que existen en idiomas distintos del inglés sustituya la orden anterior por la que se muestra a continuación:

```
# make LOCALIZED_LANG=su_idioma install clean
```

Sustituya *`su_idioma`* por el código ISO del idioma que quiera. Tiene una lista de códigos de idioma disponibles dentro del directorio del port en `files/Makefile.localized` .

Hecho esto puede ejecutar OpenOffice.org con la orden:

```
% openoffice.org
```

6.4. Visores de documentos

Desde la aparición de UNIX® han ido apareciendo formatos de documento de lo más diverso, alguno de los cuales se ha hecho muy popular. Los visores de documentos estándar que requieren no tienen por qué estar en el sistema base. En esta sección veremos cómo instalar estos visores.

Esta sección cubre las siguientes aplicaciones:

Nombre de la aplicación	Recursos necesarios	Instalación desde los ports	Dependencias importantes
Acrobat Reader®	ligero	ligero	Compatibilidad binaria de Linux
gv	ligero	ligero	Xaw3d
Xpdf	ligero	ligero	FreeType
GQview	ligero	ligero	Gtk+ o GNOME

6.4.1. Acrobat Reader®

Muchos documentos se distribuyen en formato PDF, siglas de «Portable Document Format». Uno de los visores recomendados para este tipo de ficheros es Acrobat Reader®, un producto de Adobe para Linux. Dado que FreeBSD puede ejecutar binarios de Linux también existe para FreeBSD.

Para instalar Acrobat Reader® 7 desde la Colección de Ports ejecute:

```
# cd /usr/ports/print/acroread7
# make install clean
```

No hay package debido a las restricciones que impone la licencia del software.

6.4.2. gv

gv es un visor de PostScript® y PDF. Tiene su origen en ghostview pero tiene un aspecto bastante mejorado gracias a la librería Xaw3d. Es rápido y tiene una interfaz limpia. gv tiene muchas opciónes como orientación, tamaño del papel, escalado o antialias. Casi todas las operaciones se pueden realizar desde el teclado o desde el ratón.

Para instalar gv como package ejecute:

```
# pkg_add -r gv
```

Si no puede obtener el package puede usar la colección de ports:

```
# cd /usr/ports/print/gv
# make install clean
```

6.4.3. Xpdf

Si quiere un visor de PDF pequeño, Xpdf es un visor ligero y eficiente. Requiere muy pocos recursos y es muy estable. Usa los tipos estándar de X y no depende de Motif® ni de ningún otro «toolkit» de X.

Para instalar el package de Xpdf use la siguiente orden:

```
# pkg_add -r xpdf
```

Si el paquete no está disponible o prefiere usar la colección de ports ejecute:

```
# cd /usr/ports/graphics/xpdf
# make install clean
```

Una vez que la instalación haya concluido puede lanzar Xpdf; use el botón derecho del ratón para activar el menú.

6.4.4. GQview

GQview es un visor de imágenes. Puede ver un fichero con un click, lanzar un editor externo, obtener imágenes de previsualización y muchas cosas más. También tiene un modo de presentación y admite algunas opciones básicas de manipulación de ficheros. Puede gestionar colecciones de imágenes y encontrar ficheros duplicados con facilidad. GQview puede mostrar las imágenes a pantalla completa y está preparado para la internacionalización.

Si quiere instalar el paquete de GQview ejecute:

```
# pkg_add -r gqview
```

Si el paquete no está disponible o prefiere usar la colección de ports ejecute:

```
# cd /usr/ports/graphics/gqview
# make install clean
```

6.5. Finanzas

Si por algun motivo quisiera gestionar sus finanzas personales en su escritorio FreeBSD hay algunas aplicaciones potentes y sencillas de utilizar que puede instalar. Algunas de ellas son compatibles con formatos muy extendidos como documentos de Quicken o Excel.

Esta sección cubre las siguientes aplicaciones:

Nombre de aplicación	Recursos necesarios	Instalación desde los ports	Dependencias importantes
GnuCash	ligero	pesado	GNOME
Gnumeric	ligero	pesado	GNOME
Abacus	ligero	pesado	Tcl/Tk
KMyMoney	ligero	pesado	KDE

6.5.1. GnuCash

GnuCash es parte del afán de GNOME por ofrecer aplicaciones de uso sencillo pero a la vez potentes a los usuarios. Con GnuCash, puede gestionar sus ingresos y sus gastos, sus cuentas bancarias o sus acciones. Tiene una interfaz intuitiva pero muy profesional.

GnuCash dispone de un registro elegante, un sistema jerárquico de cuentas, muchos atajos de teclado y métodos de autocompletado. Puede dividir una transacción en varias partes más detalladas. GnuCash puede importar y mezclar ficheros QIF de Quicken. También admite muchos formatos internacionales de fecha y moneda.

Para instalar GnuCash en su sistema ejecute:

```
# pkg_add -r gnucash
```

Si el paquete no está disponible puede usar la colección de ports:

```
# cd /usr/ports/finance/gnucash
# make install clean
```

6.5.2. Gnumeric

Gnumeric es una hoja de cálculo que forma parte del entorno de escritorio GNOME. Implementa una «predicción» automática de la entrada del usuario de acuerdo con el formato de celda y un sistema de completado automático para múltiples secuencias. Puede importar ficheros de un gran número de formatos muy extendidos, como Excel, Lotus 1-2-3 o Quattro Pro. Gnumeric soporta gráficas mediante el programa de gráficas math/guppi. Tiene un gran n´mero de funciones internas y permite todos los formatos normales de celda como número, moneda, fecha, hora y muchos más.

Para instalar Gnumeric como package ejecute:

```
# pkg_add -r gnumeric
```

Si el package no está disponible puede usar la colección de ports del siguiente modo:

```
# cd /usr/ports/math/gnumeric
# make install clean
```

6.5.3. Abacus

Abacus es una hoja de cálculo pequeña y fácil de usar. Tiene muchas funciones útiles en varios ámbitos como la estadística, las finanzas y las matemáticas. Puede importar y exportar ficheros con formato Excel. Abacus puede generar ficheros PostScript®.

Para instalar Abacus como package ejecute:

```
# pkg_add -r abacus
```

Si no hay package al alcance puede usar la colección de ports:

```
# cd /usr/ports/deskutils/abacus
# make install clean
```

6.6. Resumen

FreeBSD está muy extendido en ISP por su alto rendimiento y estabilidad, pero también se puede usar a diario como escritorio. Hay miles de aplicaciones disponibles como packages o ports con las que puede crear un escritorio que cubra todas sus necesidades.

Aqui hay un resumen de todas las aplicaciones cubiertas en este cápitulo:

Nombre De La Aplicación	Nombre Del Paquete	Nombre Del Port
Mozilla	`mozilla`	www/mozilla
Opera	`opera`	www/opera
Firefox	`firefox`	www/firefox
KOffice	`koffice-kde3`	editors/koffice-kde3
AbiWord	`AbiWord2`	editors/AbiWord2
GIMP	`gimp`	graphics/gimp
OpenOffice.org	`openoffice`	editors/openoffice-1.1
Acrobat Reader®	`acroread`	print/acroread7
gv	`gv`	print/gv
Xpdf	`xpdf`	graphics/xpdf
GQview	`gqview`	graphics/gqview
GnuCash	`gnucash`	finance/gnucash
Gnumeric	`gnumeric`	math/gnumeric
Abacus	`abacus`	deskutils/abacus
KMyMoney	`kmymoney2`	finance/kmymoney2

Capítulo 7. Multimedia

Editado por Ross Lippert.

7.1. Sinopsis

FreeBSD admite una gran variedad de tarjetas de sonido, lo que permite disfrutar de sonido de alta fidelidad en su sistema. Ésto incluye la posibilidad de grabar y reproducir ficheros de audio tanto en los formatos MPEG Audio Layer 3 (MP3), WAV y Ogg Vorbis, así como muchos otros. La Colección de Ports de FreeBSD contiene también aplicaciones que le permitirán editar audio previamente grabado, añadir efectos y controlar dispositivos MIDI.

Tras unas cuantas pruebas podrá reproducir ficheros de video y DVD en FreeBSD. La cantidad de aplicaciones disponibles para codificar, convertir y reproducir los diversos soportes de video es más limitada que lo que hay disponible para el sonido. Por ejemplo, en el momento de escribir este texto no hay en la Colección de Ports de FreeBSD una buena aplicación que permita pasar datos de un formato a otro, el equivalente a lo que hay en el campo del sonido con audio/sox. A pesar de eso la situación en este área está cambiando rápidamente.

Este capítulo muestra los pasos necesarios para configurar una tarjeta de sonido. La configuración e instalación de X11 (Capítulo 5, *El sistema X Window*) habrá solucionado ya todos los problemas que hubiera podido darle su tarjeta gráfica, aunque quizás haya algún pequeño detalle que añadir para mejorar la reproducción.

Tras leer éste capítulo sabrá usted:

- Cómo configurar su sistema de modo que su tarjeta de sonido sea reconocida por el mismo.
- Confirmar el funcionamiento de su tarjeta usando herramientas de muestreo.
- Cómo enfrentarse a problemas de configuración de sonido.
- Cómo reproducir y crear ficheros MP3 y otros.
- Cómo gestiona el video el servidor X.
- Identificar unos cuantos ports para reproducir y codificar video que le darán muy buenos resultados.
- Cómo reproducir DVD y ficheros `.mpg` y `.avi`.
- Cómo convertir CD y DVD en ficheros de datos.

- Cómo configurar una tarjeta de TV.
- Cómo configurar un escáner de imágenes.

Antes de leer éste capítulo debería:

- Saber configurar e instalar un nuevo kernel (Capítulo 8, *Configuración del kernel de FreeBSD*).

Aviso

Si intenta montar un cd con mount(8) provocará (como mínimo) un error en el sistema, pero puede llegar a provocar un *kernel panic*. Dichos soportes tienen codificaciones especiales sustancialmente diferentes de los típicos sistemas de ficheros ISO.

7.2. Configuración de la tarjeta de sonido

Escrito por Moses Moore.
Ampliado para FreeBSD 5.X por Marc Fonvieille.

7.2.1. Configuración del sistema

Antes de nada tendrá que saber a ciencia cierta el modelo de tarjeta que tiene, qué chip utiliza y si es PCI o ISA. FreeBSD admite una amplia variedad de tarjetas, tanto PCI como ISA. Consulte la lista de dispositivos de sonido que puede usar en FreeBSD en las Hardware Notes. Las «Hardware Notes» le dirán también qué controlador es el que hace funcionar su tarjeta.

Para poder usar su dispositivo de sonido en FreeBSD tiene que cargar el controlador de dispositivo adecuado, cosa que puede hacerse de dos maneras. La forma más fácil es cargar el módulo del kernel correspondiente a su tarjeta de sonido mediante kldload(8). Puede hacerlo en su shell con la siguiente orden

```
# kldload snd_emu10k1
```

o añadiendo la línea apropiada al fichero `/boot/loader.conf` :

```
snd_emu10k1_load="YES"
```

Ambos ejemplos son para una tarjeta de sonido Creative SoundBlaster® Live!. Encontrará una lista de módulos de sonido que puede cargar del mismo modo en `/boot/de-`

`faults/loader.conf` . Si no está del todo seguro sobre el controlador que debe usar pruebe con el módulo `snd_driver` :

```
# kldload snd_driver
```

Se trata de un «metacontrolador» que carga en memoria todos los controladores más habituales. De este modo acelerará notablemente el proceso de detectar cuál es el controlador perfecto para su hardware. También puede cargar todos los controladores de sonido desde `/boot/loader.conf` .

Si quiere saber qué controlador se corresponde con su tarjeta de sonido una vez cargado el «metadriver» `snd_driver` debe comprobar el contenido de `/dev/sndstat` mediante la orden `cat /dev/sndstat` .

El segundo método es compilar de forma estática el soporte para su tarjeta de sonido dentro de un kernel personalizado. En la siguiente sección encontrará la información necesaria para añadir a su kernel el soporte que necesita su hardware. Para más información sobre la compilación de un kernel consulte el Capítulo 8, *Configuración del kernel de FreeBSD*.

7.2.1.1. Configuración de un kernel personalizado con soporte de sonido

Lo primero que hay que hacer es añadir el controlador del «framework» de sonido sound(4) al kernel; tendrá que añadir la siguiente línea a su fichero de configuración del kernel:

```
device sound
```

Una vez hecho esto tiene que añadir el soporte para su tarjeta de sonido. Evidentemente tendrá que saber exactamente qué controlador es el que hace funcionar su tarjeta. Consulte la lista de dispositivos de sonido soportados que encontrará en las Hardware Notes y elija el controlador perfecto para su tarjeta. Por ejemplo una una tarjeta de sonido Creative SoundBlaster® Live! usará el controlador snd_emu10k1(4). Siguiendo con el ejemplo, si queremos añadir soporte para esta tarjeta agregaremos lo siguiente:

```
device snd_emu10k1
```

Por favor, lea la página de manual del controlador si tiene dudas con la sintaxis. La sintaxis específica para la configuración en el kernel de cada tarjeta de sonido soportada existente puede encontrarse también en `/usr/src/sys/conf/NOTES` .

Las tarjetas ISA no PnP pueden requerir que se le facilite al kernel la información de la configuración de la tarjeta (IRQ, puerto de E/S, etc). Esto se hace modificando el fichero `/boot/device.hints` . Durante el proceso de arranque loader(8) leerá este fichero y le pasará los parámetros al kernel. Veamos un ejemplo: una vieja tarjeta Creative SoundBlaster® 16 ISA no PnP utilizará el controlador snd_sbc(4) junto con `snd_sb16` . En el caso de esta tarjeta habrá que añadir las siguientes líneas al fichero de configuración del kernel

```
device snd_sbc
device snd_sb16
```

y añadir lo siguiente a `/boot/device.hints` :

```
hint.sbc.0.at="isa"
hint.sbc.0.port="0x220"
hint.sbc.0.irq="5"
hint.sbc.0.drq="1"
hint.sbc.0.flags="0x15"
```

La tarjeta del ejemplo usa el puerto de E/S `0x220` I/O port y la IRQ 5.

La sintaxis a utilizar en `/boot/device.hints` se explica en la página de manual de sound(4) y en la del controlador en cuestión.

En el ejemplo se muestran los parámetros por omisión. En ciertos casos habrá que usar otra IRQ o usar otros parámetros para hacer funcionar la tarjeta de sonido. Consulte la página de manual de snd_sbc(4) para más información sobre esta tarjeta.

7.2.2. Cómo probar su tarjeta de sonido

Una vez que haya reiniciado con su kernel personalizado, o tras cargar el mólo del kernel necesario la tarjeta de sonido aparecerá en el búfer de mensajes del sistema (dmesg(8)); deberí ser algo muy parecido a esto:

```
pcm0: <Intel ICH3 (82801CA)> port 0xdc80-0xdcbf,0xd800-0xd8ff irq 5 ↺
at device 31.5 on pci0
pcm0: [GIANT-LOCKED]
pcm0: <Cirrus Logic CS4205 AC97 Codec>
```

Puede comprobar el estado de la tarjeta de sonido en el fichero `/dev/sndstat` :

```
# cat /dev/sndstat
FreeBSD Audio Driver (newpcm)
Installed devices:
pcm0: <Intel ICH3 (82801CA)> at io 0xd800, 0xdc80 irq 5 bufsz 16384
kld snd_ich (1p/2r/0v channels duplex default)
```

El resultado en su sistema seguramente será distinto. Si no aparecen dispositivos `pcm` revise todo el proceso a ver dónde está el error. Revise su fichero de configuración del kernel nuevamente y compruebe que eligió el controlador de dispositivo adecuado. Tiene una lista de problemas habituales en la Sección 7.2.2.1, “Problemas habituales”.

Si todo ha ido bien ya tiene una tarjeta de sonido funcionando en su sistema. Si los pins de salida de sonido de su CD-ROM o DVD-ROM están colocados correctamente puede poner un CD en la unidad y reproducirlo con cdcontrol(1).

```
% cdcontrol -f /dev/acd0 play 1
```

Hay muchas aplicaciones como audio/workman que le ofrecerán un interfaz más amigable. Si quiere escuchar ficheros MP3 puede instalar, por ejemplo, audio/mpg123.

Hay otra via aún más rápida de probar su tarjeta de sonido. Envíe datos a to `/dev/dsp` del siguiente modo:

```
% cat fichero > /dev/dsp
```

donde `fichero` puede ser cualquier fichero. Esta orden producirá ruido, lo que confirmará que la tarjeta de sonido funciona.

Puede manejar los niveles de la tarjeta de sonido con mixer(8). Tiene todos los detalles en la página de manual de mixer(8).

7.2.2.1. Problemas habituales

Error	Solution
sb_dspwr(XX) timed out	El puerto de E/S no ha sido configurado correctamente.
bad irq XX	No ha elegido correctamente la IRQ. Asegúrese de que la IRQ que ha elegido y la IRQ del sonido son la misma.
xxx: gus pcm not attached, out of memory	No queda memoria suficiente en el sistema para poder usar el dispositivo.
xxx: can't open /dev/dsp!	Compruebe si hay otra aplicación utilizando (y por tanto ocupando) el dispositivo. Escriba `fstat | grep dsp`. Los sospechosos habituales son esound y el sistema de sonido de KDE.

7.2.3. Cómo utilizar múltiples fuentes de sonido

Escrito por Munish Chopra.

Disponer de múltiples fuentes de sonido puede ser muy útil, por ejemplo cuando esound o artsd no permiten compartir el dispositivo de sonido con otra aplicación.

FreeBSD permite hacerlo gracias a los *Virtual Sound Channels* (`canales virtuales de sonido`), que se activan mediante sysctl(8). Los canales virtuales permiten multiplexar la reproducción de sonido de su tarjeta mezclando el sonido en el kernel.

El número de canales virtuales se configura utilizando dos sysctl; como `root` escriba lo siguiente:

```
# sysctl hw.snd.pcm0.vchans=4
```

```
# sysctl hw.snd.maxautovchans=4
```

En el ejemplo anterior se han creado cuatro canales virtuales, una cantidad muy razonable para el uso normal. `hw.snd.pcm0.vchans` es el número de canales virtuales de que dispone `pcm0`; puede configurarlo en cuanto el dispositivo sea reconocido por el sistema. `hw.snd.maxautovchans` es el número de canales virtuales que se le asignan a un nuevo dispositivo de audio cuando se conecta al sistema mediante kldload(8). Puesto que el módulo `pcm` puede cargarse independientemente de los controladores de hardware `hw.snd.maxautovchans` puede almacenar cuántos canales virtuales tienen asignados los dispositivos que se vayan conectando al sistema.

Nota

No es posible modificar el número de canales virtuales de un dispositivo que esté en marcha. Cierre antes todos los programas que estén utilizando el dispositivo, como reproductores de música o dæmons de sonido.

Si no usa devfs(5) tendrá que configurar sus aplicaciones para que apunten a `/dev/dsp0.`*x*, donde *x* es un número entre 0 y 3 si `hw.snd.pcm.0.vchans` tiene un valor de 4 como en el ejemplo anterior. Si se estuviera usando devfs(5) en el ejemplo anterior cualquier programa que pidiera `/dev/dsp0` sería atendido de forma totalmente transparente.

7.2.4. Asignación a los canales de mezcla de valores por omisión

Escrito por Josef El-Rayes.

Los valores por defecto de los distintos canales de mezcla están fijados en el código del controlador pcm(4). Hay distintas aplicaciones y dæmons que le permitirán asignar valores al mezclador y que permanezcan entre uso y uso, pero no es una solución demasiado limpia. Puede asignar valores por omisión a las mezclas a nivel de controlador definiendo los valores apropiados en `/boot/device.hints` . Veamos un ejemplo:

```
hint.pcm.0.vol="50"
```

Esto asignará al canal de volumen un valor por omisión de 50 al cargar el módulo pcm(4).

7.3. Sonido MP3

Escrito por Chern Lee.

El formato MP3 (MPEG Layer 3 Audio) llega casi al nivel de calidad del CD, por lo que no hay motivo por el deba faltar en su escritorio FreeBSD.

7.3.1. Reproductores de MP3

El reproductor de MP3 para X11 más extendido es, de lejos, XMMS (X Multimedia System). Puede usar en él «skins»de Winamp puesto que el interfaz gráfico de XMMS es casi idéntico al de Nullsoft Winamp. XMMS incluye también soporte nativo para «plug-ins».

Puede instalar XMMS como port (multimedia/xmms) o como package.

La interfaz de XMMS es intuitiva e incluye una lista de reproducción, un ecualizador gráfico y muchas otras cosas. Si está acostumbrado a Winamp XMMS le será muy fácil de usar.

El port audio/mpg123 es un reproductor MP3 que se usa desde la shell.

Puede especificarle desde la misma línea de órdenes a mpg123 el dispositivo de sonido que debe usar y el fichero MP3 que debe reproducir. Veamos un ejemplo:

```
# mpg123 -a /dev/dsp1.0 ASaber-GrandesÉxitos.mp3
High Performance MPEG 1.0/2.0/2.5 Audio Player for Layer 1, 2 and 3.
Version 0.59r (1999/Jun/15). Written and copyrights by Michael Hipp.
Uses code from various people. See 'README' for more!
THIS SOFTWARE COMES WITH ABSOLUTELY NO WARRANTY! USE AT YOUR OWN ↺
RISK!




Playing MPEG stream from ASaber-GrandesÉxitos.mp3 ...
MPEG 1.0 layer III, 128 kbit/s, 44100 Hz joint-stereo
```

Sustituya `/dev/dsp1.0` por el dispositivo `dsp` que haya en su sistema.

7.3.2. Extracción de pistas de sonido de un CD

Antes de convirtir una pista o todo un CD a MP3 la información sonido debe volcarse al disco duro. Esto se hace convirtiendo los datos «raw CDDA» (CD Digital Audio) a ficheros WAV.

La herramienta `cdda2wav` , que forma parte del port sysutils/cdrtools, permite la extracción de información de sonido en CD así como la información asociada a él.

Con el CD en la unidad la siguiente orden (como `root`) un CD entero en ficheros WAV individuales, uno por pista:

```
# cdda2wav -D 0,1,0 -B
```

cdda2wav soporta unidades CDROM ATAPI (IDE). En el caso de unidades IDE especifique el número de dispositivo en el lugar correspondiente al núro de unidad SCSI. Por ejemplo, para extraer la pista 7 desde una unidad IDE:

```
# cdda2wav -D /dev/acd0 -t 7
```

La opción `-D 0,1,0` indica el dispositivo SCSI `0,1,0`, que corresponde con la salida de `cdrecord -scanbus`.

Para extraer pistas de forma selectiva use la opción `-t`:

```
# cdda2wav -D 0,1,0 -t 7
```

Esta opción extrae la séptima pista de un CDROM de sonido. Si quiere extraer un rango de pistas, por ejemplo las pistas de la uno a la siete, declare el rango del siguiente modo:

```
# cdda2wav -D 0,1,0 -t 1+7
```

La herramienta dd(1) le permite también extraer pistas de sonido desde unidades ATAPI; consulte la Sección 18.6.5, “Copiar CD de audio” para más información.

7.3.3. Codificación de MP3

Hoy por hoy el codificador de MP3 por antonomasia es lame. Puede encontrar lame en audio/lame.

Partiendo de ficheros WAV previamente extraidos la siguiente orden convierte `audio01.wav` en `audio01.mp3` :

```
# lame -h -b 128 \
--tt "Título de copla o cantar" \
--ta "Grupo, solista o coral sinfónica" \
--tl "Título del álbum" \
--ty "2001" \
--tc "Autor de la extracción y paso a MP3" \
--tg "Estilo" \
audio01.wav audio01.mp3
```

Los 128 kbits pasan por se el estándar de ratio de bits de los MP3, aunque puede que usted sea de los que prefieren calidades más altas, 160 o incluso 192. A mayor ratio de bits, más espacio en disco necesitará, aunque la calidad también será mayor. La opción `-h` usa el modo «más calidad pero algo más lento». Las opciones que empiezan por `--t` indican etiquetas ID3, que suelen contener información de la canción que contiene el fichero MP3. Tiene información sobre otras opciones de codificación en la página de manual de lame.

7.3.4. Decodificación de MP3

Para grabar un CD de sonido desde ficheros MP3 hay que convertirlos a formato WAV no comprimido. Tanto XMMS como mpg123 soportan la conversión de MP3 a un formato de fichero no comprimido.

Escritura a disco desde XMMS:

1. Arranque XMMS.
2. Botón derecho del raón en la ventana para acceder al menú de XMMS.
3. Elija `Preferences` en `Options`.
4. Cambie el «Output Plugin» a «Disk Writer Plugin».
5. Pulse `Configure`.
6. Elija un directorio en el que guardar los ficheros descomprimidos.
7. Cargue el fichero MP3 en XMMS de la forma habitual, pero asegúrese de tener el volumen al 100% y las opciones de ecualización desactivadas.
8. Pulse `Play` — parecerá que XMMS está reproduciendo música pero no sonará nada. Lo que está haciendo es volcar el MP3 a un fichero.
9. Recuerde que antes de poder volver a escuchar ficheros MP3 de nuevo tendrá que devolver el «Plugin de salida» por omisión a la posición inicial.

Escritura de la salida estándar en mpg123:

- Ejecute `mpg123 -s audio01.mp3 > audio01.pcm`

XMMS genera un fichero WAV, mientras que mpg123 convierte MP3 a datos de sonido raw PCM. Puede usar cdrecord con ambos formatos para crear CD de sonido. Tendrá que usar burncd(8) para raw PCM. Si lo que tiene son ficheros WAV escuchará un ligero clic al principio de cada pista: se trata de la cabecera del fichero WAV. Puede eliminar esta cabecera mediante SoX (que puede instalar como port desde audio/sox o si lo prefiere como package):

```
% sox -t wav -r 44100 -s -w -c 2 pista.wav pista.raw
```

Consulte la Sección 18.6, “Creación y uso de medios ópticos (CD)” para más información sobre el uso de grabadoras de CD en FreeBSD.

7.4. Reproducción de vídeo

Contributed by Ross Lippert.

La reproducción de vídeo es un área muy nueva y en la que se producen grandes cambios a gran velocidad. Tenga paciencia. Aquí no va a ser todo tan sencillo como al configurar el sonido.

Antes de comenzar es imprescindible que sepa con certeza cuál es el modelo de su tarjeta gráfica y qué chip usa. Tanto Xorg como XFree86™ soportan gran cantidad de tarjetas gráficas, pero de pocas se puede decir que den un buen rendiemiento de reproducción.

Puede consultar una lista de extensiones que el servidor X puede hacer funcionar en su tarjeta ejecutando xdpyinfo(1) desde una sesión X11.

Le recomendamos que tenga a mano un fichero MPEG no muy grande para usarlo en las pruebas que haga con los diversos reproductores y opciones. Algunos reproductores de DVD buscarán por omisión soportes DVD en `/dev/dvd` (algunos incluso tienen fijado en el código el nombre de dicha unidad) seguramente le resultará útil ejecutar los siguientes enlaces simbólicos:

```
# ln -sf /dev/acd0 /dev/dvd
# ln -sf /dev/acd0 /dev/rdvd
```

Tenga en cuenta que debido al diseño de devfs(5) los enlaces creados a mano no sobrevivirán a un reinicio del sistema. Para que así suceda tras cada reinicio debe añadir las siguientes líneas a `/etc/devfs.conf` :

```
link acd0 dvd
link acd0 rdvd
```

Además, el descifrado de DVD (una tarea que hará necesario llamar a funciones especiales de los DVD-ROM) requiere permisos de escritura en unidades DVD.

Le recomendamos que aumente la memoria compartida del interfaz X11, para lo cual tendrá que incremente los valores asignados a varias sysctl(8):

```
kern.ipc.shmmax=67108864
kern.ipc.shmall=32768
```

7.4.1. Cómo determinar las características de video

Hay varias formas de reproducir vídeo en X11, pero cuál de ellas sea la que funcione depende en gran medida del hardware que intervenga. Cada uno de los métodos que se describen más son mejores o peores según en qué hardware se ejecute. Además el procesamiento de video en X11 es un asunto que está recibiendo mucha atención últimamente y en cada nueva versión de Xorg y de XFree86™ no es extraño encontrar avances significativos.

Veamos una lista de interfaces de vídeo muy comunes:

1. X11: salida normal de X11 utilizando memoria compartida.
2. XVideo: una extensión del interfaz X11 que permite la reproducción de vídeo en las ventanas de X11.
3. SDL: Simple Directmedia Layer.
4. DGA: Direct Graphics Access (`acceso directo a gráficos`.
5. SVGAlib: capa de consola de gráficos a bajo nivel.

7.4.1.1. XVideo

Xorg y XFree86™ 4.X tienen una extensión llamada *XVideo* (también conocida como Xvideo, Xv o xv) que permite reproducir vídeo en ventanas del sistema X11 a través de un tipo de aceleración especial. Dicha extensión ofrece una calidad de reproducción muy alta incluso en máquinas muy cortas de recursos.

Utilice `xvinfo` para saber qué extensión está utilizando:

```
% xvinfo
```

XVideo is supported for your card if the result looks like:

```
X-Video Extension version 2.2
screen #0
  Adaptor #0: "Savage Streams Engine"
    number of ports: 1
    port base: 43
    operations supported: PutImage
    supported visuals:
      depth 16, visualID 0x22
      depth 16, visualID 0x23
    number of attributes: 5
      "XV_COLORKEY" (range 0 to 16777215)
              client settable attribute
              client gettable attribute (current value is 2110)
      "XV_BRIGHTNESS" (range -128 to 127)
              client settable attribute
              client gettable attribute (current value is 0)
      "XV_CONTRAST" (range 0 to 255)
              client settable attribute
              client gettable attribute (current value is 128)
      "XV_SATURATION" (range 0 to 255)
              client settable attribute
              client gettable attribute (current value is 128)
      "XV_HUE" (range -180 to 180)
              client settable attribute
              client gettable attribute (current value is 0)
    maximum XvImage size: 1024 x 1024
    Number of image formats: 7
      id: 0x32595559 (YUY2)
        guid: 59555932-0000-0010-8000-00aa00389b71
        bits per pixel: 16
        number of planes: 1
        type: YUV (packed)
      id: 0x32315659 (YV12)
        guid: 59563132-0000-0010-8000-00aa00389b71
        bits per pixel: 12
        number of planes: 3
        type: YUV (planar)
      id: 0x30323449 (I420)
        guid: 49343230-0000-0010-8000-00aa00389b71
```

```
        bits per pixel: 12
        number of planes: 3
        type: YUV (planar)
      id: 0x36315652 (RV16)
        guid: 52563135-0000-0000-0000-000000000000
        bits per pixel: 16
        number of planes: 1
        type: RGB (packed)
        depth: 0
        red, green, blue masks: 0x1f, 0x3e0, 0x7c00
      id: 0x35315652 (RV15)
        guid: 52563136-0000-0000-0000-000000000000
        bits per pixel: 16
        number of planes: 1
        type: RGB (packed)
        depth: 0
        red, green, blue masks: 0x1f, 0x7e0, 0xf800
      id: 0x31313259 (Y211)
        guid: 59323131-0000-0010-8000-00aa00389b71
        bits per pixel: 6
        number of planes: 3
        type: YUV (packed)
      id: 0x0
        guid: 00000000-0000-0000-0000-000000000000
        bits per pixel: 0
        number of planes: 0
        type: RGB (packed)
        depth: 1
        red, green, blue masks: 0x0, 0x0, 0x0
```

Tenga en cuenta que los formatos que aparecen en el listado (YUV2, YUV12, etc) no aparecen en todas las implementaciones de de XVideo y su ausencia puede despistar a algunos reproductores.

Si el resultado se parece mucho a esto

```
X-Video Extension version 2.2
screen #0
no adaptors present
```

lo más probable es que su tarjeta no soporte XVideo.

Si su tarjeta gráfica no soporta XVideo lo único que significa es que para su sistema será más complicado alcanzar los requisitos computacionales necesarios para el procesamiento de vídeo. Dependiendo de la tarjeta gráfica y el procesador que tenga es posible que de todos modos pueda salvar la papeleta. Consulte la sección sobre recursos avanzados en la Sección 7.4.3, “Lecturas adicionales”.

7.4.1.2. Simple Directmedia Layer

SDL (Simple Directmedia Layer) se creó pensando en una capa que permitiera aplicaciones fáciles de portar entre las plataformas Microsoft® Windows®, BeOS y UNIX® y que

permitiera al mismo tiempo un uso eficiente del sonido y el vídeo. La capa SDL ofrece abstracción de hardware a bajo nivel, lo que a veces puee ser más eficiente que la propia interfaz X11.

SDL está en devel/sdl12.

7.4.1.3. Direct Graphics Access

«Direct Graphics Access» es una extensión de X11 que permite que un programa sortée el servidor X y acceda directamente al «framebuffer». Utiliza memoria de bajo nivel, así es imprescindible que se ejecute como `root`.

Puede probar y analizar el rendimiento de la extensión DGA con dga(1). Al ejecutar `dga` la pantalla cambia de color cada vez que pulsa una tecla. Para salir de la aplicación pulse q.

7.4.2. Ports y packages relacionados con el vídeo

Esta sección expone el software existente en la Colección de Ports de FreeBSD que puede usar para la reproducción de vídeo. La reproducción de vídeo es un área del desarrollo de software donde hay muchísima actividad, de manera que las características de más de una de las aplicaciones que vamos a describir pueden ser distintas.

Lo primero que hay que conocer es el importante detalle de que la mayoría de las aplicaciones de vídeo que funcionan en FreeBSD se desarrollaron para Linux. Muchas de esas aplicaciones todavía están en versiones beta. Estos son algunos de los problemas que puede encontrarse durante el uso de software de vídeo en FreeBSD:

1. Una aplicación no puede reproducir un fichero que ha generado otra aplicación distinta.

2. Una aplicación no puede reproducir un fichero que ella misma a generado.

3. La misma aplicación ejecutándose en dos máquinas distintas, compilada en cada una de esas máquinas; la reproducción del mismo fichero da resultados diferentes.

4. El uso de un filtro aparentemente trivial como es redimensionar una imagen aplicando una escala da como fruto un resultado horroroso debido a que la rutina encargada de aplicarla es defectuosa.

5. Que una aplicación de volcados de memoria («core dumps»).

6. La documentación no se instala con el port, pero sí que existe en la web o en el directorio `work` del port.

La mayoría de estas aplicaciones muestra síntomas evidente de «Linuxismo», es decir, hay problemas derivados del modo en el que se implementan las bibliotecas en las distribu-

ciones de Linux, o del hecho de que ciertas características del kernel de Linux han sido dadas por omnipresentes por los autores de las aplicaciones. Estos problemas no siempre han sido advertidas y solucionadas de algún modo por los responsables de los ports, lo que puede desembocar en problemas como:

1. El uso de `/proc/cpuinfo` para detectar las características del procesador.

2. El uso erróneo de los hilos puede causar que ciertos programas se cuelguen al intentar cerrarse en lugar de cerrarse correctamente.

3. Software que aún no está en la Colección de Ports de FreeBSD usado frecuentemente con la aplicación.

Hasta ahora los autores de las aplicaciones han colaborado de buen grado con los responsables de los ports para hacer más sencillo el trabajo de portar estas aplicaciones a FreeBSD.

7.4.2.1. MPlayer

MPlayer es un reproductor de vídeo desarrollado muy recientemente y con un desarrollo rapidísimo. Los objetivos del equipo MPlayer son la velocidad y la flexibilidad en Linux y otros sistemas Unix. El proyecto comenzó cuando el fundador del equipo se cansó de padecer el pobre rendimiento de los reproductores de vídeo que había disponibles en ese momento. Hay quien dice que el interfaz gráfico fue sacrificado en pos de un diseño más ligero, aunque una vez que se acostumbre a las opciones desde la línea de órdenes y a los atajos de teclado todo irá como la seda.

7.4.2.1.1. Compilación de MPlayer

Encontrará MPlayer en multimedia/mplayer. MPlayer ejecuta una serie de pruebas de hardware durante el proceso de compilación, dando como resultado un binario que no tiene por qué funcionar en otros sistemas. Recuerde, por tanto, compilar la aplicación como port y no utilice el package binario. Además hay múltiples parámetros que puede pasarle a `make`, tal y como verá al comenzar la compilación:

```
# cd /usr/ports/multimedia/mplayer
# make
N - O - T - E

Take a careful look into the Makefile in order
to learn how to tune mplayer towards you personal preferences!
For example,
make WITH_GTK1
builds MPlayer with GTK1-GUI support.
If you want to use the GUI, you can either install
/usr/ports/multimedia/mplayer-skins
or download official skin collections from
http://www.mplayerhq.hu/homepage/dload.html
```

Las opciones por omisión del port deberían bastar para la mayoría de los usuarios, aunque si va a necesitar el codec XviD tendrá que especificar `WITH_XVID` en la línea de órdenes. La unidad de DVD por defecto se puede definir con la opción `WITH_DVD_DEVICE` si la opción por omisión, `/dev/acd0` , no le sirve.

En el momento de escribir esto el port de MPlayer genera la documentación en formato HTML y dos ejecutables: `mplayer` y `mencoder`, que es una herramienta para recodificar vídeo.

La documentación en HTML de MPlayer es muy informativa. Si el lector no cubre todas sus dudas sobre hardware de vídeo e interfaces con el contenido de este capítulo la documentación de MPlayer es el sitio donde debe buscar. Si quiere más información sobre el soporte de vídeo en UNIX® el mejor sitio para consultar es, sin duda alguna, la documentación de MPlayer.

7.4.2.1.2. Uso de MPlayer

Los usuarios de MPlayer deben crear un subdirectorio `.mplayer` en su «home». Para crear este directorio (necesario para el funcionamiento de la aplicación) teclée lo siguiente:

```
% cd /usr/ports/multimedia/mplayer
% make install-user
```

Las opciones de `mplayer` para la línea de órdenes se detallan en la página de manual. Si necesita todavía más detalles consulte la documentación en HTML. En esta sección se va a explicar solamente lo más básico.

Para reproducir el fichero `ficherodeprueba.avi` , mediante una de las diversas interfaces de vídeo utilice la opción `-vo`:

```
% mplayer -vo xv ficherodeprueba.avi
```

```
% mplayer -vo sdl ficherodeprueba.avi
```

```
% mplayer -vo x11 ficherodeprueba.avi
```

```
# mplayer -vo dga ficherodeprueba.avi
```

```
# mplayer -vo 'sdl:dga' ficherodeprueba.avi
```

Merece la pena que pruebe todas, puesto que el rendimiento relativo depende en gran medida de muchos factores y será muy distinto el resultado dependiendo del hardware.

Si quiere reproducir un DVD sustituya `ficherodeprueba.avi` por `dvd://`*`N`* `-dvd-device` *`DISPOSITIVO`*, donde *`N`* es el número de título que quiere reproducir y `DISPOSITIVO` es el nodo de dispositivo del DVD-ROM. Veamos un ejemplo: para reproducir el título 3 desde `/dev/dvd` ejecutaremos:

```
# mplayer -vo xv dvd://3 -dvd-device /dev/dvd
```

Nota

La unidad de DVD por omisión puede definirse durante la compilación del port de MPlayer con la opción `WITH_DVD_DEVICE` . Por defecto apunta a `/dev/acd0` . Tiene todos los detalles en el `Makefile` del port.

Consulte los atajos de teclado para parar la reproducción, ponerla en pausa, etc. consulte la salida de `mplayer -h` o mejor aún lése la página de manual.

Hay otras opciones que le resultarán de interés para la reproducción: `-fs -zoom` , para la reproducción a pantalla completa y `-framedrop`, que mejora el rendimiento.

Para evitar que la línea de órdenes se haga demasiado larga puede crear un fichero `.mplayer/config` e introducir en él las opciones que quiera usar desde el arranque de la aplicación:

```
vo=xv
fs=yes
zoom=yes
```

Para terminar, puede usar `mplayer` para extraer un título de un DVD a un fichero `.vob`. Por ejemplo, si quiere extraer el segundo título de un DVD escriba esto:

```
# mplayer -dumpstream -dumpfile salida.vob dvd://2 -dvd-device /dev/
dvd
```

El fichero de salida, `salida.vob` , un MPEG y puede manipularlo con las aplicaciones que se describen en esta sección.

7.4.2.1.3. mencoder

Antes de empezar a usar `mencoder` le recomendamos que se familiarice con las opciones que se detallan en la documentación HTML. Hay también una página de manual, pero no es de mucha utilidad sin la documentación HTML. Hay una ingente cantidad de formas de mejorar la calidad, reducir el ratio de bits y cambiar formatos; alguno de esos pequeños trucos pueden marcar la diferencia entre un buen y un mal rendimiento. Le mostramos un par de ejemplos por los que puede empezar. El primero es una simple copia:

```
% mencoder entrada.avi -oac copy -ovc copy -o salida.avi
```

Ciertas combinaciones erróneas de opciones en la línea de órdenes pueden dar como fruto un fichero de salida que no puede reproducir ni siquiera el propio `mplayer`, así que si lo

que quiere es extraer datos a un fichero siga usando tranquilamente la opción `-dumpfile` de `mplayer`.

Para convertir `entrada.avi` con codificación MPEG4 y audio con codificación MPEG3 (necesitará audio/lame):

```
% mencoder entrada.avi -oac mp3lame -lameopts br=192 \
  -ovc lavc -lavcopts vcodec=mpeg4:vhq -o salida.avi
```

El fichero de salida puede verse con `mplayer` y con `xine`.

Si sustituye `entrada.avi` por `dvd://1 -dvd-device /dev/dvd` y lo ejecuta como `root` puede recodificar un título de DVD directamente. Es posible que el resultado no sea muy satisfactorio tras los primeros intentos, así que le recomendamos que vuelque el título a un fichero y haga todas las pruebas que necesite sobre ese fichero.

7.4.2.2. El reproductor de vídeo xine

El reproductor de vídeo xine es un ambicioso proyecto que tiene como meta ofrecer no solamente una aplicación de vídeo «todo en uno», sino crear una biblioteca base reutilizable y un ejecutable modular que puede ampliarse mediante «plugins». Existe como package y también como port: multimedia/xine.

El reproductor xine está todavía en mantillas, pero sus primeros pasos están siendo muy interesantes. xine requiere o una CPU rápida o una tarjeta gráfica, o bien soporte para la extensión XVideo. El GUI es intuitivo, aunque un tanto pedestre.

En el momento de escribir esto xine no incorpora ningún módulo de entrada que sirva para reproducir DVD codificados con CSS. Hay versiones compiladas por terceros que tienen módulos que lo permiten, pero ninguna de ellas está en la Colección de Ports de FreeBSD.

Comparado con MPlayer xine deja las cosas más hechas para el usuario, pero al mismo tiempo pone más difícil el control exhaustivo sobre todo lo que pasa. El reproductor xine da un mejor rendimiento en interfaces XVideo.

Por omisión xine arranca con un intefaz gráfico de usuario. Los menús pueden usarse para abrir un fichero en concreto:

```
% xine
```

También puede invocar la aplicación desde la línea de órdenes para que reproduzca un fichero inmediatamente y sin utilizar el GUI:

```
% xine -g -p fichero.avi
```

7.4.2.3. Las herramientas transcode

transcode no es un reproductor, sino una suite de herramientas para recodificar (también se podría usar la palabra «transcodificar», de ahí su nombre) ficheros de sonido y de vídeo.

Con transcode es posible fundir ficheros de video y reparar ficheros deteriorados desde la línea de órdenes y en las interfaces de flujo `stdin/stdout`.

Puede configurar gran cantidad de opciones en el momento de la compilación de multimedia/transcode; le recomendamos que use la siguiente línea de órdenes para compilar transcode:

```
# make WITH_OPTIMIZED_CFLAGS=yes WITH_LIBA52=yes WITH_LAME=yes ↺
WITH_OGG=yes \
WITH_MJPEG=yes -DWITH_XVID=yes
```

La configuración que le proponemos deberí satisfacer a la mayoría de los usuarios.

Veamos un ejemplo ilustrativo de las posibilidades de `transcode`: convertir un fichero DivX en un fichero PAL MPEG-1 (PAL VCD):

```
% transcode -i input.avi -V --export_prof vcd-pal -o output_vcd
% mplex -f 1 -o output_vcd.mpg output_vcd.m1v output_vcd.mpa
```

Puede reproducir el fichero MPEG resultante, `output_vcd.mpg` , con MPlayer. Si quiere también puede grabar el fichero en un CD-R y tendrá un VideoCD, aunque para eso tendrá que instalar multimedia/vcdimager y sysutils/cdrdao.

Consulte la página de manual de `transcode`, pero no olvide consultar también el wiki de transcode, en el que encontrará más información y muchos ejemplos.

7.4.3. Lecturas adicionales

El desarrollo de las diversas aplicaciones de vídeo disponibles en FreeBSD avanza rápidamente. Es muy posible que en un futuro cercano la mayoría de los problemas de los que hemos hablado aquí hayan sido resueltos. Mientras tanto quien quiera exprimir las capacidades de FreeBSD en las áreas del sonido y el vídeo tendrán que recopilar la información repartida en varias FAQ y tutoriales y utilizar unos cuantos programas diferentes. En esta sección se le proponen al lector interesado varias fuentes de información adicional.

La documentación de MPlayer contiene una gran cantidad de información técnica. Cualquiera que quiera alcanzar un alto grado de conocimiento en vídeo sobre UNIX® debe consultar esa documentación. La lista de correo de MPlayer es sumamente hostil hacia quien no se ha molestado en leer la documentación *antes* de preguntar, así que si va a enviar informes de errores a esa lista, por favor, *léase la documentación.*.

El texto xine HOWTO contiene un capítulo sobre mejora del rendimiento que afecta a todos los reproductores.

Hay unas cuantas aplicaciones muy prometedoras que el lector debería al menos probar:

- Avifile, que encontrará en los ports:multimedia/avifile.
- Ogle, también en la colección de ports: multimedia/ogle.

- Xtheater
- multimedia/dvdauthor, una aplicación libre para la producción de contenidos en DVD.

7.5. Configuración de tarjetas de TV

Texto original de Josef El-Rayes.
Ampliado y adaptado por Marc Fonvieille.

7.5.1. Introducción

Las tarjetas de TV le permiten ver en su sistema emisiones de TV tradicionales o incluso por cable. La mayoría aceptan vídeo compuesto mediante RCA o una entrada de S-video; algunas de estas tarjetas incorporan también un sintonizador de radio FM.

FreeBSD puede utilizar tarjetas de TV PCI que tengan el chip Brooktree Bt848/849/878/879 o el chip de captura de vídeo Conexant CN-878/Fusion 878a gracias al controlador bktr(4). Asegúrese de que la tarjeta tiene un sintonizador que esté soportado: consulte la lista de sintonizadores soportados que encontrará en la página de manual de bktr(4).

7.5.2. Añadir el controlador

Poder usar la tarjeta tendrá que cargar el controlador bktr(4). Una forma de hacerlo es añdir la siguiente línea al fichero `/boot/loader.conf` :

```
bktr_load="YES"
```

También puede hacerlo compilando estáticamente el soporte para la tarjeta de TV en su kernel; añada las siguientes líneas al fichero de configuración de su kernel:

```
device  bktr
device iicbus
device iicbb
device smbus
```

Es necesario incluir todas esas lineas adicionales porque los componentes de la tarjeta están conectados entre sí por un bus I2C. Hecho esto compile e instale su nuevo kernel.

Una vez añadido el soporte de la tarjeta al sistema tendrá que reiniciar. Durante el arranque deberían aparecer rastros de su tarjeta de TV. Veamos un ejemplo:

```
bktr0: <BrookTree 848A> mem 0xd7000000-0xd7000fff irq 10 at device ↺
10.0 on pci0
iicbb0: <I2C bit-banging driver> on bti2c0
iicbus0: <Philips I2C bus> on iicbb0 master-only
iicbus1: <Philips I2C bus> on iicbb0 master-only
```

```
smbus0: <System Management Bus> on bti2c0
bktr0: Pinnacle/Miro TV, Philips SECAM tuner.
```

Es obvio que estos mensajes serán distintos según sea el hardware del que se trate. Asegúrese de que el sistema detecta correctamente el sintonizador. Se puede sobreescribir alguno de los parámetros detectados por el sistema mediante MIB sysctl(8) y opciones del fichero de configuración del kernel. Si por ejemplo quiere obligar al sintonizador para que se comporte como un Phillips SECAM añada lo siguiente al fichero de configuración de su kernel:

```
options OVERRIDE_TUNER=6
```

Puede utilizar directamente sysctl(8):

```
# sysctl hw.bt848.tuner=6
```

Consulte en bktr(4) y en `/usr/src/sys/conf/NOTES` si necesita más detalles sobre las opciones disponibles.

7.5.3. Aplicaciones útiles

Tendrá que instalar una de estas aplicaciones para poder utilizar su tarjeta de TV:

- multimedia/fxtv ofrece «TV en una ventana» y captura de imagen/sonido/vídeo.
- multimedia/xawtv: otra aplicación de TV; ofrece lo mismo que fxtv.
- misc/alevt decodifica y reproduce videotexto y teletexto.
- audio/xmradio permite utilizar el sintonizador de radio FM que incorporan algunas tarjetas de TV.
- audio/wmtune, una aplicación de escritorio muy útil para el uso de sintonizadores de radio.

Hay más aplicaciones útiles en la Colección de de Ports de FreeBSD.

7.5.4. Solución de problemas

Si tiene algún problema con su tarjeta de TV compruebe en primer lugar que el chip de captura de vídeo y el sintonizador realmente funcionan con el controlador bktr(4) y si ha utilizado las opciones de configuración más adecuadas. Si necesita más ayuda o necesita respuestas sobre su tarjeta de TV puede recurrir a la lista de correo freebsd-multimedia.

7.6. Escáneres de imágenes

Escrito por Marc Fonvieille.

7.6.1. Introducción

El acceso a escáneres de imágenes en FreeBSD funciona gracias a la API SANE (siglas de «Scanner Access Now Easy», «el acceso a escáneres ahora es fácil»), disponible en la Colección de Ports de FreeBSD. SANE utiliza también ciertos controladores de FreeBSD para acceder al hardware del escáner.

FreeBSD puede utilizar tanto escáneres SCSI como USB. Asegúrese de que el suyo funciona con SANE antes de intentar ningún tipo de configuración. SANE tiene una lista de dispositivos soportados donde encontrará información actualizada sobre hasta dónde funciona el escáner y su estatus. La página de manual de uscanner(4) incluye una listas de escáneres USB que funcionan con SANE.

7.6.2. Configuración del kernel

Como ya se ha dicho tanto los interfaces USB como las SCSI funcionan. Según sea el interfaz de su escáner necesitará un tipo diferente de controlador:

7.6.2.1. Interfaz USB

El kernel `GENERIC` incluye por omisión los controladores de dispositivo necesarios para poder usar escáneres USB. Si va a personalizar su kernel asegúrese de que esto está en el fichero de configuración del mismo:

```
device usb
device uhci
device ohci
device uscanner
```

Según cual sea el chipset de su placa base tendrá que usar `device uhci` o `device ohci`, aunque si tiene ambos no tendrá problemas por eso.

Si por cualquier motivo no quiere recompilar su kernel y su kernel no es el `GENERIC` puede cargar el módulo controlador de dispositivo uscanner(4) con kldload(8):

```
# kldload uscanner
```

Si quiere cargar este módulo cada vez que arranque el sistema debe añadir la siguiente línea a `/boot/loader.conf` :

```
uscanner_load="YES"
```

Tras reiniciar con el kernel correcto (o una vez que el módulo esté cargado) conecte su escáner USB. Debe aparecer en el «buffer» de mensajes del sistema (dmesg(8)) un mensaje una línea como la siguiente:

```
uscanner0: EPSON EPSON Scanner, rev 1.10/3.02, addr 2
```

La línea del ejemplo muestra que nuestro escáner está utilizando el nodo de dispositivo/`dev/uscanner0`.

7.6.2.2. Interfaz SCSI

Si su escáner tiene un interfaz SCSI es importante saber qué controladora SCSI incorpora. El chipset SCSI que tenga influirá en la configuración del kernel que tenga que personalizar. El kernel `GENERIC` tiene controladores de dispositivo para las controladoras SCSI más comunes. Consulte el fichero `NOTES` y añada la línea correcta al fichero de configuración de su kernel. Tendrá que añadir también la siguiente línea:

```
device scbus
device pass
```

Una vez compilado e instalado el kernel correctamente tiene ya que ver sus nuevos dispositivos en el «buffer» de mensajes del sistema:

```
pass2 at aic0 bus 0 target 2 lun 0
pass2: <AGFA SNAPSCAN 600 1.10> Fixed Scanner SCSI-2 device
pass2: 3.300MB/s transfers
```

Si su escáner estaba apagado durante el arranque puede forzar manualmente la detección si ejecuta una búsqueda de buses SCSI con camcontrol(8):

```
# camcontrol rescan all
Re-scan of bus 0 was successful
Re-scan of bus 1 was successful
Re-scan of bus 2 was successful
Re-scan of bus 3 was successful
```

El escáner aparecerá en la lista de dispositivos SCSI:

```
# camcontrol devlist
<IBM DDRS-34560 S97B>              at scbus0 target 5 lun 0 ↺
(pass0,da0)
<IBM DDRS-34560 S97B>              at scbus0 target 6 lun 0 ↺
(pass1,da1)
<AGFA SNAPSCAN 600 1.10>           at scbus1 target 2 lun 0 (pass3)
<PHILIPS CDD3610 CD-R/RW 1.00>     at scbus2 target 0 lun 0 ↺
(pass2,cd0)
```

Tiene todos los detalles sobre los dispositivos SCSI en las páginas de manual de scsi(4) y camcontrol(8).

7.6.3. Configuración de SANE

El sistema SANE se divide en dos partes: los «backends» (graphics/sane-backends) y los «frontends» (graphics/sane-frontends). Los «backends» facilitan el acceso al escáner propiamente dicho. La lista de dispositivos que funcionan con SANE le dirá qué «backend» hará funcionar su escáner de imágenes. No hay más remedio que identificar correctamente el «backend» de su escáner para poder usarlo. Los «frontends» ofrecen una interfaz gráfica para escanear (xscanimage).

Lo primero que hay que hacer es instalar el port o package de graphics/sane-backends. Después ejecute la orden `sane-find-scanner` para comprobar el funcionamiento de la detección de escáneres del sistema SANE:

```
# sane-find-scanner -q
found SCSI scanner "AGFA SNAPSCAN 600 1.10" at /dev/pass3
```

En la salida de la orden anterior se le mostrará el tipo de interfaz del escáner y el nodo de dispositivo a través del cual el sistema accede a él. Es posible que no aparezcan la marca y el modelo, pero eso no tiene importancia.

Nota

Algunos escáneres USB necesitan que les instale un «firmware»; esto se explica en la página de manual del «backend» correspondiente. También debe leer las páginas de manual sane-find-scanner(1) y sane(7).

Ahora hay que comprobar que el «frontend» de escaneo puede identificar el escáner. Por omisión los «backends» de SANE incorporan una herramienta de línea de órdenes llamada scanimage(1), que le mostrará una lista de dispositivos e incluso una captura de imagen desde la shell. La opción `-L` muestra una lista de escáners:

```
# scanimage -L
device `snapscan:/dev/pass3' is a AGFA SNAPSCAN 600 flatbed scanner
```

Si el software no lanza ningún mensaje, o si el mensaje indica que no se han encontrado escáneres significa que scanimage(1) no puede identificar su escáner. Si este es el caso tendrá que editar el fichero de configuración del «backend»y definir en él el escáner que quiera utilizar. Encontrará todos los ficheros de configuración de los «backends» en el directorio `/usr/local/etc/sane.d/` . Este problema con la identificación del hardware es relativamente frecuente con ciertos modelos de escáneres USB.

Por ejemplo, con el escáner que hemos usado en la Sección 7.6.2.1, “Interfaz USB” la orden `sane-find-scanner` nos da la siguiente información:

```
# sane-find-scanner -q
found USB scanner (UNKNOWN vendor and product) at device /dev/
uscanner0
```

El escáner ha sido detectado sin problemas, utiliza el interfaz USB y está conectado al nodo de dispositivo `/dev/uscanner0` . Ahora comprobaremos que podemos identificar correctamente el escáner:

```
# scanimage -L
```

```
No scanners were identified. If you were expecting something ↺
different,
check that the scanner is plugged in, turned on and detected by the
sane-find-scanner tool (if appropriate). Please read the ↺
documentation
which came with this software (README, FAQ, manpages).
```

No hemos podido identificar el escáner, así que hemos de editar el fichero `/usr/local/etc/sane.d/epson.conf` . Estamos usando el modelo de escáner EPSON Perfection® 1650, así que ya sabemos que el escáner utilizará el «backend» `epson`. Lea atentamente los comentarios que hay en los ficheros de configuración de los «backends». Lo que hay que hacer es muy sencillo: basta con comentar las líneas en las que aparecen interfaces que no sean el perfecto para su escáner. En nuestro caso comentaremos todas las líneas que comiencen por `scsi` puesto que nuestro escáner tiene interfaz USB. Después de esto hay que añadir una lína al final del fichero declarando el interfaz y el nodo de dispositivo que se van a usar. En el caso del ejemplo esto es lo que añadimos:

```
usb /dev/uscanner0
```

Por favor, lea los comentarios que encontrará en el fichero de configuración del «backend» y la pángina de manual de del «backend» si necesita más información sobre la sintaxis que debe usar. Hecho esto podemos verificar si ya podemos identificar el escáner:

```
# scanimage -L
device `epson:/dev/uscanner0' is a Epson GT-8200 flatbed scanner
```

Hemos identificado nuestro escáner USB. Ni tiene importancia si la marca y modelo que hemos visto no coinciden con nuestro escáner. Lo único realmente importante es el campo `` `epson:/dev/uscanner0' ``, que nos da el nombre correcto del «backend» y el nodo de dispositivo correcto.

Una vez que `scanimage -L` llega a ver al escáner hemos terminado con la configuración. El dispositivo está listo para escanear imángenes.

scanimage(1) permite ejecutar una captura de imágen desde la línea de órdenes, pero siempre es mejor utilizar un interfaz gráfico de usuario para estas tareas. SANE ofrece un interfaz gráfico simple pero eficiente: xscanimage (graphics/sane-frontends).

Otro «frontend» gráfico para escanear que se ha hecho muy popular es Xsane (graphics/xsane). Este «frontend» tiene características muy avanzadas como varios modos de escaneo (fotocopia, fax, etc.), corrección de color, escaneo de imágenes por lotes, etc. Ambas aplicaciones pueden utilizarse además como «plugins» de GIMP.

7.6.4. Dar acceso al escáner a otros usuarios

Todo lo que se ha mostrado en esta sección se ha ejecutado con privilegios de `root`. Lo normal es que haya otros usuarios que tengan que tener acceso al escáner. Un usuario que

quiera usar el escáner tiene que tener acceso de lectura y escritura al nodo de dispositivo que usa el escáner. Por ejemplo, nuestro escáner USB utiliza el nodo de dispositivo `/dev/uscanner0` propiedad del grupo `operator`. Al añadir al usuario *joe* al grupo `operator` éste podrá acceder al esc´ner:

```
# pw groupmod operator -m joe
```

Para más información sobre el uso de pw(8) lea su página de manual. Tendrá también que configurar correctamente los permisos de escritura (0660 o 0664) en el nodo de dispositivo `/dev/uscanner0`; por omisión los miembros del grupo `operator` sólo tienen acceso de lectura en este nodo de dispositivo. Añada las siguientes líneas al fichero lines to the `/etc/devfs.rules` :

```
[system=5]
add path uscanner0 mode 660
```

Ahora añada lo siguiente a `/etc/rc.conf` y reinicie la máquina:

```
devfs_system_ruleset="system"
```

Para más información sobre lo que acaba de leer consulte la página de manual de devfs(8).

Nota

Por seguridad debería pensarse dos veces el hecho mismo de añadir cualquier usuario a un grupo y muy especialmente al grupo `operator`.

Capítulo 8. Configuración del kernel de FreeBSD

Actualizado y reestructurado por Jim Mock.
Texto original por Jake Hamby.

8.1. Sinopsis

El kernel de FreeBSD es el corazón del sistema operativo. Es el responsable de la gestión y control de la memoria, reforzamiento de los controles de seguridad, gestión de redes, acceso a dispositivos y muchas cosas más. FreeBSD es un sistema dinámico y configurable, pero a veces sigue siendo necesario reconfigurar y recompilar un kernel hecho a la medida de nuestras necesidades.

Una vez leido este capítulo sabrá usted:

- Por qué puede ser necesario que compile un kernel personalizado.
- Cómo crear un fichero de configuración del kernel, o bien editar un fichero existente.
- Cómo utilizar el fichero de configuración para crear y compilar un nuevo kernel.
- Cómo instalar su nuevo kernel.
- Cómo resolver algunos problemas si algo sale mal.

8.2. ?Qué razones hay para compilar un kernel personalizado?

Tradicionalmente FreeBSD ha contado con lo que se conoce como un kernel “monolítico”. Esto quiere decir que el kernel era un gran y único programa que se comunicaba con una lista previa de dispositivos, y que si se deseaba modificar el comportamiento del kernel se debía compilar un nuevo kernel y reiniciar el sistema con el nuevo kernel.

Hoy en día FreeBSD evoluciona muy rápidamente hacia un modelo donde la funcionalidad del kernel se basa en módulos, los cuales pueden cargarse y descargarse dinámicamente de acuerdo a las necesidades del kernel. Esto permite al kernel adaptarse al nuevo hardware que sale al mercado (como las tarjetas PCMCIA en sistemas portátiles), o bien añadir nuevas funcionalidades al kernel que no eran al compilarlo la vez anterior. Esto es lo que conocemos como kernel modular.

A pesar de lo dicho aún es necesario llevar a cabo cierta configuración estática en la configuración del kernel. A veces se debe a que la funcionalidad del sistema está tan ligada al kernel que no puede hacerse mediante carga dinámica de módulos. En otros casos puede tratarse simplemente de que nadie ha programado un módulo para esa funcionalidad concreta que pueda cargarse dinámicamente en el kernel.

Compilar un kernel personalizado es uno de los ritos de iniciación más importantes a los que los usuarios de casi todos los BSD han de enfrentarse. Este proceso lleva su tiempo, pero le aportará grandes beneficios a su sistema FreeBSD. A diferencia del kernel `GENERIC`, con el que puede funcionar una cantidad apabullante de hardware disponible en el mercado, un kernel personalizado contiene únicamente lo necesario para que funcione el hardware de *su* sistema. Obviamente esto tiene sus ventajas:

- Mayor rapidez en el arranque del sistema. Dado que el kernel sólo tiene que probar el hardware que realmente está en el sistema el tiempo que necesitará para arrancar se reducirá visiblemente.
- Menor uso de memoria. Generalmente un kernel personalizado utiliza menos memoria que un kernel `GENERIC`, lo cual es importante ya que el kernel debe encontrarse siempre en memoria real. Por ésta razón un kernel personalizado puede ser de gran utilidad en sistemas con una cantidad limitada de memoria RAM.
- Hacer funcionar hardware específico. Un kernel personalizado le permite añadir al sistema dispositivos como tarjetas de sonido, que no están incluidas en el kernel `GENERIC`.

8.3. Inventario de hardware del sistema

Escrito por Tom Rhodes.

Antes de aventurarnos en la configuración del kernel nunca está de más disponer de un inventario completo del hardware que hay en la máquina. En los casos en los que FreeBSD no es el sistema operativo principal de la máquina puede conseguir el inventario consultando la configuración de otro sistema operativo. Por ejemplo, el Gestor de dispositivos de Microsoft® suele contener información importante sobre los dispositivos que haya en la máquina. El Gestor de dispositivos está en el panel de control.

Nota

Algunas versiones de Microsoft® Windows® tienen un icono de Sistema con forma de pantalla desde el que se puede acceder al Gestor de dispositivos.

Si no hay otro sistema operativo en la máquina el administrador tendrá que conseguir la información por su cuenta. Una forma de hacerlo es mediante dmesg(8) y man(1). La mayoría de los controladores de dispositivo que hay en FreeBSD tienen una página de manual, que suele incluir una lista del hardware que puede hacer funcionar. En la prueba del sistema durante el arranque va apareciendo una lista del harware que hay en la máquina. Por ejemplo, las siguientes líneas indican que el controlador `psm` ha encontrado un ratón:

```
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: [GIANT-LOCKED]
psm0: [ITHREAD]
psm0: model Generic PS/2 mouse, device ID 0
```

Puede añadir este controlador al fichero de configuración de su kernel personalizado o cargarlo utilizando loader.conf(5).

A veces los datos que muestra `dmesg` solamente muestran mensajes del sistema en lugar de la salida de la prueba de arranque. En esos casos puede ver la verdadera salida de `dmesg` en el fichero `/var/run/dmesg.boot` .

Otro método para identificar el hardware del sistema es usar pciconf(8), que tiene una salida mucho más verbosa. Veamos un ejemplo:

```
ath0@pci0:3:0:0:        class=0x020000 card=0x058a1014 ↺
chip=0x1014168c rev=0x01 hdr=0x00
    vendor     = 'Atheros Communications Inc.'
    device     = 'AR5212 Atheros AR5212 802.11abg wireless'
    class      = network
    subclass   = ethernet
```

Este fragmento, obtenido con `pciconf -lv` muestra cómo el controlador `ath` ha encontrado un dispositivo Ethernet inalámbrico. Escriba `man ath` para consultar la página de manual de ath(4).

La opción `-k` de man(1) le suministrará valiosa información. Sigamos con el ejemplo anterior:

```
# man -k Atheros
```

Con esto obtendremos una lista de páginas de manual que contienen esa palabra en particular:

```
ath(4)                   - Atheros IEEE 802.11 wireless network ↺
driver
ath_hal(4)               - Atheros Hardware Access Layer (HAL)
```

Una vez que tenemos nuestra flamante lista de hardware del sistema el proceso de compilar un kernel personalizado debería parecer un poco menos inquietante.

8.4. Controladores del kernel, subsistemas y módulos

Antes de compilar un kernel personalizado conviene pensar antes los motivos. Si lo que se necesita es añadir hardware específico al sistema es posible que sea posible utilizarlo mediante un módulo.

Los módulos del kernel están en el directorio `/boot/kernel` y se pueden cargar dinámicamente en un kernel en funcionamiento con kldload(8). La mayoría, por no decir todos, los controladores del kernel tienen un módulo específico y una página de manual. Por ejemplo, en la sección anterior hablábamos del controlador Ethernet inalámbrico `ath`. La página de manual de este dispositivo contiene la siguiente información:

```
Alternatively, to load the driver as a module at boot time, place the
     following line in loader.conf(5):

           if_ath_load="YES"
```

Si hace lo que dice ahí, es decir, añadir la línea `if_ath_load="YES"` al fichero `/boot/loader.conf` hará que el módulo se cargue dinámicamente cada vez que el sistema arranque.

Hay casos, no obstante, en los que no existe el módulo correspondiente. Esto sucede con ciertos subsistemas y con controladores muy importantes, como por ejemplo el de FFS, que necesita el kernel. Igual sucede con la posibilidad de utilizar redes (INET). Por desgracia, en estos casos la única forma de saber si hace falta un controlador es buscar el módulo.

Aviso

Es sumamente fácil quitar un dispositivo o una opción del kernel y encontrarse de manos a boca con un kernel que no funciona. Por ejemplo, si elimina el controlador ata(4) del fichero de configuración del kernel un sistema que utilice unidades de disco ATA no podrá arrancar a menos que se añada la línea correspondiente al fichero `loader.conf` . Si tiene dudas busque el módulo y deje el kernel como está.

8.5. Compilación e instalación de un kernel personalizado

En primer lugar, hagamos un breve recorrido por el directorio donde se lleva a cabo la compilación del kernel. Todos los directorios mencionados hacen referencia al directorio

principal, que es `/usr/src/sys` , al que también podemos acceder desde `/sys`. Existen gran cantidad de subdirectorios que representan diferentes partes del kernel, pero el más importante para lo que deseamos hacer son `arch` y `/conf`, que es donde se llevara a cabo la edición del fichero de configuración y la compilación propiamente dicha del mismo, el punto de partida para la personalización del kernel. El directorio `arch` representa la arquitectura del sistema, por lo que puede ser `i386`, `alpha`, `amd64`, `ia64`, `powerpc`, `sparc64` o `pc98` (una arquitectura alternativa, similar a PC y muy popular en Japón). Todo lo que existe dentro de un directorio de una arquitectura en particular, es específico para dicha arquitectura; el resto del código es común para todas las plataformas en las que FreeBSD puede funcionar. Observe la organización lógica de la estructura de los directorios, con cada dispositivo utilizable en el sistema, sistema de ficheros y opciones, cada cosa en su propio subdirectorio.

A lo largo de este capítulo asumiremos que está utilizando la arquitectura i386, puesto que sobre ella son los ejemplos que vamos a ir exponiendo. Si no es este su caso debe hacer los cambios adecuados a su arquitectura de hardware en las rutas que vayamos utilizando.

Nota

Si *no existe* el directorio `/usr/src/sys` en su sistema quiere decir que no están instaladas las fuentes del kernel. La forma más sencilla de tenerlas es mediante `/stand/sysinstall`. Como usuario `root` seleccione `Configure`, luego `Distributions`, después `src` y finalmente `sys`. Si no le gusta mucho sysinstall y tiene acceso a un CDROM "oficial" de FreeBSD puede instalar las fuentes por medio de la siguiente línea de órdenes:

```
# mount /cdrom
# mkdir -p /usr/src/sys
# ln -s /usr/src/sys /sys
# cat /cdrom/sys/ssys.[a-d]* | tar -xzvf
```

Ahora vaya al al directorio `arch/conf` y copie el fichero de configuración `GENERIC` con el nombre que desee. Por ejemplo:

```
# cd /usr/src/sys/i386/conf
# cp GENERIC MIKERNEL
```

Por tradición el nombre se escribe con mayúsuculas y si tiene varias maquinas FreeBSD, con diferente hardware es recomendable darle a cada kernel el mismo nombre que la máquina en la que va a ejecutarse. En este ejemplo usaremos el nombre de `MIKERNEL` .

Sugerencia

Guardar su fichero de configuración directamente bajo el directorio `/usr/src` , puede que no sea una idea muy astuta. Si empieza a tener problemas puede ser muy tentador borrar `/usr/src` y comenzar desde cero. Cinco segundos después de haber hecho esto se dará cuenta de que ha eliminado el fichero de configuración de su kernel, en el que quizás llevaba horas trabajando.

Le proponemos una alternativa: guardar su fichero de configuración cualquier otro sitio y crear un enlace simbólico al fichero que hay en el directorio `i386`.

Por ejemplo:

```
# cd /usr/src/sys/i386/conf
# mkdir /root/kernels
# cp GENERIC /root/kernels/MIKERNEL
# ln -s /root/kernels/MIKERNEL
```

Ahora edite el fichero de configuración `MIKERNEL` con su editor de textos favorito. Si se trata de una instalación reciente probablemente el único editor disponible sea vi, del cual es complicado explicar su uso detallado en este documento, pero existen bastantes libros que detallan su uso; puede ver algunos en la bibliografía. FreeBSD dispone de un editor de uso muy sencillo, llamado ee; si es usted principiante le será de mucha ayuda. Cambie los comentarios al principio del fichero con algo que refleje los cambios realizados o al menos para diferenciarlo del fichero `GENERIC`.

Si ha compilado un kernel en SunOS™ o algún otro sistema operativo BSD seguramente la mayor parte de fichero le será muy familiar. Si viene usted de otros sistemas operativos como DOS, el fichero `GENERIC` puede parecerle intimidante, así que siga las instrucciones descritas en El Fichero de Configuración detenidamente y con sumo cuidado.

Nota

Asegúrese siempre de verificar el fichero `/usr/src/UPDATING` antes de realizar cualquier actualización del sistema si ha sincronizado sus fuentes para disponer de la última versión de los mismos. En el fichero `/usr/src/UPDATING` está toda la información importante sobre las actualizaciones. Al distribuirse con FreeBSD dicha infor-

mación deberá estar más actualizada que la que hay en este mismo texto.

Ahora es momento de llevar a cabo la compilación del código fuente del kernel.

Procedimiento 8.1. Compilación del kernel

1. Vaya al directorio `/usr/src` :

```
# cd /usr/src
```

2. Compile el kernel:

```
# make buildkernel KERNCONF=MIKERNEL
```

3. Instale el nuevo kernel:

```
# make installkernel KERNCONF=MIKERNEL
```

Nota

Es imprescindible que disponga del código fuente completo de FreeBSD para poder compilar el kernel.

Sugerencia

Por omisión, si compila un kernel personalizado *todos* los módulos del kernel serán recompilados también. Si quiere que la actualización de su kernel sea mucho más rápida o compilar solamente ciertos módulos personalizados, edite kernel modules will be rebuilt as well. `/etc/make.conf` antes de compilar el kernel:

```
MODULES_OVERRIDE = linux acpi sound/sound sound/
driver/ds1 ntfs
```

Esta variable crea una lista de módulos listos para ser compilados, sin tocar los demás.

```
WITHOUT_MODULES = linux acpi sound/sound sound/
driver/ds1 ntfs
```

Esta variable crea una lista de módulos que serán excluidos del proceso de compilación. Hay otras variables que pueden resultar-

le útiles durante este proceso; consulte la página de manual de make.conf(5).

El nuevo kernel se copiará al directorio raíz como `/kernel` y el kernel viejo tendrá el nombre de `/kernel.old` . Ahora reinicie su sistema para poder probar su nuevo kernel. En caso de que se presente algun problema hay algunos consejos para la resolución de problemas al final de este capítulo. Asegúrese de leer la sección que explica cómo recuperar el sistema en caso de que su kernel no quiera arrancar.

Nota

Ciertos ficheros relacionados con el proceso de arranque, como loader(8)y su configuración, se guardan en `/boot`. Los módulos personalizados o de terceros se suelen guardar en `/boot/kernel` , aunque debe tener presente que es muy importante que los módulos y el kernel estén sincronizados. Si intenta usar módulos con un kernel para el que no están destinados puede haber errores o incluso producirse intesatabilidades.

8.6. El fichero de configuración

Actualizado a FreeBSD 6.X por Joel Dahl.

El formato de un fichero de configuración es bastante simple. Cada línea contiene una palabra clave con uno o más argumentos. Para simplificar, hay muchas líneas que solamente contienen un argumento. Cualquier cosa detrás de un `#` se considerará un comentario y en consecuencia será ignorado. Las siguientes secciones describen todas las palabras clave en el orden en el que aparecen en `GENERIC`. Si quiere una lista exhaustiva de de opciones dependientes de arquitectura y de dispositivos puede consultar el fichero `NOTES` en el mismo directorio donde está el fichero `GENERIC`. Si quiere ver las opciones independientes de arquitectura consulte `/usr/src/sys/conf/NOTES` .

Nota

Puede generar un fichero que contenga todas las opciones disponibles en un kernel. Esto es algo que solamente se usa para hacer pruebas. Si quiere generarlo ejecute, como `root`, lo siguiente:

```
# cd /usr/src/sys/i386/conf && make LINT
```

A continuación veremos un ejemplo de fichero de configuración de un kernel `GENERIC` al que se han añadido comentarios adicionales donde se ha visto que era necesario abundar un poco para mayor claridad. Este ejemplo es igual (o, en el peor de los casos, casi igual) que la copia del mismo que tiene usted en `/usr/src/sys/i386/conf/GENERIC` .

```
machine  i386
```

Esta es la arquitectura de la máquina. Debe ser `alpha`, `amd64`, `i386`, `ia64`, `pc98`, `powerpc` o `sparc64`.

```
cpu          I486_CPU
cpu          I586_CPU
cpu          I686_CPU
```

Las opciones anteriores definen el tipo de CPU que haya en su sistema. Puede dejar varias líneas de CPU (si, por ejemplo, no está seguro de usar `I586_CPU` o `I686_CPU`), pero si está personalizando su kernel es mucho mejor que solamente ponga la CPU que tenga. Si no está seguro de la CPU que tiene busque en los mensajes de arranque que se guardan en `/var/run/dmesg.boot` .

```
ident          GENERIC
```

Este es el identificador del kernel. Modifíquelo para que cuadre con el nombre que le haya dado a su kernel, es decir, `MIKERNEL` si ha seguido los ejemplos anteriores. El valor que asigne a la cadena `ident` será el que se muestre cuando arranque con su kernel, así que es útil darle a su kernel un nombre distintivo que permita distinguirlo fácilmente de otros, por ejemplo, si está compilando un kernel experimental.

```
#To statically compile in device wiring instead of /boot/device.hints
#hints          "GENERIC.hints"         # Default places to look ↺
for devices.
```

device.hints(5) se usa para configurar opciones de controladores de dispositivo. La ubicación por defecto en la que loader(8) buscará durante el arranque es `/boot/device.hints` . Si usa la opción `hints` puede compilar el contenido de `device.hints` en su kernel, lo que hará innecesario crear ese fichero en `/boot`.

```
makeoptions     DEBUG=-g          # Build kernel with gdb(1) debug ↺
symbols
```

El proceso normal de compilación en FreeBSD incluye información de depuración de errores si se compila un kernel con la opción `-g`, que activa la información de depuración de errores al pasar a gcc(1).

```
options          SCHED_4BSD         # 4BSD scheduler
```

El planificador de tareas tradicional y por omisión de FreeBSD. Déjelo como está.

```
options         PREEMPTION          # Enable kernel thread preemption
```

Permite que hilos que están en el kernel puedan asociarse con hilos cuya prioridad sea más alta. Ayuda con la interactividad y permite que los hilos interrumpidos puedan ejecutarse antes en lugar de tener que esperar.

```
options         INET                # InterNETworking
```

Networking. Deje esto como está, incluso si no tiene en mente conectar la máquina a una red. Muchos programas necesitan al menos disponer de lo que se llama «loopback networking» (esto es, poder efectuar conexiones de red con su propia máquina) así que tener esto es obligatorio.

```
options         INET6               # IPv6 communications protocols
```

Activa los protocolos de comunicación IPv6.

```
options         FFS                 # Berkeley Fast Filesystem
```

El sistema de ficheros básico para discos duros. Debe dejarlo como está si pretende poder arrancar desde disco duro.

```
options         SOFTUPDATES         # Enable FFS Soft Updates support
```

Activa Soft Updates en el kernel, lo que acelerará los accesos de escritura a sus discos. Esta funcionalidad la facilita el kernel, pero debe activarse para cada disco de forma específica. Revise la salida de mount(8) y verá si Soft Updates está activado en los discos de su sistema. Si no aparece la opción `soft-updates` actívela mediante tunefs(8) (para sistemas de ficheros ya existentes) o newfs(8) (en el caso de sistemas de ficheros nuevos).

```
options         UFS_ACL             # Support for access control lists
```

Esta opción activa en el kernel las listas de control de acceso. Consiste en el uso de atributos extendidos y UFS2 junto con las características que se describen detalladamente en la Sección 14.12, “Listas de control de acceso a sistemas de ficheros”. Las ACL por omisión están activadas, y no deben desactivarse del kernel una vez que hayan sido usadas en un sistema de ficheros puesto que eliminará las listas de control de acceso y el modo en el que se protegen esos ficheros de un modo totalmente fuera de control.

```
options         UFS_DIRHASH         # Improve performance on big ↺
directories
```

This option includes functionality to speed up disk operations on large directories, at the expense of using additional memory. You would normally keep this for a large server, or interactive workstation, and remove it if you are using FreeBSD on a smaller system where memory is at a premium and disk access speed is less important, such as a firewall.

```
options         MD_ROOT             # MD is a potential root device
```

Esta opción permite que la partición raíz esté en un disco virtual basado en memoria.

```
options         NFSCLIENT       # Network Filesystem Client
options         NFSSERVER       # Network Filesystem Server
options         NFS_ROOT        # NFS usable as /, requires ↺
NFSCLIENT
```

NFS, el sistema de ficheros en red. Salvo que tenga intención de montar particiones de sistemas de ficheros UNIX® de un servidor a través de TCP/IP puede comentar estas opciones.

```
options         MSDOSFS         # MSDOS Filesystem
```

El sistema de ficheros MS-DOS®. Salvo que tenga en mente montar particiones de disco duro con formato DOS durante el arranque puede comentar esta opción. En caso de necesidad esta funcionalidad se cargará automáticamente. También tiene a su emulators/mtools, que le permitirá acceder a disquetes DOS sin tener que montarlos y desmontarlos (y ni siquiera requiere `MSDOSFS`).

```
options         CD9660          # ISO 9660 Filesystem
```

El sistema de ficheros ISO 9660 para CDROM. Coméntelo si no tiene unidad CDROM o solamente monta CD muy de vez en cuando; cuando lo necesite el sistema lo cargará dinámicamente. Los CD de sonido no utilizan este sistema de ficheros.

```
options         PROCFS          # Process filesystem (requires ↺
PSEUDOFS)
```

El sistema de ficheros de procesos. Es un sistema de ficheros "simulado" que se monta en `/proc` y permite a programas como ps(1) suministrar más información sobre qué procesos están ejecutándose. En la mayoaría de los casos no es necesario usar `PROCFS`, puesto que la mayoría de las herramientas de monitorización y depuración han sido adaptadas para que funcionen sin `PROCFS`. De hecho aunque lo instale el sistema no lo montará por omisión.

```
options         PSEUDOFS        # Pseudo-filesystem framework
```

Los kernel 6.X pueden usar `PSEUDOFS` al utilizar `PROCFS`.

```
options         GEOM_GPT        # GUID Partition Tables.
```

Con esta opción se puede tener una gran cantidad de particiones en un único disco.

```
options         COMPAT_43       # Compatible with BSD 4.3 [KEEP ↺
THIS!]
```

Compatibilidad con 4.3BSD. Déjelo como está; ciertos programas pueden comportarse de formas muy extrañas si comenta esta opción.

```
options         COMPAT_FREEBSD4 # Compatible with FreeBSD4
```

FreeBSD 5.X en sistemas i386™ y Alpha necesita esta opción para poder usar aplicaciones compiladas en versiones antiguas de FreeBSD que utilizan, por tanto, llamadas al sistema

más antiguas. Esta opción no es necesaria en plataformas en las que funciona FreeBSD desde 5.X, como ia64 y Sparc64®.

```
options         COMPAT_FREEBSD5   # Compatible with FreeBSD5
```

Esta opción hace falta en sistemas FreeBSD 6.X y versiones posteriores para poder ejecutar aplicaciones compiladas en FreeBSD 5.X, que usan interfaces de llamada al sistema FreeBSD 5.X.

```
options         SCSI_DELAY=5000  # Delay (in ms) before probing SCSI
```

Hace que el kernel haga una pausa de 5 segundos antes de probar los dispositivos SCSI del sistema. Si solamente tiene discos IDE puede ignorar esta opción, o también puede asignarle un valor menos para evitar el retardo en el arranque. Si lo hace y FreeBSD tiene problemas para reconocer dispositivos SCSI en el sistema es obvio que tendrá que incrementar el valor.

```
options         KTRACE            # ktrace(1) support
```

Activa las trazas en el kernel, algo muy útil para la depuración de errores.

```
options         SYSVSHM           # SYSV-style shared memory
```

Facilita memoria compartida System V. El uso más habitual es la extensión XSHM de X, que utiliza la mayoría de programas que hacen uso intensivo de los gráficos para incrementar la velocidad. Si usa X es casi seguro que le vendrá bien esta opción.

```
options         SYSVMSG           # SYSV-style message queues
```

Mensajes System V. Esta opción añade solamente unos cuantos bytes al kernel.

```
options         SYSVSEM           # SYSV-style semaphores
```

Semáforos System V. No es demasiado frecuente que se utilicen, pero solamente añaden unos cuantos cientos de bytes al kernel.

Nota

La opción `-p` de ipcs(1) le mostrará una lista de procesos que estén utilizando características System V.

```
options       _KPOSIX_PRIORITY_SCHEDULING # POSIX P1003_1B real-
time extensions
```

Extensiones en tiempo real añadidas en 1993 POSIX®. Ciertas aplicaciones de la Colección de Ports las utilizan, por ejemplo StarOffice™.

```
options         KBD_INSTALL_CDEV  # install a CDEV entry in /dev
```

This option is required to allow the creation of keyboard device nodes in `/dev`.

```
options         ADAPTIVE_GIANT    # Giant mutex is adaptive.
```

Giant es un mecanismo de exclusión mutua («sleep mutex») que protege un gran conjunto de recursos del kernel. Hoy en dia no es asumible tener un cuello de botella así por el impacto que tiene en el rendimiento, así que está siendo reemplazado por bloqueos que protegen los recursos de manera individual. `ADAPTIVE_GIANT` hace que Giant sea incluido en un conjunto de «mutextes» que va rotando Esto es, cuando un hilo quiere bloquear el Giant mutex (pero ya está bloqueado por un hilo de otra CPU) el primer hilo seguirá ejecutándose a la espera de que se libere el bloqueo. Generalmente el hilo volverá al estado de reposo y esperará hasta que aparezca otra oportunidad de ejecutarse. Si no está seguro de lo que está haciendo es mejor que deje esta opción tal y como está.

Nota

Tenga en cuenta que en FreeBSD 8.0-CURRENT y versiones siguientes todos los «mutexes» son adaptables por omisión, salvo que se use la opción `NO_ADAPTIVE_MUTEXES` . El resultado evidente que es Giant es adaptable por omisión, así que la opción `ADAPTIVE_GIANT` ha sido eliminada de la configuración.

```
device          apic              # I/O APIC
```

El dispositivo apic activa la E/S APIC en la entrega de interrupciones. El dispositivo apic puede usarse tanto en kernels para un procesador (UP) como para sistemas multiprocesador (SMP). Si añada `options SMP` funcionará en sistemas multiprocesador.

Nota

El dispositivo apic existe solamente en la arquitectura i386, de modo que no debe usarse esta línea en otras arquitecturas.

```
device          eisa
```

Use esta opción si tiene una placa base EISA. Activa la detección automática y permite la configuración de todos los dispositivos que estén en el bus EISA.

```
device          pci
```

Use esta opción si tiene una placa PCI. Permite la detección automática de tarjetas PCI y permite la configuración entre el bus ISA y el PCI.

```
# Floppy drives
device          fdc
```

Este dispositivo es el controlador de la unidad de disquetes.

```
# ATA and ATAPI devices
device          ata
```

Este controlador permite utilizar dispositivos ATA y ATAPI. Si añade al kernel one `device ata` éste detectará cualquier dispositivo ATA/ATAPI PCI que conecte a una máquina moderna.

```
device          atadisk                 # ATA disk drives
```

Si usa `device ata` tendrá que añadir también esto para poder usar unidades de disco ATA.

```
device          ataraid                 # ATA RAID drives
```

Si usa `device ata` tendrá que añadir también esto para poder usar unidades de disco ATA RAID.

```
device          atapicd                 # ATAPI CDROM drives
```

Si usa `device ata` tendrá que añadir también esto para poder usar unidades ATAPI CDROM.

```
device          atapifd                 # ATAPI floppy drives
```

Si usa `device ata` tendrá que añadir también esto para poder usar unidades de disquete ATAPI.

```
device          atapist                 # ATAPI tape drives
```

Si usa `device ata` tendrá que añadir también esto para poder usar unidades de cinta ATAPI.

```
options         ATA_STATIC_ID           # Static device numbering
```

Con esta opción hace que el número de controladores sea estático; si no se usa los números de dispositivo se asignan dinámicamente.

```
# SCSI Controllers
device          ahb        # EISA AHA1742 family
device          ahc        # AHA2940 and onboard AIC7xxx devices
options         AHC_REG_PRETTY_PRINT    # Print register bitfields ↺
in debug
                                        # output.  Adds ~128k to ↺
driver.
device          ahd        # AHA39320/29320 and onboard AIC79xx ↺
devices
```

```
options         AHD_REG_PRETTY_PRINT    # Print register bitfields ↺
in debug
                                        # output.  Adds ~215k to ↺
driver.
device          amd         # AMD 53C974 (Teckram DC-390(T))
device          isp         # Qlogic family
#device         ispfw       # Firmware for QLogic HBAs- normally a ↺
module
device          mpt         # LSI-Logic MPT-Fusion
#device         ncr         # NCR/Symbios Logic
device          sym         # NCR/Symbios Logic (newer chipsets + ↺
those of `ncr')
device          trm         # Tekram DC395U/UW/F DC315U adapters

device          adv         # Advansys SCSI adapters
device          adw         # Advansys wide SCSI adapters
device          aha         # Adaptec 154x SCSI adapters
device          aic         # Adaptec 15[012]x SCSI adapters, ↺
AIC-6[23]60.
device          bt          # Buslogic/Mylex MultiMaster SCSI adapters

device          ncv         # NCR 53C500
device          nsp         # Workbit Ninja SCSI-3
device          stg         # TMC 18C30/18C50
```

Controladoras SCSI. Coméntelas si no las tiene en su sistema. Si en su sistema tiene solamente unidades IDE puede borrarlas todas. Las líneas `*_REG_PRETTY_PRINT` son opciones de depuración de errores de sus respectivos controladores de dispositivo.

```
# SCSI peripherals
device          scbus       # SCSI bus (required for SCSI)
device          ch          # SCSI media changers
device          da          # Direct Access (disks)
device          sa          # Sequential Access (tape etc)
device          cd          # CD
device          pass        # Passthrough device (direct SCSI access)
device          ses         # SCSI Environmental Services (and SAF-TE)
```

Periféricos SCSI. Estos también puede borrarlos sin problemas si no los tiene en su sistema o si solamente tiene hardware IDE.

Nota

El controlador USB umass(4) (y unos cuantos controladores más) utilizan el subsistema SCSI aunque no sean dispositivos SCSI reales. No elimine el subsistema SCSI del kernel si va a utilizar cualquiera de estos controladores.

```
# RAID controllers interfaced to the SCSI subsystem
device          amr        # AMI MegaRAID
device          arcmsr     # Areca SATA II RAID
device          asr        # DPT SmartRAID V, VI and Adaptec SCSI ↺
RAID
device          ciss       # Compaq Smart RAID 5*
device          dpt        # DPT Smartcache III, IV - See NOTES for ↺
options
device          hptmv      # Highpoint RocketRAID 182x
device          rr232x     # Highpoint RocketRAID 232x
device          iir        # Intel Integrated RAID
device          ips        # IBM (Adaptec) ServeRAID
device          mly        # Mylex AcceleRAID/eXtremeRAID
device          twa        # 3ware 9000 series PATA/SATA RAID

# RAID controllers
device          aac        # Adaptec FSA RAID
device          aacp       # SCSI passthrough for aac (requires CAM)
device          ida        # Compaq Smart RAID
device          mfi        # LSI MegaRAID SAS
device          mlx        # Mylex DAC960 family
device          pst        # Promise Supertrak SX6000
device          twe        # 3ware ATA RAID
```

Controladoras RAID que pueden utilizarse en FreeBSD. Si no las tiene en su sistema puede borrarlas.

```
# atkbdc0 controls both the keyboard and the PS/2 mouse
device          atkbdc     # AT keyboard controller
```

El controlador de teclado (`atkbdc`) ofrece servicios de E/S con teclados AT y ratones PS/2. El controlador de teclado (`atkbd`) y el controlador de ratón PS/2 (`psm`) necesitan este dispositivo.

```
device          atkbd      # AT keyboard
```

El controlador `atkbd`, junto con el controlador `atkbdc`, permiten utilizar el teclado AT 84 u otros tipos de teclados AT mejorados que se conecten mediante el controlador de teclado AT.

```
device          psm        # PS/2 mouse
```

Utilice este dispositivo si conecta su ratón en el puerto PS/2.

```
device          kbdmux        # keyboard multiplexer
```

Funcionalidad básica para múltiples teclados. Si no tiene en mente usar más de un teclado en el sistema puede borrar esta línea sin mayor problema.

```
device          vga        # VGA video card driver
```

El controlador de la tarjeta gráfica.

```
device          splash     # Splash screen and screen saver support
```

«Splash screen» en el arranque. Los salvapantallas necesitan este dispositivo.

```
# syscons is the default console driver, resembling an SCO console
device          sc
```

`sc` por omisión es el controlador de dispositivo de la consola; se parece mucho a una consola de SCO. Dado que muchos programas de pantalla completa acceden a la consola a través de la biblioteca de bases de datos de terminal `termcap` no tiene demasiada importancia si usa `vt`, el controlador de consola compatible `VT220`. Cuando acceda al sistema asigne a su variable `TERM` el valor `scoansi`si los programas a pantalla completa tienen algún problema para acceder a la consola.

```
# Enable this for the pcvt (VT220 compatible) console driver
#device          vt
#options         XSERVER          # support for X server on a vt ↺
console
#options         FAT_CURSOR       # start with block cursor
```

El controlador de dispositivo VT220-compatible; es compatible con VT100/102, anterior a él. Funciona bien en ciertos sistemas portátiles que adolecen de incompatibilidad de harware con `sc`. Asigne a su variable de entorno `TERM` el valor `vt100` o `vt220` cuando acceda al sistema. Este controlador le puede ser de utilidad si tiene que acceder a gran cantidad de máquinas a través de una red, una situación en la que suele suceder que `termcap` o `terminfo` no están ahí para que las use `sc`. `vt100`, por el contrario, debería aparecer en práticamente cualquier plataforma.

```
device          agp
```

Utilice esta opción si tiene en el sistema una tarjeta AGP. Activará AGP y también AGP GART si su tarjeta puede usarla.

```
# Power management support (see NOTES for more options)
#device          apm
```

Gestión avanzada de la energía. Muy útil en sistemas portátiles. Viene desactivada por omisión en el kernel `GENERIC` .

```
# Add suspend/resume support for the i8254.
device          pmtimer
```

Controlador del reloj para eventos de gestión de la energía, como APM y ACPI.

```
# PCCARD (PCMCIA) support
# PCMCIA and cardbus bridge support
device          cbb             # cardbus (yenta) bridge
device          pccard          # PC Card (16-bit) bus
device          cardbus         # CardBus (32-bit) bus
```

Dispositivos PCMCIA. Si el sistema es portátil necesita tener esto activado.

```
# Serial (COM) ports
device          sio             # 8250, 16[45]50 based serial ports
```

Estos son los puertos serie a los que se conoce como puertos `COM` en entornos MS-DOS®/Windows®.

Nota

Si tiene un módem interno en `COM4` un puerto serie en `COM2` tendrá que asignar a la IRQ del módem el 2 (por razones técnicas ignotas IRQ2 = IRQ 9) para que pueda acceder al dispositivo desde FreeBSD. Si tiene una tarjeta serie multipuerto consulte la página de manual de sio(4) si quiere más detalles sobre los valores que debe añadir a `/boot/device.hints` . Algunas tarjetas gráficas (sobre todo las que usan chips S3) utilizan direcciones IO del tipo `0x*2e8` y dado que muchas tarjetas serie de baja calidad no decodifican correctamente el espacio de direcciones de 16 bits chocan con estas tarjetas, haciendo que el puerto `COM4` sea prácticamente inútil.

Es necesario que cada puerto serie tenga una IRQ única (salvo que use una tarjeta multipuerto que permita compartir interrupciones), así que las IRQ de `COM3` y de `COM4` no se pueden utilizar.

```
# Parallel port
device          ppc
```

El interfaz del puerto paralelo de bus ISA.

```
device          ppbus       # Parallel port bus (required)
```

El bus del puerto paralelo.

```
device          lpt         # Printer
```

Permite usar el puerto paralelo para conectar impresoras.

Nota

Necesitará disponer de los tres anteriores para poder utilizar impresoras mediante el puerto paralelo.

```
device          plip       # TCP/IP over parallel
```

Este controlador es para la interfaz de red a través del puerto paralelo.

```
device          ppi        # Parallel port interface device
```

La E/S de propósito general (conocida también como “puerto geek”) + E/S IEEE1284.

```
#device         vpo        # Requires scbus and da
```

Este dispositivo se usa con unidades Iomega Zip. Necesita `scbus` y `da` . El mejor rendimiento se alcanza con el uso de los puertos en modo EPP 1.9.

```
#device         puc
```

Puede utilizar este dispositivo si tiene una tarjeta PCI “tonta” (por puerto serie o paralelo) que funcione mediante el controlador puc(4).

```
# PCI Ethernet NICs.
device          de         # DEC/Intel DC21x4x (“Tulip”)
device          em         # Intel PRO/1000 adapter Gigabit ↺
Ethernet Card
device          ixgb       # Intel PRO/10GbE Ethernet Card
device          txp        # 3Com 3cR990 (“Typhoon”)
device          vx         # 3Com 3c590, 3c595 (“Vortex”)
```

Diversos controladores para tarjetas de red PCI. Puede borrar todas las que no estén en su sistema.

```
# PCI Ethernet NICs that use the common MII bus controller code.
# NOTE: Be sure to keep the 'device miibus' line in order to use ↺
these NICs!
device          miibus     # MII bus support
```

El poder utilizar bus MII es necesario para ciertas tarjetas Ethernet PCI 10/100, más concretamente las que usan transceptores compatibles con MII o implementan interfaces de control de transceptores que funcionan como si fueran MII. Si incluye `device miibus` al kernel dispondrá de la API miibus genérica y todos los controladores PHY, incluyendo uno que hará funcionar hardware que, siendo del tipo PHY, no está bajo ninguno de los controladores PHY específicos.

```
device          bce        # Broadcom BCM5706/BCM5708 Gigabit ↺
Ethernet
device          bfe        # Broadcom BCM440x 10/100 Ethernet
device          bge        # Broadcom BCM570xx Gigabit Ethernet
device          dc         # DEC/Intel 21143 and various workalikes
device          fxp        # Intel EtherExpress PRO/100B (82557, ↺
82558)
device          lge        # Level 1 LXT1001 gigabit ethernet
```

```
device          msk         # Marvell/SysKonnect Yukon II Gigabit ↺
Ethernet
device          nge         # NatSemi DP83820 gigabit ethernet
device          nve         # nVidia nForce MCP on-board Ethernet ↺
Networking
device          pcn         # AMD Am79C97x PCI 10/100 (precedence over
 'lnc')
device          re          # RealTek 8139C+/8169/8169S/8110S
device          rl          # RealTek 8129/8139
device          sf          # Adaptec AIC-6915 (“Starfire”)
device          sis         # Silicon Integrated Systems SiS 900/SiS ↺
7016
device          sk          # SysKonnect SK-984x & SK-982x gigabit ↺
Ethernet
device          ste         # Sundance ST201 (D-Link DFE-550TX)
device          stge        # Sundance/Tamarack TC9021 gigabit ↺
Ethernet
device          ti          # Alteon Networks Tigon I/II gigabit ↺
Ethernet
device          tl          # Texas Instruments ThunderLAN
device          tx          # SMC EtherPower II (83c170 “EPIC”)
device          vge         # VIA VT612x gigabit ethernet
device          vr          # VIA Rhine, Rhine II
device          wb          # Winbond W89C840F
device          xl          # 3Com 3c90x (“Boomerang”, “Cyclone”)
```

Controladores que utilizan el código del controlador de bus MII.

```
# ISA Ethernet NICs.  pccard NICs included.
device          cs          # Crystal Semiconductor CS89x0 NIC
# 'device ed' requires 'device miibus'
device          ed          # NE[12]000, SMC Ultra, 3c503, DS8390 ↺
cards
device          ex          # Intel EtherExpress Pro/10 and Pro/10+
device          ep          # Etherlink III based cards
device          fe          # Fujitsu MB8696x based cards
device          ie          # EtherExpress 8/16, 3C507, StarLAN 10 ↺
etc.
device          lnc         # NE2100, NE32-VL Lance Ethernet cards
device          sn          # SMC's 9000 series of Ethernet chips
device          xe          # Xircom pccard Ethernet

# ISA devices that use the old ISA shims
#device         le
```

Controladores Ethernet ISA. Consulte `/usr/src/sys/i386/conf/NOTES` para más detalles sobre qué tarjetas hace funcionar qué controlador.

```
# Wireless NIC cards
device          wlan            # 802.11 support
```

802.11 genérico. Necesitará esta línea si va a usar redes inalámbricas.

```
device          wlan_wep        # 802.11 WEP support
device          wlan_ccmp       # 802.11 CCMP support
device          wlan_tkip       # 802.11 TKIP support
```

Criptografía en dispositivos 802.11. Necesita estas líneas si quiere utilizar criptografía y protocolos de seguridad 802.11.

```
device          an         # Aironet 4500/4800 802.11 wireless NICs.
device          ath             # Atheros pci/cardbus NIC's
device          ath_hal         # Atheros HAL (Hardware Access Layer)
device          ath_rate_sample # SampleRate tx rate control for ath
device          awi        # BayStack 660 and others
device          ral        # Ralink Technology RT2500 wireless NICs.
device          wi         # WaveLAN/Intersil/Symbol 802.11 ↺
wireless NICs.
#device         wl         # Older non 802.11 Wavelan wireless NIC.
```

Diversas tarjetas inalámbricas.

```
# Pseudo devices
device   loop           # Network loopback
```

El dispositivo de «loopback» para TCP/IP. Si accede por telnet o FTP or FTP to `localhost` también conocido como `127.0.0.1`) lo hará a través de este dispositivo. Es *imprescindible* tenerlo en el sistema.

```
device   random         # Entropy device
```

Generador de números criptográficamente seguro.

```
device   ether          # Ethernet support
```

`ether` solo es necesario si tiene alguna tarjeta Ethernet. Incluye código genérico del protocolo Ethernet.

```
device   sl             # Kernel SLIP
```

`sl` permite utilizar SLIP. Ha sido sustituido casi totalmente por PPP, que es más fácil de usar, está mejor capacitado para la conexión de módem a módem y es, en general, claramente mejor.

```
device   ppp            # Kernel PPP
```

Este dispositivo incluye en el kernel la capacidad de gestionar conexiones de llamada entrante «dial-up». Hay también una versión de PPP implementada como aplicación de usuario; utiliza `tun` y ofrece más flexibilidad y características como la llamada bajo petición.

```
device   tun            # Packet tunnel.
```

Este dispositivo lo usa el software PPP de usuario. Consulte la sección sobre PPP de este mismo libro.

```
device   pty            # Pseudo-ttys (telnet etc)
```

Este dispositivo es una "pseudoterminal", o un puerto de entrada al sistema simulado. Se usa en sesiones entrantes de `telnet` y `rlogin`; también lo usan xterm y otras aplicaciones, entre las que encontramos a Emacs.

```
device   md             # Memory "disks"
```

Pseudodispositivos de disco basados en memoria.

```
device   gif            # IPv6 and IPv4 tunneling
```

Este dispositivo implementa túneles de IPv6 sobre IPv4, IPv4 sobre IPv6, IPv4 sobre IPv4 e IPv6 sobre IPv6. El dispositivo `gif` se puede clonar a sí mismo, así que los nodos de dispositivo se van creando a medida que van haciendo falta.

```
device   faith          # IPv6-to-IPv4 relaying (translation)
```

Este pseudodispositivo captura paquetes que se le hayan enviado y los dirige hacia el dæmon de traducción IPv4/IPv6.

```
# The `bpf' device enables the Berkeley Packet Filter.
# Be aware of the administrative consequences of enabling this!
# Note that 'bpf' is required for DHCP.
device   bpf            # Berkeley packet filter
```

El filtro de paquetes de Berkeley. Este pseudodispositivo permite poner interfaces de red en modo promíscuo, lo que significa que capturan todos los paquetes que circulen por una red broadcast (por ejemplo una Ethernet). Dichos paquetes pueden guardarse en disco para su posterior examen mediante tcpdump(1) (el análisis con tcpdump(1) no puede hacerse directamente también).

Nota

El dispositivo bpf(4) también lo usa dhclient(8) para obtener direcciones IP del encaminador (gateway) por omisión. Si usa DHCP deje esta opción como está.

```
# USB support
device          uhci          # UHCI PCI->USB interface
device          ohci          # OHCI PCI->USB interface
device          ehci          # EHCI PCI->USB interface (USB 2.0)
device          usb           # USB Bus (required)
```

```
#device         udbp            # USB Double Bulk Pipe devices
device          ugen            # Generic
device          uhid            # “Human Interface Devices”
device          ukbd            # Keyboard
device          ulpt            # Printer
device          umass           # Disks/Mass storage - Requires scbus ↺
and da
device          ums             # Mouse
device          ural            # Ralink Technology RT2500USB ↺
wireless NICs
device          urio            # Diamond Rio 500 MP3 player
device          uscanner        # Scanners
# USB Ethernet, requires mii
device          aue             # ADMtek USB Ethernet
device          axe             # ASIX Electronics USB Ethernet
device          cdce            # Generic USB over Ethernet
device          cue             # CATC USB Ethernet
device          kue             # Kawasaki LSI USB Ethernet
device          rue             # RealTek RTL8150 USB Ethernet
```

Diversos dispositivos USB.

```
# FireWire support
device          firewire        # FireWire bus code
device          sbp             # SCSI over FireWire (Requires scbus ↺
and da)
device          fwe             # Ethernet over FireWire (non-
standard!)
```

Diversos dispositivos Firewire.

Tiene más información y una lista con más dispositivos que funcionan en FreeBSD consulte `/usr/src/sys/i386/conf/NOTES` .

8.6.1. Configuraciones con grandes cantidades de memoria (PAE)

Las máquinas que tienen configuraciones con grandes cantidades de memoria necesitan acceder a más de 4 gigabytes de espacio de direcciones KVA (User+Kernel Virtual Address). Debido a esta limitación Intel añadió a las CPU Pentium® Pro y modelos posteriores la posibilidad de acceso al espacio de direcciones físicas de 36 bits.

PAE (Physical Address Extension) a las CPU Intel® Pentium® Pro y los modelos posteriores configuraciones de memoria de hasta 64 gigabytes. Para poder aprovechar esto en FreeBSD existe la opción del kernel `PAE`, disponible en todas las versiones modernas de FreeBSD. A causa de esta limitación de memoria en los Intel no hay nada que distinga de algún modo la memoria situada por debajo del límite de los 4 gigabytes. La memoria que esté por encima de los 4 gigabytes se coloca en el «pool» de memoria disponible.

Si quiere activar PAE en el kernel tiene que añadir la siguiente liínea al fichero de configuración del kernel:

```
options      PAE
```

Nota

En FreeBSD PAE solamente existe en procesadores Intel® IA-32. Hemos de advertirle de que PAE no ha sido probado todo lo necesario, así que debe considerarse de calidad beta, sobre todo si se le compara con otras características de FreeBSD.

PAE en FreeBSD tiene varias limitaciones:

- Un solo proceso no puede acceder a más de 4 gigabytes de espacio VM.
- No puede cargar módulos KLD en un kernel que tenga PAE activado debido a las diferencias existentes entre el «framework» de compilación del módulo y el del kernel mismo.
- Los controladores de dispositivo que utilizan el interfaz bus_dma(9) pueden provocar corrupción de datos en un kernel con PAE activado, una excelente razón para no utilizarlos. Esta es la razón de que FreeBSD incorpore un fichero de configuración de un kernel `PAE` del que se han extraído todos los módulos que se sabe que no funcionan en un kernel con PAE activado.
- Algunos «system tunables» determinan el uso de recursos de memoria basándose en la memoria física disponible. Estos «tunables» pueden asignar más memoria de la que realmente debieran debido a que el sistema PAE está íntimamente ligado a cantidades bastante importantes de memoria. Un ejemplo de esto es la sysctl `kern.maxvnodes` , que controla el número máximo de vnodes permitidos en el kernel. Le recomendamos que ajuste este y otros tunables dentro valores razonables.
- Es posible que tenga que aumentar el espacio virtual de direcciones del kernel (el KVA) o reducir la cantidad de recursos exclusivos del kernel que se utilicen exhaustivamente (ver más arriba) para evitar que KVA literalmente se ahogue. La opción del kernel `KVA_PAGES` permite incrementar el espacio KVA.

Si quiere saber más sobre la estabilidad del sistema consulte la página de manual de tuning(7). La página de manual de pae(4) contiene información actualizada sobre PAE y FreeBSD.

8.7. Qué hacer si algo va mal

Hay cuatro categorías de problemas que podemos encontrarnos en el proceso de compilación de un kernel personalizado:

Fallo de `config`:

Si config(8) falla cuando le pasa la descripción de su kernel seguramente ha cometido algún pequeño error. Por suerte config(8) les mostrará el número de la línea que le está dando problemas, así que no tendrá mayor problema para localizarla. Veamos un ejemplo:

```
config: line 17: syntax error
```

Si ve algo así asegúrese de que ha tecleado la palabra clave que hay en esa lína. Compare la de su fichero de configuración del kernel con la de `GENERIC`.

Fallo de `make`:

If the `make` command fails, it usually signals an error in your kernel description which is not severe enough for config(8) to catch. Again, look over your configuration, and if you still cannot resolve the problem, send mail to the lista de correo para preguntas generales sobre FreeBSD with your kernel configuration, and it should be diagnosed quickly.

El kernel no arranca:

Si su nuevo kernel no arranca o no reconoce sus dispositivos de hardware no desespere. FreeBSD dispone de excelentes mecanismos de recuperación ante kernels defectuosos. Elija el kernel con el que quiere arrancar en el gestor de arranque de FreeBSD. Puede acceder al gestor de arranque en el momento en el que aparece el menú de arranque. Elija la opción número seis, “Escape to a loader prompt” option, Escriba en el prompt `unload kernel` y después escriba `boot /boot/kernel.old/kernel` , o el nombre de cualquier otro kernel que tenga en el sistema y del que sepa que puede fiarse. Si va a reconfigurar un kernel es una buenísima idea guardar un kernel que sepa que funciona.

Tras arrancar con un kernel de fiar verifique el fichero de configuración e intente de nuevo una compilación. El fichero `/var/log/messages` es una fuente de información muy valiosa puesto que registra, entre otras cosas, todos los mensajes que deja el kernel cada vez que hay un arranque satisfactorio. dmesg(8) le mostrará los mensajes que el kernel ha generado durante el último arranque.

Nota

Si tiene problemas compilando un kernel recuerde que es de vital importancia que guarde una copia de un kernel `GENERIC` u otro del que pueda fiarse y que (esto es muy importante) tenga un nombre distinto de `kernel.old` para evitar que el sistema lo borre una vez que termine una nueva compilación. No puede confiar en su kernel `kernel.old` porque al instalar un nuevo kernel (que aún no sabe si será el que funcione tal y como espera de él) el kernel `kernel.old` se sobreescribe con el ker-

nel que instale. Otra cosa importante es que copie ese kernel de fiar a `/boot/kernel` , o ciertas herramientas como ps(1) no funcionarán. Basta con que haga lo siguiente:

```
# mv /boot/kernel /boot/kernel.malo
# mv /boot/kernel.bueno /boot/kernel
```

El kernel funciona, pero ps(1) no:

Si el kernel que tiene instalado es de una versión de FreeBSD y las utilidades del sistema son de otra, por ejemplo un kernel -CURRENT en una -RELEASE, hay muchas herramientas de monitorización del sistema como ps(1) y vmstat(8) no funcionarán. Ejecute un make buildworld y un make install world con la misma versión de código fuente con la que ha compilado su kernel. Esta es una de las razones por las que no es una idea demasiado buena utilizar versiones diferentes de kernel y de sistema operativo.

Capítulo 9. Imprimir

Escrito por Sean Kelly.
Reestructurado y actualizado por Jim Mock.

9.1. Sinopsis

FreeBSD puede utilizarse para imprimir en una gran variedad de impresoras, desde las antiguas impresoras de matriciales hasta las más modernas impresoras laser y todo lo que existe entre unas y otras, con lo que podrá imprimir con una calidad muy alta desde cualquier aplicación.

FreeBSD puede usarse también como servidor de impresión en red; FreeBSD puede recibir peticiones de impresión de una gran cantidad de sistemas, bien sean FreeBSD, Windows® y Mac OS®. FreeBSD se encarga de los trabajos se vayan imprimiendo uno tras otro y puede mantener estadísticas sobre qué usuarios y qué máquinas están imprimiendo más, generar páginas de «encabezado» para colocarlas antes de cada trabajo y distinguir así de quién es la impresión, etc.

Tras leer el presente capítulo sabrá usted:

- Cómo configurar la cola de impresión en FreeBSD.
- Cómo instalar filtros de impresión para gestionar diversos trabajos especiales como la conversión de documentos a formatos aptos para su impresora.
- Cómo habilitar encabezados en sus impresiones.
- Cómo imprimir en impresoras conectadas a otras máquinas.
- Cómo imprimir en impresoras conectadas directamente a la red.
- Cómo controlar restricciones de impresión, como por ejemplo limitaciones del tamaño de trabajos o evitar que ciertos usuarios puedan imprimir.
- Cómo mantener estadísticas de impresión y cuentas de usuario de impresión.
- Cómo solucionar los problemas de impresión más comunes.

Antes de leer este capítulo usted debería:

- Saber cómo compilar e instalar un kernel personalizado (Capítulo 8, *Configuración del kernel de FreeBSD*).

9.2. Introducción

Si quiere utilizar impresoras con FreeBSD debe configurarlas de forma que funcionen con el sistema de cola de impresión de Berkeley, también conocido como sistema de cola LPD. Es el sistema de control de impresión estándar en FreeBSD. Este capítulo trata sobre el sistema LPD, al que en adelante nos referiremos simplemente como LPD, y le introducirá en la configuración del mismo.

Si está familiarizado con el sistema de cola de impresión LPD, o bien con algún otro sistema de impresión, puede pasar a la sección Configuración del sistema de cola de impresión.

LPD se encarga de controlar todo lo relacionado con la impresión, lo que en consecuencia implica encargarse de lo siguiente:

- Controlar el acceso a las impresoras conectadas directamente al sistema y a impresoras conectadas a otras máquinas de la red.
- Permitir que ciertos usuarios puedan enviar ficheros para su impresión; estos envíos reciben el nombre de *trabajos* (“jobs”).
- Evita que usuarios diferentes accedan a la impresora al mismo tiempo manteniendo un orden estricto en la cola de impresión de cada impresora.
- Imprimir *encabezados* de forma que los usuarios puedan distinguir fácilmente los trabajos que han impreso.
- Se ocupa de los parámetros para impresoras conectadas a puertos serie.
- Enviar trabajos de impresión a través de una red a la cola LPD de una máquina remota.
- Ejecutar filtros especiales para formatear trabajos que requieren la gestión de lenguajes de impresión o impresoras de diferentes características.
- Mantener un registro del uso de la impresora.

Gracias al fichero de configuración (`/etc/printcap`) y mediante programas especiales de filtrado puede hacerse que el sistema LPD cumpla todas las tareas citadas o si lo prefiere solo unas cuantas; además puede hacerlo en una gran variedad de impresoras.

9.2.1. ?Por qué utilizar una cola de impresión?

Si es usted la única persona que usa el sistema se estará por qué preocuparse de utilizar un sistema de cola de impresión si es evidente que no necesita restricciones de acceso, encabezados ni cuentas de impresión. Puede utilizar el acceso directo a impresión, pero debería utilizar una cola de impresión por las siguientes razones:

- El sistema LPD ejecuta los trabajos en segundo plano, esto es, no tiene que esperar a que la impresora procese los datos para seguir trabajando.
- LPD puede procesar un trabajo de forma que pase por filtros, añadir cabeceras de fecha y hora o convertir un fichero de formato especial (por ejemplo TeX DVI) a un formato que la impresora entienda y que no tenga usted que ocuparse de ello manualmente.
- Muchos programas, tanto libres como comerciales, desde los que se puede imprimir están hechos pensando en que haya disponible un sistema de impresión. Una vez que configure un sistema de colas podrá utilizar software que ya tiene instalado y preparar el campo para el que instale en el futuro.

9.3. Configuración básica

Pendiente de traducción.

9.4. Configuración avanzada de impresoras

Pendiente de traducción

9.5. Cómo utilizar impresoras

Pendiente de traducción.

9.6. Alternativas a LPD

Pendiente de traducción

9.7. Solución de problemas

Pendiente de traducción

Capítulo 10. Compatibilidad binaria con Linux

Reestructurado y actualizado por Jim Mock.
Escrito por Brian N. Handy y Rich Murphey.

10.1. Sinopsis

FreeBSD proporciona compatibilidad de binarios con muchos otros sistemas operativos tipo UNIX®, incluyendo Linux. Puede estarse preguntando ?por qué necesita FreeBSD ejecutar binarios de Linux? La respuesta a esa pregunta es muy simple. Muchos desarrolladores y compañías desarrollan sólo para Linux, ya que últimamente es el blanco de todas las miradas dentro del mundo de las tecnologís de la información. Esto hace que la comunidad FreeBSD tenga que exigir a esas compañías y desarrolladores que produzcan versiones nativas de sus aplicaciones para FreeBSD. El problema es que la mayoría de esas compañías no suelen saber realmente cuánta gente utilizaría su producto si existieran esas versiones para FreeBSD, y la mayoría continúa desarrollando únicamente para Linux. Vista la situación ?que puede hacer un usuario de FreeBSD? Aquí es donde entra en juego la compatibilidad binaria con Linux.

Para expresarlo en pocas palabras, dicha compabitilidad permite a los usuarios de FreeBSD cerca del 90% de las aplicaciones de Linux sin tener que modificarlas en absoluto. Entre estas está StarOffice™, la versión Linux de Netscape®, Adobe® Acrobat®, RealPlayer, VMware, Oracle®, WordPerfect, Doom, Quake y muchas más. En determinados casos los binarios Linux rinden mejor en FreeBSD que en Linux.

Existen, por desgracia, ciertas características específicas de Linux que no funcionan en FreeBSD. Los binarios Linux no funcionarán en FreeBSD si recurren a llamadas específicas de i386™ como la activación del modo virtual 8086.

Tras leer este capítulo sabrá usted:

- cómo activar la compatibilidad binaria con Linux en su sistema.
- cómo instalar bibliotecas compartidas de Linux que pueda necesitar.
- cómo instalar aplicaciones de Linux en su sistema FreeBSD.
- cuáles son los detalles de la implementación de compatibilidad binaria con Linux en FreeBSD.

Antes de leer este capítulo es necesario que sepa:

- cómo instalar software de terceros (Capítulo 4, *Instalación de aplicaciones: «packages» y ports*).

10.2. Instalación

La compatibilidad binaria con Linux no viene activada por omisión. La forma más sencilla de habilitarla es cargar el KLD ("objeto cargable en el kernel") `linux`. Como usuario `root` proceda del siguiente modo:

```
# kldload linux
```

Si quiere que la compatibilidad con Linux esté siempre activada tendrá que añadir la siguiente línea en `/etc/rc.conf` :

```
linux_enable="YES"
```

Utilice kldstat(8) para verificar que el KLD esté cargado:

```
% kldstat
Id Refs Address    Size     Name
 1    2 0xc0100000 16bdb8   kernel
 7    1 0xc24db000 d000     linux.ko
```

Si por alguna razón no desea o no puede cargar el KLD, entonces puede enlazar estáticamente la compatibilidad de binarios Linux en el kernel agregando `options COMPAT_LINUX` a su fichero de configuración del kernel. Luego instale su nuevo kernel como se describe en Capítulo 8, *Configuración del kernel de FreeBSD*.

10.2.1. Instalación de bibliotecas de ejecución Linux

Puede hacerse de dos maneras, ya sea usando el port linux_base, o instalándolas de forma manual.

10.2.1.1. Instalación usando el port linux_base

Este es con mucho el método mas sencillo para instalar bibliotecas de ejecución. Es como instalar cualquier otro port de la Colección de Ports. Es tan sencillo como esto:

```
# cd /usr/ports/emulators/linux_base
# make install distclean
```

Hecho esto debería disponer de compatibilidad binaria con Linux. Algunos programas pueden "quejarse" por la presencia de versiones antiguas de algunas bibliotecas del sistema. Generalmente esto no suele ser un problema muy grave.

Nota

Pueden coexistir múltiples versiones del port emulators/linux_base disponibles correspondientes a distintas versiones de diversas distribuciones de Linux. Tendrá que instalar el port que más se ajuste a las necesidades de las aplicaciones de Linux que quiera instalar.

10.2.1.2. Instalación manual de bibliotecas

Si, por el motivo que fuese, no tiene instalada la colección de ports puede instalar las bibliotecas que necesite de forma manual. Necesitará las bibliotecas compartidas Linux de las que depende el programa y el enlazador en tiempo de ejecución ("runtime linker"). Necesitará también crear un directorio `/compat/linux` donde alojar las bibliotecas Linux en su sistema FreeBSD Cualquier biblioteca compartida a la que haya recurrido un programa de Linux ejecutado en FreeBSD buscará en primer lugar en dicho directorio. Por lo tanto, si se carga un programa Linux, por ejemplo `/lib/libc.so` , FreeBSD intentará en primer lugar abrir `/compat/linux/lib/libc.so` y, si no existe, lo intentará con `/lib/libc.so`. Las bibliotecas compartidas deben instalarse en `/compat/linux/lib` en lugar de las rutas que el `ld.so` de Linux proporcione.

En general, necesitará buscar las bibliotecas compartidas de las que los binarios Linux dependen sólamente las primeras veces que instale un programa Linux en su FreeBSD. Más adelante tendrá un conjunto suficiente de bibliotecas compartidas Linux en su sistema para poder ejecutar binarios Linux sin que tenga que hacer nada más.

10.2.1.3. Cómo instalar bibliotecas compartidas adicionales

?Que pasaría si instalara el port `linux_base` y su aplicación todavía tuviera problemas debido a bibliotecas compartidas que no encuentra en el sistema? ?Cómo saber qué bibliotecas compartidas necesitan los binarios Linux? Básicamente hay dos posibilidades (para poder ejecutar las siguientes instrucciones necesitará estar como `root`

Si tiene acceso a un sistema Linux busque en él qué bibliotecas necesita la aplicación, y cópielas a su sistema FreeBSD. Veamos unos ejemplos:

Asumiremos que utilizó FTP para conseguir los binarios Linux de Doom y los puso en un sistema Linux. Para ver qué bibliotecas compartidas necesitará ejecute `ldd linuxdoom`:

```
% ldd linuxdoom
libXt.so.3 (DLL Jump 3.1) => /usr/X11/lib/libXt.so.3.1.0
libX11.so.3 (DLL Jump 3.1) => /usr/X11/lib/libX11.so.3.1.0
libc.so.4 (DLL Jump 4.5pl26) => /lib/libc.so.4.6.29
```

Necesitaría todos los ficheros de la segunda columna, y tendrá que ponerlos en `/compat/linux` con los nombres de la primera columna como enlaces simbólicos apuntando hacia ellos. De este modo tendría en su sistema FreeBSD los siguientes ficheros:

```
/compat/linux/usr/X11/lib/libXt.so.3.1.0
/compat/linux/usr/X11/lib/libXt.so.3 -> libXt.so.3.1.0
/compat/linux/usr/X11/lib/libX11.so.3.1.0
/compat/linux/usr/X11/lib/libX11.so.3 -> libX11.so.3.1.0
/compat/linux/lib/libc.so.4.6.29
/compat/linux/lib/libc.so.4 -> libc.so.4.6.29
```

Nota

Recuerde que si ya tiene una biblioteca compartida Linux con un número de versión mayor que coincida con la primera columna de la salida de `ldd` no necesitará copiar el fichero que aparece en la última columna; el que tiene debería funcionar, aunque se aconseja copiar la biblioteca compartida de todas maneras si es una nueva versión. Puede eliminar la vieja siempre que haga que el enlace simbólico apunte a la nueva. Si tiene estas bibliotecas en su sistema:

```
/compat/linux/lib/libc.so.4.6.27
/compat/linux/lib/libc.so.4 -> libc.↺
so.4.6.27
```

y un binario requiere una versión más reciente (como indica la siguiente salida de `ldd`):

```
libc.so.4 (DLL Jump 4.5pl26) -> ↺
libc.so.4.6.29
```

si solo ve una o dos versiones desfasadas en los últimos dígitos no se preocupe de copiar `/lib/libc.so.4.6.29` , el programa debería funcionar bien con una versión ligeramente antigua. De todas formas, si así lo prefiere, puede actualizar `libc.so` ; el resultado sería este:

```
/compat/linux/lib/libc.so.4.6.29
/compat/linux/lib/libc.so.4 -> libc.↺
so.4.6.29
```

Nota

El mecanismo de enlazado simbólico *sólamente* es necesario con binarios Linux. El enlazador en tiempo de ejecución de FreeBSD se encarga de buscar él mismo las versiones correctas, así que no tendrá que preocuparse usted de hacerlo.

10.2.2. Instalar binarios ELF Linux

Los binarios ELF algunas veces requieren un paso extra de "marcado". Si trata de ejecutar un binario ELF no marcado recibirá un mensaje de error como el siguiente:

```
% ./mi-binario-elf
ELF binary type not known
Abort
```

Para ayudar al kernel de FreeBSD a distinguir entre un binario ELF de FreeBSD y uno de Linux utilice brandelf(1).

```
% brandelf -t Linux mi-binario-elf-de-linux
```

Las herramientas GNU se encargan de ubicar automáticamente la marca apropiada en los binarios ELF, por lo tanto este paso será innecesario en un futuro próximo.

10.2.3. Configuración de la resolución de nombres de equipos

Si el DNS no funciona u obtiene este mensaje:

```
resolv+: "bind" is an invalid keyword resolv+:
"hosts" is an invalid keyword
```

Necesitará un fichero `/compat/linux/etc/host.conf` con el siguiente contenido:

```
order hosts, bind
multi on
```

Significa que `/etc/hosts` seráanalizado en primer lugar y después se usará DNS. Si `/compat/linux/etc/host.conf` no está instalado, las aplicaciones Linux usan el `/etc/host.conf` de FreeBSD y chocan con la sintaxis (incompatible) de FreeBSD. Borre `bind` de su `/etc/resolv.conf` si no tiene configurado un servidor de nombres.

10.3. Instalación de Mathematica®

Actualizado para Mathematica 5.X por Hollas Boris.

Este documento describe el proceso de instalación de la versión para Linux de Mathematica® 5.X en un sistema FreeBSD.

Puede pedir a Wolfram, el fabricante, La versión para para Linux de Mathematica® o la versión de Mathematica® para estudiantes en su sitio web, `http://www.wolfram.com/` .

10.3.1. El instalador de Mathematica®

Lo primero que tiene que hacer es decirle a FreeBSD que los binarios de Mathematica® para Linux utilizan la ABI Linux. La forma más sencilla de hacerlo es marcar por omisión todos los binarios sin marcas como Linux ELF.

```
# sysctl kern.fallback_elf_brand=3
```

Hecho esto FreeBSD asumirá que cualquier binario sin marca que encuentre utiliza la ABI Linux; de este modo podrá ejecutar el binario directamente desde el CDROM.

Copie el fichero `MathInstaller` en su disco duro

```
# mount /cdrom
# cp /cdrom/Unix/Installers/Linux/MathInstaller /directoriolocal/
```

Edite este fichero y sustituya la primera línea, `/bin/sh` , por `/compat/linux/bin/sh` para asegurarnos de que lo que ejecute el instalador sea la verión de sh(1) de Linux. El siguiente paso es sustituir todos los `Linux)` por `FreeBSD)` con un editor de texto on con el script que encontrará en la siguiente sección. Esto se hace para ayudar al instalador de Mathematica®, el cual en un cierto momento invoca a `uname -s` para determinar el sistema operativo, a tratar a FreeBSD como si fuera un sistema operativo muy similar a Linux. Hecho todo esto, cuando ejecute `MathInstaller` podrá instalar Mathematica®.

10.3.2. Modificación de los ejecutables de Mathematica®

Debe modificar los scripts de shell que Mathematica® creó durante la instalación antes de usarlos. Si eligió ubicar en `/usr/local/bin` los ejecutables de Mathematica® verá que en ese directorio hay enlaces simbólicos a ficheros como `math`, `mathematica`, `Mathematica` y `MathKernel`. En cada uno de esos ficheros debe sustituir `Linux)` por `FreeBSD)` con un editor de texto o bien con el siguiente script de shell:

```
#!/bin/sh
cd /usr/local/bin
for i in math mathematica Mathematica MathKernel
  do sed 's/Linux)/FreeBSD)/g' $i > $i.tmp
  sed 's/\/bin\/sh/\/compat\/linux\/bin\/sh/g' $i.tmp > $i
  rm $i.tmp
```

```
  chmod a+x $i
done
```

10.3.3. Cómo obtener una contraseña de Mathematica®

Cuando arranque Mathematica® por primera vez se le pedirá una contraseña. Si Wolfram no le ha enviado ya necesita un "machine ID", para lo cual debe ir al directorio de instalación y ejecutar `mathinfo`. Este "machine IDE" se obtiene de la dirección MAC de la primera tarjeta Ethernet de la máquina y tiene como objetivo que no pueda ejecutar Mathematica® en más de una máquina.

Durante el proceso de registro en Wolfram (ya sea por correo electrónico, teléfono o fax) les dará el "machine ID" y Wolfram le enviará una contraseña relacionada con él, consistente en grupos de números.

10.3.4. Ejecución del «frontend» de Mathematica® través de una red

Mathematica® usa unos cuantos tipos especiales para mostrar caracteres que no están en ningún conjunto estándar de tipos: integrales, sumas, letras griegas, etc. El protocolo X exige que los tipos estén instalados *en local*, es decir, tiene que copiar los tipos del CDROM o la máquina desde la que ha instalado Mathematica® a su máquina. Los tipos están en el directorio del CDDROM `/cdrom/Unix/Files/SystemFiles/Fonts` y se supone que deben estar en su disco duro en el directorio `/usr/local/mathematica/SystemFiles/Fonts`. Los tipos están realmente en los subdirectorios `Type1` y `X`. Hay varias formas de utilizarlos.

La primera es copiarlos en uno de los directorios de tipos que hay en `/usr/X11R6/lib/X11/fonts`, antes de lo cual tendrá que añadir a `fonts.dir` los nombres de los tipos; tendrá también que cambiar el número de tipos en la primera línea. Por otra parte, todo esto puede hacerse ejecutando mkfontdir(1) en el directorio donde haya copiado los tipos.

La segunda forma de utilizar estos tipos es copiarlos bajo `/usr/X11R6/lib/X11/fonts` :

```
# cd /usr/X11R6/lib/X11/fonts
# mkdir X
# mkdir MathType1
# cd /cdrom/Unix/Files/SystemFiles/Fonts
# cp X/* /usr/X11R6/lib/X11/fonts/X
# cp Type1/* /usr/X11R6/lib/X11/fonts/MathType1
# cd /usr/X11R6/lib/X11/fonts/X
# mkfontdir
# cd ../MathType1
# mkfontdir
```

Añada los nuevos directorios de tipos a su ruta de tipos:

```
# xset fp+ /usr/X11R6/lib/X11/fonts/X
# xset fp+ /usr/X11R6/lib/X11/fonts/MathType1
```

```
# xset fp rehash
```

Si usa el servidor Xorg puede cargar los tipos automáticamente añadiéndolos al fichero `xorg.conf` .

Nota

En servidores XFree86™ el fichero de configuración es `XF86Config` .

Si *no* tiene ya en su sistema un directorio `/usr/X11R6/lib/X11/fonts/Type1` puede cambiarle el nombre al directorio `MathType1` del ejemplo anterior por `Type1`.

10.4. Instalación de Maple™

Escrito por Aaron Kaplan.
Gracias a Robert Getschmann.

Maple™ es un programa comercial de matemáticas similar a Mathematica®. Puede adquirir este software en `http://www.maplesoft.com/` ; tras registrarlo recibirá un fichero de licencia. Si quiere instalar este software en FreeBSD siga los siguienes pasos:

1. Ejecute el "script" de shell `INSTALL` desde el lugar de instalación del producto. Elija la opción "RedHat" cuando le pregunte el programa de instalación. `/usr/local/maple` es un buen sitio para instalar el software.

2. Si no lo ha hecho ya, solicite una licencia para Maple™ a Maple Waterloo Software (`http://register.maplesoft.com/`) y cópiela a `/usr/local/maple/license/license.dat` .

3. Instale el gestor de licencias FLEXlm ejecutando el "script" de shell de instalación `INSTALL_LIC` que viene con Maple™. Introduzca el nombre de su máquina (el servidor de licencias lo necesita).

4. Parchée el fichero `/usr/local/maple/bin/maple.system.type` con lo siguiente:

```
   ----- snip ------------------
*** maple.system.type.orig      Sun Jul  8 16:35:33 2001
--- maple.system.type   Sun Jul  8 16:35:51 2001
***************
*** 72,77 ****
--- 72,78 ----
          # the IBM RS/6000 AIX case
          MAPLE_BIN="bin.IBM_RISC_UNIX"
```

```
          -;;
+     "FreeBSD"|\
      "Linux")
          # the Linux/x86 case
        # We have two Linux implementations, one for Red Hat and
   ----- snip end of patch -----
```

Tenga muy presente que después de `"FreeBSD"|\` no debe haber ningún espacio en blanco.

Este parche le dice a Maple™ que interprete "FreeBSD" como un tipo de sistema Linux. El "script" de shell `bin/maple` llama al "script" de shell `bin/maple.system.type` , que a su vez recurre a `uname -a` para dictaminar el nombre del sistema operativo. Dependiendo de cuál sea sabrá qué binarios utilizar.

5. Inicio del servidor de licencias.

   El siguiente "script", sito en `/usr/local/etc/rc.d/lmgrd.sh` , le permitirá arrancar `lmgrd`:

```
   ----- snip ------------

#! /bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/
bin:/usr/X11R6/bin
PATH=${PATH}:/usr/local/maple/bin:/usr/local/maple/FLEXlm/UNIX/
LINUX
export PATH

LICENSE_FILE=/usr/local/maple/license/license.dat
LOG=/var/log/lmgrd.log

case "$1" in
start)
 lmgrd -c ${LICENSE_FILE} 2<< ${LOG} 1<&2
 echo -n " lmgrd"
 ;;
stop)
 lmgrd -c ${LICENSE_FILE} -x lmdown 2<< ${LOG} 1<&2
 ;;
*)
 echo "Usage: `basename $0` {start|stop}" 1<&2
 exit 64
 ;;
esac

exit 0
   ----- snip ------------
```

6. Prueba de arranque de Maple™:

```
% cd /usr/local/maple/bin
```

```
% ./xmaple
```

Todo debería funcionar perfectamente. Si es así aún le queda un último paso: escribir a Maplesoft y decirles que sería genial una versión nativa para FreeBSD.

10.4.1. Problemas frecuentes

- El gestor de licencias FLEXlm puede ser un tanto difícil de usar. En caso de necesitarla tiene más información en `http://www.globetrotter.com/` .
- `lmgrd` tiene una reconocida fama de ser muy meticuloso en todo lo relacionado con el fichero de licencia; suele generar volcados de memoria si se encuentra con algún problema. Un fichero de licencia correcto tiene que parecerse mucho a este:

```
# ====================================================
# License File for UNIX Installations ("Pointer File")
# ====================================================
SERVER chillig ANY
#USE_SERVER
VENDOR maplelmg

FEATURE Maple maplelmg 2000.0831 permanent 1 XXXXXXXXXXX \
         PLATFORMS=i86_r ISSUER="Waterloo Maple Inc." \
         ISSUED=11-may-2000 NOTICE=" Technische Universitat Wien" \
         SN=XXXXXXXXX
```

Nota

El número de serie y la clave han sido sobreescritos con X. `chillig` es el nombre de un equipo.

Puede editar el fichero de licencia siempre que no toque la línea "FEATURE" (que está protegida por la clave de la licencia).

10.5. Instalación de MATLAB®

Escrito por Dan Pelleg.

Este documento describe el proceso de instalación de la versión para Linux de MATLAB® version 6.5 en FreeBSD. En general funciona bastante bien, excepción hecha de Java Virtual Machine™ (consulte la Sección 10.5.3, "Enlace del entorno de ejecución Java™").

La versión Linux de MATLAB® puede pedirse directamente en el sitio de The MathWorks, `http://www.mathworks.com` . Tiene que recibir también el fichero de licencia o instruc-

ciones de cómo crearlo. Al hacer su pedido aproveche para decirles que sería muy buena idea que ofrecieran una versión nativa de su software para FreeBSD.

10.5.1. Instalación de MATLAB®

Para instalar MATLAB® haga lo siguiente:

1. Inserte el CD de instalación y móntelo. Conviértase en `root` e inicie la instalación:

```
# /compat/linux/bin/sh /cdrom/install
```

Sugerencia

El instalador es gráfico. Si obtiene errores acerca de no ser capaz de abrir un display teclée `setenv HOME ~USUARIO`, donde *`USUARIO`* es el nombre del usuario con el que hizo su(1).

2. Teclée **`/compat/linux/usr/local/matlab`** donde el instalador le pida el directorio raíz de MATLAB®.

Sugerencia

Esto último le facilitará la entrada de datos durante el resto de la instalación. Introduzca lo siguiente en el "prompt" de su shell: `set MATLAB=/compat/linux/usr/local/matlab`

3. Edite el fichero de licencia tal y como consta en las instrucciones de la licencia de MATLAB®.

Sugerencia

Puede tenerlo ya editado y copiado a `$MATLAB/license.dat` desde antes de que el instalador se lo pida.

4. Complete el proceso de instalación.

La instalación MATLAB® ha finalizado. Los siguientes pasos aplicarán el "pegamento" necesario para conectarlo a su sistema FreeBSD.

10.5.2. Inicio del administrador de licencias

1. Crée los enlaces simbólicos que necesitan los "scripts" del administrador de licencias:

```
# ln -s $MATLAB/etc/lmboot /usr/local/etc/lmboot_TMW
# ln -s $MATLAB/etc/lmdown /usr/local/etc/lmdown_TMW
```

2. Crée un fichero de inicio en `/usr/local/etc/rc.d/flexlm.sh` . El siguiente ejemplo es una versión modificada de `$MATLAB/etc/rc.lm.glnx86` que viene con la distribución de MATLAB®. Los cambios que se han hecho en él obedecen a la ubicación de los ficheros y el arranque del administrador de licencias bajo emulación de Linux.

```
#!/bin/sh
case "$1" in
  start)
        if [ -f /usr/local/etc/lmboot_TMW -]; then
              /compat/linux/bin/sh /usr/local/etc/lmboot_TMW -
u nombre-de-usuario  && echo 'MATLAB_lmgrd'
        fi
        -;;
  stop)
 if [ -f /usr/local/etc/lmdown_TMW -]; then
            /compat/linux/bin/sh /usr/local/etc/lmdown_TMW  > /
dev/null 2>&1
 fi
        -;;
  *)
 echo "Usage: $0 {start|stop}"
 exit 1
 ;;
esac

exit 0
```

Importante

El fichero debe ser ejecutable:

```
# chmod +x /usr/local/etc/rc.d/flexlm.sh
```

Tendrá que reemplazar la entrada *`nombre-de-usuario`* de nuestro ejemplo por un nombre de usuario válido en su sistema (que no sea `root`).

3. Arranque el administrador de licencias:

```
# /usr/local/etc/rc.d/flexlm.sh start
```

10.5.3. Enlace del entorno de ejecución Java™

Cambie el enlace del entorno de ejecución Java™ (JRE) a uno que funcione en FreeBSD:

```
# cd $MATLAB/sys/java/jre/glnx86/
# unlink jre; ln -s ./jre1.1.8 ./jre
```

10.5.4. Creación de un "script" de arranque para MATLAB®

1. Coloque el siguiente "script" de arranque en `/usr/local/bin/matlab` :

```
#!/bin/sh
/compat/linux/bin/sh /compat/linux/usr/local/matlab/bin/matlab
 "$@"
```

2. Escriba `chmod +x /usr/local/bin/matlab` .

Sugerencia

Dependiendo de su versión de emulators/linux_base tal vez obtenga errores al ejecutar este "script". Para evitarlo edite `/compat/linux/usr/local/matlab/bin/matlab` y cambie la línea en la que aparece:

```
if [ `expr "$lscmd" : '.*->.*'` -ne 0 -]; then
```

(en la versión 13.0.1 es en la línea 410) por esta otra línea:

```
if test -L $newbase; then
```

10.5.5. Creación de un "script" para detener MATLAB®

Este "script" solucionará las dificultades que pueda tener para detener MATLAB® correctamente.

1. Crée un fichero llamado `$MATLAB/toolbox/local/finish.m` y ponga en él una sola línea con este texto:

```
! $MATLAB/bin/finish.sh
```

Nota

`$MATLAB` debe escribirse tal cual.

Sugerencia

En el mismo directorio encontrará los ficheros `finishsav.m` y `finishdlg.m` , que le permiten guardar su trabajo antes de salir de la aplicación. Si quiere usar alguno de ellos, inserte la línea de arriba inmediatamente después de `save`.

2. Crée un fichero `$MATLAB/bin/finish.sh` con el siguiente contenido:

```
#!/usr/compat/linux/bin/sh
(sleep 5; killall -1 matlab_helper) &
exit 0
```

3. El fichero tiene que ser ejecutable:

```
# chmod +x $MATLAB/bin/finish.sh
```

10.5.6. Uso de MATLAB®

Desde este momento ya puede usted teclear `matlab` y empezar a usarlo.

10.6. Instalación de Oracle®

Escrito por Marcel Moolenaar.

10.6.1. Prefacio

Este texto describe el proceso de instalación de Oracle® 8.0.5 y Oracle® 8.0.5.1 Enterprise Edition para Linux en una máquina FreeBSD.

10.6.2. Instalación del entorno Linux

Debe tener instalados los ports emulators/linux_base y devel/linux_devtools. Si tiene dificultades con estos ports es posible que tenga que usar los paquetes o quizás versiones más antiguas de dichas aplicaciones que encontrará en la Colección de Ports.

Si quiere usar el agente inteligente también tendrá que instalar el paquete Tcl de Red Hat, `tcl-8.0.3-20.i386.rpm` . La orden genérica para instalar paquetes con el port oficial de RPM (archivers/rpm) es:

```
# rpm -i --ignoreos --root /compat/linux --dbpath /var/lib/rpm paquete
```

La instalación de dicho *`paquete`* no debe generar ningún error.

10.6.3. Creación del entorno Oracle®

Antes de instalar Oracle® tendrá que configurar un entorno apropiado. Este documento solamente explica lo que hay que hacer *especialmente* para utilizar la versión de Linux para Oracle® FreeBSD, no lo que figura en la guía de instalación de Oracle®.

10.6.3.1. Personalización del kernel

Tal y como consta en la guía de instalación de Oracle®, debe configurar la cantidad máxima de memoria compartida. No utilice `SHMMAX` en FreeBSD. `SHMMAX` se calcula a partir de `SHMMAXPGS` y `PGSIZE`, así que defina `SHMMAXPGS`. Todas las demás opciones pueden usarse tal y como se describen en la guía. Por ejemplo:

```
options SHMMAXPGS=10000
options SHMMNI=100
options SHMSEG=10
options SEMMNS=200
options SEMMNI=70
options SEMMSL=61
```

Configure estas opciones para que se ajusten al uso que pretenda darle a Oracle®.

Asegúrese también de que las siguientes opciones están en el fichero de configuración de su kernel:

```
options SYSVSHM #SysV shared memory
options SYSVSEM #SysV semaphores
options SYSVMSG #SysV interprocess communication
```

10.6.3.2. Cuenta Oracle®

Crée una cuenta `oracle` según el procedimiento habitual de creación de usuarios. La cuenta `oracle`, empero, tiene algo especial puesto que debe teer una shell *de* Linux. Añada `/compat/linux/bin/bash` a `/etc/shells` y asigne a la cuenta `oracle` `/compat/linux/bin/bash` como shell por omisión.

10.6.3.3. Entorno

Además de las variables normales para Oracle®, como `ORACLE_HOME` y `ORACLE_SID` , debe configurar las siguientes variables de entorno:

Variable	Valor
LD_LIBRARY_PATH	$ORACLE_HOME/lib
CLASSPATH	$ORACLE_HOME/jdbc/lib/classes111.zip
PATH	/compat/linux/bin /compat/linux/sbin /compat/linux/usr/bin /compat/linux/usr/sbin /bin /sbin /usr/bin /usr/sbin /usr/local/bin $ORACLE_HOME/bin

Le aconsejamos configurar todas las variables de entorno en `.profile`. Veamos un ejemplo completo:

```
ORACLE_BASE=/oracle; export ORACLE_BASE
ORACLE_HOME=/oracle; export ORACLE_HOME
LD_LIBRARY_PATH=$ORACLE_HOME/lib
export LD_LIBRARY_PATH
ORACLE_SID=ORCL; export ORACLE_SID
ORACLE_TERM=386x; export ORACLE_TERM
CLASSPATH=$ORACLE_HOME/jdbc/lib/classes111.zip
export CLASSPATH
PATH=/compat/linux/bin:/compat/linux/sbin:/compat/linux/usr/bin
PATH=$PATH:/compat/linux/usr/sbin:/bin:/sbin:/usr/bin:/usr/sbin
PATH=$PATH:/usr/local/bin:$ORACLE_HOME/bin
export PATH
```

10.6.4. Instalación de Oracle®

Debido a una pequeña inconsistencia en el emulador Linux tendrá que crear un directorio llamado `.oracle` en `/var/tmp` antes de iniciar el instalador. Haga que sea propiedad del usuario `oracle`. Hecho esto deberí poder instalar Oracle® sin ningún problema. Si no es así *revise su distribución* Oracle® y su configuración. Una vez finalizada la instalación de Oracle® aplique los parches que se detallan en las dos siguientes subsecciones.

Un problema que se da con una cierta frecuencia es que el adaptador del protocolo TCP no está correctamente instalado. Como consecuencia no puede iniciarse ninguna escucha TCP, a las que también se les llama directamente «listeners». Esto le ayudará a resolver el problema.:

```
# cd $ORACLE_HOME/network/lib
# make -f ins_network.mk ntcontab.o
# cd $ORACLE_HOME/lib
# ar r libnetwork.a ntcontab.o
# cd $ORACLE_HOME/network/lib
# make -f ins_network.mk install
```

No se olvide de ejecutar `root.sh` de nuevo.

10.6.4.1. Cómo parchear root.sh

Durante la instalación de Oracle® algunas acciones que requieren ser ejecutadas como `root` deben almacenarse en un "script" de shell llamado `root.sh`. Dicho "script" está en el directorio `orainst`. Aplique el siguiente parche a `root.sh` para que utilice la ruta correcta de `chown` o ejecute el "script" bajo una shell nativa de Linux.

```
*** orainst/root.sh.orig Tue Oct 6 21:57:33 1998
--- orainst/root.sh Mon Dec 28 15:58:53 1998
***************
*** 31,37 ****
# This is the default value for CHOWN
# It will redefined later in this script for those ports
# which have it conditionally defined in ss_install.h
! CHOWN=/bin/chown
#
# Define variables to be used in this script
--- 31,37 ----
# This is the default value for CHOWN
# It will redefined later in this script for those ports
# which have it conditionally defined in ss_install.h
! CHOWN=/usr/sbin/chown
#
# Define variables to be used in this script
```

Si no está instalando Oracle® desde un CD puede parchear las fuentes de `root.sh`. Se llama `rthd.sh` y está en el directorio `orainst` del árbol de fuentes.

10.6.4.2. Cómo parchear genclntsh

El "script" `genclntsh` se usa para crear una biblioteca de cliente compartida y para construir los demos. Al aplicar el siguiente parche comentará la definición de `PATH`:

```
*** bin/genclntsh.orig Wed Sep 30 07:37:19 1998
--- bin/genclntsh Tue Dec 22 15:36:49 1998
***************
*** 32,38 ****
#
# Explicit path to ensure that we're using the correct commands
#PATH=/usr/bin:/usr/ccs/bin export PATH
! PATH=/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin export PATH
#
# each product MUST provide a $PRODUCT/admin/shrept.lst
--- 32,38 ----
#
# Explicit path to ensure that we're using the correct commands
#PATH=/usr/bin:/usr/ccs/bin export PATH
! #PATH=/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin export PATH
#
# each product MUST provide a $PRODUCT/admin/shrept.lst
```

10.6.5. Ejecución de Oracle®

Una vez seguidas estas instrucciones podrá ejecutar Oracle® como si la hubiera instalado en Linux.

10.7. Instalación de SAP® R/3®

Escrito por Holger Kipp.
Versión original convertida a SGML por Valentino Vaschetto.

Las instalaciones de sistemas SAP® en FreeBSD no reciben soporte técnico de SAP®. SAP® solamente lo ofrece si se usan plataformas certificadas.

10.7.1. Introducción

Este texto expone una forma de instalar un SAP® R/3® System con una Oracle® Database para Linux en una máquina FreeBSD, incluyendo la instalación de FreeBSD y Oracle®. Se muestran dos configuraciones diferentes:

- SAP® R/3® 4.6B (IDES) con Oracle® 8.0.5 en FreeBSD 4.3-STABLE
- SAP® R/3® 4.6C con Oracle® 8.1.7 en FreeBSD 4.5-STABLE

Aunque este documento trate de describir todos los pasos importantes con detalle no ha sido escrito como sustituto de las guías de instalación de Oracle® y SAP® R/3®.

Por favor, consulte la documentación de SAP® R/3® que se incluye en la edición para Linux de SAP® y las preguntas específicas sobre Oracle®, así como los recursos que estén a su disposición sobre Oracle® y SAP® OSS.

10.7.2. Software

Durante la instalación de SAP® se han utilizado los siguientes CD-ROM:

10.7.2.1. SAP® R/3® 4.6B, Oracle® 8.0.5

Nombre	Número	Descripción
KERNEL	51009113	SAP Kernel Oracle / Installation / AIX, Linux, Solaris
RDBMS	51007558	Oracle / RDBMS 8.0.5.X / Linux
EXPORT1	51010208	IDES / DB-Export / Disco 1 de 6

Nombre	Número	Descripción
EXPORT2	51010209	IDES / DB-Export / Disco 2 de 6
EXPORT3	51010210	IDES / DB-Export / Disco 3 de 6
EXPORT4	51010211	IDES / DB-Export / Disco 4 de 6
EXPORT5	51010212	IDES / DB-Export / Disco 5 de 6
EXPORT6	51010213	IDES / DB-Export / Disco 6 de 6

También utilizamos el CD de Oracle® 8 Server (versión pre-producción 8.0.5 para Linux, versión de kernel 2.0.33), que no es realmente necesario y FreeBSD 4.3-STABLE (a unos cuantos días de la liberación de 4.3-RELEASE).

10.7.2.2. SAP® R/3® 4.6C SR2, Oracle® 8.1.7

Nombre	Número	Descripción
KERNEL	51014004	SAP Kernel Oracle / SAP Kernel Version 4.6D / DEC, Linux
RDBMS	51012930	Oracle 8.1.7/ RDBMS / Linux
EXPORT1	51013953	Release 4.6C SR2 / Export / Disco 1 de 4
EXPORT1	51013953	Release 4.6C SR2 / Export / Disco 2 de 4
EXPORT1	51013953	Release 4.6C SR2 / Export / Disco 3 de 4
EXPORT1	51013953	Release 4.6C SR2 / Export / Disco 4 de 4
LANG1	51013954	Release 4.6C SR2 / Language / DE, EN, FR / Disco 1 de 3

Según los idiomas que quiera usar es posible que necesite otros CD de idiomas. Sólo hemos utilizado DE y EN, así que nos bastó con el primer CD. Para su información, los números de los cuatro CD EXPORT son idénticos. Los tres CD de idiomas también tienen el mismo número, aunque esto es distinto en los CD de la versión 4.6B IDES. Al escribir este texto (20.03.2002) ejecutamos la instalación en FreeBSD 4.5-STABLE (20.03.2002).

10.7.3. Notas SAP®

Las siguientes notas han resultado ser muy útiles durante la instalación, así que le recomendamos encarecidamente que las lea antes de instalar SAP® R/3®:

10.7.3.1. SAP® R/3® 4.6B, Oracle® 8.0.5

Número	Título
0171356	SAP Software on Linux: Essential Comments
0201147	INST: 4.6C R/3 Inst. on UNIX - Oracle
0373203	Update / Migration Oracle 8.0.5 --> 8.0.6/8.1.6 LINUX
0072984	Release of Digital UNIX 4.0B for Oracle
0130581	R3SETUP step DIPGNTAB terminates
0144978	Your system has not been installed correctly
0162266	Questions and tips for R3SETUP on Windows NT / W2K

10.7.3.2. SAP® R/3® 4.6C, Oracle® 8.1.7

Número	Título
0015023	Initializing table TCPDB (RSXP0004) (EBCDIC)
0045619	R/3 with several languages or typefaces
0171356	SAP Software on Linux: Essential Comments
0195603	RedHat 6.1 Enterprise version: Known problems
0212876	The new archiving tool SAPCAR
0300900	Linux: Released DELL Hardware
0377187	RedHat 6.2: important remarks
0387074	INST: R/3 4.6C SR2 Installation on UNIX
0387077	INST: R/3 4.6C SR2 Inst. on UNIX - Oracle
0387078	SAP Software on UNIX: OS Dependencies 4.6C SR2

10.7.4. Requisitos de hardware

El siguiente equipo es suficiente para la instalación de un sistema SAP® R/3®. Si pretende darle uso productivo necesitará hacer un estudio detallado de sus necesidades:

Componente	4.6B	4.6C
Procesador	Pentium® III 800MHz x 2	Pentium® III 800MHz x 2
Memoria	1GB ECC	2GB ECC
Espacio en disco	50-60GB (IDES)	50-60GB (IDES)

Para su uso en producción le recomendamos procesadores Xeon™ con una caché grande, discos de alta velocidad (SCSI, controlador de RAID por hardware), USV y ECC-RAM. Un espacio en disco tan grande se debe al sistema IDES preconfigurado, que crea ficheros de bases de datos de 27 GB durante la instalación. Este espacio también es suficiente para sistemas de producción iniciales y datos de aplicación.

10.7.4.1. SAP® R/3® 4.6B, Oracle® 8.0.5

Este es el hardware que utilizamos al escribir este texto: placa base dual con 2 procesadores Pentium® III a 800 MHz, adaptador SCSI Adaptec® 29160 Ultra160 (para acceder a una unidad de cinta 40/80 GB DLT y CDROM), Mylex® AcceleRAID™ (2 canales, firmware 6.00-1-00 con 32 MB RAM). La controladora Mylex® RAID tiene conectados dos discos duros de 17 GB (replicados) y cuatro discos duros de 36 GB (RAID nivel 5).

10.7.4.2. SAP® R/3® 4.6C, Oracle® 8.1.7

Para esta instalación se usó un Dell™ PowerEdge™ 2500: placa base dual con 2 procesadores Pentium® III a 1000 MHz (256 kB de Caché), 2 GB PC133 ECC SDRAM, controladora RAID PERC/3 DC PCI con 128 MB y una unidad EIDE DVD-ROM. La controladora RAID tiene conectados dos discos duros 18 GB (replicados) y cuatro discos duros de 36 GB (RAID nivel 5).

10.7.5. Instalación de FreeBSD

Lo primero que tiene que hacer es instalar FreeBSD. Hay muchas formas de hacerlo. Nosotros instalamos FreeBSD 4.3 desde un FTP y FreeBSD 4.5 desde el CD de la distribución. Si necesita más información sobre los medios de instalación de FreeBSD consulte la Sección 2.13, “Cómo preparar su propio medio de instalación”.

10.7.5.1. Esquema de disco

Quisimos hacer el proceso lo más simple posible, así que usamos el esquema de disco de SAP® R/3® 46B y SAP® R/3® 46C SR2. Solo cambiamos los nombres de dispositivo debido a que las instalaciones tuvieron lugar en hardware diferente (`/dev/da` y `/dev/amr` respectivamente. Si utiliza una AMI MegaRAID® verá en pantalla `/dev/amr0s1a` en lugar de `/dev/da0s1a`):

Sistema de ficheros	Tamaño (bloques de 1k)	Tamaño (GB)	Montado en
`/dev/da0s1a`	1.016.303	1	`/`

Sistema de ficheros	Tamaño (bloques de 1k)	Tamaño (GB)	Montado en
/dev/da0s1b		6	swap
/dev/da0s1e	2.032.623	2	/var
/dev/da0s1f	8.205.339	8	/usr
/dev/da1s1e	45.734.361	45	/compat/linux/oracle
/dev/da1s1f	2.032.623	2	/compat/linux/sapmnt
/dev/da1s1g	2.032.623	2	/compat/linux/usr/sap

Configure e inicialice antes que nada las dos unidades lógicas con el software Mylex® o PERC/3 RAID. El software puede iniciarse durante la fase de arranque de BIOS.

Por favor, tenga en cuenta que el esquema de disco que utilizamos difiere ligeramente de las recomendaciones de SAP®, ya que SAP® sugiere montar los subdirectorios Oracle® (y algunos otros) por separado. Decidimos crearlos como subdirectorios reales para simplificar.

10.7.5.2. `make world` y un nuevo kernel

Descargue las fuentes -STABLE más recientes. Ejecute `make world` y compile su kernel personalizado. Recuerde incluir en él tanto los parámetros del kernel requeridos por SAP® R/3® como los que necesita Oracle®.

10.7.6. Instalación del entorno Linux

10.7.6.1. Instalación del sistema base Linux

Primero instale el port linux_base (como `root`):

```
# cd /usr/ports/emulators/linux_base
# make install distclean
```

10.7.6.2. Instalación del entorno de desarrollo Linux

El entorno de desarrollo Linux es imprescindible si quiere instalar Oracle® en FreeBSD según se explica en la Sección 10.6, “Instalación de Oracle®”:

```
# cd /usr/ports/devel/linux_devtools
# make install distclean
```

El entorno de desarrollo Linux solo ha de instalarse si sigue el proceso para instalar SAP® R/3® 46B IDES. No es necesario si Oracle® DB no está reenlazado («relinked») con el sistema FreeBSD. Este sería su caso si está usa el fichero comprimido tar de Oracle® de un sistema Linux.

10.7.6.3. Instalación de los RPM necesarios

Necesitará soporte PAM para iniciar el programa R3SETUP . Durante la primera instalación de SAP® en FreeBSD 4.3-STABLE intentamos instalar PAM con todas las dependencias y finalmente forzamos la instalación del paquete PAM, y funcionó. En SAP® R/3® 4.6C SR2 forzamos la instalación del RPM PAM, que también funcionó, así que parece que las dependencias no lo son tanto:

```
# rpm -i --ignoreos --nodeps --root /compat/linux --dbpath /var/lib/
rpm \
pam-0.68-7.i386.rpm
```

Para que Oracle® 8.0.5 pueda lanzar el agente inteligente también tendremos que instalar el paquete Tcl de RedHat `tcl-8.0.5-30.i386.rpm` (si no, cuando lo reenlace durante la instalación de Oracle® no funcionará). Existen otros aspectos relacionados con el reenlazado de Oracle® a tener en cuenta durante la instalación, pero atañen a la versión para Linux de Oracle® y no son específicos de FreeBSD.

10.7.6.4. Sugerencias

Le recomendamos añadir `linprocfs` a `/etc/fstab` . Consulte linprocfs(5) para más información. Otro parámetro que debería configurar es `kern.fallback_elf_brand=3` en `/etc/sysctl.conf` .

10.7.7. Creación del entorno SAP® R/3®

10.7.7.1. Creación de los sistemas de ficheros y puntos de montaje necesarios

Para una instalación sencilla es suficiente con crear los siguientes sistemas de ficheros:

punto de montaje	tamaño en GB
`/compat/linux/oracle`	45 GB
`/compat/linux/sapmnt`	2 GB
`/compat/linux/usr/sap`	2 GB

También es necesario crear algunos enlaces. Si no, el instalador SAP® tendrá problemas ya que buscará los siguientes enlaces:

```
# ln -s /compat/linux/oracle /oracle
# ln -s /compat/linux/sapmnt /sapmnt
# ln -s /compat/linux/usr/sap /usr/sap
```

Veamos unos cuantos errores que se le pueden presentar durante la instalación (en este caso con el sistema *PRD* y la instalación de SAP® R/3® 4.6C SR2):

```
INFO 2002-03-19 16:45:36 R3LINKS_IND_IND SyLinkCreate:200
    Checking existence of symbolic link /usr/sap/PRD/SYS/exe/dbg to
    /sapmnt/PRD/exe. Creating if it does not exist...

WARNING 2002-03-19 16:45:36 R3LINKS_IND_IND SyLinkCreate:400
    Link /usr/sap/PRD/SYS/exe/dbg exists but it points to file
    /compat/linux/sapmnt/PRD/exe instead of /sapmnt/PRD/exe. The
    program cannot go on as long as this link exists at this
    location. Move the link to another location.

ERROR 2002-03-19 16:45:36 R3LINKS_IND_IND Ins_SetupLinks:0
    can not setup link '/usr/sap/PRD/SYS/exe/dbg' with content
    '/sapmnt/PRD/exe'
```

10.7.7.2. Creación de usuarios y directorios

SAP® R/3® necesita dos usuarios y tres grupos. Los nombres de usuario dependen del "SAP® system ID" (SID), y consisten en tres letras. Algunos de estos SID están reservados por SAP® (por ejemplo `SAP` y `NIX`. Tiene una lista completa de ellos en la documentación de SAP®). Para la instalación de IDES usamos `IDS` y para la instalación de 4.6C SR2 `PRD`, dado que ese sistema está pensado para un uso de producción. Tenemos por lo tanto los siguientes grupos (Los ID de grupo pueden ser diferentes, estos son solamente los valores que utilizamos en nuestra instalación):

ID de grupo	nombre de grupo	descripción
100	dba	Administrador de base de datos
101	sapsys	Sistema SAP®
102	oper	Operador de base de datos

En una instalación por omisión de Oracle® solo se usa el grupo `dba`. Puede usar el grupo `oper` como grupo `dba` (consulte la documentación de Oracle® y SAP® para más información).

También necesitaremos los siguientes usuarios:

ID de usuario	nombre de usuario	nombre genérico	grupo	grupos adicionales	descripción
1000	idsadm/prdadm	*sid*adm	sapsys	oper	Administrador SAP®
1002	oraids/oraprd	ora*sid*	dba	oper	Administrador Oracle®

Al añadir dichos usuarios mediante adduser(8) tenga en cuenta que debe incluir las siguientes entradas (observe la shell y el directorio home) al crear el "administrador SAP®":

```
Name: sidadm
Password: ******
Fullname: SAP Administrator SID
Uid: 1000
Gid: 101 (sapsys)
Class:
Groups: sapsys dba
HOME: /home/sidadm
Shell: bash  (/compat/linux/bin/bash)
```

y para el "Administrador Oracle®":

```
Name: orasid
Password: ******
Fullname: Oracle Administrator SID
Uid: 1002
Gid: 100 (dba)
Class:
Groups: dba
HOME: /oracle/sid
Shell: bash  (/compat/linux/bin/bash)
```

Esto también incluye al grupo `oper` en caso de que esté usando el grupo `dba` y el grupo `oper`.

10.7.7.3. Creación de directorios

Estos directorios se crean como sistemas de ficheros independientes. Esto depende totalmente de sus necesidades. Nosotros decidimos crearlos como directorios ya que todos están en el mismo RAID 5:

Primero vamos a configurar los propietarios y los derechos de algunos directorios (como `root`):

```
# chmod 775 /oracle
# chmod 777 /sapmnt
# chown root:dba /oracle
# chown sidadm:sapsys /compat/linux/usr/sap
# chmod 775 /compat/linux/usr/sap
```

Luego vamos a crear directorios como el usuario `orasid`. Estos serán todos subdirectorios de `/oracle/SID`:

```
# su - orasid
# cd /oracle/SID
# mkdir mirrlogA mirrlogB origlogA origlogB
# mkdir sapdata1 sapdata2 sapdata3 sapdata4 sapdata5 sapdata6
# mkdir saparch sapreorg
```

```
# exit
```

Para la instalación de Oracle® 8.1.7 tendrá que crear unos cuantos directorios más:

```
# su - orasid
# cd /oracle
# mkdir 805_32
# mkdir client stage
# mkdir client/80x_32
# mkdir stage/817_32
# cd /oracle/SID
# mkdir 817_32
```

Nota

El directorio `client/80x_32` tiene que tener exactamente este nombre. No sustituya la *x* por un número ni por ninguna otra cosa.

En el tercer paso creamos directorios como usuario *sid*`adm`:

```
# su - sidadm
# cd /usr/sap
# mkdir SID
# mkdir trans
# exit
```

10.7.7.4. Entradas en `/etc/services`

SAP® R/3® requiere algunas entradas en `/etc/services` que es posible que no estén correctamente activadas durante la instalalación. Añada las siguientes entradas (necesita al menos las entradas correspondientes al número de instancia, en este caso, `00`. No hará ningún daño añadir todas las entradas de `00` hasta `99` para `dp`, `gw`, `sp` y `ms`). Si va a utilizar un SAProuter o necesita acceder a SAP® OSS, también necesitará `99`, ya que el puerto 3299 se usa generalmente para el proceso SAProuter en el sistema destino:

```
sapdp00    3200/tcp # SAP Dispatcher.      3200 + Instance-Number
sapgw00  3300/tcp # SAP Gateway.         3300 + Instance-Number
sapsp00  3400/tcp #                      3400 + Instance-Number
sapms00  3500/tcp #                      3500 + Instance-Number
sapms<i>SID</i> 3600/tcp # SAP Message Server.  3600 + Instance-Number
sapgw00s   4800/tcp # SAP Secure Gateway   4800 + Instance-Number
```

10.7.7.5. Locales necesarios

SAP® requiere al menos dos locales que no forman parte de la instalación por defecto de RedHat. SAP® dispone de los paquetes RPMs que pueda necesitar; puede descargalos

desde su FTP, aunque tenga en cuenta que solo pueden acceder al mismo los clientes con acceso OSS). Consulte la nota 0171356, que contiene una lista de los RPM que necesitará.

También puede crear enlaces (por ejemplo desde *de_DE* y *en_US*), pero no se lo recomendamos si pretende configurar un sistema de producción (no obstante, hemos de reconocer que a nosotros nos ha funcionado con el sistema IDES sin ningún problema). Necesitará al menos los siguientes locales:

```
de_DE.ISO-8859-1
en_US.ISO-8859-1
```

Haga los enlaces de esta manera:

```
# cd /compat/linux/usr/share/locale
# ln -s de_DE de_DE.ISO-8859-1
# ln -s en_US en_US.ISO-8859-1
```

Si no están habrá algunos problemas durante la instalación. Si se ignoran (es decir, si configura el `STATUS` de los pasos relacionados con esos locales a `OK` en el fichero `CENTRDB.R3S`) será imposible entrar al sistema SAP® sin tener que recurrir a ciertas triquiñuelas.

10.7.7.6. Personalización del kernel

Los sistemas SAP® R/3® necesitan muchos recursos, por eso hemos añadido los siguientes parámetros al fichero de configuración de su kernel:

```
# Set these for memory pigs (SAP and Oracle):
options MAXDSIZ="(1024*1024*1024)"
options DFLDSIZ="(1024*1024*1024)"
# System V options needed.
options SYSVSHM #SYSV-style shared memory
options SHMMAXPGS=262144 #max amount of shared mem. pages
#options SHMMAXPGS=393216 #use this for the 46C inst.parameters
options SHMMNI=256 #max number of shared memory ident if.
options SHMSEG=100 #max shared mem.segs per process
options SYSVMSG #SYSV-style message queues
options MSGSEG=32767 #max num. of mes.segments in system
options MSGSSZ=32 #size of msg-seg. MUST be power of 2
options MSGMNB=65535 #max char. per message queue
options MSGTQL=2046 #max amount of msgs in system
options SYSVSEM #SYSV-style semaphores
options SEMMNU=256 #number of semaphore UNDO structures
options SEMMNS=1024 #number of semaphores in system
options SEMMNI=520 #number of semaphore identifiers
options SEMUME=100       #number of UNDO keys
```

Puede consultar los valores mínimos en la documentación de SAP®. Como no hay detalles sobre Linux, consulte para mayor información la sección de HP-UX (32-bit). El sistema de instalación 4.6C SR2 tiene más memoria principal, asín que los segmentos compartidos pueden ser más extensos tanto para SAP® como para Oracle®; elija, por tanto, un número mayor de páginas de memoria compartida.

Nota

En la instalación por omisión de FreeBSD 4.5 en i386™, configure `MAXDSIZ` y `DFLDSIZ` como máximo a 1 GB. Si no lo hace podrían aparecer errores extraños como ORA-27102: out of memory y Linux Error: 12: Cannot allocate memory.

10.7.8. Instalación de SAP® R/3®

10.7.8.1. Preparación de los CDROM

Hay que montar y desmontar muchos CD-ROM durante la instalación. Si tiene suficientes unidades de CDROM, podría montarlos todos. Nosotros decidimos copiar el contenido de los CD-ROM a los directorios correspondientes:

```
/oracle/SID/sapreorg/nombre-cd
```

Donde *nombre-cd* era `KERNEL`, `RDBMS`, `EXPORT1`, `EXPORT2`, `EXPORT3`, `EXPORT4`, `EXPORT5` y `EXPORT6` para la instalación 4.6B/IDES, y `KERNEL`, `RDBMS`, `DISK1`, `DISK2`, `DISK3`, `DISK4` y `LANG` para la instalación 4.6C SR2. Todos los nombres de fichero en los CDs montados deben estar en mayúsculas; si no es así use la opción `-g` al montar. Utilice lo siguiente:

```
# mount_cd9660 -g /dev/cd0a /mnt
# cp -R /mnt/* /oracle/SID/sapreorg/nombre-cd
# umount /mnt
```

10.7.8.2. Ejecución del "script" de instalación

Primero tendrá que preparar un directorio `install`:

```
# cd /oracle/SID/sapreorg
# mkdir install
# cd install
```

Una vez arrancado el "script" de instalación copiará casi todos los ficheros relevantes en el directorio `install`:

```
# /oracle/SID/sapreorg/KERNEL/UNIX/INSTTOOL.SH
```

La instalación IDES (4.6B) incluye un sistema de demostración SAP® R/3® totalmente personalizado, así que hay seis CD EXPORT en lugar de solo tres. La plantilla de instalación `CENTRDB.R3S` está pensada para una instancia central estándar (R/3® y base de datos), no la instancia central IDES, así que hay que copiar el `CENTRDB.R3S` correspondiente del directorio `EXPORT1`. Si no lo hace `R3SETUP` solo pedirá tres CD EXPORT.

La nueva versión de SAP® 4.6C SR2 incluye cuatro CDs EXPORT. El fichero de parámetros que controla los pasos de la instalación es `CENTRAL.R3S` . A diferencia de versiones versiones anteriores, no existen patrones de instalación por separado para una instancia central con o sin base de datos. SAP® utiliza un patrón separado para la instalación de la base de datos. Para reiniciar la instalación después es suficiente reiniciarla con el fichero original.

Durante y después de la instalación, SAP® necesita que `hostname` devuelva *sólamente* el nombre del sistema, no el nombre cualificado de dominio. Configure el nombre del equipo de ese modo, o active un alias mediante `alias hostname='hostname -s'` para `ora`*`sid`* y para *`sid`*`adm` (y para `root` al menos durante los pasos de la instalación realizados como `root`). También puede configurar los ficheros `.profile` y `.login` de los usuarios que se crean durante la instalación SAP®.

10.7.8.3. Inicio de `R3SETUP` 4.6B

Asegúrese de que `LD_LIBRARY_PATH` esté configurada correctamente:

```
# export LD_LIBRARY_PATH=/oracle/IDS/lib:/sapmnt/IDS/exe:/
oracle/805_32/lib
```

Inicie `R3SETUP` como `root` desde el directorio de instalación:

```
# cd /oracle/IDS/sapreorg/install
# ./R3SETUP -f CENTRDB.R3S
```

El "script" le preguntará algunas cosas; le mostramos aquí entre corchetes la respuesta por defecto, y después la respuesta que nosotros introdujimos:

Pregunta	Por omisión	Entrada
Enter SAP System ID	[C11]	IDSIntro
Enter SAP Instance Number	[00]	Intro
Enter SAPMOUNT Directory	[/sapmnt]	Intro
Enter name of SAP central host	[troubadix.domain.de]	Intro
Enter name of SAP db host	[troubadix]	Intro
Select character set	[1] (WE8DEC)	Intro
Enter Oracle server version (1) Oracle 8.0.5, (2) Oracle 8.0.6, (3) Oracle 8.1.5, (4) Oracle 8.1.6		1Intro
Extract Oracle Client archive	[1] (Yes, extract)	Intro
Enter path to KERNEL CD	[/sapcd]	/oracle/IDS/sapreorg/KERNEL

Pregunta	Por omisión	Entrada
Enter path to RDBMS CD	[/sapcd]	/oracle/IDS/sapreorg/RDBMS
Enter path to EXPORT1 CD	[/sapcd]	/oracle/IDS/sapreorg/EXPORT1
Directory to copy EXPORT1 CD	[/oracle/IDS/sapreorg/CD4_DIR]	Intro
Enter path to EXPORT2 CD	[/sapcd]	/oracle/IDS/sapreorg/EXPORT2
Directory to copy EXPORT2 CD	[/oracle/IDS/sapreorg/CD5_DIR]	Intro
Enter path to EXPORT3 CD	[/sapcd]	/oracle/IDS/sapreorg/EXPORT3
Directory to copy EXPORT3 CD	[/oracle/IDS/sapreorg/CD6_DIR]	Intro
Enter path to EXPORT4 CD	[/sapcd]	/oracle/IDS/sapreorg/EXPORT4
Directory to copy EXPORT4 CD	[/oracle/IDS/sapreorg/CD7_DIR]	Intro
Enter path to EXPORT5 CD	[/sapcd]	/oracle/IDS/sapreorg/EXPORT5
Directory to copy EXPORT5 CD	[/oracle/IDS/sapreorg/CD8_DIR]	Intro
Enter path to EXPORT6 CD	[/sapcd]	/oracle/IDS/sapreorg/EXPORT6
Directory to copy EXPORT6 CD	[/oracle/IDS/sapreorg/CD9_DIR]	Intro
Enter amount of RAM for SAP + DB		850Intro (en Megabytes)
Service Entry Message Server	[3600]	Intro
Enter Group-ID of sapsys	[101]	Intro
Enter Group-ID of oper	[102]	Intro
Enter Group-ID of dba	[100]	Intro
Enter User-ID of *sid*adm	[1000]	Intro
Enter User-ID of ora*sid*	[1002]	Intro

Pregunta	Por omisión	Entrada
Number of parallel procs	[2]	Intro

Si no ha copiado los CD a su disco duro el instalador SAP® no podrá encontrar el CD que necesite (identifica los contenidos mediante fichero `LABEL.ASC` de cada CD) y por lo tanto le pedirá que introduzca y monte el CD, o que confirme o introduzca la ruta de montaje.

`CENTRDB.R3S` puede contener algún error. En nuestro caso, solicitó el CD EXPORT4 más de una vez, pero se le indicó la clave correcta (6_LOCATION, luego 7_LOCATION, etc), así que pudimos continuar introduciendo los valores correctos.

Aparte de algunos problemas que se detallan más adelante deberíamos ir llegando a la instalación del software de base de datos Oracle®.

10.7.8.4. Iniciar `R3SETUP` 4.6C SR2

Asegúrese de que `LD_LIBRARY_PATH` esté correctamente configurada. Tenga en cuenta de que es un valor diferente de la instalación 4.6B con Oracle® 8.0.5:

```
# export LD_LIBRARY_PATH=/sapmnt/PRD/exe:/oracle/PRD/817_32/lib
```

Arranque `R3SETUP` como el usuario `root` desde el directorio de instalación:

```
# cd /oracle/PRD/sapreorg/install
# ./R3SETUP -f CENTRAL.R3S
```

El "script" le preguntará algunas cosas. Le presentamos la respuesta por omisión entre corchetes y después la respuesta que dimos nosotros):

Pregunta	Por omisión	Entrada
Enter SAP System ID	[C11]	PRDIntro
Enter SAP Instance Number	[00]	Intro
Enter SAPMOUNT Directory	[/sapmnt]	Intro
Enter name of SAP central host	[majestix]	Intro
Enter Database System ID	[PRD]	PRDIntro
Enter name of SAP db host	[majestix]	Intro
Select character set	[1] (WE8DEC)	Intro
Enter Oracle server version (2) Oracle 8.1.7		2Intro
Extract Oracle Client archive	[1] (Yes, extract)	Intro
Enter path to KERNEL CD	[/sapcd]	/oracle/PRD/sapreorg/KERNEL

Pregunta	Por omisión	Entrada
Enter amount of RAM for SAP + DB	2044	1800Intro (in Megabytes)
Service Entry Message Server	[3600]	Intro
Enter Group-ID of sapsys	[100]	Intro
Enter Group-ID of oper	[101]	Intro
Enter Group-ID of dba	[102]	Intro
Enter User-ID of `oraprd`	[1002]	Intro
Enter User-ID of `prdadm`	[1000]	Intro
LDAP support		3Intro (no support)
Installation step completed	[1] (continue)	Intro
Choose installation service	[1] (DB inst,file)	Intro

La creación de usuarios da un error durante la instalación en las fases OSUSERDBSID_IND_ORA (al crear al usuario `ora`*`sid`*) y OSUSERSIDADM_IND_ORA (al crear el usuario *`sid`*`adm`).

Más adelante hablaremos de cierto problemas que aún tenemos pendientes, pero ha llegado el momento de instalar el software de base de datos Oracle®.

10.7.9. Instalación de Oracle® 8.0.5

Consulte los `Readme` de Oracle® y las notas de de SAP® sobre Linux y Oracle® DB por si hubiera algo que le pueda afectar. La mayoría de los problemas, por no decir todos, tienen su origen en bibliotecas incompatibles.

Para mayor información sobre la instalación de Oracle® diríjase al capítulo de instalación de Oracle®.

10.7.9.1. Instalación de Oracle® 8.0.5 con `orainst`

Si quiere instalar Oracle® 8.0.5 necesitará unas cuantas bibliotecas para el enlazado, ya que Oracle® 8.0.5 fué enlazado con una glibc antigua (la de RedHat 6.0), pero RedHat 6.1 usa una nueva glibc. Tendrá que instalar los siguientes paquetes para asegurarse que el reenlazado funcione:

```
compat-libs-5.2-2.i386.rpm

compat-glibc-5.2-2.0.7.2.i386.rpm

compat-egcs-5.2-1.0.3a.1.i386.rpm
```

`compat-egcs-c++-5.2-1.0.3a.1.i386.rpm`

`compat-binutils-5.2-2.9.1.0.23.1.i386.rpm`

Para más información consulte las notas correspondientes de SAP® o los `Readme` de Oracle®. Si no es posible (durante la instalación no tuvimos tiempo suficiente para ello), se podrían utilizar los binarios originales, o los binarios reenlazados de un sistema original RedHat.

Instale el paquete Tcl de RedHat para compilar el agente inteligente. Si no puede conseguir `tcl-8.0.3-20.i386.rpm` debería funcionar una versión más reciente, por ejemplo `tcl-8.0.5-30.i386.rpm` para RedHat.

Aparte del reenlazado, la instalación es muy sencilla:

```
# su - oraids
# export TERM=xterm
# export ORACLE_TERM=xterm
# export ORACLE_HOME=/oracle/IDS
# cd $ORACLE_HOME/orainst_sap
# ./orainst
```

Confirme todas las pantallas con Intro hasta que el software esté instalado; todas excepto en la que debe quitar la marca de instalación al *visualizador de textos Oracle® en línea*, ya que no existe para Linux. Oracle® intentará, gracias a esto, reenlazar con `i386-glibc20-linux-gcc` en lugar de `gcc`, `egcs` o `i386-redhat-linux-gcc` .

Debido a la falta de tiempo decidimos usar los binarios de una versión Oracle® 8.0.5 Pre-Production, después de que nuestro primer intento de que funcionara la versión del CD RDBMS fallara y viendo que encontrar y utilizar los RPM correctos hubiera sido una pesadilla.

10.7.9.2. Instalación de Oracle® 8.0.5 “Pre-production Release” para Linux (Kernel 2.0.33)

La instalación es bastante fácil. Monte el CD e inicie el instalador. Le preguntará por la ubicación del directorio home de Oracle®, y copiará en él todos los binarios. (Nosotros no eliminamos los restos de una instalación RDBMS anterior que no terminó de llegó a terminar).

Tras esto la base de datos Oracle® puede arrancar.

10.7.10. Instalación desde el fichero comprimido de Linux Oracle® 8.1.7

Descomprima el fichero `oracle81732.tgz` (creado en el directorio de instalación en un sistema Linux) y descomprímalo en `/oracle/SID/817_32/` .

10.7.11. Continúe con la instalación SAP® R/3®

Revise las configuraciones del entorno de los usuarios `idsamd` (*sid*adm) y `oraids` (ora*sid*). Ambos deben tener los ficheros `.profile`, `.login` y `.cshrc` con `hostname` correctamente configurado. En caso que el nombre del sistema sea el nombre cualificado completo tendrá que cambiar `hostname` a `hostname -s` en los tres ficheros anteriormente citados.

10.7.11.1. Carga de la base de datos

Hecho esto puede rearrancar `R3SETUP` o volver a arrancar la instalación (dependiendo si eligió salir o no). `R3SETUP` crea las tablas y carga los datos (en 46B IDES, desde EXPORT1 a EXPORT6, en 46C desde DISK1 a DISK4) mediante `R3load`.

Cuando se termina la carga de la base de datos (que puede llevar un par de horas) se le pedirán algunas contraseñas. En una instalación de prueba puede usar unas contraseñas de compromiso. (*use una contraseña de verdad si le preocupa siquiera ligeramente la seguridad*):

Pregunta	Entrada
Enter Password for sapr3	sapIntro
Confirum Password for sapr3	sapIntro
Enter Password for sys	change_on_installEnter
Confirm Password for sys	change_on_installEnter
Enter Password for system	managerIntro
Confirm Password for system	managerIntro

Aquí tuvimos problemas con `dipgntab` en la instalación de 4.6B.

10.7.11.2. Las escuchas

Arranque las escuchas de Oracle® con el usuario `ora`*sid* de la siguiente manera:

```
% umask 0; lsnrctl start
```

Si no lo hace así verá un error ORA-12546, ya que los sockets no tendrán los permisos correctos. Consulte la nota 072984 de SAP®.

10.7.11.3. Actualización de tablas MNLS

Si tiene previsto importar idiomas que no sean Latin-1 en SAP® tiene que actualizar las tablas "Multi National Language Support". Tiene más información sobre esto en las notas de SAP® OSS 15023 y 45619. Si no es su caso puede saltarse esta parte de la instalación de SAP®.

Nota

Aunque no necesite soporte MNLS sigue siendo necesario que revise la tabla TCPDB y que la inicialice si no lo ha hecho ya. Consulte las notas 0015023 y 0045619 de SAP® para más información.

10.7.12. Pasos para después de la instalación

10.7.12.1. Solicitar una licencia SAP® R/3®

Tiene que solicitar una licencia de SAP® R/3®. No tendrá más remedio, puesto que la licencia temporal que se usa durante la instalación tiene un límite de validez de cuatro semanas. Necesitará la llave hardware. Entre al sistema como usuario `idsadm` y ejecute `saplicense`:

```
# /sapmnt/IDS/exe/saplicense -get
```

Si ejecuta `saplicense` sin parámetros verá una lista de opciones. Una vez que tenga la licencia en su poder la podrá instalar del siguiente modo:

```
# /sapmnt/IDS/exe/saplicense -install
```

Se le solicitará que introduzca los siguientes valores:

```
SAP SYSTEM ID   = SID, 3 caracteres
CUSTOMER KEY    = llave hardware, 11 caracteres
INSTALLATION NO = instalación, 10 caracteres
EXPIRATION DATE = yyyymmdd, normalmente "99991231"
LICENSE KEY     = licencia, 24 caracteres
```

10.7.12.2. Crear usuarios

Cree un usuario dentro del cliente `000` (es necesario para algunas tareas que requieren hacerse dentro del cliente `000`, pero con un usuario que no sea ni `sap*` ni `ddic`). Nosotros solemos elegir para este usuario el nombre de `wartung` (o `service`, ambos "servicio" en castellano). Los perfiles son `sap_new` y `sap_all`. Para mayor seguridad las contraseñas para usuarios por defecto dentro de todos los clientes deben cambiarse (incluidos los usuarios `sap*` y `ddic`).

10.7.12.3. Configurar sistema de transporte, perfil, modos de operación, etc.

Dentro del cliente `000` y con un usuario que no sea `ddic` ni `sap*`, haga al menos lo siguiente:

Tarea	Transacción
Configurar sistema de transporte, por ejemplo como *Stand-Alone Transport Domain Entity*	STMS
Crear / editar perfil para el sistema	RZ10
Mantener modos de operación e instancias	RZ04

Todos estos (y muchos más) pasos para ejecutar después de la instalación se explican de forma detallada en las guías de instalación de SAP®.

10.7.12.4. Editar `initsid.sap` (`initIDS.sap`)

El fichero `/oracle/IDS/dbs/initIDS.sap` contiene la copia de seguridad del perfil de de SAP®. Aquí es donde debe definir el tamaño de la cinta a utilizar, tipo de conpresión, etc. Las siguientes modificaciones nos permitirían ejecutar `sapdba` / `brbackup`:

```
compress = hardware
archive_function = copy_delete_save
cpio_flags = "-ov --format=newc --block-size=128 --quiet"
cpio_in_flags = "-iuv --block-size=128 --quiet"
tape_size = 38000M
tape_address = /dev/nsa0
tape_address_rew = /dev/sa0
```

Explicación:

`compress`: La cinta que usamos es una HP DLT1 que tiene compresión por hardware.

`archive_function` : Define el comportamiento por omisión del almacenaje de los logs de Oracle®: los nuevos ficheros de log se guardan en cinta, los ficheros de log que ya han sido guardados se guardan de nuevo y luego se borran. Así se evitan muchos problemas si necesita recuperar la base de datos y una de las cintas está dañada.

`cpio_flags` : por omisión se usa `-B`, que asigna un tamaño de bloque de 5120 Bytes. HP recomienda un tamaño de bloque de 32 K como mínimo; usamos `--block-size=128` para que sea de 64 K. Necesitaremos usar `--format=newc` porque tenemos números de inodo mayores a 65535. La última opción (`--quiet`) se necesita ya que `brbackup` se queja en cuanto `cpio` imprime los números de bloque guardados.

`cpio_in_flags` : Parámetros necesarios para cargar datos desde la cinta. El formato es reconocido automáticamente.

`tape_size` : La capacidad de almacenaje de la cinta. Por razones de seguridad (nosotros usamos compresión por hardware) el valor es ligeramente menor que el valor real.

`tape_address` : El dispositivo (que no permite el rebobinado) que se usará con `cpio`.

`tape_address_rew` : El dispositivo (que permite el rebobinado) que se usará con `cpio`.

10.7.12.5. Aspectos de la configuración una vez concluida la instalación

Los siguientes parámetros SAP® deben personalizarse una vez concluída la instalación (los ejemplos son para IDES 46B, 1 GB de memoria):

Nombre	Valor
ztta/roll_extension	250000000
abap/heap_area_dia	300000000
abap/heap_area_nondia	400000000
em/initial_size_MB	256
em/blocksize_kB	1024
ipc/shm_psize_40	70000000

SAP® Note 0013026:

Nombre	Valor
ztta/dynpro_area	2500000

SAP® Note 0157246:

Nombre	Valor
rdisp/ROLL_MAXFS	16000
rdisp/PG_MAXFS	30000

Nota

En un sistema con 1 GB de memoria y los parámetros arriba expuestos puede esperarse encontrar un consumo de memoria similar al siguiente:

```
Mem: 547M Active, 305M Inact, 109M Wired, 40M ↺
Cache, 112M Buf, 3492K Free
```

10.7.13. Problemas durante la instalación

10.7.13.1. Reiniciar `R3SETUP` una vez arreglado el problema

`R3SETUP` se detiene si encuentra un error. Si ha revisado los logs y ha corregido el error reinicie `R3SETUP`; hágalo seleccionando la opción REPEAT en el paso donde `R3SETUP` se detuvo.

Cuando quiera reiniciar R3SETUP inícielo con el fichero R3S correspondiente:

```
# ./R3SETUP -f CENTRDB.R3S
```

en el caso de 4.6B, o con

```
# ./R3SETUP -f CENTRAL.R3S
```

en 4.6C; no importa si el error ocurrió con CENTRAL.R3S o con DATABASE.R3S .

Nota

En algunas etapas, R3SETUP asume que la base de datos y los procesos SAP® están en marcha (como aquellos en los cuales se completaron los pasos). Si hay errores y por ejemplo la base de datos no se puede iniciar tendrá que arrancar la base de datos y SAP® manualmente una vez haya corregido los errores y antes de iniciar R3SETUP nuevamente.

No olvide iniciar también la escucha de Oracle® (como `ora`*`sid`* con `umask 0; lsnrctl start`) si también tuvo que detenerlo (si por ejemplo hubo que reiniciar el sistema).

10.7.13.2. OSUSERSIDADM_IND_ORA durante R3SETUP

Si R3SETUP se queja en este etapa edite la plantilla R3SETUP que esté usando en ese momento (CENTRDB.R3S (en 4.6B) o CENTRAL.R3S o DATABASE.R3S (en 4.6C)). Ubique [OSUSERSIDADM_IND_ORA] o busque la única entrada STATUS=ERROR y edite los siguientes valores:

```
HOME=/home/sidadm (was empty)
STATUS=OK (had status ERROR)
```

Hecho esto, reinicie R3SETUP .

10.7.13.3. OSUSERDBSID_IND_ORA durante R3SETUP

Es posible que R3SETUP se queje también en esta etapa. El error aquí es similar al de la fase OSUSERSIDADM_IND_ORA. Edite la plantilla R3SETUP que esté usando (CENTRDB.R3S (en 4.6B) o CENTRAL.R3S o DATABASE.R3S (en 4.6C)). Ubique [OSUSERDBSID_IND_ORA] o busque la única entrada STATUS=ERROR y edite los siguientes valores en esa sección:

```
STATUS=OK
```

Hecho esto reinicie R3SETUP .

10.7.13.4. oraview.vrf FILE NOT FOUND durante la instalación de Oracle®

No ha dejado sin seleccionar la opción de instalar el *visualizador de texto en línea de Oracle®* antes de iniciar la instalación. Está seleccionado para ser instalado, aunque la aplicación no existe para Linux. Deje sin seleccionar el producto en el menú de instalación de Oracle® y reinicie la instalación.

10.7.13.5. TEXTENV_INVALID durante `R3SETUP`, o inicio de RFC o SAPgui

Si se encuentra con este error significa que falta el locale correcto. La nota 0171356 de SAP® contiene una lista de RPM que deben instalarse (p.ej. `saplocales-1.0-3`, `saposcheck-1.0-1` para RedHat 6.1). En caso de que ignore todos los errores relacionados y configure los `STATUS` correspondientes de `ERROR` a `OK` (en `CENTRDB.R3S`) cada vez que `R3SETUP` se queje y simplemente reinicie `R3SETUP` ; el sistema SAP® no estará configurado correctamente y no podrá conectarse al sistema con SAPgui, aunque el sistema pueda arrancar. Si intenta conectar con el antiguo SAPgui de Linux recibirá los siguientes mensajes:

```
Sat May 5 14:23:14 2001
*** ERROR => no valid userarea given [trgmsgo. 0401]
Sat May 5 14:23:22 2001
*** ERROR => ERROR NR 24 occured [trgmsgi. 0410]
*** ERROR => Error when generating text environment. [trgmsgi. 0435]
*** ERROR => function failed [trgmsgi. 0447]
*** ERROR => no socket operation allowed [trxio.c 3363]
Speicherzugriffsfehler
```

Este comportamiento se debe a que SAP® R/3® es incapaz de asignar correctamente un locale y tampoco puede configurarse a sí mismo correctamente (faltan entradas en algunas tablas de la base de datos). Añada las siguientes entradas al fichero `DEFAULT.PFL` y podrá conectarse a SAP® (vea la nota 0043288):

```
abap/set_etct_env_at_new_mode = 0
install/collate/active = 0
rscp/TCP0B = TCP0B
```

Reinicie el sistema SAP®. Puede conectar al sistema, aunque la configuración de idioma o de país puede que no funcione como se espera de ella. Una vez corregidas las configuraciones de país (y proporcionados los locales adecuados) puede eliminar estas entradas de `DEFAULT.PFL` y el sistema SAP® puede reiniciarse.

10.7.13.6. ORA-00001

Este error solo aparece con Oracle® 8.1.7 en FreeBSD 4.5. Se debe a que la base de datos Oracle® no puede inicializarse correctamente y se viene abajo, dejando semáforos y memoria compartida en el sistema. El siguiente intento de iniciar la base de datos produce el error ORA-00001.

Encuéntrelos con `ipcs -a` y elimínelos con `ipcrm`.

10.7.13.7. ORA-00445 (Brackground Process PMON Did Not Start)

Este error tuvo lugar con Oracle® 8.1.7. Aparece si se arranca la base de datos con el "script" `startsap` (por ejemplo `startsap_majestix_00`) con el usuario `prdadm`.

Una solución (entre otras) es iniciar la base de datos con el usuario `oraprd` en lugar de hacerlo con `svrmgrl` :

```
% svrmgrl
SVRMGR> connect internal;
SVRMGR> startup ;
SVRMGR> exit
```

10.7.13.8. ORA-12546 (Start Listener with Correct Permissions)

Inicie la escucha de Oracle® como usuario `oraids` con la siguiente orden:

```
# umask 0; lsnrctl start
```

De no ser así puede encontrarse con el error ORA-12546, ya que los sockets no tendrán los permisos adecuados. Consulte la nota de SAP® 0072984.

10.7.13.9. ORA-27102 (Out of Memory)

Este error ocurre al tratar de usar valores mayores a 1 GB (1024x1024x1024) en `MAXDSIZ` y `DFLDSIZ` . Recibiremos, además, este otro error: Linux Error 12: Cannot allocate memory.

10.7.13.10. [DIPGNTAB_IND_IND] during `R3SETUP`

Consulte la nota de SAP® 0130581 de (`R3SETUP` step (`DIPGNTAB` terminates). Por alguna razón durante la instalación específica IDES el proceso de instalación no usaba el nombre de sistema correcto SAP® "IDS" sino la cadena vacía `""`. Esto provocaba algunos errores menores en el acceso a directorios, ya que las rutas se generan dinámicamente en base a dicho *`SID`* (en este caso IDS). En lugar de ejecutar los accesos del siguiente modo :

```
/usr/sap/IDS/SYS/...
/usr/sap/IDS/DVMGS00
```

se usaron las siguientes rutas:

```
/usr/sap//SYS/...
/usr/sap/D00
```

Para continuar con la instalación creamos un enlace y un directorio adicional:

```
# pwd
/compat/linux/usr/sap
# ls -l
total 4
drwxr-xr-x 3  idsadm sapsys 512 May 5 11:20 D00
drwxr-x--x 5  idsadm sapsys 512 May 5 11:35 IDS
```

```
lrwxr-xr-x 1  root   sapsys 7 May 5 11:35 SYS -> IDS/SYS
drwxrwxr-x 2  idsadm sapsys 512 May 5 13:00 tmp
drwxrwxr-x 11 idsadm sapsys 512 May 4 14:20 trans
```

Encontramos una descripción de este comportamiento en las notas de SAP®. (0029227 y 0008401). En la instalación de SAP® 4.6C no tuvimos estos problemas.

10.7.13.11. [RFCRSWBOINI_IND_IND] during R3SETUP

Durante la instalación de SAP® 4.6C nos encontramos con este error, cuyo origen está en un error que tuvo lugar anteriormente durante la propia instalación. Busque en sus ficheros de log y corrija el problema.

Si despues de buscar en los logs el error resulta ser el correcto (revise las notas de SAP®), puede poner el `STATUS` del paso donde se produce el error de `ERROR` a `OK` (en el fichero `CENTRDB.R3S`) y reiniciar `R3SETUP` . Una vez finalizada la instalación, tiene que ejecutar el informe `RSWBOINS` de la transacción SE38. Consulte la nota SAP® 0162266 para más información sobre las fases `RFCRSWBOINI` y `RFCRADDBDIF`.

10.7.13.12. [RFCRADDBDIF_IND_IND] during R3SETUP

Aquí sucede lo mismo de antes, asegúrese, revisando los logs, de que la causa de este error no esté en algún problema previo.

Si en la nota de SAP® 0162266 está la solución ponga el `STATUS` del paso donde se produce el error de `ERROR` a `OK` (en el fichero `CENTRDB.R3S`) y reinicie `R3SETUP` . Una vez finalizada la instalación ejecute el informe `RADDBDIF` desde la transacción SE38.

10.7.13.13. sigaction sig31: File size limit exceeded

Este error ocurrió durante el inicio del proceso SAP® *disp+work*. Si inicia SAP® con el "script" `startsap` se inician los subprocesos que se separan y hacen el "trabajo sucio" de iniciar el resto de procesos de SAP®, pero es importante saber que el propio "script" no notará si algo ha ido mal.

Puede revisar si los procesos SAP® se iniciaron correctamente con `ps ax | grep SID` , que le proporcionará una lista de todos los procesos de Oracle® y de SAP®. Si parece que algunos procesos no están, o si no puede conectarse al sistema SAP® revise los logs que encontrará en `/usr/sap/SID/DVEBMGSnr/work/` . Los ficheros que debe revisar son `dev_ms` y `dev_disp` .

La señal 31 aparece si la cantidad de memoria compartida asignada a Oracle® y SAP® supera la definida dentro del fichero de configuración del kernel y puede resolverse usando un valor mayor:

```
# larger value for 46C production systems:
options SHMMAXPGS=393216
# smaller value sufficient for 46B:
```

```
#options SHMMAXPGS=262144
```

10.7.13.14. Start of `saposcol` Failed

Hay algunos problemas con el programa `saposcol` (version 4.6D). El sistema SAP® utiliza `saposcol` para recoger datos del rendimiento del sistema. Este programa no es necesario para usar el sistema SAP®, así que el problema puede considerarse como poco importante. La versión más antigua (4.6B) funciona, pero no recoge todos los datos (muchas llamadas devolverán un 0, por ejemplo el uso de CPU).

10.8. Temas avanzados

Si siente curiosidad por saber cómo funciona la compatibilidad con Linux esta es la sección que debe leer. La mayor parte de lo que sigue está basado casi en su totalidad en un mensaje enviado por Terry Lambert <tlambert@primenet.com > a la lista lista de charla de FreeBSD (Message ID: <199906020108.SAA07001@usr09.primenet.com>).

10.8.1. ?Cómo funciona?

FreeBSD dispone de una abstracció denominada “cargador de clase en ejecución”. Esto no es más que un bloque de código incrustado en la llamada execve(2) del sistema.

Históricamente las plataformas UNIX® disponían de un único cargador de binarios, que en última instancia (*fallback*) recurría al cargador `#!` para ejecutar cualesquiera intérpretes o scripts de la shell. Ese cargador único examinaba el número mágico (generalmente los 4 u 8 primeros bytes del fichero) para ver si era un binario reconocible por el sistema y, en tal caso, invocaba al cargador binario.

Si no era de tipo binario, la llamada execve(2) devolvía un error y la shell intentaba empezar a ejecutarlo como órdenes shell, tomando por defecto como punto de partida “la shell actual, sea cual sea”.

Posteriormente se pensó en hacer una modificación de manera que sh(1) examinara los dos primeros caracteres, de modo que si eran `:\n` se llamaba a la shell csh(1) en su lugar (parece ser que en SCO fueron los primeros en utilizar ese truco).

Lo que ocurre ahora es que FreeBSD dispone de una lista de cargadores, en lugar de uno solo. FreeBSD recorre esa lista de cargadores, con un cargador genérico `#!` que sabe reconocer los intérpretes en base a los caracteres que siguen al siguiente espacio en blanco, con `/bin/sh` como último recurso.

Para dar soporte a la ABI (“Application Binary Interface”) de Linux, FreeBSD interpreta el número mágico como un binario ELF (“Executable and Linking Format”): En este punto no hace distinción entre FreeBSD, Solaris™, Linux® o cualquier otro SO que tenga un tipo de imagen ELF.

El cargador ELF busca entonces una marca (*brand*) especial, una sección de comentarios en la imagen ELF que no está presente en los binarios ELF de SVR4/Solaris™.

Para que los binarios de Linux funcionen deben estar marcados con brandelf(1) como tipo `Linux`:

```
# brandelf -t Linux file
```

Hecho esto el cargador ELF verá la marca `Linux` en el fichero.

Cuando el cargador ELF ve la marca `Linux` sustituye un puntero en la estructura `proc`. Todas las llamadas del sistema se indexan a través de este puntero (en un sistema UNIX® tradicional sería el «array» de estructura `sysent[]` que contiene las llamadas del sistema). Además, el proceso se marca con unos indicadores ("flags") para que el vector trampa del código de envío señales lo maneje de una forma determinada, así como otros arreglos (menores) que serán utilizados por el módulo Linux del kernel.

El vector de llamada del sistema Linux contiene, entre otras cosas, una lista de entradas `sysent[]` cuyas direcciones residen en el módulo del kernel.

Cuando el binario Linux realiza una llamada al sistema, el código trampa extrae el puntero a la función de la llamada del sistema de la estructura `proc`, y así obtiene los puntos de entrada a las llamadas del sistema Linux, no las de FreeBSD.

Además, el modo Linux cambia la raíz de las búsquedas de una forma dinámica. En efecto, esto es lo que hace la opción `union` cuando se monta un sistema de ficheros (¡y que *no* es lo mismo que el sistema de ficheros `unionfs` !). Primero se hace un intento de buscar el fichero en el directorio `/compat/linux/ruta-original` y *solo después*, si lo anterior falla, se repite la búsqueda en el directorio `/ruta-original`. Esto permite que se puedan ejecutar binarios que necesitan de otros binarios (por ejemplo las herramientas de programación ("toolchain") de Linux pueden ejecutarse en su totalidad bajo la ABI de Linux). Esto significa también que los binarios Linux pueden cargar y ejecutar binarios FreeBSD si los binarios Linux equivalentes no se hallan presentes y que se puede poner una orden uname(1) en el árbol de directorios `/compat/linux` para poder estar seguros de que los binarios Linux no puedan decir que no estaban ejecutándose en Linux.

En efecto, hay un kernel Linux en el kernel FreeBSD; las distintas funciones subyacentes que implementan todos los servicios proporcionados por el kernel son idénticas en ambas, las tablas de entradas de llamadas del sistema en FreeBSD y en Linux: operaciones del sistema de ficheros, operaciones de memoria virtual, envío de señales IPC System V, etc. La única diferencia es que los binarios FreeBSD reciben sus funciones de conexión ("*glue*") y los binarios Linux las suyas (la mayoría de los sistemas operativos más antiguos solo tienen sus propias funciones de conexión: direcciones de funciones en un "array" de estructura `sysent[]` estática y global, en lugar de direcciones de funciones que se extraen a partir de un puntero inicializado dinámicamente en la estructura `proc` del proceso que hace la llamada).

?Cuál es entonces la ABI nativa de FreeBSD? No importa. Básicamente, la única diferencia es (ahora mismo; esto podría cambiar y probablemente lo hará en una release futura) que las funciones de conexión de FreeBSD están enlazadas estáticamente en el kernel mientras que las de Linux pueden estarlo también estáticamente o se puede acceder a ellas por medio de un módulo del kernel.

Bien, pero ?de verdad es esto una emulación? No. Es una implementación ABI, no una emulación. No hay un emulador involucrado (ni un simulador, para adelantarnos a la siguiente pregunta).

Entonces ?por qué a veces se le llama "emulación Linux"? ¡Para hacer más difícil el vender FreeBSD! En serio, se debe a que la primera implementación se hizo en un momento en que realmente no había ninguna palabra distinta a esa para describir lo que se estaba haciendo; decir que FreeBSD ejecutaba binarios Linux no era cierto si no se compilaba el código o se cargaba un módulo; hacía falta una forma de describir todo esto y acabamos usando "emulador Linux".

Parte III. Administración del sistema.

Los restantes capítulos de este libro cubren todos los aspectos de administración de un sistema FreeBSD. Cada capítulo comienza describiendo lo que será aprendido una vez finalizada la lectura, explicando también los conocimientos mínimos necesarios para una comprensión satisfactoria del texto.

Estos capítulos están diseñados para leerse cuando se necesita la información. No es necesario leerlos en un determinado orden ni es necesario leerlos todos antes de comenzar a usar FreeBSD.

Tabla de contenidos

11. Configuración y Adaptación del Sistema 317
    11.1. Configuración de Tarjetas de Red 317
    11.2. “Arrancar servicios” 317
    11.3. “Soft Updates” 317
    11.4. Añadir espacio swap 317
12. El proceso de arranque en FreeBSD 319
    12.1. Sinopsis 319
    12.2. El problema que representa arrancar el sistema 319
    12.3. El RMA y las etapas de arranque uno, dos y tres 321
    12.4. Interacción con el kernel durante el arranque 326
    12.5. Device Hints 326
    12.6. Init: inicialización del proceso de control 327
    12.7. Secuencia de apagado 328
13. Usuarios y administración básica de cuentas 329
    13.1. Sinopsis 329
    13.2. Introducción 329
    13.3. La cuenta superusuario 331
    13.4. Cuentas de sistema 332
    13.5. Cuentas de usuario 332
    13.6. Modificación de cuentas 332
    13.7. Limitar a los usuarios 339
    13.8. Personalizar a los usuarios 342
    13.9. Grupos 342
14. Seguridad 345
    14.1. Sinopsis 345
    14.2. Introducción 346
    14.3. Asegurar FreeBSD 348
    14.4. DES, MD5 y Crypt 357
    14.5. Contraseñas de un solo uso 358
    14.6. TCP Wrappers 364
    14.7. KerberosIV 368
    14.8. Kerberos5 376
    14.9. OpenSSL 386
    14.10. VPN sobre IPsec 390
    14.11. OpenSSH 404
    14.12. Listas de control de acceso a sistemas de ficheros 411
    14.13. Monitorización de fallos de seguridad de aplicaciones 413
    14.14. FreeBSD Security Advisories 414
    14.15. Contabilidad de procesos 417
15. Jaulas 419
    15.1. Sinopsis 419
    15.2. Términos relacionados con las jaulas 419
    15.3. Introducción 420
    15.4. Creación y gestión de jaulas 421

15.5. Administración y personalización a fondo 424
15.6. Uso de las jaulas 425
16. Mandatory Access Control 433
16.1. Sinopsis 433
16.2. Términos clave en este capítulo 433
16.3. Explicación de MAC 433
16.4. Las etiquetas MAC 433
16.5. Configuración de módulos 433
16.6. El módulo MAC ifoff 433
16.7. El módulo MAC portacl 433
16.8. Políticas de etiquetas MAC 434
16.9. El módulo MAC partition 434
16.10. El módulo de seguridad multinivel MAC 434
16.11. El módulo MAC Biba 434
16.12. El módulo MAC LOMAC 434
16.13. Implementación de un entorno seguro con MAC 434
16.14. Otro ejemplo: Uso de MAC para restringir un servidor web 434
16.15. Depuración de errores en MAC 434
17. Auditoría de eventos de seguridad 435
17.1. * 435
18. Almacenamiento 437
18.1. Sinopsis 437
18.2. Nombres de dispositivo 437
18.3. Añadir discos 438
18.4. RAID 441
18.5. Dispositivos de almacenamiento USB 447
18.6. Creación y uso de medios ópticos (CD) 449
18.7. Crear y utilizar medios ópticos (DVDs) 456
18.8. Creación y uso de disquetes (floppies) 462
18.9. Creación y uso de cintas de datos 464
18.10. Respaldos en disquetes 468
18.11. Bases para respaldos 469
18.12. Sistemas de ficheros en red, memoria y respaldados en fichero 478
18.13. Instantáneas (“snapshots”) de sistemas de ficheros 483
18.14. Cuotas en sistemas de ficheros 484
18.15. Cifrado de particiones de disco 488
19. GEOM: Marco de trabajo modular de transformación de discos 495
19.1. Sinopsis 495
19.2. Introducción a GEOM 495
19.3. RAID0 - Distribución por bandas 496
19.4. RAID1 - Replicación 497
20. El Gestor de Volúmenes Vinum 503
20.1. Sinopsis 503
20.2. Los Discos son Demasiado Pequeños 503
20.3. Cuellos de Botella en el Acceso 503
20.4. Integridad de Datos 504

20.5. Objetos Vinum 504
20.6. Ejemplos 505
20.7. Esquema de Nombres de los Objetos 508
20.8. Configuración de Vinum 508
20.9. Uso de Vinum en el Sistema de Ficheros Raíz 509
21. Virtualización 511
21.1. * 511
22. Localización - Uso y configuración de I18N/L10N 513
22.1. Sinopsis 513
22.2. Lo básico 513
22.3. Uso de la localización 514
22.4. Compilación de programas con soporte para I18N 521
22.5. Localización de FreeBSD a idiomas específicos 522
23. Lo último de lo último 527
23.1. Sinopsis 527
23.2. FreeBSD-CURRENT vs. FreeBSD-STABLE 527
23.3. Sincronización de su código fuente 527
23.4. Uso de `make world` 527
23.5. Redes pequeñas 527

Capítulo 11. Configuración y Adaptación del Sistema

Configuración y Adaptación del Sistema

11.1. Configuración de Tarjetas de Red

Pendiente de Traducción

11.2. “Arrancar servicios”

Pendiente de Traducción

11.3. “Soft Updates”

Pendiente de Traducción

11.4. Añadir espacio swap

Pendiente de traducción

Capítulo 12. El proceso de arranque en FreeBSD

12.1. Sinopsis

Al proceso de inicio del sistema y carga del sistema operativo se le conoce como "mecanismo de arranque" (bootstrap), o simplemente "arranque" (booting). El proceso de arranque de FreeBSD provee de gran flexibilidad al configurar lo que ocurre cuando se inicia el sistema, permitiendole seleccionar de diferentes sistemas operativos instalados en el mismo ordenador, o inclusive diferentes versiones del mismo sistema oprativo o kernels instalados.

Este capítulo detalla las opciones de configuración que puede manejar y como personalizar el proceso de arranque de FreeBSD. Esto incluye todo lo que sucede hasta que el kernel de FreeBSD comienza con la comprobación de dispositivos y se inicializa init(8). En caso de que usted desconozca cuando ocurre todo esto, esto sucede cuando el color del texto que aparece al iniciar el sistema, cambia de blanco a gris.

Una vez que concluya con la lectura de este capítulo, usted sabrá:

- Cuales son los componentes del mecanismo de arranque de FreeBSD, y como es que interactuan entre sí.
- Las opciones que puede manejar con los componentes del mecanismo de arranque de FreeBSD, para controlar el proceso de inicio del sistema.

Sólo x86

El presente capítulo describe únicamente el proceso de inicio, para sistemas FreeBSD que corren en plataformas Intel x86.

12.2. El problema que representa arrancar el sistema

El encender una computadora e iniciar el sistema operativo, trae consigo un dilema interesante. Por definición, la computadora no sabe hacer nada, hasta que el sistema operativo ha sido cargado. Esto incluye la ejecución de programas desde el disco duro. Así

que este es el dilema; si la computadora no sabe hacer nada hasta que se cargue el sistema operativo, y el sistema operativo es un conjunto de programas que residen en el disco duro, ?Cómo es que arranca el sistema operativo?

Este problema se asemeja a un problema del libro *Las Aventuras del Barón Munchausen*. Donde un personaje ha caido parcialmente en un hoyo, y ha podido salir al tomarse de las cintas de sus botas y jalarse hacia fuera. En los años mozos de la computación, el término utilizado para hablar sobre el proceso de carga del sistema operativo era *mecanismo de arranque* (bootstrap), que por efectos de simplificación ahora conocemos como "arranque" (booting).

En equipos con arquitectura x86, el Sistema Básico de Entrada/Salida (BIOS) es el responsable de cargar el sistema operativo. Para hacer esto, el BIOS busca en el disco duro el Registro Maestro de Arranque (RMA) (N de T. Conocido como MBR—Master Boot Record), el cual debe localizarse en un lugar específico del disco. El BIOS cuenta con suficiente información, para cargar y ejecutar el RMA, y asumir que el RMA puede encargarse del resto de las tareas necesarias en la carga del sistema operativo.

Si usted sólo cuenta con un sistema operativo instalado en su disco duro, el RMA estándar será suficiente. Este RMA buscará la primer partición del disco que pueda arrancar, y posteriormente ejecutará el código restante de dicha partición, para efecto de completar la carga del sistema operativo.

Si usted cuenta con varios sistemas operativos instalados en su disco, entonces puede hacer uso de un RMA diferente, uno que despliegue una lista de los diferentes sistemas operativos, y le permita escoger cual de ellos desea que se cargue. FreeBSD cuenta con un RMA de este tipo que puede ser instalado, así como otros distribuidores de sistemas operativos cuentan con RMAs alternativos.

En el caso de FreeBSD, el resto del mecanismo de arranque, está dividido en tres etapas. La primer etapa es ejecutada por al RMA, que sabe lo suficiente como para poner a la computadora en un estado específico y ejecutar la segunda etapa. La segunda etapa puede hacer un poco más que esto, antes de ejecutar la tercer etapa. La tercer etapa finaliza el trabajo de carga del sistema operativo. El trabajo es dividido en tres etapas, debido a las limitantes que tiene una PC, en cuanto al tamaño de los programas a ejecutar, durante las etapas uno y dos. El encadenar estas tareas, le permiten a FreeBSD contar con un arrancador más flexible.

Posteriormente el kernel es inicializado y comienza con la comprobación de dispositivos, y prepararlos para su uso. Una vez que el proceso de arranque del kernel ha finalizado, el kernel transfiere el control al proceso de usuario init(8), quien se asegura de que los discos se encuentren en buen estado para su uso. Posteriormente init(8) inicia la configuración fuente a nivel de usuario, que monta los sistemas de ficheros, configura las tarjetas de red para que pueden comunicarse en la red, y comunmente inicia todos los procesos que normalmente son ejecutados en un sistema FreeBSD al arrancar el mismo.

12.3. El RMA y las etapas de arranque uno, dos y tres

12.3.1. RMA, `/boot/boot0`

El RMA de FreeBSD, se localiza en `/boot/boot0` . Este es una *copia* del RMA, ya que el RMA real debe localizarse en una parte especial del disco duro, fuera de la área de FreeBSD.

El fichero `boot0` es muy simple, dado que el programa en el RMA sólo puede ser de 512 bytes. Si usted ha instalado el RMA de FreeBSD y ha instalado varios sistemas operativos en sus discos duros, entonces al momento de arrancar el sistema, visualizará una pantalla similar a la siguiente.

Ejemplo 12.1. Pantalla **`boot0`**

```
F1 DOS
F2 FreeBSD
F3 Linux
F4 ??
F5 Drive 1

Default: F2
```

Es sabido que otros sistemas, en particular Windows 95, sobreescriben el RMA con el suyo. Si esto le sucede, o bien desea reemplazar su RMA actual con el RMA de FreeBSD, entonces puede utilizar las siguientes órdenes.

```
# fdisk -B -b /boot/boot0 dispositivo
```

Donde *`dispositivo`* es aquel, desde el cual usted pretende arrancar el sistema, tal como `ad0` para el disco conectado al primer IDE, `ad2` para el disco maestro conectado al IDE secundario, `da0` para el primer disco SCSI, y así sucesivamente.

Por otro lado, si usted es un usuario de Linux, y prefiere que la aplicación LILO controle el proceso de arranque, puede editar el fichero `/etc/lilo.conf` para incluir a FreeBSD, o bien seleccionar la opción `Leave The Master Boot Record Untouched` durante el proceso de instalación. Si ha instalado el gestor de arranque de FreeBSD, puede arrancar Linux y modificar el fichero de configuración de LILO, `/etc/lilo.conf` , añadiendo la siguiente opción:

```
other=/dev/hdXY
table=/dev/hdb
loader=/boot/chain.b
label=FreeBSD
```

lo que permitirá el arranque de FreeBSD y Linux, por medio de LILO. En nuestro ejemplo hemos utilizado *XY* para especificar el disco utilizado y su partición. Si usted utiliza un sistema SCSI, deberá cambiar */dev/hdXY* por algo similar a */dev/sdXY* , que nuevamente utiliza la sintáxis *XY*. La opción `loader=/boot/chain.b` puede omitirse si usted cuenta con ambos sistemas operativos en el mismo disco. Una vez que esto se ha completado, puede ejecutar `/sbin/lilo -v` para que se actualicen los cambios en el sistema, lo cual deberá verificarse con los mensajes que aparezcan en pantalla.

Nota

N de T: La opción mencionada como; `Leave The Master Boot Record Untouched`, se mostrará "tal cual" aparece en este documento, una vez que ha terminado la fase de partición del disco duro.

12.3.2. Etapa uno, /boot/boot1, y etapa dos, /boot/boot2

Conceptualmente las estapas uno y dos, son parte del mismo programa, en la misma área del disco. Por cuestiones de espacio se han dividido en dos, pero estas siempre se instalaran de manera conjunta.

Estas son localizadas en el sector de arranque, de la partición de arranque, que es donde boot0, o cualquier otro programa del RMA espera encontrar el programa que dará continuación al proceso de arranque. Los ficheros localizados bajo el directorio `/boot` son copias de los ficheros reales, que se localizan fuera del sistema de ficheros de FreeBSD.

El fichero `boot1` es muy simple, en virtud de que sólo puede tener un tamaño de 512 bytes, y conocer simplemente del etiquetador de discos de FreeBSD (*disklabel*), el cual almacena la información de las particiones, para efecto de localizar y ejecutar `boot2`.

El fichero `boot2` es un poco más sofisticado, y comprende lo suficiente del sistema de ficheros de FreeBSD como para localizar ficheros en el, y puede proveer una interfaz simple, para seleccionar el kernel o cargador que deberá ejecutarse.

En virtud de que el cargador (loader) es mucho más sofisticado, y provee una configuración de arranque más sencilla de utilizar, `boot2` la ejecuta normalmente, una vez que ha terminado de solicitar el arranque del kernel directamente.

Ejemplo 12.2. Pantalla de **boot2**

```
>> FreeBSD/i386 BOOT
Default: 0:ad(0,a)/kernel
```

```
boot:
```

Si alguna vez requiere reemplazar los ficheros `boot1` y `boot2` instalados, utilice disklabel(8).

```
# disklabel -B partición
```

Donde *`partición`* es el disco y partición del cual pretende arrancar el sistema, tal como `ad0s1` para la primer partición del disco Mastro—Primario.

Modo peligrosamente dedicado

Si sólo utiliza el nombre del disco, tal como `ad0`, al usar disklabel(8) creará un disco peligrosamente dedicado, sin partición alguna. Seguramente esto no es lo que desea hacer, así que asegurese dos veces antes de presionar la tecla Return cuando utilice disklabel(8).

12.3.3. Etapa tres, `/boot/loader` (cargador de arranque)

El cargador es la etapa final de las tres etapas del mecanismo de arranque, y esta localizado en el sistema de ficheros, normalmente como `/boot/loader` .

El cargador pretende ser un metodo amistoso de configuración, utilizando una serie de órdenes integradas de fácil uso, respaldado por un intérprete más poderoso, con una serie de órdenes de mayor complejidad.

12.3.3.1. Flujo del programa cargador

Durante la inicialización del sistema, el cargador hará una comprobación en busca de una consola y discos, y sabra de que disco se encuentra arrancando. Establecerá las variables necesarias, y posteriormente es iniciado el intérprete donde se pueden introducir órdenes desde un “script” o de manera interactiva.

Posteriormente el cargador leerá el fichero `/boot/loader.rc` , que por default lee la información del fichero `/boot/defaults/loader.conf` que a su vez, establece las variables correspondientes y verifica en el fichero `/boot/loader.conf` los cambios locales que se hayan hecho, para establecer valores de las variables modificadas. Una vez llevado a cabo esto, `loader.rc` actua sobre estas variables, cargando cualquier módulo y kernel seleccionado.

Finalmente, y por default, el cargador hace una pausa contando 10 segundos y en espera de que al presionar una tecla se interrumpa el proceso, de no ser así, procederá a arrancar el kernel. En el caso de que al hacer esta pausa, se haya presionado una tecla (diferente

de Return), el proceso será interrumpido y se nos mostrará un "prompt", que entiende el conjunto de órdenes de fácil-uso, y donde el usuario puede ajustar ciertas variables, cargar y descargar todos los módulos, y también arrancar o reiniciar el sistema.

12.3.3.2. Órdenes internas del cargador

A continuación se presentan las órdenes más comunes del cargador. Para ver una descripción detallada de los mismos, por favor consulte la página de manual de loader(8)

autoboot *segundos*
: Procede a iniciar el arranque del sistema, si es que no es interrumpido el periodo dado, en segundos. Despliega una cuenta regresiva, y el tiempo dado es de 10 segundos.

boot [-opciones] [nombre_del_kernel]
: Procede a iniciar el kernel de manera inmediata, con las opciones dadas, si es que fuera el caso y el kernel especificado, si es que se especifica alguno.

boot-conf
: Rehace la configuración automática de modulos en función a las variables definidas, como sucede al arrancar. Esta opción sólo tiene sentido utilizarla, si en primer lugar hemos usado `unload`, y hemos modificado alguna variable, siendo lo más común el `kernel`.

help [tema]
: Muestra la ayuda de un tema específico, que lee del fichero `/boot/loader.help` . Si el tema que se indica es `index`, entonces se mostrará una lista de todos los temas disponibles.

include *fichero* ...
: Procesa el fichero que se ha especificado. El fichero se lee e interpreta línea por línea. Cualquier error detendrá inmediatamente a include.

load [-t tipo] *fichero*
: Carga el kernel, modulo del kernel, o el fichero del tipo dado, en base al fichero especificado. Cualquier argumento que se añada, será pasado al fichero.

ls [-l] [ruta]
: Despliega un listado de todos los ficheros que se localizan en la ruta especificada, o en el directorio raíz, si es que no se le especifica ruta alguna. Si se utiliza la opción `-l`, también se mostrara el tamaño de los ficheros.

lsdev [-v]
: Nos muestra una lista de todos los dispositivos desde los cuales puede ser posible cargar modulos. Si se incluye la opción `-v`, el listado que se obtiene cuenta con más detalle.

lsmod [-v]
: Despliega los módulos cargados. Si se utiliza la opción `-v`, se mostraran más detalles.

more *fichero*
: Despliega el contenido del fichero especificado, haciendo una pausa a cada numero determinado de `LINEAS` mostradas.

reboot
: Reinicia el sistema de forma inmediata.

set *variable*, set *variable* =*valor*
: Especifica los valores de las variables de entorno del cargador.

unload
: Remueve todos los módulos cargados.

12.3.3.3. Ejemplos de uso del cargador

He aqui unos ejemplos prácticos sobre el uso correcto del cargador.

- Para arrancar simplemente su kernel usual, pero en modo mono-usuario, deberá hacer lo siguiente:

```
boot -s
```

- Para descargar su kernel usual y sus módulos correspondientes, y posteriormente cargar su kernel anterior (o cualquier otro):

```
unload
load kernel.old
```

  Puede utilizar `kernel.GENERIC` para referirse al kernel generico actual que viene con la instalación, o bien puede utilizar `kernel.old` para hacer referencia al kernel anterior (por ejemplo, cuando ha actualizado su sistema o ha recompilado su propio kernel).

Nota

Utilice lo siguiente para cargar sus módulos actuales con otro kernel:

```
unload
set kernel="kernel.old"
boot-conf
```

- Para cargar un escrito de configuración (script que de forma automática hará todo lo que normalmente hace usted de forma manual al momento de ejecutarse el configurador de arranque):

```
load -t escrito_de_configuración /boot/kernel.conf
```

12.4. Interacción con el kernel durante el arranque

Una vez que el kernel ha sido iniciado, ya sea por el cargador (que es lo común) o bien por boot2 (sobrepasando el cargador), examinará las opciones de arranque, en busca de cambios, y ajustar su comportamiento en caso de ser necesario.

12.4.1. Opciones de arranque del kernel

He aqui las opciones de arranque más comunes:

`-a`
: durante la inicialización del kernel, pregunta por los dispositivos a utilizar, para montar el sistema de ficheros raíz.

`-C`
: arranque desde una unidad de CDROM.

`-c`
: ejecuta UserConfig, la utilidad de configuración de arranque del kernel.

`-s`
: arranca el sistem en modo mono-usuario.

`-v`
: imprime mensajes informativos durante el arranque del kernel

Nota

Existen otras opciones de arranque, por favor vea la página de ayuda boot(8) para más información al respecto.

12.5. Device Hints

Escrito por Tom Rhodes.

Pendiente de Traducción

12.6. Init: inicialización del proceso de control

Ya que el kernel ha finalizado de arrancar, pasará el control a un proceso de usuario llamado `init`, el cual se localiza en `/sbin/init` , o bien en la ruta especificada por la variable de entorno `init_path` del cargador.

12.6.1. Secuencia automática de reinicio

La secuencia automática de reinicio se asegura de que los sistemas de ficheros disponibles en el sistema sean consistentes. Si no lo son, y el programa `fsck` no puede arreglar estas inconsistencias, `init` envia el sistema a modo monousuario, de tal forma que el administrador pueda ingresar en él y arreglar los problemas directamente.

12.6.2. Modo monousuario

A este modo se puede llegar por medio de la secuencia automática de reinicio, o por medio de la opción `-s` en el arranque de usuario o al establecer la variable `boot_single` en el cargador.

También desde el modo multi-usuario se puede acceder, al utilizar `shutdown` sin la opción de reinicio (`-r`) o la de apagado (`-h`) del sistema.

Si la consola del sistema esta configurada de modo `inseguro` en el fichero `/etc/ttys` , entonces el sistema solicitará la contraseña del `superusuario` (root), antes de ingresar al sistema en modo mono-usuario.

Ejemplo 12.3. Una consola insegura en /etc/ttys

```
# name  getty                           type    status
  comments
#
# If console is marked "insecure", then init will ask for the ↺
root password
# when going to single-user mode.
console none                            unknown off insecure
```

Nota

Una consola `insegura` significa que usted considera como insegura físicamente su consola, por lo que desea asegurarse de que sólo

quien conoce la contraseña del `superusuario` puede ingresar al sistema en modo mono-usuario, y no que desea ejecutar la consola inseguramente. Esto es, si desea contar con seguridad escoja la opción `insecure`, y no `secure`.

12.6.3. Modo multiusuario

En el caso de que `init` encuentre en buen estado al sistema de ficheros, o una vez que el usuario ha terminado del modo mono-usuario, el sistema entrará al modo multi-usuario, en donde comienzan los ficheros de configuración-fuente del sistema.

12.6.3.1. Configuración-Fuente(rc)

La configuración fuente lee la configuración por default del fichero `/etc/defaults/rc.conf` , y detalles específicos del sistema desde el fichero `/etc/rc.conf` , y posteriormente procede a montar los sistemas de ficheros del sistema, descritos en `/etc/fstab`, iniciar servicios de red, así como varios demonios del sistema, para finalmente ejecutar los escritos (scripts) de configuración instalados por paquetes, localmente.

La página de ayuda rc(8) es una buena referencia para conocer más de este tipo de ficheros, así como examinar los mismos ficheros.

12.7. Secuencia de apagado

Una vez que el apagado sea controlado, por medio de `shutdown`, `init` ejecutará el escrito `/etc/rc.shutdown` , para posteriormente enviar a todos los procesos la señal `TERM`, y subsecuentemente la señal `KILL` a cualquiera que no haya terminado en tiempo.

Capítulo 13. Usuarios y administración básica de cuentas

Contributed by Neil Blakey-Milner.

13.1. Sinopsis

FreeBSD permite que varios usuarios usen el mismo ordenador. Obviamente, sólo uno de estos usuarios puede sentarse frente al monitor y al teclado en un momento dado [1] , pero cualquier número de usuarios puede entrar por la red para hacer su trabajo. Para usar el sistema cada usuario ha de tener una cuenta.

Después de leer este capítulo sabrás:

- Cuáles son las diferencias entre las distintas cuentas de usuario en sistemas FreeBSD.
- Cómo añadir cuentas.
- Cómo eliminar cuentas.
- Cómo cambiar detalles de las cuentas, como el nombre completo del usuario, o su shell preferida.
- Cómo establecer límites por cuenta, para controlar los recursos como memoria o tiempo de CPU que las cuentas y grupos de cuentas pueden emplear.
- Cómo usar grupos para facilitar la administración de cuentas.

Antes de leer este capítulo deberías:

- Entender los conceptos básicos de Unix y FreeBSD (Capítulo 3, *Conceptos básicos de Unix*).

13.2. Introducción

Todos los accesos al sistema se consiguen vía cuentas, y todos los procesos son ejecutados por usuarios, por ello la administración de usuarios y cuentas es de una gran importancia en sistemas FreeBSD.

[1] Bueno, a menos que conectes múltiples terminales, pero dejaremos ese tema para el Capítulo 24, *Comunicaciones serie*.

Cada cuenta en un sistema FreeBSD tiene cierta información asociada que la identifica.

Nombre de usuario
: El nombre de usuario como se le entraría al prompt `login:`. Los nombres de usuario han de ser únicos en la computadora; no puede haber dos usuarios con el mismo nombre de usuario. Existen algunas reglas para la creación de nombres de usuario válidos documentadas en passwd(5); típicamente se usarían nombres de usuario de a lo sumo ocho caracteres, todos ellos en minúscula.

Contraseña
: Cada cuenta tiene una contraseña asociada. La contraseña puede ser vacía, en cuyo caso no se requerirá ninguna para acceder al sistema. Esto normalmente es una mala idea; cada cuenta debería tener una constraseña no vacía.

Identificador de usuario (UID)
: El UID es un numero entre 0 y 65536 que sirve para identificar unívocamente al usuario en el sistema. Internamente, FreeBSD usa el UID para identificar usuarios y cualquier comando de FreeBSD que permita especificar un nombre de usuario convertirá éste al UID antes de trabajar con él. Esto significa que puedes tener varias cuentas con nombres de usuario distintos pero con el mismo UID. En lo que a FreeBSD respecta, tales cuentas son un solo usuario. Es improbable que alguna vez tengas que hacer algo así.

Identificador de grupo (GID)
: El GID es un número entre 0 y 65536 que sirve para identificar unívocamente el grupo principal al cual pertenece un usuario. Los grupos son un mecanismo para controlar el acceso a recursos del sistema en base al GID, en vez del UID. Esto puede reducir significativamente el tamaño de algunos ficheros de configuración. Un usuario puede pertencer a más de un grupo.

Clase de login
: Las clases de login son una extensión al mecanismo de grupos que ofrecen una mayor flexibilidad a la hora de adaptar el sistema a distintos usuarios.

Tiempo de cambio de contraseña
: Por defecto FreeBSD no obliga a los usuarios a cambiar su contraseña periódicamente. Se puede requerir esto a determinados usuarios, haciendo que algunos o todos deban cambiar sus contraseñas al cabo de cierto periodo de tiempo.

Tiempo de expiración de cuentas
: Por defecto las cuentas en FreeBSD no expiran. Si estás creando cuentas que sabes que van a tener un tiempo limitado de vida, por ejemplo, las cuentas de los estudiantes de una escuela, entonces puedes especificar cuándo expiran. Una vez vencido su tiempo de expiración una cuenta no puede ser usada para entrar en el sistema, si bien sus directorios y archivos serán conservados.

Nombre completo de usuario
: El nombre de usuario identifica unívocamente a una cuenta para FreeBSD, pero no refleja su verdadero nombre necesariamente. Esta información puede ser asociada a la cuenta.

Directorio home
: El directorio home es el camino completo de un directorio en el sistema en el que el usuario se hallará cuando entre. Una convención usual consiste en poner todos los directorios home en `/home/nombre_de_usuario` o en `/usr/home/nombre_de_usuario`. Los usuarios guardarían sus archivos personales en sus directorios home, y en cualquier directorio que creasen allí.

Shell de usuario
: La shell provee el entorno por defecto mediante el cual los usuarios interactúan con el sistema. Existen varios tipos de shell y los usuarios experimentados tendrán sus propias preferencias, que pueden expresarse en la configuración de su cuenta.

Existen principalmente tres tipos de cuentas; la cuenta de superusuario, las cuentas de usuarios del sistema, y las de usuarios. La cuenta de superusuario, normalmente llamada `root`, se usa para administrar el sistema sin limitaciones en los privilegios. Los usuarios del sistema utilizan servicios del mismo. Finalmente, las cuentas de usuarios son usadas por gente real, aquellos que entran, leen correo, etcétera.

13.3. La cuenta superusuario

La cuenta superusuario, normalmente llamada `root`, viene preconfigurada para facilitar la administración del sistema, y no debería ser utilizada para tareas cotidianas como enviar o recibir correo, exploración general del sistema, o programación.

Esto es así porque el superusuario, a diferencia de las cuentas de usuario, puede operar sin límites, y un mal uso de la cuenta de superusuario puede conllevar desastres espectaculares. Las cuentas de usuario no pueden destruir el sistema por un error, por ello es generalmente mejor utilizar cuentas de usuario normales cuando sea posible, a no ser que especialmente necesites privilegios extra.

Deberías comprobar siempre un par o tres de veces los comandos que ejecutas como superusuario, ya que un espacio de más o un carácter omitido pueden significar una pérdida de datos irreparable.

Así pues, lo primero que deberías hacer después de leer este capítulo es crear una cuenta sin privilegios de uso general para ti si aún no la tienes. Esto aplica tanto si trabajas en una máquina con varios usuarios como si trabajas en una máquina con un solo usuario. Más adelante, en este mismo capítulo, explicamos cómo crear cuentas adicionales, y cómo cambiar de usuario normal a superusuario.

13.4. Cuentas de sistema

Los usuarios de sistema son aquéllos que corren servicios como DNS, correo, servidores web, etc. Esto es así por seguridad; si todos los servicios corrieran como superusuario podrían actuar sin ninguna restricción.

Algunos ejemplos de usuarios de sistema son `daemon`, `operator`, `bind` (para el DNS), y `news`. Con frecuencia, los administradores de sistemas crean el usuario `httpd` para que ejecute los servidores web que instalan.

`nobody` es el usuario de sistema sin privelegios genérico. No obstante, es importante tener en cuenta que cuantos más servicios use `nobody`, más ficheros y procesos estarán asociados con dicho usuario, y en consecuencia más privilegiado será.

13.5. Cuentas de usuario

Las cuentas de usuario constituyen la principal vía de acceso al sistema para la gente real. Estas cuentas aíslan al usuario del entorno, impidiendo que pueda dañar al sistema o a otros usuarios, y permitiendo a su vez que pueda personalizar su entorno sin que esto afecte a otros.

Cada persona que acceda a tu sistema debería tener una sola cuenta de usuario. Esto te permite averiguar quién está haciendo qué, evita que interfieran las configuraciones de distintos usuarios, que unos puedan leer el correo de otros, etcétera.

Cada usuario puede configurar su entorno para acomodarlo al uso que hace del sistema, utilizando shells, editores, atajos de teclado e idioma alternativos.

13.6. Modificación de cuentas

Existe una variedad de comandos disponible en el entorno Unix para modificar cuentas de usuario. Los comandos más comunes se hallan resumidos a continuación, seguidos de ejemplos más detallados de su uso.

Comando	Resumen
`adduser`	La aplicación de línea de comandos recomendada para añadir nuevos usuarios.
`rmuser`	La aplicación de línea de comandos recomendada para eliminar usuarios.

Comando	Resumen
`chpass`	Una herramienta flexible para modificar la base de datos de usuarios.
`passwd`	Una herramienta de línea de comandos simple para cambiar contraseñas de usuario.
`pw`	Una herramienta potente y flexible para modificar cualquier aspecto de las cuentas de usuario.

13.6.1. adduser

`adduser` es un programa simple para añadir usuarios. Crea entradas en los archivos de sistema `passwd` y `group`. También crea un directorio home para el nuevo usuario, copia allí ficheros de configuración por defecto ("dotfiles") de `/usr/share/skel` , y opcionalmente puede enviar al usuario un mensaje de bienvenida.

Para crear el fichero inicial de configuración usa `adduser -s -config_create .`[2] A continuación configuramos valores por defecto para adduser y creamos nuestra primera cuenta de usuario, dado que utilizar `root` para uso normal del sistema es pernicioso y peligroso.

Ejemplo 13.1. Configuración de adduser

```
# adduser -v
Use option ``-silent'' if you don't want to see all warnings ↺
and questions.
Check /etc/shells
Check /etc/master.passwd
Check /etc/group
Enter your default shell: csh date no sh tcsh zsh [sh]: zsh
Your default shell is: zsh -> /usr/local/bin/zsh
Enter your default HOME partition: [/home]:
Copy dotfiles from: /usr/share/skel no [/usr/share/skel]:
Send message from file: /etc/adduser.message no
[/etc/adduser.message]: no
Do not send message
Use passwords (y/n) [y]: y

Write your changes to /etc/adduser.conf? (y/n) [n]: y

Ok, let's go.
```

[2] La `-s` hace que `adduser` sea silencioso por defecto. Más tarde usamos `-v` cuando queremos cambiar valores por defecto.

```
Don't worry about mistakes. I will give you the chance later ↺
to correct any input.
Enter username [a-z0-9_-]: jru
Enter full name []: J. Random User
Enter shell csh date no sh tcsh zsh [zsh]:
Enter home directory (full path) [/home/jru]:
Uid [1001]:
Enter login class: default []:
Login group jru [jru]:
Login group is ``jru''. Invite jru into other groups: guest no
[no]: wheel
Enter password []:
Enter password again []:

Name:     jru
Password: ****
Fullname: J. Random User
Uid:      1001
Gid:      1001 (jru)
Class:
Groups:   jru wheel
HOME:     /home/jru
Shell:    /usr/local/bin/zsh
OK? (y/n) [y]: y
Added user ``jru''
Copy files from /usr/share/skel to /home/jru
Add another user? (y/n) [y]: n
Goodbye!
#
```

En resumen, cambiamos la shell por defecto a zsh (una shell alternativa incluida en la colección de ports), y deshabilitamos el envío de un mensaje de bienvenida a nuevos usuarios. Luego grabamos la configuración, creamos una cuenta para `jru`, y nos aseguramos de que `jru` esté en el grupo `wheel` (de modo que puede asumir el papel de `root` vía el comando `su`).

Nota

La contraseña que escribes no se muestra, tampoco se muestran asteriscos. Asegúrate de no entrar dos veces una contraseña equivocada.

Nota

Usa `adduser` sin argumentos en adelante, no necesitarás cambiar las opciones por defecto. Si el programa te pide modificarlas sal y prueba con la opción `-s`.

13.6.2. rmuser

Puedes usar `rmuser` para eliminar completamente del sistema a un usuario. `rmuser` efectúa los siguientes pasos:

1. Elimina la entrada del usuario en crontab(1) (si tiene alguna).
2. Elimina las tareas at(1) pertenecientes al usuario.
3. Mata todos los procesos pertenecientes al usuario.
4. Elimina al usuario del fichero local de contraseñas del sistema.
5. Borra el directorio home del usuario (si le pertenece).
6. Elimina los archivos de correo entrante del usuario de `/var/mail` .
7. Borra todos los ficheros del usuario de áreas en las que se guardan archivos temporales como `/tmp`.
8. Finalmente, elimina el nombre de usuario de todos aquellos grupos a los que pertenece en `/etc/group` .

Nota

Si un grupo queda vacío y el nombre del grupo coincide con el del usuario, el grupo es eliminado; esto complementa la creación de grupos por usuario de adduser(8).

`rmuser` no puede ser usado para eliminar cuentas de superusuario, dado que algo así es casi siempre señal de masiva destrucción.

Por defecto existe un modo interactivo que intenta asegurar que uno sabe lo que hace.

Ejemplo 13.2. Eliminación interactiva de cuenta con rmuser

```
# rmuser jru
Matching password entry:
jru:*:1001:1001::0:0:J. Random User:/home/jru:/usr/local/bin/
tcsh
Is this the entry you wish to remove? y
Remove user's home directory (/home/jru)? y
Updating password file, updating databases, done.
Updating group file: trusted (removing group jru -- personal ↺
group is empty) done.
Removing user's incoming mail file /var/mail/jru: done.
Removing files belonging to jru from /tmp: done.
Removing files belonging to jru from /var/tmp: done.
Removing files belonging to jru from /var/tmp/vi.recover: done.
#
```

13.6.3. chpass

`chpass` cambia información de los usuarios en base de datos como contraseñas, shells y datos personales.

Los administradores del sistema, como el superusuario, son los únicos que pueden modificar la información y contraseñas de otros usuarios con `chpass`.

Cuando no le pasamos más opciones, salvo un nombre de usuario opcional, `chpass` muestra un editor con información de usuario. Cuando se sale del editor la base de datos de usuarios se actualiza con la nueva información.

Ejemplo 13.3. chpass interactivo ejecutado por el superusuario

```
#Changing user database information for jru.
Login: jru
Password: *
Uid [#]: 1000
Gid [# or name]: 1000
Change [month day year]:
```

```
Expire [month day year]:
Class:
Home directory: /home/jru
Shell: /usr/local/bin/tcsh
Full Name: J. Random User
Office Location:
Office Phone:
Home Phone:
Other information:
```

Un usuario normal puede modificar un pequeño subconjunto de esta información, y sólo para sí mismo.

Ejemplo 13.4. chpass interactivo ejecutado por un usuario normal

```
#Changing user database information for jru.
Shell: /usr/local/bin/tcsh
Full Name: J. Random User
Office Location:
Office Phone:
Home Phone:
Other information:
```

Nota

`chfn` y `chsh` son enlaces a `chpass`, como también lo son `ypchpass`, `ypchfn`, e `ypchsh`. El soporte para NIS es automático, por lo que no es necesario especificar el `yp` antes del comando. Si esto te resulta algo confuso no te preocupes, NIS será tratado en el Capítulo 29, *Networking avanzado*.

13.6.4. passwd

`passwd` es el comando que se usa normalmente para cambiar tu propia contraseña como usuario o, como superusuario, la de otros usuarios.

Nota

Los usuarios han de introducir su contraseña original antes de cambiarla para prevenir que gente no autorizada pueda hacerlo cuando no se encuentren en la consola.

Ejemplo 13.5. Cambio de tu contraseña

```
% passwd
Changing local password for jru.
Old password:
New password:
Retype new password:
passwd: updating the database...
passwd: done
```

Ejemplo 13.6. Cambio de la contraseña de otro usuario como superusuario

```
# passwd jru
Changing local password for jru.
New password:
Retype new password:
passwd: updating the database...
passwd: done
```

Nota

Como ocurre con `chpass`, `yppasswd` es un enlace a `passwd`, de manera que NIS funciona con ambos comandos.

13.6.5. pw

pw(8) es una utilidad de línea de comandos para crear, eliminar, modificar, y mostrar usuarios y grupos. Hace de interfaz a los archivos del sistema de usuarios y grupos. `pw` tiene un conjunto de opciones de línea de comandos bastante potente que lo hacen adecuado para su uso en scripts de shell, aunque los nuevos usuarios puede que lo encuentren algo más complicado que el resto de comandos que presentamos aquí.

13.7. Limitar a los usuarios

En un sistema multiusuario es probable que no confíes en que el sistema no vaya a ser dañado por ningún usuario.

Las cuotas de disco permiten al administrador decirle al sistema de ficheros qué cantidad de espacio de disco puede utilizar un usuario; además, ofrecen una manera rápida de comprobar el uso de disco de un usuario sin tener que calcularlo cada vez. Las cuotas se estudian en el capítulo de quotas.

El resto de límites de recursos incluyen cantidad de CPU, memoria, y otros recursos que el usuario puede utilizar.

Las clases de login se definen en `/etc/login.conf` . La semántica precisa está fuera del alcance de esta sección, pero se describe con detalle en la página de manual login.conf(5). Es suficiente decir que cada usuario es asignado a una clase de login (`default` por defecto), y que cada clase de login tiene un conjunto de capacidades asociado. Una capacidad de login es un par `nombre=valor` , donde *`nombre`* es un identificador conocido y *`valor`* una cadena de texto arbitraria que se procesa en función del nombre. Establecer clases y capacidades de login es bastante sencillo y también se describe en login.conf(5).

Los límites de recursos son diferentes de las capacidades de login en dos sentidos. En primer lugar, para cada límite existe un límite blando (actual) y uno duro. Un límite blando puede ser ajustado por el usuario o una aplicación, pero no puede ser más alto que el límite duro. Éste último puede ser disminuido por el usuario pero nunca aumentado. En segundo lugar, la mayoría de los límites de recursos aplican a un usuario concreto por proceso, no globalmente. Nótese, no obstante, que estas diferencias vienen impuestas por cómo se tratan los límites específicamente, no por la implementación del marco de capacidades de login (es decir, en realidad no constituyen un caso especial de capacidades de login).

Sin más, a continuación veremos los límites de recursos más comúnmente usados (el resto, junto con el resto de capacidades de login, puede encontrarse en login.conf(5)).

`coredumpsize`

> El tamaño de un fichero core generado por un programa está, por razones obvias, subordinado a otros límites sobre uso de disco (p. ej., `filesize`, o cuotas de disco). Aun y así, se usa frecuentemente como un método menos severo de controlar con-

sumo de espacio de disco, dado que los usuarios no generan ficheros core por ellos mismos, y a menudo no los borran, activar este límite puede evitar que agoten el espacio de disco de que disponen si algún programa grande (p. ej., emacs) deja de funcionar abruptamente.

`cputime`

Ésta es la máxima cantidad de tiempo de CPU que los procesos de un usuario pueden consumir.

Nota

Éste es un límite sobre el *tiempo* de CPU consumido, no el porcentaje de uso de CPU que se muestra en algunos campos de top(1) y ps(1). Un límite de ese tipo no es posible a día de hoy, y sería bastante inútil: un compilador —probablemente una tarea legítima— puede usar prácticamente el 100% de la CPU durante algún tiempo con facilidad.

`filesize`

Éste es el tamaño máximo que puede llegar a tener un fichero del usuario. A diferencia de las cuotas de disco, este límite se especifica para ficheros individuales, no para el conjunto de todos los archivos que posee.

`maxproc`

Éste es el máximo número de procesos que un usuario puede ejecutar a la vez, incluidos tanto los procesos en primer plano como los procesos en segundo plano. Por razones obvias, este límite no puede ser mayor que el límite de sistema especificado por `kern.maxproc sysctl`. Obsérvese también que si asignamos un valor demasiado bajo a este límite podemos mermar la productividad de un usuario: frecuentemente es útil entrar múltiples veces en el sistema o ejecutar pipelines. Algunas tareas, como compilar largos programas, lanzan múltiples procesos (p. ej., make(1), cc(1), y demás preprocesadores intermedios).

`memorylocked`

Ésta es la máxima cantidad de memoria que un proceso puede haber solicitado tener bloqueada en memoria principal (p. ej., ver mlock(2)). Algunos programas críticos para el sistema, como amd(8), se quedan bloqueados en la memoria principal de manera que en caso de ser llevados a swap no contribuyan a la basura del sistema si hay algún problema.

`memoryuse`

Ésta es la mayor cantidad de memoria que un proceso puede consumir en todo momento. Incluye tanto memoria normal como uso de swap. No se trata de un límite para restringir el consumo de memoria en general, pero es un buen comienzo.

`openfiles`

Ésta es la máxima cantidad de archivos que un proceso puede tener abiertos. En FreeBSD, los archivos se usan tambien para representar sockets y canales IPC; así, cuida de no poner este límite demasiado bajo. A nivel de sistema, el límite para esto lo define `kern.maxfiles sysctl`.

`sbsize`

Éste es el límite de cantidad de memoria de red, y por lo tanto mbufs, que un usuario puede consumir. Se originó como respuesta a un viejo ataque DoS que creaba muchos sockets, pero puede ser usado en general para limitar las comunicaciones por red.

`stacksize`

Éste es el tamaño máximo que puede alcanzar la pila de un proceso. Por sí solo no es suficiente para limitar la cantidad de memoria que puede usar un programa; en consecuencia, debería ser usado junto con otros límites.

Hay unas pocas cosas más a recordar cuando establecemos límites de recursos. A continuación vienen algunas recomendaciones, sugerencias, y comentarios varios.

- Los procesos que se ponen en marcha cuando arranca el sistema por `/etc/rc` están asignados a la clase de login `daemon`.
- Aunque el `/etc/login.conf` que viene con el sistema tiene valores razonables para la mayoría de los límites, sólo tú, el administrador, puedes saber lo que es apropiado para tu sistema.
- A los usuarios del X Window System (X11) probablemente se les debería conceder más recursos que al resto. X11 de por sí consume muchos recursos, pero además contribuye a que los usuarios ejecuten más programas simultáneamente.
- Recuerda que hay muchos límites que aplican a procesos individuales, no al usuario en general. Por ejemplo, poner `openfiles` a 50 significa que cada uno de los procesos que ejecute un usuario puede abrir a lo máximo 50 ficheros. Así, la cantidad de ficheros que un usuario puede abrir es el valor de `openfiles` multiplicado por el valor de `maxproc`. Esto también aplica al uso de memoria.

Para más información acerca de límites de recursos y clases y capacidades de login en general, consulta las páginas de manual relevantes: cap_mkdb(1), getrlimit(2), login.conf(5).

13.8. Personalizar a los usuarios

La localización es un entorno establecido por el administrador o el usuario para dar soporte a distintos lenguajes, juegos de caracteres, estándares sobre fechas y horas, etcétera. Éste tema se trata en el capítulo Localización.

13.9. Grupos

Un grupo es simplemente una lista de usuarios. Los grupos se identifican por su nombre de grupo y gid (ID de grupo). En FreeBSD (y en la mayoría de sistemas Unix), los dos factores que tiene en cuenta el núcleo para decidir si un proceso puede hacer algo es su ID de usuario y la lista de grupos a los que pertenece. A diferencia del ID de usuario, un proceso tiene una lista de grupos asociados. En ocasiones encontrarás menciones al "ID de grupo" de un usuario o de un proceso; la mayoría de las veces referirán simplemente al primero de los grupos de la lista.

La correspondencia entre nombres e IDs de grupo está en `/etc/group` . Se trata de un fichero de texto plano con cuatro campos separados por el signo dos puntos. El primer campo es el nombre de grupo, el segundo la contraseña encriptada, el tercero el ID de grupo, y el cuarto la lista de miembros separados por comas. Puede ser editado a mano sin peligro (¡suponiendo, por supuesto, que no se cometan errores de sintaxis!). Para una descripción más completa de la sintaxis, ver la página de manual group(5).

Si no quieres editar `/etc/group` manualmente, puedes usar el comando pw(8) para añadir y modificar grupos. Por ejemplo, para añadir un grupo llamado `teamtwo` y luego confirmar que existe puedes usar:

Ejemplo 13.7. Añadir un grupo usando pw(8)

```
# pw groupadd teamtwo
# pw groupshow teamtwo
teamtwo:*:1100:
```

El número `1100` en el ejemplo anterior es el ID de grupo del grupo `teamtwo`. Ahora mismo `teamtwo` no tiene miembros, y es por tanto bastante inútil. Cambiemos eso invitando a `jru` a formar parte del grupo `teamtwo`.

Ejemplo 13.8. Añadir a alguien a un grupo usando pw(8)

```
# pw groupmod teamtwo -M jru
# pw groupshow teamtwo
teamtwo:*:1100:jru
```

El argumento de la opción `-M` es una lista con los usuarios que son miembros del grupo separados por comas. Sabemos de secciones anteriores que el fichero de contraseñas también contiene un grupo para cada usuario. El usuario es automáticamente añadido a la lista de grupos por el sistema; no constará como miembro cuando usemos el comando `groupshow` con pw(8), pero sí cuando la información se consulte con id(1) u otra herramienta similar. En otras palabras, pw(8) sólo manipula el fichero `/etc/group` ; nunca tratará de leer datos adicionales de `/etc/passwd` .

Ejemplo 13.9. Determinar pertenencia a grupos con id(1)

```
% id jru
uid=1001(jru) gid=1001(jru) groups=1001(jru), 1100(teamtwo)
```

Como puedes ver, `jru` es miembro de los grupos `jru` y `teamtwo`.

Para más información acerca de pw(8), consulta su página de manual, y para más información acerca del formato de `/etc/group` , consulta la página de manual de group(5).

Capítulo 14. Seguridad

Gran parte del contenido de este capítulo procede de la página de manual de security(7), de Matthew Dillon.

14.1. Sinopsis

Este capítulo contiene una introducción básica a los conceptos de seguridad del sistema, unas cuantas normas básicas de uso y algunos avanzados del tema en FreeBSD. Muchos de los temas expuestos se aplican a la seguridad del sistema y de Internet en general. Internet ya no es aquél lugar "amistoso" en el que todo el mundo se comportaba como un buen ciudadano. Si quiere proteger sus datos, su propiedad intelectual, su tiempo y muchas más cosas de manos malintencionadas debe hacer que su sistema sea seguro.

FreeBSD proporciona un variado arsenal de utilidades y mecanismos para asegurar la integridad y la seguridad de su sistema y red.

Después de leer este capítulo:

- conocerá conceptos básicos de la seguridad relacionados con FreeBSD.
- Tendrá información sobre los diversos mecanismos de cifrado disponibles en FreeBSD, entre los cuales están DES y MD5.
- Sabrá cómo configurar la autentificación de contraseñas de un solo uso.
- Sabrá cómo configurar TCP Wrappers y usarlos con `inetd`.
- Sabrá cómo instalar KerberosIV en versiones de FreeBSD anteriores a 5.0.
- Sabrá cómo instalar Kerberos5 en versiones de FreeBSD posteriores a 5.0.
- Podrá configurar IPsec y crear una VPN entre máquinas FreeBSD/Windows®.
- Sabrá cómo configurar y utilizar OpenSSH, la implementación de SSH en FreeBSD.
- Sabrá en qué consisten las ACL del sistema de ficheros y cómo utilizarlas.
- Sabrá cómo usar Portaudit, con la que podrá auditar el software que instale desde la desde la colección de ports.
- Sabrá cómo sacar partido de los avisos de seguridad que publica FreeBSD.
- Podrá hacerse una idea clara de en qué consiste la contabilidad de procesos y de cómo activarla en FreeBSD.

Antes de leer este capítulo:

- Comprender conceptos básicos de FreeBSD e Internet.

En otras secciones de este manual se cubren aspectos adicionales sobre seguridad. Por ejemplo, MAC (controles de acceso obligatorio) se explica en el Capítulo 16, *Mandatory Access Control* y los cortafuegos en el Capítulo 26, *Cortafuegos*.

14.2. Introducción

La seguridad es un trabajo que que comienza y termina en el administrador de sistema. Aunque que los sistemas multiusuario BSD UNIX® posean una seguridad inherente, el trabajo de construir y mantener mecanismos de seguridad adicionales para que los usuarios sean aún más "honestos" es probablemente una de las mayores tareas de la administración de sistemas. Los sistemas son tan seguros como uno los haga, y no hay que olvidar que los problemas de seguridad compiten con la comodidad a la que tendemos los humanos. Los sistemas UNIX® son capaces de ejecutar una gran cantidad de procesos simultáneamente, muchos de los cuales son servidores, lo que significa que las entidades externas pueden conectarse y "hablar" con ellos. Del mismo modo que las minicomputadoras de ayer se convirtieron en los sistemas de escritorio de hoy en día, la seguridad se va convirtiendo en un problemas más y más acuciante.

La seguridad bien entendida se implementa en capas, a la manera de una "cebolla". Básicamente lo que se hace es crear la mayor cantidad posible de capas de seguridad, para más tarde monitorizar el sistema en busca de intrusos. No es conveniente exagerar la seguridad, ya que interferiría con la detección, y la detección es uno de los aspectos más importantes de cualquier mecanismo de seguridad. Por ejemplo, no tiene mucho sentido activar la bandera `schg` (consulte chflags(1)) en cada binario del sistema, ya que aunque protegería en cierto modo los binarios, haría que cualquier cambio que pudiera realizar un atacante una vez dentro del sistema fuera más difícil de detectar o incluso hacerlo del todo imposible.

La seguridad del sistema depende también de estar preparados para distintos tipos de ataque, incluyendo intentos de "tirar" la máquina o dejarla en un estado inutilizable, pero que no impliquen intentos de comprometer el usuario `root` Los problemas de seguridad pueden dividirse en diferentes categorías:

1. Ataques de denegación de servicio (DoS).
2. Comprometer cuentas de usuarios.
3. Comprometer root a través de servidores accesibles.
4. Comprometer root desde cuentas de usuario.
5. Creación de puertas traseras ("Backdoors").

Un ataque de denegación de servicio es una acción que priva al sistema de los recursos requeridos para su funcionamiento normal. Generalmente, los ataques DoS son mecanismos

de fuerza bruta que intentan "tumbar" el sistema o hacerlo inutilizable sobrecargando la capacidad de sus servidores o de la pila de red. Algunos ataques DoS intentan aprovechar errores en la pila de red para "tumbar" el sistema con un solo paquete. Estos últimos únicamente pueden solucionarse aplicando al kernel una actualización que subsane el error. Los ataques a servidores muchas veces pueden solucionarse configurando las opciones apropiadas para limitar la carga del sistema en condiciones adversas. Los ataques de fuerza bruta a redes son más complicados. Los ataques con paquetes enmascarados, por ejemplo, son casi imposibles de detener, a menos que desconecte el sistema de Internet. Puede ser que no "tiren" el sistema, pero saturarán la conexión a Internet.

Comprometer una cuenta de usuario es mucho más común que un ataque DoS. Muchos administradores de sistemas todavía ejecutan servidores estándar telnetd, rlogind, rshd y ftpd en sus máquinas. Estos servidores, por defecto no operan a través de conexiones cifradas. El resultado es que se si se tiene una base de usuarios de tamaño medio, tarde o temprando la contraseña de uno (o más) de sus usuarios será descubierta durante sus accesos al sistema desde ubicaciones remotas.(que es, por otra parte, la forma más común y más cómoda de acceder a un sistema). El administrador de sistemas atento analizará sus logs de acceso remoto en busca de direcciones origen spspechosas, incluso entre los accesos al sistema.

Se debe asumir *siempre* que, una vez que el atacante tiene acceso a una cuenta de usuario, el atacante puede comprometer la cuenta `root`. En realidad en un sistema bien mantenido y asegurado el acceso a una cuenta de usuario no necesariamente da al atacante acceso a `root`. Esta precisión es importante porque sin acceso a `root` el atacante difícilmente podrá esconder sus huellas; podrá, como mucho, hacer poco más que sembrar el caos en los ficheros del usuario o "tirar" la máquina. Comprometer cuentas de usuario es muy común porque los usuarios tienden a no tomar las precauciones que toma el administrador.

Los administradores de sistemas deben tener presente que existen muchas formas potenciales de comprometer la cuenta `root` de una máquina. El atacante puede conocer la contraseña de `root`, el atacante puede encontrar un error en un servidor que se ejecuta como root y ser capaz de comprometer `root` a través de una conexión de red a ese servidor; puede ser que el atacante sepa de la existencia de un error en un programa suid-root que le permita comprometer `root` una vez dentro de una cuenta de usuario. Si un atacante encuentra la manera de comprometer la cuenta `root` de una máquina puede que no necesite instalar una puerta trasera. Muchos de los agujeros `root` encontrados y cerrados hasta la fecha implican una cantidad considerable de trabajo para el atacante limpiando todo después del ataque, así que la mayoría de los atacantes instalan puertas traseras. Una puerta trasera facilita al atacante una forma sencilla de recuperar el acceso de `root` al sistema, pero también proporciona al administrador de sistemas inteligente una forma de detectar la intrusión. Si hace imposible a un atacante la instalación de una puerta trasera puede estar actuando en detrimento de su seguridad, porque no cerrará el agujero que el atacante encontró para accder al sistema la primera vez que lo hizo.

Las medidas de seguridad se implementan en un modelo multicapa (tipo "cebolla"), que puede categorizarse del siguiente modo:

1. Asegurar `root` y cuentas administrativas.
2. Asegurar los servidores que se ejecuten como `root` los binarios suid/sgid.
3. Asegurar cuentas de usuario.
4. Asegurar el fichero de contraseñas.
5. Asegurar el núcleo del kernel, los dispositivos en bruto y el sistema de ficheros.
6. Detección rápida de cambios hechos al sistema.
7. Paranoia.

La siguiente sección de este capítulo tratará los puntos de arriba con mayor profundidad.

14.3. Asegurar FreeBSD

Orden vs. protocolo

En este capítulo usaremos el texto en negrita para referirnos a una orden o aplicación, y una fuente en `cursiva` para referirnos a órdenes específicas. Usaremos un tipo normal para los protocolos. Esta diferencia tipográfica nos será útil por ejemplo con ssh, que es tanto un protocolo como una orden.

Las siguientes secciones cubren los métodos a seguir para asegurar su sistema FreeBSD que se mencionados en la sección anterior de este capítulo.

14.3.1. Asegurar la cuenta `root` y las cuentas administrativas

En primer lugar, no se moleste en asegurar las cuentas administrativas (o "staff") si no ha asegurado la cuenta `root`. La mayoría de los sistemas tienen una contraseña asignada para la cuenta `root`. Lo primero que se hace es asumir que la contraseña está *siempre* amenazada. Esto no significa que deba eliminar la contraseña. La contraseña es casi siempre necesaria para el acceso por consola a la máquina; significa que no se debe permitir el uso de la contraseña fuera de la consola o, mejor aún, mediante su(1). Por ejemplo, asegúrese de que sus ptys aparezcan como *inseguras* en el fichero `/etc/ttys` , con lo que hará que los accesos como `root` vía `telnet` o `rlogin` no sean posibles. Si utiliza otros tipos de login como sshd asegúrese de que los accesos al sistema como `root` estén también deshabilitados. Para ello edite su `/etc/ssh/sshd_config` y asegúrese de que `PermitRootLogin` esté puesto a `NO`. Estudie cada método de acceso: hay servicios como FTP que frecuentemente

son origen de grietas en la estructura del sistema. El acceso directo como usuario `root` sólamente debe permitirse a través de la consola.

Es evidente que, como administrador del sistema, debe usted tener la posibilidad de acceder a `root`, así que tendrá que abrir algunos agujeros, pero debe asegurarse de que estos agujeros necesiten contraseñas adicionales para verificar su correcto uso. Puede hacer que `root` sea accesible añadiendo cuentas administrativas al grupo `wheel` (en `/etc/group`). El personal que administra los sistemas que aparezcan en el grupo en el grupo `wheel` pueden hacer `su` a `root`. Nunca debe de proporcionar al personal administrativo el acceso nativo a `wheel` poniéndolos en el grupo `wheel` en su entrada de contraseña. Las cuentas administrativas deben colocarse en un grupo `staff`, y agregarse después al grupo `wheel` en `/etc/group` . Sólo aquellos administradores que realmente necesiten acceder a `root` deben pertenecer al grupo `wheel`. También es posible, mediante un método de autentificación como Kerberos, usar el fichero `.k5login` en la cuenta `root` para permitir un ksu(1) a `root` sin tener que colocar a nadie en el grupo `wheel`. Puede ser una mejor solución, ya que el mecanismo `wheel` aún permite a un atacante comprometer `root` si el intruso ha conseguido el fichero de contraseñas y puede comprometer una cuenta de administración. Recurrir al mecanismo `wheel` es mejor que no tener nada, pero no es necesariamente la opción más segura.

Una manera indirecta de asegurar las cuentas de staff y el acceso a `root` es utilizar un método de acceso alternativo: es lo que se conoce como "estrellar" las contraseñas cifradas de las cuentas administrativas. Use vipw(8) para reemplazar cada contraseña cifrada por un sólo caracter asterisco ("*"). Esto actualizará `/etc/master.passwd` y la base de datos de usuario/contraseña y deshabilitará los accesos al sistema validados mediante contraseñas.

Veamos una cuenta administrativa típica:

```
foobar:R9DT/Fa1/LV9U:1000:1000::0:0:Foo Bar:/home/foobar:/usr/local/
bin/tcsh
```

y cómo debería quedar:

```
foobar:*:1000:1000::0:0:Foo Bar:/home/foobar:/usr/local/bin/tcsh
```

Este cambio evitará que se efectúen logins normales, ya que la contraseña cifrada nunca se corresponderá con "*". Hecho esto, el personal de administración tendrá que usar otro mecanismo de validación como kerberos(1) o ssh(1) que use un par de llave pública/privada. Si decide usar algo como Kerberos tendrá que asegurar la máquina que ejecuta los servidores Kerberos y su estación de trabajo. Si usa un par de llave pública/privada con ssh, debe asegurar la máquina *desde* desde la que se hace el login (normalmente nuestra estación de trabajo). Puede añadir una capa adicional de protección al par de llaves protegiéndolas con contraseña al crearlo con ssh-keygen(1). El "estrellado" de las contraseñas administrativas también garantiza que dicho personal sólo pueda entrar a través de métodos de acceso que haya usted configurado. Así obligará al personal administrativo a usar conexiones seguras, cifradas, en todas sus sesiones, lo que cierra un importante

agujero de seguridad al que recurren muchos intrusos: usar un sniffer (olfateador) de red desde una máquina que le permita hacer tal cosa.

Los mecanismos de seguridad más indirectos también asumen que está validando su identidad desde un servidor más restrictivo un servidor menos restrictivo. Por ejemplo, si su máquina principal ejecuta toda clase de servidores su estación de trabajo no debe ejecutar ninguno. Para que su estación de trabajo sea razonablemente segura debe ejecutar los mínimos servidores posibles, si es posible ninguno, y debe usar un salvapantallas protegido por contraseña. Es evidente que un atancante con acceso físico al sistema puede romper cualquier barrera de seguridad que se disponga. Es un problema a tener en cuenta, pero la mayoría de las intrusiones tienen lugar de forma remota, a través de la red, por parte de gente que no tiene acceso físico a su estación de trabajo ni a sus servidores.

Usar Kerberos le ofrece también el poder de deshabilitar o cambiar la contraseña para una cuenta administrativa en un lugar, y que tenga un efecto inmediato en todas las máquinas en las cuales ese administrador pueda tener una cuenta. Si una de esas cuentas se ve comprometida la posibilidad para cambiar instantáneamente su contraseña en todas las máquinas no debe ser desestimada. Con contraseñas distintas, el cambio de una contraseña en N máquinas puede ser un problema. También puede imponer restricciones de re-contraseñas con Kerberos: no sólo se puede hacer un ticket de Kerberos que expire después de un tiempo, sino que el sistema Kerberos puede requerir al usuario que escoja una nueva contraseña después de cierto tiempo (digamos una vez al mes).

14.3.2. Asegurar servidores que se ejecutan como `root` y binarios SUID/SGID

Un administrador de sistemas prudente sólo ejecutará los servidores que necesita, ni uno más ni uno menos. Dese cuenta de que los servidores ajenos son los más propensos a contener errores. Por ejemplo, ejecutando una versión desfasada de imapd o popper es como dar una entrada universal de `root` al mundo entero. Nunca ejecute un servidor que no haya revisado cuidadosamente. Muchos servidores no necesitan ejecutarse como `root`. Por ejemplo, los dæmons ntalk, comsat y finger pueden ejecutarse en una *caja de arena (sandbox)* especial de usuario. Una caja de arena no es perfecta, a menos que pase por muchos problemas, pero la aproximación de cebolla a la seguridad prevalece aún y todo: Si alguien es capaz de penetrar a través de un servidor ejecutándose en una caja de arena, todavía tendrá que salir de la caja de arena. Cuantas más capas tenga que romper el atacante menor será la posibilidad de éxito que tenga. Se han encontrado vías de entrada a `root` en virtualmente todos los servidores que se haya ejecutado como `root`, incluyendo servidores básicos del sistema. Si está tiene una máquina a través de la cual la gente sólo entra por sshd, y nunca entra por telnetd, rshd, o rlogind *apague esos servicios*.

FreeBSD ejecuta por defecto ntalkd, comsat y finger en una caja de arena. Otro programa que puede ser candidato para ejecutarse en una caja de arena es named(8). `/etc/defaults/rc.conf` contiene las directrices necesarias (con comentarios) para usar named en una caja de arena. Dependiendo de si está instalando un nuevo sistema o actualizando

un sistema ya existente, las cuentas especiales de usuario que usan estas cajas de arena puede que no estén instaladas. El administrador de sistemas prudente debe investigar e implementar cajas de arena para servidores siempre que sea posible.

Existen numerosos servidores que no se suelen ejecutar en cajas de arena: sendmail, imapd, ftpd, y otros. Existen alternativas para algunos de ellos, pero instalarlas puede requerir más trabajo del que tal vez esté dispuesto a realizar (el factor comodidad ataca de nuevo). Tal vez tenga que ejecutar estos servidores como `root` y depender de otros mecanismos para detectar intrusiones que puedan tener lugar a través de ellos.

Los otros grandes agujeros potenciales de `root` que encontramos en un sistema son los binarios suid-root y sgid. La mayoría de estos binarios, como rlogin, están en `/bin`, `/sbin`, `/usr/bin` o `/usr/sbin`. Aunque no hay nada absolutamente seguro los binarios suid y sgid del sistema por defecto pueden considerarse razonablemente seguros. Aún así, de vez en cuando aparecen agujeros `root` en estos binarios. En 1998 se encontró un agujero `root` en `Xlib`, que hacía a xterm (que suele ser suid) vulnerable. Es mejor prevenir que curar, y el administrador de sistemas prudente restringirá los binarios suid, que sólo el personal de administración debe ejecutar, a un grupo especial al que sólo dicho personal pueda acceder, y deshacerse de cualquier binario suid (`chmod 000`) que no se use. Un servidor sin pantalla generalmente no necesita un binario xterm. Los binarios sgid pueden ser igual de peligrosos. Si un intruso logra comprometer un binario sgid-kmem, el intruso podría leer `/dev/kmem` y llegar a leer el fichero cifrado de contraseñas, poniendo en compromiso potencial cualquier cuenta con contraseña. Por otra parte, un intruso que comprometa el grupo `kmem` puede monitorizar las pulsaciones de teclado que se envien a través de ptys, incluyendo las ptys a las que acceden usuarios que emplean métodos seguros. Un intruso que comprometa el grupo `tty` puede escribir en la pty de casi cualquier usuario. Si un usuario ejecuta un programa de terminal o un emulador capaz de simular un teclado, el intruso podría generar un flujo de datos que provoque que la terminal del usuario muestre una orden en pantalla, orden que el usuario ejecutará.

14.3.3. Asegurar las cuentas de usuario

Las cuentas de usuario suelen ser las más difíciles de asegurar. Aunque puede imponer restricciones de acceso draconianas a su personal administrativo y "estrellar" sus contraseñas, tal vez no pueda hacerlo con todas las cuentas de todos sus usuarios. Si mantiene el control en un grado suficiente quizás lo logre y sea capaz de hacer que las cuentas de sus usuarios sean seguras. Si no, tendrá que ser más cuidadoso (aún) en la monitorización de esas cuentas. Usar ssh y Kerberos en cuentas de usuario da más problemas debido al soporte técnico y administrativo que requerirá, pero sigue siendo mejor solución que un fichero de contraseñas cifradas.

14.3.4. Asegurar el fichero de contraseñas

La única manera segura es ponerle * a tantas contraseñas como sea posible y utilizar ssh o Kerberos para acceder a esas cuentas. Aunque el fichero cifrado de contraseñas (`/etc/`

`spwd.db`) sólo puede ser legible para `root`, puede que un intruso consiga acceso de lectura a ese fichero, incluso sin haber alcanzado el acceso de escritura como root.

Sus "scripts" de seguridad deben buscar siempre cambios en el fichero de contraseñas (consulte Revisión de integridad de ficheros más abajo) e informar de ellos.

14.3.5. Asegurar el Kernel, dispositivos en bruto y el sistema sistema de ficheros

Si un atacante compromete `root` puede hacer cualquier cosa, pero hay ciertas cosas que puede usted preparar para "curarse en salud". Por ejemplo, la mayoría de los kernel modernos tienen un dispositivo de los Kernels modernos tienen un integrado un dispositivo de paquetes. En FreeBSD se llama `bpf`. Un intruso típico tratará de ejecutar un "sniffer" de paquetes en una máquina comprometida. No debería darle a ese intruso tal recurso, y la mayoría de los sistemas no necesitan el dispositivo `bpf`.

Pero si desactiva el dispositivo `bpf` todavía tendrá que preocuparse por `/dev/mem` y `/dev/kmem`. Desde ellos el intruso podría en dispositivos de disco en bruto. También hay que tener muy en cuenta una opción del kernel llamada cargador de módulos, kldload(8). Un intruso con iniciativa puede usar un módulo KLD para instalar su propio dispositivo `bpf`, u otro dispositivo que le permita el "sniffing" en un kernel en ejecución. Para prevenir estos problemas debe ejecutar el kernel en un nivel de seguridad mayor, al menos en securelevel 1. Puede configurar el securelevel mediante una `sysctl` en la variable `kern.securelevel` . Una vez que tiene su securelevel a 1, los accesos de escritura a dispositivos en bruto se denegarán y se impondrán las banderas especiales `schg`. También debe cerciorarse de activar la bandera `schg` en binarios críticos para el arranque, directorios y scripts (dicho de otro modo, todo aquello que se ejecuta *antes* de que se active el securelevel). Puede ser que todo esto sea una exageración, sobre todo teniendo en cuenta que la actualización del sistema se complica bastante a medida que se incrementa el nivel de seguridad. Puede ejecutar el sistema a un nivel de seguridad superior pero no activar la bandera `schg` en cada fichero y directorio del sistema. Otra posibilidad es montar `/` y `/usr` como sólo lectura. Recuerde que siendo demasiado draconiano en aquello que busca proteger puede dificultar mucho la detección de una intrusión.

14.3.6. Revisión de integridad de ficheros: binarios, ficheros de configuración, etc.

Cuando se piensa de protecćon, sólo se puede proteger la configuración central del sistema y los ficheros de control hasta el momento en el que el factor comodidad salta a la palestra. Por ejemplo, si usa `chflags` para activar el bit `schg` en la mayoría de los ficheros de `/` y `/usr` probablemente sea contraproducente; puede proteger los ficheros haciéndolo, pero también cierra una vía de detección. La última capa de su modelo de seguridad tipo cebolla es quizás la más importante: la detección. El resto de su estructura de seguridad será inútil (o peor aún, le proporcionará un sentimiento de seguridad totalmente infundado) si no puede detectar posibles intrusiones. La mitad del trabajo de la cebolla es

alentar al atacante, en lugar de detenerlo, para darle a la parte de la ecuación de detección una oportunidad de atraparlo con las manos en la masa.

La mejor manera de detectar una intrusión es buscar ficheros modificados, perdidos, o cuya presencia o estado sea inesperado. La mejor forma de buscar ficheros modificados es desde otro sistema (que muchas veces es centralizado) con acceso restringido. Escribir sus "scripts" de seguridad en un sistema "extraseguro" y con acceso restringido los hace casi invisibles a posibles atacantes, y esto es algo muy importante. potenciales, y esto es importante. Para poderle sacar el máximo partido debe proporcionar a esa máquina con acceso restringido un acceso preferente al contenido de las otras máquinas de su entorno; suele hacerse mediante la importación vía NFS de sólo lectura de las demás máquinas, o configurando pares de llaves ssh para acceder a las otras máquinas desde la que tiene el acceso restringido. Si exceptuamos el tráfico de red, NFS es el método menos visible y le permite monitorizar los sistemas de ficheros de cada máquina cliente de forma prácticamente indetectable. Si su servidor de acceso restringido está conectado a las máquinas clientes a través de un concentrador o a través de varias capas de encaminamiento el método NFS puede ser muy inseguro, por lo que ssh puede ser la mejor opción, incluso con las huellas de auditoría que ssh va dejando.

Una vez que le da a una máquina de acceso restringido (al menos) acceso de lectura a los sistemas cliente que va a monitorizar, tendrá que escribir "scripts" para efectuar la monitorización. Si va a usar un montaje NFS puede escribir "scripts" utilizando simples herramientas del sistema como find(1) y md5(1). Es aconsejable ejecutar MD5 físicamente en los ficheros de las máquinas cliente al menos una vez al día, y comprobar los ficheros de control (los que hay en `/etc` y `/usr/local/etc`) con una frecuencia incluso mayor. Si aparecen discrepancias al compararlos con la información basada en MD5 que la máquina de acceso restringido usa como base debe hacer una comprobación inmediata y profunda. Un buen "script" también debe buscar binarios que sean suid sin razón aparente, y ficheros nuevos o borrados en particiones del sistema como `/` y `/usr`.

Si usa ssh en lugar de NFS será mucho más complicado escribir el "script" de seguridad. En esencia, tiene que pasar por `scp` los "scripts" a la máquina cliente para poder ejecutarlos, haciéndolos visibles; por seguridad, también tendrá que pasar vía `scp` los binarios (por ejemplo find) que utilizan dichos "scripts". El cliente ssh de la máquina cliente puede estar ya bajo el control del intruso. Con todo y con eso, puede ser necesario usar ssh si trabaja sobre enlaces inseguros, también es mucho más difícil de manejar.

Un buen "script" de seguridad buscará también cambios en la configuración de los ficheros de acceso de usuarios y miembros del personal de administración: `.rhosts` , `.shosts` , `.ssh/authorized_keys` , etc; en resumen, ficheros fuera del rango de revisión `MD5`.

Si tiene que vérselas con una cantidad enorme de espacio en disco para usuarios le llevará mucho tiempo recorrer cada fichero de cada partición. En su caso sería una buena idea configurar mediante opciones de montaje la deshabilitación de binarios y dispositivos suid en esas particiones. Revise las opciones `nodev` y `nosuid` de mount(8). Debería

comprobarlos de todas maneras al menos una vez por semana, ya que el objeto de esta capa es detectar intrusiones, efectivas o no.

La contabilidad de procesos (vea accton(8)) es una opción con una carga relativamente ligera para el sistema operativo, y puede ayudarle como mecanismo de evaluación tras una intrusión. Es especialmente útil para rastrear cómo consiguión realmente acceder el intruso al sistema (asumiendo que el fichero esté intacto después de la intrusión).

Los "scripts" de seguridad deben procesar los logs, y los propios logs deben generarse de la forma más segura posible: un syslog remoto puede ser muy útil. Un intruso trata de cubrir sus huellas, los logs son un recurso crítico cuando el administrador de sistemas intenta determinar la hora y el método de la intrusión inicial. La ejecución de la consola del sistema en un puerto serie y recolectar la información de forma periódica en una máquina segura de monitorización de consolas es una forma de cumplir esta tarea.

14.3.7. Paranoia

Un poco de paranoia nunca está de más. Como norma, un administrador de sistemas puede añadir cualquier tipo de mecanismo de seguridad siempre y cuando no afecte a la comodidad, y puede añadir mecanismos de seguridad que *sí* afecten a la comodidad si tiene una buena razón para hacerlo. Más aún, un administrador de seguridad debe mezclar un poco de ambas cosas: si sigue al pie de la letra las recomendaciones que se dan en este documento también está sirviendo en bandeja de plata al posible atancante su metodología. Ese posible atacante también tiene acceso a este documento.

14.3.8. Ataques de denegación de servicio

Esta sección cubre ataques de denegación de servicio. Un ataque DoS suele consistir en un ataque mediante paquetes. NO hay mucho que pueda hacerse contra un ataque mediante paquetes falsificados ("spoofed") que busque saturar su red, pero puede limitar el daño asegurándose de que los ataques no tiren sus servidores.

1. Limitación de forks en el servidor.

2. Limitación de ataques "springboard" (ataques de respuesta ICMP, ping broadcast, etc.)

3. Caché de rutas del kernel.

Un típico ataque DoS contra un servidor con instancias (forks) sería tratar de provocar que el servidor consuma procesos, descriptores de fichero y memoria hasta tirar la máquina. inetd (consulte inetd(8)) dispone de varias opciones para limitar este tipo de ataque. Recuerde que aunque es posible evitar que una máquina caiga, generalmente no es posible evitar que un servicio sea vea interrumpido a causa el ataque. Consulte la página de manual de inetd atentamente y sobre todo estudie las las opciones `-c`, `-C`, y `-R`. Observe que los ataques con direcciones IP falsificadas sortearán la opción `-C` de inetd, así que

debe usar una combinación de opciones. Algunos servidores autónomos ("standalone") cuentan con parámetros de autolimitación de instancias.

Sendmail tiene la opción `-OMaxDaemonChildren`, que tiende a funcionar mucho mejor que las opciones de límite de carga de sendmail debido al retraso que provoca la carga. Debe especificar un parámetro `MaxDaemonChildren` al inicio de sendmail que sea lo suficientemente alto como para gestionar la carga esperada, pero no tan alto que la computadora no pueda absorber tal número de sendmails sin caerse de boca. También es prudente ejecutar sendmail en modo de cola (`-ODeliveryMode=queued`) y ejecutar el dæmon (`sendmail -bd`) de manera independiente de las ejecuciones de cola (`sendmail -q15m`). Si a pesar de todo necesita entregas en tiempo real puede ejecutar la cola a un intervalo menor, como `-q1m`, pero asegúrese de especificar una opción `MaxDaemonChildren` razonable para *ese* sendmail y así evitar fallos en cascada.

Syslogd puede recibir ataques directos y se recomienda encarecidamente que utilice la opción `-s` siempre que sea posible, y si no la opción `-a`.

También debe ser extremadamente cuidadoso con servicios de conexión inversa como el ident inverso de TCP Wrapper, que puede recibir ataques directos. No se suele usar el ident inverso de TCP Wrapper por esa misma razón.

Es una muy buena idea proteger los servicios internos de acceso externo protegiéndolos vía con un cortafuegos en los routers de frontera. La idea es prevenir ataques de saturación desde el exterior de la LAN, y no tanto para proteger servicios internos de compromisos `root` basados en red. Configure siempre un cortafuegos exclusivo, esto es, "restringir todo *menos* los puertos A, B, C, D y M-Z". De esta manera restringirá todos sus puertos con números bajos excepto ciertos servicios específicos como named (si es el servidor primario de una zona), ntalkd, sendmail, y otros servicios accesibles desde Internet. Si configura el cortafuegos de la otra manera (como un cortafuegos inclusivo o permisivo), tiene grandes posibilidades de que olvide "cerrar" un par de servicios, o de que agregue un nuevo servicio interno y olvide actualizar el cortafuegos. Puede incluso abrir el rango de números de puerto altos en el cortafuegos para permitir operaciones de tipo permisivo sin comprometer sus puertos bajos. Recuerde también que FreeBSD le permite controlar el rango de números de puerto utilizados para asignación dinámica a través de las numerosas `net.inet.ip.portrange` de `sysctl` (`sysctl -a | fgrep portrange`), lo cual también facilita la complejidad de la configuración de su cortafuegos. Por ejemplo, puede utilizar un rango normal primero/último de 4000 ó 5000, y un rango de puerto alto de 49152 a 65535; bloquée todo por debajo de 4000 (excepto para ciertos puertos específicos accesibles desde Internet, por supuesto).

Otro ataque DoS común es llamado ataque "springboard": atacar un servidor de forma que genere respuestas que lo sobrecarguen, sobrecarguen la red local o alguna otra máquina. Los ataques más comunes de este tipo son los *ataques ICMP ping broadcast*. El atacante falsifica paquetes ping enviados a la dirección broadcast de su LAN simulando que la dirección IP origen es la de la máquina que desean atacar. Si sus routers de frontera no están

configurados para lidiar con pings a direcciones de broadcast su LAN termina generando suficientes respuestas a la dirección origen falsificada como para saturar a la víctima, especialmente cuando el atacante utiliza el mismo truco en varias docenas de direcciones broadcast en varias docenas de redes diferentes a la vez. Se han medido ataques de broadcast de más de ciento veinte megabits. Un segundo tipo de ataque "springboard" bastante común se da contra el sistema de informe de error de ICMP. Un atacante puede saturar la conexión entrante de red de un servidor mediante la construcción de paquetes que generen respuestas de error ICMP, provocando que el servidor sature su conexión saliente de red con respuestas ICMP. Este tipo de ataque también puede tumbar el servidor agotando sus "mbufs", especialmente si el servidor no puede drenar lo suficientemente rápido las respuestas ICMP que genera. El kernel de FreeBSD tiene una opción de compilación llamada `ICMP_BANDLIM` , que limita la efectividad de este tipo de ataques. La última gran categoría de ataques "springboard" está relacionada con ciertos servicios de inetd, como el servicio de eco udp. El atacante simplemente imita un paquete UDP con el puerdo de eco del servidor A como dirección de origen, y el puerto eco del servidor B como dirección de destino, estando ambos servidores en la misma LAN. Un atacante puede sobrecargar ambos servidores y la propia LAN inyectando simplemente un par de paquetes. Existen problemas similares con el puerto chargen. Un administrador de sistemas competente apagará todos estos servicios internos de verificación de inetd.

Los ataques con paquetes falsificados pueden utilizarse también para sobrecargar la caché de rutas del kernel. Consulte los parámetros de `sysctl net.inet.ip.rtexpire` , `rtminexpire`, y `rtmaxcache`. Un ataque de paquetes falsificados que utiliza una dirección IP origen aleatoria provocará que el kernel genere una ruta temporal en caché en su tabla de rutas, visible con `netstat -rna | fgrep W3` . Estas rutas suelen expiran en 1600 segundos más o menos. Si el kernel detecta que la tabla de rutas en caché es ya demasiado grande reducirá dinámicamente `rtexpire`, pero nunca la reducirá a un valor que sea menor que `rtminexpire`. Esto nos presenta dos problemas:

1. El kernel no reacciona con suficiente rapidez cuando un servidor ligeramente cargado es atacado.

2. El `rtminexpire` no es lo suficientemente bajo para que el kernel sobreviva a un ataque sostenido.

Si sus servidores están conectados a Internet mediante mediante una línea T3 o superior puede ser prudente corregir manualmente `rtexpire` y `rtminexpire` por medio de sysctl(8). Nunca ponga ambos parámetros a cero (a menos que desée estrellar la máquina). Configurar ambos parámetros a 2 segundos debería bastar para proteger de ataques la tabla de rutas.

14.3.9. Otros aspectos del acceso con Kerberos y SSH

Existen un par de detalles con respecto a Kerberos y ssh que debe analizar sy pretende usarlos. Kerberos V es un excelente protocolo de protocolo de autentificación, pero hay

errores en la versión kerberizada de telnet y rlogin que las hacen inapropiadas para gestionar flujos binarios. Ademé Kerberos no cifra por defecto una sesión a menos que utilice la opción `-x`. ssh cifra todo por defecto.

ssh funciona bastante bien en todos los casos, con la sola salvedad de que por defecto reenvía llaves de cifrado. Esto significa que si usted tiene una estación de trabajo segura, que contiene llaves que le dan acceso al resto del sistema, y hace ssh a una máquina insegura, sus llaves se pueden utilizar. Las llaves en sí no se exponen, pero ssh crea un puerto de reenvío durante el login, y si un atacante ha comprometido el `root` de la máquina insegura, puede utilizar ese puerto para usar sus llaves y obtener acceso a cualquier otra máquina que sus llaves abran.

Le recomendamos que, siempre que sea posible, use ssh combinado con Kerberos en los login de su personal de administración. para logins de staff. Puede compilar ssh con soporte de Kerberos. Esto reducirá su dependencia de llaves ssh expuestas, al mismo tiempo que protege las contraseñas vía Kerberos. Las llaves ssh deben usarse sólamente para tareas automáticas desde máquinas seguras (algo que Kerberos no hace por incompatibilidad). Recomendamos también que desactive el reenvío de llaves en la configuración de ssh, o que use la opción `from=IP/DOMAIN` que ssh incluye en `authorized_keys` ; así la llave sólo podrá ser utilizada por entidades que se validen desde máquinas específicas.

14.4. DES, MD5 y Crypt

Secciones reescritas y actualizadas por Bill Swingle.

Cada usuario de un sistema UNIX® tiene una contraseña asociada a su cuenta. Parece obvio que estas contraseñas sólo deben ser conocidas por el usuario y por el sistema operativo. Para que estas contraseñas permanezcan en secreto se cifran con lo que se conoce como un “hash de una pasada”, esto es, sólo pueden ser fácilmente cifradas pero no descifradas. En otras palabras, lo que acabamos de decir es tan obvio que ni siguiera es verdad: el propio sistema operativo no sabe cuál es *realmente* la contraseña. Lo único que conoce es la versión *cifrada* de la contrasenña. La única manera de obtener la contraseña en “texto plano” es por medio de una búsqueda de fuerza bruta en el espacio de contraseñas posibles.

Por desgracia la única manera segura de cifrar contraseñas cuando UNIX® empezó a hacerlo estaba basada en DES, (“Data Encryption Standard”, “estándar de cifrado de datos”). Esto no era un gran problema para usuarios residentes en los EEUU, pero el código fuente de FreeBSD no se podía exportar desde los EEUU, así que FreeBSD hubo de buscar una forma de complir las leyes de EEUU y al mismo tiempo mantener la compatibilidad con otras variantes de UNIX® que que todavía utilizaban DES.

La solución fué dividir las bibliotecas de cifrado para que los usuarios de EEUU pudieran instalar las bibliotecas DES pero los usuarios del resto del mundo tuvieran un método de cifrado que pudiera ser exportado. Así es como FreeBSD comenzó a usar MD5 como su

método de cifrado por defecto. MD5 se considera más seguro que DES, así que se mantiene la opción de poder instalar DES por motivos de compatibilidad.

14.4.1. Cómo reconocer su mecanismo de cifrado

En versiones anteriores a FreeBSD 4.4 `libcrypt.a` era un enlace simbólico que apuntaba a la biblioteca que se usaba para el cifrado. En FreeBSD 4.4 se cambió `libcrypt.a` para ofrecer una biblioteca hash configurable de validación de contraseñas. Actualmente la biblioteca permite funciones hash DES, MD5 y Blowfish. FreeBSD utiliza por defecto MD5 para cifrar contraseñas.

Es muy sencillo identificar qué método usa FreeBSD para cifrar. Una forma es examinando las contraseñas cifradas en `/etc/master.passwd` . Las contraseñas cifradas con el hash MD5 son más largas que las cifradas con el hash DES, y también comienzan por los caracteres `$1$`. Las contraseñas que comienzan por `$2a$` están cifradas con la función hash de Blowfish. Las contraseñas DES no tienen ninguna característica particular, pero son más cortas que las contraseñas MD5, y están codificadas en un alfabeto de 64 caracteres que no incluye el caracter `$`; es por esto que una cadena relativamente corta que comience con un signo de dólar es muy probablemente una contraseña DES.

El formato de contraseña a usar en nuevas contraseñas se define en `/etc/login.conf` mediante `passwd_format` , pudiendo tener los valores `des`, `md5` o `blf`. Consulte la página de manual login.conf(5) para más información.

14.5. Contraseñas de un solo uso

S/Key es un esquema de contraseña de un solo uso basado en una función de hash de sentido único. FreeBSD utiliza el hash MD4 por compatibilidad, pero otros sistemas usan MD5 y DES-MAC. S/Key forma parte del sistema base de FreeBSD desde la versión 1.1.5 y se usa también en un número creciente de otros sistemas operativos. S/Key es una marca registrada de Bell Communications Research, Inc.

A partir de la versión 5.0 de FreeBSD S/Key fué reemplazado por su equivalente OPIE (“One-time Passwords In Everything”, “Contraseñas de un solo uso para todo”). OPIE usa por defecto hash MD5.

En esta sección se explican tres tipos de contraseña. La primera es la típica contraseña al estilo UNIX® o Kerberos; las llamaremos “contraseñas UNIX®”. El segundo tipo es la contraseña de un solo uso, que se genera con el programa `key` de S/Key o con opiekey(1) de OPIE, y que aceptan los programas `keyinit`, opiepasswd(1), y el prompt de login; llamaremos a esta una “contraseña de un solo uso”. El último tipo de contraseña es la contraseña secreta que le da usted a los programas `key/opiekey` (y a veces `keyinit/opiepasswd`), que se usa para generar contraseñas de un solo uso; a estas las llamaremos “contraseñas secretas”, o simplemente “contraseña”.

La contraseña secreta no tiene nada que ver con su contraseña UNIX®; pueden ser la misma, pero no es recomendable. Las contraseñas secretas S/Key y OPIE no están limitadas a 8 caracteres como las contraseñas UNIX® antiguas[1], pueden ser tan largas como se quiera. Las contraseñas con frases de seis o siete palabras muy largas son bastante comunes. El funcionamiento del sistema S/Key o el OPIE es en gran parte completamente independiente del sistema de contraseñas UNIX®.

Además de la contraseña hay dos datos que son importantes para S/Key y OPIE. Uno es lo que se conoce como "semilla" o "llave", que consiste en dos letras y cinco dígitos. El otro dato importante se llama la "cuenta de iteración", que es un número entre 1 y 100. S/Key genera la contraseña de un solo uso concatenando la semilla y la contraseña secreta, aplica el hash MD4/MD5 tantas veces como especifique la cuenta de iteración y convierte el resultado en seis palabras cortas en inglés. Estas seis palabras en inglés son su contraseña de un solo uso. El sistema de autentificación (principalmente PAM) mantiene un registro del uso de contraseñas de un solo uso, y el usuario puede validarse si el hash de la contraseña que proporciona es igual a la contraseña previa. Como se utiliza un hash de sentido único es imposible generar futuras contraseñas de un solo uso si una contraseña que ya ha sido usada fuera capturada; la cuenta de iteración se reduce después de cada login correcto para sincronizar al usuario con el programa login. Cuanto la iteración llega a 1, S/Key y OPIE deben reinicializar.

Hay tres programas involucrados en cada uno de estos sistemas. Los programas `key` y `opiekey` aceptan una cuenta iterativa, una semilla y una contraseña secreta, y generan una contraseña de un solo uso o una lista consecutiva de contraseñas de un solo uso. Los programas `keyinit` y `opiepasswd` se usan respectivamente para inicializar S/Key y OPIE, y para cambiar contraseñas, cuentas iterativas o semillas; toman ya sea una frase secreta, o una cuenta iterativa y una contraseña de un solo uso. Los programas `keyinfo` y `opieinfo` examinan los ficheros de credenciales correspondientes (`/etc/skeykeys` o `/etc/opiekeys`) e imprimen la cuenta iterativa y semilla del usuario invocante.

Explicaremos cuatro tipos de operaciones diferentes. La primera es usar `keyinit` o `opiepasswd` a través de una conexión segura para configurar contraseñas de un solo uso por primera vez, o para cambiar su contraseña o semilla. La segunda operación es utilizar `keyinit` o `opiepasswd` a través de una conexión insegura, además de usar `key` u `opiekey` sobre una conexión segura para hacer lo mismo. La tercera es usar `key/opiekey` para conectarse a través de una conexión insegura. La cuarta es usar `opiekey` o `key` para generar numerosas llaves, que pueden ser escritas para llevarlas con usted al ir a algún lugar desde el que no se puedan hacer conexiones seguras a ningún sitio.

14.5.1. Inicialización de conexiones seguras

Para inicializar S/Key por primera vez cambie su contraseña, o cambie su semilla mientras está conectado a través de una conexión segura (esto es, en la consola de una máquina o vía ssh); use `keyinit` sin ningún parámetro:

[1] En FreeBSD la contraseña del login estándar puede ser de hasta 128 caracteres de longitud.

```
% keyinit
Adding unfurl:
Reminder - Only use this method if you are directly connected.
If you are using telnet or rlogin exit with no password and use ↺
keyinit -s.
Enter secret password:
Again secret password:

ID unfurl s/key is 99 to17757
DEFY CLUB PRO NASH LACE SOFT
```

En OPIE se utiliza `opiepasswd`:

```
% opiepasswd -c
[grimreaper] ~ $ opiepasswd -f -c
Adding unfurl:
Only use this method from the console; NEVER from remote. If you ↺
are using
telnet, xterm, or a dial-in, type ^C now or exit with no password.
Then run opiepasswd without the -c parameter.
Using MD5 to compute responses.
Enter new secret pass phrase:
Again new secret pass phrase:
ID unfurl OTP key is 499 to4268
MOS MALL GOAT ARM AVID COED
```

En `Enter new secret pass phrase:` o `Enter secret password:`, debe introducir una contraseña o frase. Recuerde que no es la contraseña que utilizará para entrar, se usará para generar sus llaves de un solo uso. La línea "ID" da los parámetros de su instancia en particular: su nombre de login, la cuenta iterativa y semilla. En el momento del login el sistema recordará estos parámetros y los presentará de nuevo para que no tenga que recordarlos. La última línea proporciona las contraseéas de un solo uso que corresponden a esos parámetros y su contraseña secreta; si fuera a hacer login de manera inmediata, debería usar esta contraseña de una sola vez.

14.5.2. Inicialización de conexiones inseguras

Para inicializar o cambiar su contraseña secreta a través de una conexión insegura, necesitará tener alguna conexión segura a algún lugar donde pueda ejecutar `key` u `opiekey`; puede ser gracias a un accesorio de escritorio o en una Macintosh®, o un prompt de shell en una máquina en la que confíe. Necesitará también una cuenta iterativa (100 probablemente sea un buen valor), y puede usar su propia semilla, o usar una generada aleatoriamente. Siguiendo con la conexión insegura (hacia la máquina que está inicializando), ejecute `keyinit -s`:

```
% keyinit -s
Updating unfurl:
Old key: to17758
Reminder you need the 6 English words from the key command.
Enter sequence count from 1 to 9999: 100
```

```
Enter new key [default to17759]:
s/key 100 to 17759
s/key access password:
s/key access password:CURE MIKE BANE HIM RACY GORE
```

En OPIE debe usar `opiepasswd`:

```
% opiepasswd

Updating unfurl:
You need the response from an OTP generator.
Old secret pass phrase:
        otp-md5 498 to4268 ext
        Response: GAME GAG WELT OUT DOWN CHAT
New secret pass phrase:
        otp-md5 499 to4269
        Response: LINE PAP MILK NELL BUOY TROY

ID mark OTP key is 499 gr4269
LINE PAP MILK NELL BUOY TROY
```

Para aceptar la semilla por defecto (la que el programa `keyinit` llama `key`, "llave", para terminar de complicar las cosas), pulse Enter. Antes de introducir una una contraseña de acceso cambie a su conexión o accesorio de escritorio S/Key y dele el mismo parámetro:

```
% key 100 to17759
Reminder - Do not use this program while logged in via telnet or ↺
rlogin.
Enter secret password: <secret password>
CURE MIKE BANE HIM RACY GORE
```

O para OPIE:

```
% opiekey 498 to4268
Using the MD5 algorithm to compute response.
Reminder: Don't use opiekey from telnet or dial-in sessions.
Enter secret pass phrase:
GAME GAG WELT OUT DOWN CHAT
```

Vuelva a la conexión insegura y copie la contraseña de un solo uso generada al programa que quiera usar.

14.5.3. Generación una sola contraseña de un solo uso

Una vez que ha inicializado S/Key u OPIE, cuando haga login verá un "prompt" parecido al siguiente:

```
% telnet ejemplo.com
Trying 10.0.0.1...
Connected to ejemplo.com
Escape character is '^]'.
```

```
FreeBSD/i386 (ejemplo.com) (ttypa)

login: <username>
s/key 97 fw13894
Password:
```

O, en el caso de OPIE:

```
% telnet ejemplo.com
Trying 10.0.0.1...
Connected to ejemplo.com
Escape character is '^]'.

FreeBSD/i386 (ejemplo.com) (ttypa)

login: <nombre_de_usuario>
otp-md5 498 gr4269 ext
Password:
```

Como una nota aparte, el "prompt" de S/Key y OPIE cuenta con una opción útil (que no se muestra aquí): si pulsa Enter en el "prompt" de contraseña el "prompt" activará el eco para que pueda ver en pantalla lo que teclea. Esto puede ser extremadamente útil si está tecleando una contraseña a a mano o desde un la lista impresa.

Ahora necesitará generar su contraseña de un sólo uso para responder a este "prompt" de login. Debe hacerlo en un sistema digno de confianza y en el que pueda ejecutar `key` u `opiekey`. Existen versiones DOS, Windows® y también para Mac OS®. Ambos usarán la cuenta iterativa y la semilla como opciones de línea de órdenes. Puede cortarlas y pegarlas desde el "prompt" de login de la máquina en la que se está identificando.

En el sistema de confianza:

```
% key 97 fw13894
Reminder - Do not use this program while logged in via telnet or ↺
rlogin.
Enter secret password:
WELD LIP ACTS ENDS ME HAAG
```

Con OPIE:

```
% opiekey 498 to4268
Using the MD5 algorithm to compute response.
Reminder: Don't use opiekey from telnet or dial-in sessions.
Enter secret pass phrase:
GAME GAG WELT OUT DOWN CHAT
```

Ahora que tiene su contraseña de un solo uso puede proceder con el login:

```
login: <nombre_de_usuario>
s/key 97 fw13894
Password: <Enter para activar el eco>
s/key 97 fw13894
```

```
Password [echo on]: WELD LIP ACTS ENDS ME HAAG
Last login: Tue Mar 21 11:56:41 from 10.0.0.2 ...
```

14.5.4. Generación de múltiples contraseñas de un solo uso

A veces usted hay que ir a lugares donde no hay acceso a una máquina de fiar o a una conexión segura. En estos casos, puede utilizar `key` y `opiekey` para generar previamente numerosas contraseñas de un solo uso para, una vez impresas, llevárselas a donde hagan falta. Por ejemplo:

```
% key -n 5 30 zz99999
Reminder - Do not use this program while logged in via telnet or ↺
rlogin.
Enter secret password: <secret password>
26: SODA RUDE LEA LIND BUDD SILT
27: JILT SPY DUTY GLOW COWL ROT
28: THEM OW COLA RUNT BONG SCOT
29: COT MASH BARR BRIM NAN FLAG
30: CAN KNEE CAST NAME FOLK BILK
```

O para OPIE:

```
% opiekey -n 5 30 zz99999
Using the MD5 algorithm to compute response.
Reminder: Don't use opiekey from telnet or dial-in sessions.
Enter secret pass phrase: <secret password>
26: JOAN BORE FOSS DES NAY QUIT
27: LATE BIAS SLAY FOLK MUCH TRIG
28: SALT TIN ANTI LOON NEAL USE
29: RIO ODIN GO BYE FURY TIC
30: GREW JIVE SAN GIRD BOIL PHI
```

El `-n 5` pide cinco llaves en secuencia, la opción `30` especifica que ese debe ser el último número de iteración. Observe que se imprimen en el orden *inverso* de uso. Si es realmente paranoico escriba los resultados a mano; si no, puede enviar la salida a `lpr`. Observe que cada línea muestra la cuenta iterativa y la contraseña de un solo uso; puede ir tachando las contraseñas según las vaya utilizando.

14.5.5. Restricción del uso de contraseñas UNIX®

S/Key puede implantar restricciones en el uso de contraseñas UNIX® basándose en el nombre de equipo, nombre de usuario, puerto de terminal o dirección IP de una sesión de login. Consulte el fichero de configuración `/etc/skey.access` . La página de manual de skey.access(5) contiene más información sobre el formato del fichero y detalla también algunas precauciones de seguridad que hay que tener en cuenta antes de basar nuestra seguridad en este fichero.

Si `/etc/skey.access` no existiera (por defecto es así en sistemas FreeBSD 4.X) todos los usuarios podrán disponer de contraseñas UNIX®. Si el fichero existe se exigirá a todos

los usuarios el uso de S/Key, a menos que se configure de otro modo en `skey.access` . En todos los casos las contraseñas UNIX® son admiten en consola.

Aquí hay un ejemplo del fichero de configuración `skey.access` que muestra las tres formas más comunes de configuración:

```
permit internet 192.168.0.0 255.255.0.0
permit user fnord
permit port ttyd0
```

La primera línea (`permit internet`) permite a usuarios cuyas direcciones IP origen (las cuales son vulnerables a una falsificación) concuerden con los valores y máscara especificados utilizar contraseñas UNIX®. Esto no debe usarse como mecanismo de seguridad, sino como medio de recordarle a los usuarios autorizados que están usando una red insegura y necesitan utilizar S/Key para la validación.

La segunda línea (`permit user`) permite al nombre de usuario especificado, en este caso `fnord`, utilizar contraseñas UNIX® en cualquier momento. Hablando en general, esto solo debe ser usado por gente que no puede usar el programa `key`, como aquellos con terminales tontas o refractarios al aprendizaje.

La tercera línea (`permit port`) permite a todos los usuarios validados en la línea de terminal especificada utilizar contraseñas UNIX®; esto puede usarse para usuarios que se conectan mediante “dial-ups”.

OPIE puede restringir el uso de contraseñas UNIX® basándose en la dirección IP de una sesión de login igual que lo haría S/Key. El fichero que gestiona esto es `/etc/opieaccess` , que está incluído por defecto en sistemas FreeBSD 5.0 o posteriores. Revise opieaccess(5) para más información sobre este fichero y qué consideraciones de seguridad debe tener presentes a la hora de usarlo.

Veamos un ejemplo de `opieaccess`:

```
permit 192.168.0.0 255.255.0.0
```

Esta línea permite a usuarios cuya dirección IP de origen (vulnerable a falsificación) concuerde con los valores y máscara especificados, utilizar contraseñas UNIX® en cualquier momento.

Si no concuerda ninguna regla en `opieaccess` se niegan por defecto los logins no-OPIE.

14.6. TCP Wrappers

Escrito por: Tom Rhodes.

Cualquiera que esté familiarizado con inetd(8) probablemente haya oído hablar de TCP Wrappers, pero poca gente parece comprender completamente su utilidad en un entorno de red. Parece que todos quieren instalar un cortafuegos para manejar conexiones de red.

Aunque un cortafuegos tiene una amplia variedad de usos hay cosas que un cortafuegos no es capaz de gestionar, como el envío de texto como respuesta al creador de la conexión. El software TCP hace esto y más. En las siguientes secciones se explicarán unas cuantas opciones de TCP Wrappers y, cuando sea necesario, se mostrarán ejemplos de configuraciones.

El software TCP Wrappers extiende las habilidades de `inetd` para ofrecer soporte para cada servidor dæmon bajo su control. Utilizando este método es posible proveer soporte de logs, devolver mensajes a conexiones, permitir a un dæmon aceptar solamente conexiones internas, etc. Aunque algunas de estas opciones pueden conseguirse gracias a un cortafuegos, no sólo añadirá una capa extra de seguridad, sino que irá más allá del nivel de control ue un cortafuegos puede ofrecerle.

Las brillantes capacidades de TCP Wrappers no deben considerarse una alternativa a un buen cortafuegos. TCP Wrappers puede usarse conjuntamente con un cortafuegos u otro sistema de de seguridad, pues ofrece una capa extra de protección para el sistema.

Ya que es una extensión de la configuración de `inetd`, se da por hecho que el lector ha leído la sección configuración de inetd.

Nota

Aunque los programas ejecutados por inetd(8) no son exactamente "dæmons" tradicionalmente han recibido ese nombre. Dæmon es, por tanto, el término que usaremos en esta sección.

14.6.1. Configuración inicial

El único requisito para usar TCP Wrappers en FreeBSD es que el servidor `inetd` se inicie desde `rc.conf` con la opción `-Ww` (es la configuración por defecto). Por descontado, se presupone que `/etc/hosts.allow` estará correctamente configurado, pero syslogd(8) enviará mensajes a los logs del sistema si no es así.

Nota

A diferencia de otras implementaciones de TCP Wrappers, se ha dejado de usar `hosts.deny`. Todas las opciones de configuración deben ir en `/etc/hosts.allow`.

En la configuración más simple las políticas de conexión de dæmons están configuradas ya sea a permitir o bloquear, dependiendo de las opciones en `/etc/hosts.allow`. La con-

figuración por defecto en FreeBSD consiste en permitir una conexión a cada dæmon iniciado por `inetd`. Es posible modificar esta configuración, pero explicaremos cómo hacerlo después de exponer la configuración básica.

La configuración básica tiene la estructura `dæmon : dirección : acción`, donde `dæmon` es el nombre de dæmon que inicia `inetd`. La `dirección` puede ser un nombre de equipo válido, una dirección IP o IPv6 encerrada en corchetes ([ ]). El campo acción puede ser permitir o denegar para el dar el acceso apropiado. Tenga presente que la configuración funciona en base a la primera regla cuya semántica concuerde; esto significa que el fichero de configuración se lee en orden ascendente hasta que concuerde una regla. Cuando se encuentra una concordancia se aplica la regla y el proceso se detendrá.

Existen muchas otras opciones pero estas se explican en una sección posterior. Una línea de configuración simple puede generarse mediante datos así de simples. Por ejemplo, para permitir conexiones POP3 mediante el dæmon mail/qpopper, añada las siguientes líneas a `hosts.allow` :

```
# This line is required for POP3 connections:
qpopper : ALL : allow
```

Despues de añadir esta línea tendrá que reiniciar `inetd`. Use kill(1) o use el parámetro *`restart`* de `/etc/rc.d/inetd .`

14.6.2. Configuración avanzada

Las opciones avanzadas de TCP Wrappers le permiten un mayor control sobre la gestión de conexiones. En algunos casos puede convenir el enío de un comentario a ciertos equipos o conexiones de dæmons. En otros casos, quizás se deba registrar una entrada en un log o enviar un correo al administrador. Otro tipo de situaciones pueden requerir el uso de un servicio solamente para conexiones locales. Todo esto es posible gracias al uso de unas opciones de configuración conocidas como `comodines`, caracteres de expansión y ejecución de órdenes externas. Las siguientes dos secciones intentarán cubrir estas situaciones.

14.6.2.1. Órdenes externas

Imaginemos una situación en la que una conexión debe ser denegada pero se debe mandar un motivo a quien intentó establecer esa conexión. ?Cómo? Mediante la opción `twist`. Ante un intento de conexión se invoca a `twist`, que ejecuta una orden de shell o un “script”. Tiene un ejemplo en el fichero `hosts.allow` :

```
# The rest of the daemons are protected.
ALL : ALL \
        : severity auth.info \
        : twist /bin/echo "No se permite utilizar %d desde %h."
```

Este ejemplo muestra que el mensaje, “No se permite utilizar `dæmon` desde `nombre de equipo`.” se enviará en el caso de cualquier dæmon no configurado previamente en el fi-

chero de acceso. Esto es extremadamente útil para enviar una respuesta al creador de la conexión justo después de que la conexión establecida es rechazada. Observe que cualquier mensaje que se desee enviar *debe ir* entre comillas "; esta regla no tiene excepciones.

Aviso

Es posible lanzar un ataque de denegación de servicio al servidor si un atacante o grupo de atacantes pueden llegar a sobrecargar estos dæmons con peticiones de conexión.

Otra posibilidad en estos casos es usar la opción `spawn`. Igual que `twist`, `spawn` niega implícitamente la conexión, y puede usarse para ejecutar órdenes de shell externos o "scripts". A diferencia de `twist`, `spawn` no enviará una respuesta al origen de la conexión. Veamos un ejemplo; observe la siguiente línea de configuración:

```
# No permitimos conexiones desde ejemplo.com:
ALL : .ejemplo.com \
  : spawn (/bin/echo %a desde %h intento acceder a %d >> \
    /var/log/connections.log) \
  : deny
```

Esto denegará todos los intentos de conexión desde el dominio `*.ejemplo.com` ; simultáneamente creará una entrada con el nombre del equipo, dirección IP y el dæmon al que intentó conectarse al fichero `/var/log/connections.log` .

Además de la sustitución de caracteres ya expuesta más arriba (por ejemplo %a) existen unas cuantas más. Si quiere ver la lista completa consulte la página de manual hosts_access(5).

14.6.2.2. Opciones comodín

Hasta ahora se ha usado `ALL` en todos los ejemplos, pero hay otras opciones interesantes para extender un poco más la funcionalidad. Por ejemplo, `ALL` puede usarse para concordar con cualquier instancia ya sea un dæmon, dominio o dirección IP. Otro comodín es `PARANOID`, que puede utilizarse para concordar con cualquier equipo que presente una dirección IP que pueda estar falsificada. En otras palabras, `paranoid` puede usarse para definir una acción a tomar siempre que tenga lugar una conexión desde una dirección IP que difiera de su nombre de equipo. Quizás todo se vea más claro con el siguiente ejemplo:

```
# Bloquear peticiones posiblemente falsificadas a sendmail:
sendmail : PARANOID : deny
```

En ese ejemplo todas las peticiones de conexión a `sendmail` que tengan una dirección IP que varíe de su nombre de equipo serán denegadas.

Atención

Utilizando `PARANOID` puede bloquear el acceso a servidores si el cliente o el servidor tiene una configuración de DNS incorrecta. Recomendamos al administrador la máxima cautela en su uso.

Consulte hosts_access(5) si quiere saber más sobre los comodines y sus posibilidades de uso.

Si quiere que cualquiera de los ejemplos citados funcione debe comentar la primera línea de `hosts.allow` (tal y como se dijo al principio de la sección.

14.7. KerberosIV

Escrito por Mark Murray.
Basado en un texto de Mark Dapoz.

Kerberos es un sistema/protocolo de red agregado que permite a los usuarios identificarse a través de los servicios de un servidor seguro. Los servicios como login remoto, copia remota, copias de ficheros de un sistema a otro y otras tantas tareas arriesgadas pasan a ser considerablemente seguras y más controlables.

Las siguientes instrucciones pueden usarse como una guía para configurar Kerberos tal y como se distribuye con FreeBSD. De todas maneras, debe consultar diversas páginas de manual para conocer todos los detalles.

14.7.1. Instalación de KerberosIV

Kerberos es un componente opcional de FreeBSD. La manera más fácil de instalar este software es seleccionando la distribución `krb4` o `krb5` en sysinstall durante la instalación inicial de FreeBSD. Desde ahí instalará la implementación de Kerberos "eBones" (KerberosIV) o "Heimdal" (Kerberos5). Estas implementaciones se incluyen porque a que han sido desarrolladas fuera de EEUU y Canadá, lo que las convertía en accesibles para administradores de sistemas del resto del mundo durante la época restrictiva de control control de exportaciones de código criptográfico desde EEUU.

También puede instalar la implementación de Kerberos del MIT desde la colección de ports (security/krb5).

14.7.2. Creación de la base de datos inicial

Esto solo debe hacerse en el servidor Kerberos. Lo primero es asegurarse de que no tiene bases de datos de Kerberos anteriores. Entre al directorio `/etc/kerberosIV` y asegúrese de que solo estén los siguientes ficheros:

```
# cd /etc/kerberosIV
# ls
README  krb.conf        krb.realms
```

Si existe cualquier otro fichero (como `principal.*` o `master_key`) utilice `kdb_destroy` para destruir la base de datos antigua de Kerberos. Si Kerberos no está funcionando simplemente borre los ficheros sobrantes.

Edite `krb.conf` y `krb.realms` para definir su dominio Kerberos. En nuestro ejemplo el dominio será `EJEMPLO.COM` y el servidor es `grunt.ejemplo.com` . Editamos o creamos `krb.conf` :

```
# cat krb.conf
EJEMPLO.COM
EJEMPLO.COM grunt.ejemplo.com admin server
CS.BERKELEY.EDU okeeffe.berkeley.edu
ATHENA.MIT.EDU kerberos.mit.edu
ATHENA.MIT.EDU kerberos-1.mit.edu
ATHENA.MIT.EDU kerberos-2.mit.edu
ATHENA.MIT.EDU kerberos-3.mit.edu
LCS.MIT.EDU kerberos.lcs.mit.edu
TELECOM.MIT.EDU bitsy.mit.edu
ARC.NASA.GOV trident.arc.nasa.gov
```

Los demás dominios no deben estar forzosamente en la configuración. Los hemos incluido como ejemplo de cómo puede hacerse que una máquina trabaje con múltiples dominios. Si quiere mantener todo simple puede abstenerse de incluirlos.

La primera línea es el dominio en el que el sistema funcionará. Las demás líneas contienen entradas dominio/equipo. El primer componente de cada línea es un dominio y el segundo es un equipo de ese dominio, que actúa como "centro de distribución de llaves". Las palabras `admin server` que siguen al nombre de equipo significan que ese equipo también ofrece un servidor de base da datos administrativo. Si quiere consultar una explicación más completa de estos términos consulte las páginas de manual de de Kerberos.

Ahora añadiremos `grunt.ejemplo.com` al dominio `EJEMPLO.COM` y también una entrada para poner todos los equipos en el dominio `.ejemplo.com` Kerberos `EJEMPLO.COM` . Puede actualizar su `krb.realms` del siguiente modo:

```
# cat krb.realms
grunt.ejemplo.com EJEMPLO.COM
.ejemplo.com EJEMPLO.COM
.berkeley.edu CS.BERKELEY.EDU
```

```
.MIT.EDU ATHENA.MIT.EDU
.mit.edu ATHENA.MIT.EDU
```

Igual que en caso previo, no tiene por qué incluir los demás dominios. Se han incluido para mostrar cómo puede usar una máquina en múltiples dominios. Puede eliminarlos y simplificar la configuración.

La primera línea pone al sistema *específico* en el dominio nombrado. El resto de las líneas muestran cómo asignar por defecto sistemas de un subdominio a un dominio Kerberos.

Ya podemos crear la base de datos. Solo se ejecuta en el servidor Kerberos (o centro de distribución de llaves). Ejecute `kdb_init`:

```
# kdb_init
Realm name [default  ATHENA.MIT.EDU -]:   EJEMPLO.COM
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.

Enter Kerberos master key:
```

Ahora tendremos que guardar la llave para que los servidores en la máquina local puedan recogerla. Utilice `kstash`:

```
# kstash

Enter Kerberos master key:

Current Kerberos master key version is 1.

Master key entered. BEWARE!
```

Esto guarda la contraseña cifrada maestra en `/etc/kerberosIV/master_key` .

14.7.3. Puesta en marcha del sistema

Tendrá que añadir a la base de datos dos entradas en concreto para *cada* sistema que vaya a usar Kerberos: `kpasswd` y `rcmd`. Se hacen para cada sistema individualmente cada sistema, y el campo “instance” es el nombre individual del sistema.

Estos dæmons kpasswd y rcmd permiten a otros sistemas cambiar contraseñas de Kerberos y ejecutar órdenes como rcp(1), rlogin(1) y rsh(1).

Ahora vamos a añadir estas entradas:

```
# kdb_edit
Opening database...

Enter Kerberos master key:
```

```
Current Kerberos master key version is 1.

Master key entered.  BEWARE!
Previous or default values are in [brackets] ,
enter return to leave the same, or new value.

Principal name:  passwd
Instance:  grunt

<Not found>, Create [y] ?  y

Principal: passwd, Instance: grunt, kdc_key_ver: 1
New Password:                    <---- tecleo aleatorio
Verifying password

New Password:  <---- tecleo aleatorio

Random password [y] ?  y

Principal's new key version = 1
Expiration date (enter yyyy-mm-dd) [ 2000-01-01 -] ?
Max ticket lifetime (*5 minutes) [ 255 -] ?
Attributes [ 0 -] ?
Edit O.K.
Principal name:  rcmd
Instance:  grunt

<Not found>, Create [y] ?

Principal: rcmd, Instance: grunt, kdc_key_ver: 1
New Password:   <---- tecleo aleatorio
Verifying password

New Password:           <---- tecleo aleatorio

Random password [y] ?

Principal's new key version = 1
Expiration date (enter yyyy-mm-dd) [ 2000-01-01 -] ?
Max ticket lifetime (*5 minutes) [ 255 -] ?
Attributes [ 0 -] ?
Edit O.K.
Principal name:          <---- si introduce datos nulos saldrá del ↺
programa
```

14.7.4. Creación del fichero del servidor

Ahora tendremos que extraer todas las instancias que definen los servicios en cada máquina; para ello usaremos `ext_srvtab` . Esto creará un fichero que debe ser copiado o movido *por medios seguros* al directorio `/etc/kerberosIV` de cada cliente Kerberos. Este fichero debe existir en todos los servidores y clientes dada su importancia clave para el funcionamiento de Kerberos.

```
# ext_srvtab grunt
Enter Kerberos master key:

Current Kerberos master key version is 1.

Master key entered. BEWARE!
Generating 'grunt-new-srvtab'....
```

Esta orden solo genera un fichero temporal al que tendrá que cambiar el nombre a `srvtab` para que todos los servidores puedan recogerlo. Utilice mv(1) para moverlo al lugar correcto en el sistema original:

```
# mv grunt-new-srvtab srvtab
```

Si el fichero es para un sistema cliente y la red no puede considerarse segura copie el `cliente-new-srvtab` en un medio extraíble y transpórtelo por medios físicos seguros. Asegúrese de cambiar su nombre a `srvtab` en el directorio `/etc/kerberosIV` del cliente, y asegúrese también de que tiene modo 600:

```
# mv grumble-new-srvtab srvtab
# chmod 600 srvtab
```

14.7.5. Añadir entradas a la base de datos

Ahora tenemos que añadir entradas de usuarios a la base de datos. Primero vamos a crear una entrada para el usuario `jane`. Para ello usaremos `kdb_edit` :

```
# kdb_edit
Opening database...

Enter Kerberos master key:

Current Kerberos master key version is 1.

Master key entered.  BEWARE!
Previous or default values are in [brackets] ,
enter return to leave the same, or new value.

Principal name:  jane
Instance:

<Not found>, Create [y] ?  y

Principal: jane, Instance: , kdc_key_ver: 1
New Password:                <---- introduzca una constraseña segura
Verifying password

New Password:                <---- introduzca de nuevo la contraseña
Principal's new key version = 1
Expiration date (enter yyyy-mm-dd) [ 2000-01-01 -] ?
Max ticket lifetime (*5 minutes) [ 255 -] ?
```

```
Attributes [ 0 -] ?
Edit O.K.
Principal name:      <---- si introduce datos nulos saldrá del programa
```

14.7.6. Prueba del sistema

Primero tenemos que iniciar los dæmons de Kerberos. Tenga en cuenta que si su `/etc/rc.conf` está configurado correctamente el inicio tendrá ligar cuando reinicie el sistema. Esta prueba solo es necesaria en el servidor Kerberos; los clientes Kerberos tomarán lo que necesiten automáticamente desde el directorio `/etc/kerberosIV`.

```
# kerberos &
Kerberos server starting
Sleep forever on error
Log file is /var/log/kerberos.log
Current Kerberos master key version is 1.

Master key entered. BEWARE!

Current Kerberos master key version is 1
Local realm: EJEMPLO.COM
# kadmind -n &
KADM Server KADM0.0A initializing
Please do not use 'kill -9' to kill this job, use a
regular kill instead

Current Kerberos master key version is 1.

Master key entered.  BEWARE!
```

Ahora podemos probar a usar `kinit` para obtener un ticket para el ID `jane` que creamos antes:

```
% kinit jane
MIT Project Athena (grunt.ejemplo.com)
Kerberos Initialization for "jane"
Password:
```

Pruebe a listar los tokens con `klist` para ver si realmente están:

```
% klist
Ticket file:    /tmp/tkt245
Principal:      jane@EJEMPLO.COM

  Issued           Expires          Principal
Apr 30 11:23:22  Apr 30 19:23:22  krbtgt.EJEMPLO.COM@EJEMPLO.COM
```

Ahora trate de cambiar la contraseña usando passwd(1) para comprobar si el dæmon kpasswd está autorizado a acceder a la base de datos Kerberos:

```
% passwd
realm EJEMPLO.COM
```

```
Old password for jane:
New Password for jane:
Verifying password
New Password for jane:
Password changed.
```

14.7.7. Añadir privilegios de su

Kerberos nos permite dar a *cada* usuario que necesite privilegios de `root` su *propia* contraseña para su(1). Podemos agregar un ID que esté autorizado a ejecutar su(1) `root`. Esto se controla con una instancia de `root` asociada con un usuario. Vamos a crear una entrada `jane.root` en la base de datos, para lo que recurrimos a `kdb_edit`:

```
# kdb_edit
Opening database...

Enter Kerberos master key:

Current Kerberos master key version is 1.

Master key entered.  BEWARE!
Previous or default values are in [brackets] ,
enter return to leave the same, or new value.

Principal name:  jane
Instance:  root

<Not found>, Create [y] ? y

Principal: jane, Instance: root, kdc_key_ver: 1
New Password:                    <---- introduzca una contraseña ↺
SEGURA
Verifying password

New Password:         <---- introduzca otra vez la constraseña

Principal's new key version = 1
Expiration date (enter yyyy-mm-dd) [ 2000-01-01 -] ?
Max ticket lifetime (*5 minutes) [ 255 -] ?   12 <--- Keep this short!
Attributes [ 0 -] ?
Edit O.K.
Principal name:            <---- si introduce datos nulos saldrá del ↺
programa
```

Ahora trate de obtener los tokens para comprobar que todo funciona:

```
# kinit jane.root
MIT Project Athena (grunt.ejemplo.com)
Kerberos Initialization for "jane.root"
Password:
```

Hemos de agregar al usuario al `.klogin` de `root`:

```
# cat /root/.klogin
jane.root@EJEMPLO.COM
```

Ahora trate de hacer su(1):

```
% su
Password:
```

y eche un vistazo a qué tokens tenemos:

```
# klist
Ticket file: /tmp/tkt_root_245
Principal:      jane.root@EJEMPLO.COM

  Issued           Expires          Principal
May  2 20:43:12  May  3 04:43:12  krbtgt.EJEMPLO.COM@EJEMPLO.COM
```

14.7.8. Uso de otras órdenes

En un ejemplo anterior creamos un usuario llamado `jane` con una instancia `root`. Nos basamos en un usuario con el mismo nombre del "principal" (`jane`), el procedimiento por defecto en Kerberos: `<principal>.<instancia>` con la estructura `<nombre de usuario>`. `root` permitirá que `<nombre de usuario>` haga su(1) a `root` si existen las entradas necesarias en el `.klogin` que hay en el directorio home de `root`:

```
# cat /root/.klogin
jane.root@EJEMPLO.COM
```

De la misma manera, si un usuario tiene en su directorio home lo siguiente:

```
% cat ~/.klogin
jane@EJEMPLO.COM
jack@EJEMPLO.COM
```

significa que cualquier usuario del dominio `EJEMPLO.COM` que se identifique como `jane` o como `jack` (vía `kinit`, ver más arriba) podrá acceder a la cuenta de `jane` o a los ficheros de este sistema (`grunt`) vía rlogin(1), rsh(1) o rcp(1).

Veamos por ejemplo cómo `jane` se se identifica en otro sistema mediante Kerberos:

```
% kinit
MIT Project Athena (grunt.ejemplo.com)
Password:
% rlogin grunt
Last login: Mon May  1 21:14:47 from grumble
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
        The Regents of the University of California.   All rights ↺
reserved.
```

```
FreeBSD BUILT-19950429 (GR386) #0: Sat Apr 29 17:50:09 SAT 1995
```

Aquí `jack` se identifica con la cuenta de `jane` en la misma máquina (`jane` tiene configurado su fichero `.klogin` como se ha mostrado antes, y la persona encargada de la administración de Kerberos ha configurado un usuario principal *jack* con una instancia nula):

```
% kinit
% rlogin grunt -l jane
MIT Project Athena (grunt.ejemplo.com)
Password:
Last login: Mon May  1 21:16:55 from grumble
Copyright (c) 1980, 1983, 1986, 1988, 1990, 1991, 1993, 1994
        The Regents of the University of California.   All rights ↺
reserved.
FreeBSD BUILT-19950429 (GR386) #0: Sat Apr 29 17:50:09 SAT 1995
```

14.8. Kerberos5

Texto de Tillman Hodgson.
Basado en un texto de Mark Murray.

Cada versión de FreeBSD posterior a FreeBSD-5.1 incluye soporte solamente para Kerberos5. Por esta razón Kerberos5 es la única versión incluida. Su configuración es similar en muchos aspectos a la de KerberosIV. La siguiente información solo atañe a Kerberos5 en versiones de FreeBSD-5.0 o posteriores. Los usuarios que deséen utilizar KerberosIV pueden instalar el port security/krb4.

Kerberos es un sistema/protocolo agregado para red que permite a los usuarios validar su identidad a través de los servicios de un servidor seguro. Los servicios como login remoto, copia remota, copias de fichero de un sistema a otro y otras tareas generalmente consideradas poco seguras pasan a ser considerablemente seguras y más controlables.

Kerberos puede describirse como un sistema proxy identificador/verificador. También puede describirse como un sistema confiable de autentificación de terceros. Kerberos solamente ofrece una función: la validación segura de usuarios a través de una red. No proporciona funciones de autorización (es decir, lo que los usuarios tienen permitido hacer) o funciones de auditoría (lo que esos usuarios hicieron). Después de que un servidor y un cliente han usado Kerberos para demostrar su identidad pueden también cifrar todas sus sus comunicaciones, asegurando de este modo su intimidad y la integridad de sus datos durante su uso del sistema.

Es, por tanto, altamente recomendable que se use Kerberos *además* de otros métodos de seguridad que ofrezcan servicios de autorización y auditoría.

Puede usar las siguientes instrucciones como una guía para configurar Kerberos tal y como se distribuye en FreeBSD. De todas maneras, debería consultar las páginas de manual adecuadas para tener toda la información.

Vamos a mostrar una instalación Kerberos, para lo cual usaremos los siguientes espacios de nombres:

- El dominio DNS (“zona”) será ejemplo.org.
- El dominio Kerberos (realm) será EJEMPLO.ORG.

Nota

Debe utilizar nombres de dominio reales al configurar Kerberos incluso si pretende ejecutarlo internamente. Esto le evitará problemas de DNS y asegura la interoperación con otros dominios Kerberos.

14.8.1. Historia

Kerberos fué creado en el Massachusetts Institute of Technology (MIT) como una solución a los problemas de seguridad de la red. El protocolo Kerberos utiliza criptografía fuerte para que un cliente pueda demostrar su identidad en un servidor (y viceversa) a través de una conexión de red insegura.

Kerberos es el nombre de un protocolo de autentificación vía red y un adjetivo para describir programas que implementan el programa (Kerberos telnet, por ejemplo, conocido también como el “telnet kerberizado”). La versión actual del protocolo es la 5, descrita en RFC 1510.

Existen diversas implementaciones libres de este protocolo, cubriendo un amplio rango de sistemas operativos. El MIT, donde Kerberos fué desarrollado, continúa desarrollando su propio paquete Kerberos. Suele usarse en los EEUU como producto criptográfico y como tal ha sufrido las regulaciones de exportación de los EEUU. El Kerberos del MIT existe como un port en (security/krb5). Heimdal Kerberos es otra implementación de la versión 5, y fué desarrollada de forma intencionada fuera de los EEUU para sortear las regulaciones de exportación (y por eso puede incluirse en versiones no comerciales de UNIX®). La distribución Heimdal Kerberos está en el port (security/heimdal), y se incluye una instalación mínima en el sistema base de FreeBSD.

Para alcanzar la mayor audiencia estas instrucciones asumen el uso de la distribución Heimdal incluída en FreeBSD.

14.8.2. Configuración de un KDC Heimdal

El centro de distribución de llaves (KDC, Key Distribution Center) es el servicio centralizado de validación que proporciona Kerberos: es el sistema que emite tickets Kerberos. El

KDC goza del estátus de ser considerado como “confiable” por las demás computadoras del dominio Kerberos, y por eso tiene consideraciones de seguridad más elevadas.

Tenga en cuenta que, aunque la ejecución del servidor Kerberos requiere muy pocos recursos, se recomienda el uso de una máquina dedicada a KDC por razones de seguridad.

Para empezar a configurar un KDC asegúrese de que su `/etc/rc.conf` contenga la configuración adecuada para actuar como KDC (tal vez deba ajustar algunas rutas para que cuadren con su sistema):

```
kerberos5_server_enable="YES"
kadmind5_server_enable="YES"
kerberos_stash="YES"
```

Nota

`kerberos_stash` solo existe en FreeBSD 4.X.

A continuación configuraremos el fichero de configuración de Kerberos, `/etc/krb5.conf`:

```
[libdefaults]
    default_realm = EJEMPLO.ORG [realms]
    EXAMPLE.ORG = {
        kdc = kerberos.ejemplo.org
        admin_server = kerberos.ejemplo.org
    }
[domain_realm]
    .ejemplo.org = EJEMPLO.ORG
```

Tenga en cuenta que este `/etc/krb5.conf` implica que su KDC tendrá el nombre de equipo completo calificado de `kerberos.ejemplo.org`. Necesitará añadir una entrada CNAME (alias) a su fichero de zona si su KDC tiene un nombre de equipo diferente.

Nota

En grandes redes con un servidor DNS BIND bien configurado, el ejemplo de arriba puede quedar del siguiente modo:

```
[libdefaults]
      default_realm = EJEMPLO.ORG
```

Con las siguientes líneas agregadas al fichero de zona `ejemplo.org`:

```
_kerberos._udp      IN  SRV     01 00 88 kerberos.↺
ejemplo.org.
_kerberos._tcp      IN  SRV     01 00 88 kerberos.↺
ejemplo.org.
_kpasswd._udp       IN  SRV     01 00 464 kerberos.↺
ejemplo.org.
_kerberos-adm._tcp  IN  SRV     01 00 749 kerberos.↺
ejemplo.org.
_kerberos           IN  TXT     EJEMPLO.ORG
```

Nota

Para que los clientes sean capaces de encontrar los servicios Kerberos *debe* tener ya sea un `/etc/krb5.conf` configurado o un `/etc/krb5.conf` configurado de forma mínima *y* un servidor DNS configurado correctamente.

A continuación crearemos la base de datos Kerberos. Esta base de datos contiene las llaves de todos los principales cifradas con una contraseña maestra. No es necesario que recuerde esta contraseña, pues se almacenará en `/var/heimdal/m-key` . Para crear la llave maestra ejecute `kstash` e introduzca una contraseña.

Una vez que se ha creado la llave maestra puede inicializar la base de datos usando el programa `kadmin` con la opción `-l` (que significa "local"). Esta opción le dice a `kadmin` que modifique los ficheros de la base de datos directamente en lugar de ir a través del servicio de red `kadmind`. Esto gestiona el problema del huevo y la gallina de tratar de conectar a la base de datos antes de que ésta exista. Una vez que tiene el "prompt" de `kadmin`, utilice `init` para crear su base de datos de dominios iniciales.

Para terminar, mientras está todavía en `kadmin` puede crear su primer principal mediante `add`. Utilice las opciones por defecto por ahora, más tarde puede cambiarlas mediante `modify`. Recuerde que puede usar `?` en cualquier "prompt" para consultar las opciones disponibles.

Veamos un ejemplo de sesión de creación de una base de datos:

```
# kstash
Master key: xxxxxxxx
Verifying password - Master key: xxxxxxxx

# kadmin -l
kadmin> init EJEMPLO.ORG
Realm max ticket life [unlimited]:
kadmin> add tillman
```

```
Max ticket life [unlimited]:
Max renewable life [unlimited]:
Attributes []:
Password: xxxxxxxx
Verifying password - Password: xxxxxxxx
```

Ahora puede arrancar los servicios KDC. Ejecute `/etc/rc.d/kerberos start` y `/etc/rc.d/kadmind start` para levantar dichos servicios. Recuerde que no tendrá ningún dæmon kerberizado ejecutándose pero debe poder confirmar que KDC funciona por el procedimiento de obtener y listar un boleto del principal (usuario) que acaba de crear en la línea de órdenes de KDC:

```
% k5init tillman
tillman@EJEMPLO.ORG's Password:

% k5list
Credentials cache: FILE:/tmp/krb5cc_500
 Principal: tillman@EJEMPLO.ORG

  Issued           Expires          Principal
Aug 27 15:37:58  Aug 28 01:37:58  krbtgt/EJEMPLO.ORG@EJEMPLO.ORG
```

14.8.3. Creación de un servidor Kerberos con servicios Heimdal

Antes de nada necesitaremos una copia del fichero de configuración de Kerberos, `/etc/krb5.conf`. Cópielo al cliente desde KDC de forma segura (mediante scp(1), o usando un disquete).

A continuación necesitará un fichero `/etc/krb5.keytab`. Esta es la mayor diferencia entre un servidor que proporciona dæmons habilitados con Kerberos y una estación de trabajo: el servidor debe tener un fichero `keytab`. Dicho fichero contiene las llaves de equipo del servidor, las cuales le permiten a él y al KDC verificar la identidad entre ellos. Deben transmitirse al servidor de forma segura ya que la seguridad del servidor puede verse comprometida si la llave se hace pública. Por decirlo más claro, transferirla como texto plano a través de la red (por ejemplo por FTP) es una *pésima idea*.

Lo normal es que transmita su `keytab` al servidor mediante el programa `kadmin`. Esto es práctico porque también debe crear el principal del equipo (el KDC que aparece al final de `krb5.keytab`) usando `kadmin`.

Tenga en cuenta que ya debe disponer de un ticket, y que este ticket debe poder usar el interfaz `kadmin` en `kadmind.acl`. Consulte la sección "administración remota" en la página info de Heimdal (`info heimdal`), donde se exponen los detalles de diseño de las listas de control de acceso. Si no quiere habilitar acceso remoto `kadmin`, puede conectarse de forma segura al KDC (por medio de consola local, ssh(1) o telnet(1) Kerberos) y administrar en local mediante `kadmin -l`.

Después de instalar el fichero `/etc/krb5.conf` puede usar `kadmin` desde el servidor Kerberos. `add --random-key` le permitirá añadir el principal del equipo servidor, y `ext` le permitirá extraer el principal del equipo servidor a su propio keybat. Por ejemplo:

```
# kadmin
kadmin> add --random-key host/myserver.ejemplo.org
Max ticket life [unlimited]:
Max renewable life [unlimited]:
Attributes []:
kadmin> ext host/miservidor.ejemplo.org
kadmin> exit
```

Tenga en cuenta que `ext` (contracción de "extract") almacena la llave extraída por defecto en en `/etc/krb5.keytab` .

Si no tiene `kadmind` ejecutándose en KDC (posiblemente por razones de seguridad) y por lo tanto carece de acceso remoto a `kadmin` puede añadir el principal de equipo (`host/miservidor.EJEMPLO.ORG`) directamente en el KDC y entonces extraerlo a un fichero temporal (para evitar sobreescribir `/etc/krb5.keytab` en el KDC) mediante algo parecido a esto:

```
# kadmin
kadmin> ext --keytab=/tmp/ejemplo.keytab host/miservidor.ejemplo.org
kadmin> exit
```

Puede entonces copiar de forma segura el keytab al servidor (usando `scp` o un diquete). Asegúrese de usar un nombre de keytab diferente para evitar sobreescribir el keytab en el KDC.

Su servidor puede comunicarse con el KDC (gracias a su fichero `krb5.conf`) y puede probar su propia identidad (gracias al fichero `krb5.keytab`). Ahora está listo para que usted habilite algunos servicios Kerberos. En este ejemplo habilitaremos el servicio `telnet` poniendo una línea como esta en su `/etc/inetd.conf` y reiniciando el servicio inetd(8) con `/etc/rc.d/inetd restart` :

```
telnet    stream  tcp     nowait  root    /usr/libexec/telnetd  ↺
telnetd -a user
```

La parte crítica es `-a`, de autentificación de usuario. Consulte la página de manual telnetd(8) para más información.

14.8.4. Kerberos con un cliente Heimdal

Configurar una computadora cliente es extremadamente fácil. Lo único que necesita es el fichero de configuración de Kerberos que encontrará en `/etc/krb5.conf` . Simplemente cópielo de forma segura a la computadora cliente desde el KDC.

Pruebe su computadora cliente mediante `kinit`, `klist`, y `kdestroy` desde el cliente para obtener, mostrar y luego borrar un ticket para el principal que creó antes. Debería poder

usar aplicaciones Kerberos para conectarse a servidores habilitados con Kerberos, aunque si no funciona y tiene problemas al intentar obtener el boleto lo más probable es que el problema esté en el servidor y no en el cliente o el KDC.

Al probar una aplicación como `telnet`, trate de usar un "sniffer" de paquetes (como tcpdump(1)) para confirmar que su contraseña no viaja en claro por la red. Trate de usar `telnet` con la opción `-x`, que cifra el flujo de datos por entero (algo parecido a lo que hace `ssh`).

Las aplicaciones clientes Kerberos principales (llamadas tradicionalmente `kinit`, `klist`, `kdestroy` y `kpasswd`) están incluidas en la instalación base de FreeBSD. Tenga en cuenta que en las versiones de FreeBSD anteriores a 5.0 reciben los nombres de `k5init` , `k5list` , `k5destroy` , `k5passwd` y `k5stash` .

También se instalan por defecto diversas aplicaciones Kerberos que no entran dentro de la categoría de "imprescindibles". Es aquí donde la naturaleza "mínima" de la instalación base de Heimdal salta a la palestra: `telnet` es el único servicio Kerberos habilitado.

El port Heimdal añade algunas de las aplicaciones cliente que faltan: versiones Kerberos de `ftp`, `rsh`, `rcp`, `rlogin` y algunos otros programas menos comunes. El port del MIT también contiene una suite completa de aplicaciones cliente de Kerberos.

14.8.5. Ficheros de configuración de usuario: `.k5login` y `.k5users`

Suele ser habitual que los usuarios de un dominio Kerberos (o "principales") tengan su usuario (por ejemplo `tillman@EJEMPLO.ORG`) mapeado a una cuenta de usuario local (por ejemplo un usuario llamado llamado `tillman`). Las aplicaciones cliente como `telnet` normalmente no requieren un nombre de usuario o un principal.

Es posible que de vez en cuando quiera dar acceso a una una cuenta de usuario local a alguien que no tiene un principal Kerberos. Por ejemplo, `tillman@EJEMPLO.ORG` puede necesitar acceso a la cuenta de usuario local `webdevelopers`. Otros principales tal vez necesiten acceso a esas cuentas locales.

Los ficheros `.k5login` y `.k5users` , ubicados en el directorio home del usuario, pueden usarse de un modo similar a una combinación potente de `.hosts` y `.rhosts` . Por ejemplo, si pusiera un fichero `.k5login` con el siguiente contenido

```
tillman@example.org
jdoe@example.org
```

en el directorio home del usuario local `webdevelopers` ambos principales listados tendrían acceso a esa cuenta sin requerir una contraseña compartida.

Le recomendamos encarecidamente la lectura de las páginas de manual de estas órdenes. Recuerde que la página de manual de `ksu` abarca `.k5users` .

14.8.6. Kerberos Sugerencias, trucos y solución de problemas

- Tanto si utiliza el port de Heimdal o el Kerberos del MIT asegúrese de que su variable de entorno `PATH` liste las versiones de Kerberos de las aplicaciones cliente antes que las versiones del sistema.

- ?Todas las computadoras de su dominio Kerberos tienen la hora sincronizada? Si no, la autentificación puede fallar. Sección 29.12, “NTP” describe como sincronizar los relojes utilizando NTP.

- MIT y Heimdal conviven bien, con la excepción de `kadmin`, protocolo no está estandarizado.

- Si cambia su nombre de equipo debe cambiar también el “apellido” de su principal y actualizar su keytab. Esto también se aplica a entradas especiales en keytab como el principal `www/` que usa el www/mod_auth_kerb de Apache.

- Todos los equipos en su dominio Kerberos deben poder resolverse (tanto en la forma normal normal como en la inversa) en el DNS (o en `/etc/hosts` como mínimo). Los CNAME funcionarán, pero los registros A y PTR deben ser correctos y estar en su sitio. El mensaje de error que recibirá de no hacerlo así no es muy intuitivo: Kerberos5 refuses authentication because Read req failed: Key table entry not found.

- Algunos sistemas operativos que puede usar como clientes de su KDC no activan los permisos para `ksu` como setuid `root`. Esto hará que `ksu` no funcione, lo cual es muy seguro pero un tanto molesto. Tenga en cuenta que no se debe a un error de KDC.

- Si desea permitir que un principal tenga un ticket con una validez más larga que el valor por defecto de diez horas en Kerberos del MIT debe usar `modify_principal` en `kadmin` para cambiar “maxlife” tanto del principal en cuestión como del `krbtgt` del principal. Hecho esto, el principal puede utilizar la opción `-l` con `kinit` para solicitar un boleto con más tiempo de vida.

-

Nota

Si ejecuta un “sniffer” de paquetes en su KDC para ayudar con la resolución de problemas y ejecuta `kinit` desde una estación de trabajo puede encontrarse con que su TGT se envía inmediatamente después de ejecutar `kinit`: *incluso antes de que escriba su contraseña* La explicación es que el servidor Kerberos transmite tranquilamente un TGT (Ticket Granting Ticket) a cualquier petición no autorizada; de todas maneras, cada TGT está cifrado en una llave derivada de la contraseña del usuario. Por tanto, cuan-

> do un usuario teclea su contraseña no la está enviando al KDC, se está usando para descifrar el TGT que `kinit` ya obtuvo. Si el proceso de descifrado termina en un ticket válido con una marca de tiempo válida, el usuario tiene credenciales Kerberos válidas. Estas credenciales incluyen una llave de sesión para establecer comunicaciones seguras con el servidor Kerberos en el futuro, así como el TGT en sí, que se cifra con la llave del propio servidor Kerberos. Esta segunda capa de cifrado es invisible para el usuario, pero es lo que permite al servidor Kerberos verificar la autenticidad de cada TGT.

- Si desea utilizar tickets con un tiempo largo de vida (una semana, por ejemplo) y está utilizando OpenSSH para conectarse a la máquina donde se almacena su boleto asgúrese de que Kerberos `TicketCleanup` esté configurado a `no` en su `sshd_config` o de lo contrario sus tickets serán eliminados cuando termine la sesión.

- Recuerde que los principales de equipos también pueden tener tener un tiempo de vida más largo. Si su principal de usuario tiene un tiempo de vida de una semana pero el equipo al que se conecta tiene un tiempo de vida de nueve horas, tendrá un principal de equipo expirado en su caché, y la caché de ticket no funcionará como esperaba.

- Cuando esté configurando un fichero `krb5.dict` pensando específicamente en prevenir el uso de contraseñas defectuosas (la página de manual de de `kadmind` trata el tema brevemente), recuerde que solamente se aplica a principales que tienen una política de contraseñas asignada. El formato de los ficheros `krb5.dict` es simple: una cadena de texto por línea. Puede serle útil crear un enlace simbólico a `/usr/share/dict/words` .

14.8.7. Diferencias con el port del MIT

Las diferencias más grandes entre las instalaciones MIT y Heimdal están relacionadas con `kadmin`, que tiene un conjunto diferente (pero equivalente) de órdenes y utiliza un protocolo diferente. Esto tiene implicaciones muy grandes si su KDC es MIT, ya que no podrá utilizar el programa `kadmin` de Heimdal para administrar remotamente su KDC (o viceversa).

Las aplicaciones cliente pueden también disponer de diferentes opciones de línea de órdenes para lograr lo mismo. Le recomendamos seguir las instrucciones de la página web de Kerberos del MIT (`http://web.mit.edu/Kerberos/www/`). Sea cuidadoso con los parches: el port del MIT se instala por defecto en `/usr/local/` , y las aplicaciones “normales” del sistema pueden ser ejecutadas en lugar de las del MIT si su variable de entorno `PATH` lista antes los directorios del sistema.

Nota

Si usa el port del MIT security/krb5 proporcionado por FreeBSD asegúrese de leer el fichero `/usr/local/share/doc/krb5/README.FreeBSD` instalado por el port si quiere entender por qué los login vía `telnetd` y `klogind` se comportan de un modo un tanto extraño. Más importante aún, corregir la conducta de "permisos incorrectos en el fichero caché" requiere que el binario `login.krb5` se use para la validación para que pueda cambiar correctamente los permisos de propiedad de credenciales reenviadas.

14.8.8. Mitigación de limitaciones encontradas en Kerberos

14.8.8.1. Kerberos es un enfoque "todo o nada"

Cada servicio habilitado en la red debe modificarse para funcionar con Kerberos (o debe ser asegurado contra ataques de red) o de lo contrario las credenciales de usuario pueden robarse y reutilizarse. Un ejemplo de esto podría ser que Kerberos habilite todos los shells remotos (vía `rsh` y `telnet`, por ejemplo) pero que no cubra el servidor de correo POP3, que envía contraseñas en texto plano.

14.8.8.2. Kerberos está pensado para estaciones de trabajo monousuario

En un entorno multiusuario Kerberos es menos seguro. Esto se debe a que almacena los tickets en el directorio `/tmp`, que puede ser leído por todos los usuarios. Si un usuario está compartiendo una computadora con varias personas (esto es, si utiliza un sistema multiusuario) es posible que los tickets sean robados (copiados) por otro usuario.

Esto puede solventarse con la opción de línea de órdenes `-c` nombre-de-fichero o (mejor aún) la variable de entorno `KRB5CCNAME` , pero raramente se hace. Si almacena los tickets en el directorio home de los usuarios y utiliza sin mucha complicación los permisos de fichero puede mitigar este problema.

14.8.8.3. El KDC es el punto crítico de fallo

Por motivos de diseño el KDC es tan seguro como la base de datos principal de contraseñas que contiene. El KDC no debe ejecutar ningún otro servicio ejecutándose en él y debe ser físicamente seguro. El peligro es grande debido a que Kerberos almacena todas las contraseñas cifradas con la misma llave (la llave "maestra", que a su vez se guarda como un fichero en el KDC).

De todos modos una llave maestra comprometida no es algo tan terrible como parece a primera vista. La llave maestra solo se usa para cifrar la base de datos Kerberos y como semilla para el generador de números aleatorios. Mientras sea seguro el acceso a su KDC un atancante no puede hacer demasiado con la llave maestra.

Además, si el KDC no está disponible (quizás debido a un ataque de denegación de servicio o problemas de red) no se podrán utilizar los servicios de red ya que no se puede efectuar la validación, lo que hace que esta sea una buena forma de lanzar un ataque de denegación de servicio. Este problema puede aliviarse con múltiples KDCs (un maestro y uno o más esclavos) y con una implementación cautelosa de secundarios o autentificación de respaldo (para esto PAM es excelente).

14.8.8.4. Limitaciones de Kerberos

Kerberos le permite a usuarios, equipos y servicios validarse entre sí, pero no dispone de ningún mecanismo para autentificar el KDC a los usuarios, equipos o servicios. Esto significa que una versión (por ejemplo) "troyanizada" `kinit` puede grabar todos los usuarios y sus contraseñas. Puede usar security/tripwire o alguna otra herramienta de revisión de integridad de sistemas de ficheros para intentar evitar problemas como este.

14.8.9. Recursos y más información

- Las preguntas frecuentes (FAQ) de Kerberos
- Designing an Authentication System: a Dialog in Four Scenes
- RFC 1510, The Kerberos Network Authentication Service (V5)
- Página web de Kerberos del MIT
- Página web de Kerberos Heimdal

14.9. OpenSSL

Escrito por: Tom Rhodes.

El conjunto de herramientas OpenSSL es una característica de FreeBSD que muchos usuarios pasan por alto. OpenSSL ofrece una capa de cifrada de transporte sobre la capa normal de comunicación, permitiendo la combinación con con muchas aplicaciones y servicios de red.

Algunos usos de OpenSSL son la validación cifrada de clientes de correo, las transacciones basadas en web como pagos con tarjetas de crédito, etc. Muchos ports, como www/apache13-ssl y mail/sylpheed-claws ofrecen soporte de compilación para OpenSSL.

Nota

En la mayoría de los casos la colección de ports tratará de compilar el port security/openssl a menos que la variable de make `WITH_OPENSSL_BASE` sea puesta explícitamente a “yes”.

La versión de OpenSSL incluida en FreeBSD soporta los protocolos de seguridad de red Secure Sockets Layer v2/v3 (SSLv2/SSLv3) y Transport Layer Security v1 (TLSv1) y puede utilizarse como biblioteca criptográfica general.

Nota

OpenSSL soporta el algoritmo IDEA pero estáa deshabilitado por defecto debido a patentes en vigor en los Estados Unidos. Si quiere usarlo debe revisar la licencia, y si las restricciones le parecen aceptables active la variable `MAKE_IDEA` en `make.conf`.

Uno de los usos más comunes de OpenSSL es ofrecer certificados para usar con aplicaciones de software. Estos certificados aseguran que las credenciales de la compañia o individuo son válidos y no son fraudulentos. Si el certificado en cuestión no ha sido verificado por uno de las diversas “autoridades certificadoras” o CA, suele generarse una advertencia al respecto. Una autoridad de certificados es una compañia, como VeriSign, que firma certificados para validar credenciales de individuos o compañias. Este proceso tiene un costo asociado y no es un requisito imprescindible para usar certificados, aunque puede darle un poco de tranquilidad a los usuarios más paranóicos.

14.9.1. Generación de certificados

Para generar un certificado ejecute lo siguiente:

```
# openssl req -new -nodes -out req.pem -keyout cert.pem
Generating a 1024 bit RSA private key
..............++++++
.....................................++++++
writing new private key to 'cert.pem'
-----
You are about to be asked to enter information that will be ↺
incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name ↺
or a DN.
```

```
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:PA
Locality Name (eg, city) []:Pittsburgh
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Mi compañía
Organizational Unit Name (eg, section) []:Administrador de sistemas
Common Name (eg, YOUR name) []:localhost.ejemplo.org
Email Address []:trhodes@FreeBSD.org

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:UNA CONTRASEÑA
An optional company name []:Otro nombre
```

Tenga en cuenta que la respuesta directamente después de "prompt" "Common Name" muestra un nombre de dominio. Este "prompt" requiere que se introduzca un nombre de servidor para usarlo en la verificación; si escribe cualquier otra cosa producirá un certificado inválido. Otras opciones, por ejemplo el tiempo de expiración, alternan algoritmos de cifrado, etc. Puede ver una lista completa en la página de manual de openssl(1).

Debería tener dos ficheros en el directorio donde ha ejecutado la orden anterior. La petición de certificado, `req.pem`, es lo que debe enviar a una autoridad certificadora para que valide las credenciales que introdujo; firmará la petición y le devolverá el certificado. El segundo fichero es `cert.pem` y es la llave privada para el certificado, que debe proteger a toda costa; si cae en malas manos podrí usarse para suplantarle a usted o a sus servidores.

Si no necesita la firma de una CA puede crear y firmar usted mismo su certificado. Primero, genere la llave RSA:

```
# openssl dsaparam -rand -genkey -out myRSA.key 1024
```

A continuación genere la llave CA:

```
# openssl gendsa -des3 -out myca.key myRSA.key
```

Utilice esta llave para crear el certificado:

```
# openssl req -new -x509 -days 365 -key myca.key -out new.crt
```

Deberín aparecer dos nuevos ficheros en su directorio: un fichero de firma de autoridad de certificados (`myca.key`) y el certificado en sí, `new.crt`. Deben ubicarse en un directorio, que se recomienda que sea `/etc`, que es legible solo para `root`. Para terminar, es recomendable asignar permisos 0700 para el fichero con `chmod`.

14.9.2. Uso de certificados; un ejemplo

?Qué pueden hacer estos ficheros? Cifrar conexiones al MTA Sendmail es un buen sitio para usarlos. De este modo eliminará el uso de validación mediante texto en claro para los usuarios que envían correo a través del MTA local.

Nota

No es el mejor uso en el mundo, ya que algunos MUAs enviarán al usuario un mensaje de error si no tiene instalados localmente los certificados. Consulte la documentación para más datos sobre la instalación de certificados.

Debe añadir las siguientes líneas en su fichero local `.mc`:

```
dnl SSL Options
define(`confCACERT_PATH',`/etc/certs')dnl
define(`confCACERT',`/etc/certs/new.crt')dnl
define(`confSERVER_CERT',`/etc/certs/new.crt')dnl
define(`confSERVER_KEY',`/etc/certs/myca.key')dnl
define(`confTLS_SRV_OPTIONS', `V')dnl
```

`/etc/certs/` es el directorio destinado a almacenamiento de los ficheros de certificado y llave en local. El último requisito es una reconstrucción del fichero `.cf` local. Solo tiene que teclear `make` *`install`* en el directorio `/etc/mail`. A continuación ejecute un `make` *`restart`*, que debería reiniciar el dæmon Sendmail.

Si todo fué bien no habrá mensajes de error en el fichero `/var/log/maillog` y Sendmail aparecerá en la lista de procesos.

Puede probarlo todo de una forma muy sencilla; conéctese al servidor de correo mediante telnet(1):

```
# telnet ejemplo.com 25
Trying 192.0.34.166...
Connected to ejemplo.com .
Escape character is '^]'.
220 ejemplo.com  ESMTP Sendmail 8.12.10/8.12.10; Tue, 31 Aug 2004 ↺
03:41:22 -0400 (EDT)
ehlo ejemplo.com
250-ejemplo.com Hello ejemplo.com [192.0.34.166], pleased to meet you
250-ENHANCEDSTATUSCODES
250-PIPELINING
250-8BITMIME
250-SIZE
250-DSN
250-ETRN
```

```
250-AUTH LOGIN PLAIN
250-STARTTLS
250-DELIVERBY
250 HELP
quit
221 2.0.0 ejemplo.com  closing connection
Connection closed by foreign host.
```

Si la línea “STARTTLS” aparece en la salida, todo está funcionando correctamente.

14.10. VPN sobre IPsec

Escrito por Nik Clayton.

Creación de una VPN entre dos redes, a través de Internet, mediante puertas de enlace (“gateways”) FreeBSD.

14.10.1. Qué es IPsec

Escrito por Hiten M. Pandya.

Esta sección le guiará a través del proceso de configuración de IPsec, y de su uso en un entorno consistente en máquinas FreeBSD y Microsoft® Windows® 2000/XP, para hacer que se comuniquen de manera segura. Para configurar IPsec es necesario que esté familiarizado con los conceptos de construcción de un kernel personalizado (consulte el Capítulo 8, *Configuración del kernel de FreeBSD*).

IPsec es un protocolo que está sobre la capa del protocolo de Internet (IP). Le permite a dos o más equipos comunicarse de forma segura (de ahí el nombre). La “pila de red” IPsec de FreeBSD se basa en la implementación KAME, que incluye soporte para las dos familias de protocolos, IPv4 e IPv6.

Nota

FreeBSD 5.X contiene una pila IPsec “acelerada por hardware”, conocida como “Fast IPsec”, que fué obtenida de OpenBSD. Emplea hardware criptográfico (cuando es posible) a través del subsistema crypto(4) para optimizar el rendimiento de IPsec. Este subsistema es nuevo, y no soporta todas las opciones disponibles en la versión KAME de IPsec. Para poder habilitar IPsec acelerado por hardware debe añadir las siguientes opciones al fichero de configuración de su kernel:

```
options   FAST_IPSEC  # new IPsec (cannot define ↺
w/ IPSEC)
```

Tenga en cuenta que no es posible utilizar el subsistema "Fast IPsec" y la implementación KAME de IPsec en la misma computadora. Consulte la página de manual fast_ipsec(4) para más información.

IPsec consta de dos sub-protocolos:

- *Encapsulated Security Payload (ESP)*, que protege los datos del paquete IP de interferencias de terceros, cifrando el contenido utilizando algoritmos de criptografía simétrica (como Blowfish, 3DES).
- *Authentication Header (AH)*, que protege la cabecera del paquete IP de interferencias de terceros así como contra la falsificación ("spoofing"), calculando una suma de comprobación criptográfica y aplicando a los campos de cabecera IP una función hash segura. Detrás de todo esto va una cabecera adicional que contiene el hash para permitir la validación de la información que contiene el paquete.

ESP y AH pueden utilizarse conjunta o separadamente, dependiendo del entorno.

IPsec puede utilizarse para cifrar directamente el tráfico entre dos equipos (conocido como *modo de transporte*) o para construir "túneles virtuales" entre dos subredes, que pueden usarse para comunicación segura entre dos redes corporativas (conocido como *modo de túnel*). Este último es muy conocido como una *red privada virtual (Virtual Private Network, o VPN)*. ipsec(4) contiene información detallada sobre el subsistema IPsec de FreeBSD.

Si quiere añdir soporte IPsec a su kernel debe incluir las siguientes opciones al fichero de configuración de su kernel:

```
options   IPSEC         #IP security
options   IPSEC_ESP     #IP security (crypto; define w/ IPSEC)
```

Si quiere soporte para la depuración de errores no olvide la siguiente opción:

```
options   IPSEC_DEBUG  #debug for IP security
```

14.10.2. El Problema

No existe un estándar para lo que constituye una VPN. Las VPN pueden implementarse utilizando numerosas tecnologías diferentes, cada una de las cuales tiene sus pros y sus

contras. Esta sección presenta un escenario, y las estrategias usadas para implementar una VPN para este escenario.

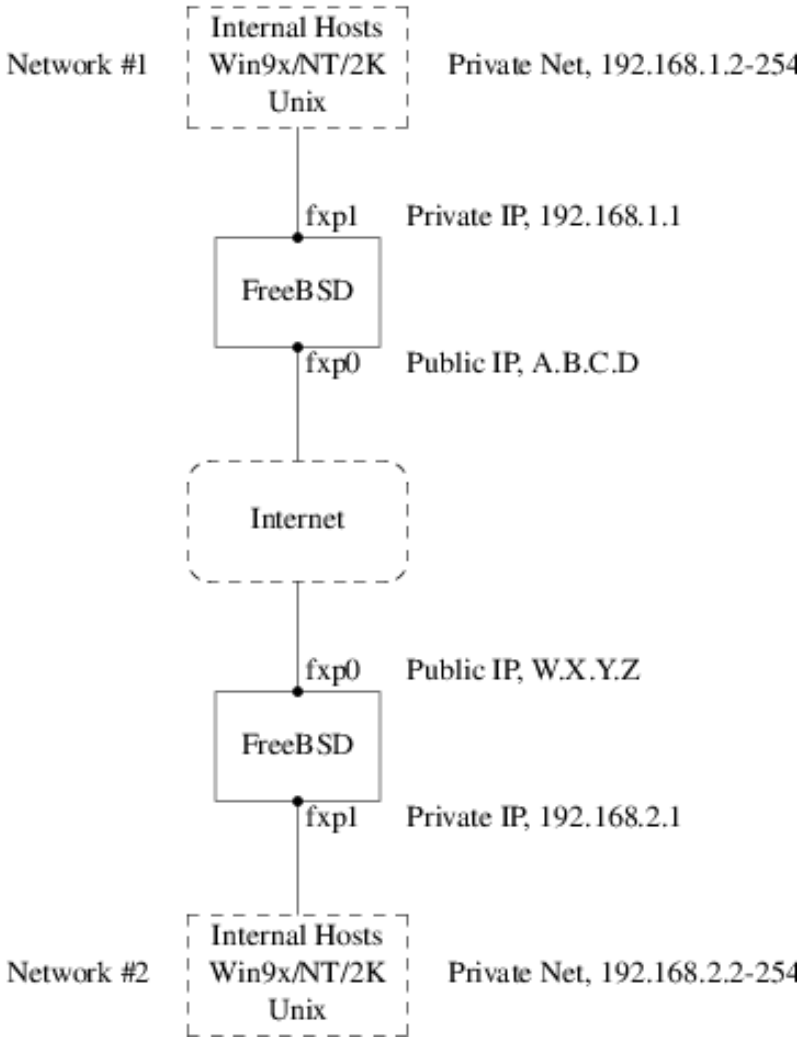
14.10.3. El escenario: dos redes, conectadas por Internet, que queremos que se comporten como una sola

Este es el punto de partida:

- Usted tiene al menos dos sitios
- Ambos sitios utilizan IP internamente
- Ambos sitios están conectados a Internet, a través de una puerta de enlace FreeBSD.
- La puerta de enlace de cada red tiene al menos una dirección IP pública.
- Las direcciones internas de las dos redes pueden ser direcciones IP públicas o privadas, no importa. Puede ejecutar NAT en la máquina que hace de puerta de enlace si es necesario.
- Las direcciones IP internas de las dos redes *no colisionan*. Aunque espero que sea teóricamente posible utilizar una combinación de tecnología VPN y NAT para hacer funcionar todo esto sospecho que configurarlo sería una pesadilla.

Si lo que intenta es conectar dos redes y ambas usan el mismo rango de direcciones IP privadas (por ejemplo las dos usan `192.168.1.x`)debería renumerar una de las dos redes.

La topología de red se parecería a esto:



Observe las dos direcciones IP públicas. Usaré letras para referirme a ellas en el resto de este artículo. El cualquier lugar que vea esas letras en este artículo reemplácelas con su propia dirección IP pública. Observe también que internamente las dos máquinas que hacen de puerta de enlace tienen la dirección IP .1, y que las dos redes tienen direcciones IP privadas diferentes (`192.168.1.x` y `192.168.2.x` respectivamente). Todas las máquinas de las redes privadas están configuradas para utilizar la máquina `.1` como su puerta de enlace por defecto.

La intención es que, desde el punto de vista de la red, cada red debe ver las máquinas en la otra red como si estuvieran directamente conectadas al mismo router (aunque aunque sea un router ligeramente lento con una tendencia ocasional a tirar paquetes).

Esto significa que (por ejemplo), la máquina `192.168.1.20` debe ser capaz de ejecutar

```
ping 192.168.2.34
```

y recibir de forma transparente una respuesta. Las máquinas Windows® deben ser capaces de ver las máquinas de la otra red, acceder a sus ficheros compartidos, etc, exactamente igual que cuando acceden a las máquinas de la red local.

Y todo debe hacerse de forma segura. Esto significa que el tráfico entre las dos redes tiene que ser cifrado.

La creación de una VPN entre estas dos redes es un proceso que requiere varios pasos. Las etapas son estas:

1. Crear un enlace de red "virtual" entre las dos redes, a través de Internet. Probarlo usando herramientas como ping(8) para asegurarse de que funcione.

2. Aplicar políticas de seguridad para asegurarse de que el tráfico entre las dos redes sea cifrado y descifrado de forma transparente. Comprobarlo mediante herramientas como tcpdump(1) para asegurarse de que el tráfico esté siendo efectivamente cifrado.

3. Configurar software adicional en las puertas de enlace FreeBSD para permitir a las máquinas Windows® verse entre ellas a través de la VPN.

14.10.3.1. Paso 1: Creación y prueba de un enlace de red "virtual"

Suponga que está en la puerta de enlace de la red red #1 (con dirección IP pública `A.B.C.D`, dirección IP privada `192.168.1.1`), y ejecuta `ping 192.168.2.1`, que es la dirección privada de la máquina con dirección IP `W.X.Y.Z`. ?Qué hace falta para esto?

1. La puerta de enlace necesita saber cómo alcanzar a `192.168.2.1`. En otras palabras, necesita tener una ruta hasta `192.168.2.1`.

2. Las direcciones IP privadas, como las que están en el rango `192.168.x` no deberían aparecer en Internet. Por eso, cada paquete que mande a `192.168.2.1` necesitará encerrarse dentro de otro paquete. Este paquete debe tener todas las características de haber sido enviado desde `A.B.C.D`, y tendrá que ser enviado a `W.X.Y.Z`. Este proceso recibe el nombre de *encapsulado*.

3. Una vez que este paquete llega a `W.X.Y.Z` necesitará ser "desencapsulado", y entregado a `192.168.2.1`.

Puede verlo como si necesitara un "túnel" entre las dos redes. Las dos "bocas del túnel" son las direcciones IP `A.B.C.D` y `W.X.Y.Z`, y debe hacer que el túnel sepa cuáles serán las direcciones IP privadas que tendrán permitido el paso a través de él. El túnel se usa para transferir tráfico con direcciones IP privadas a través de la Internet pública.

Este túnel se crea mediante la interfaz genérica, o dispositivo `gif` en FreeBSD. Como puede imaginarse la interfaz `gif` de cada puerta de enlace debe configurarse con cuatro direcciones IP: dos para las direcciones IP públicas, y dos para las direcciones IP privadas.

El soporte para el dispositivo gif debe compilarse en el kernel de FreeBSD en ambas máquinas añadiendo la línea

```
device gif
```

a los ficheros de configuración del kernel de ambas máquinas, compilarlo, instalarlo y reiniciar.

La configuración del túnel es un proceso que consta de dos partes. Primero se le debe decir al túnel cuáles son las direcciones IP exteriores (o públicas) mediante gifconfig(8). Después configure las direcciones IP con ifconfig(8).

Nota

En FreeBSD 5.X las funciones de gifconfig(8) se han incluido en ifconfig(8).

En la puerta de enlace de la red #1 debe ejecutar las siguientes dos órdenes para configurar el túnel.

```
gifconfig gif0 A.B.C.D W.X.Y.Z
ifconfig gif0 inet 192.168.1.1 192.168.2.1 netmask 0xffffffff
```

En la otra puerta de enlace ejecute las mismas órdenes, pero con el orden las direcciones IP invertido.

```
gifconfig gif0 W.X.Y.Z A.B.C.D
ifconfig gif0 inet 192.168.2.1 192.168.1.1 netmask 0xffffffff
```

Ahora ejecute:

```
gifconfig gif0
```

y podrá ver la configuración. Por ejemplo, en la puerta de enlace de la red #1 vería algo parecido a esto:

```
# gifconfig gif0
gif0: flags=8011<UP,POINTTOPOINT,MULTICAST> mtu 1280
inet 192.168.1.1 --> 192.168.2.1 netmask 0xffffffff
physical address inet A.B.C.D --> W.X.Y.Z
```

Como puede ver se ha creado un túnel entre las direcciones físicas `A.B.C.D` y `W.X.Y.Z`, y el tráfico que puede pasar a través del túnel es entre `192.168.1.1` y `192.168.2.1`.

Esto también habrá agregado una entrada en la tabla de rutas de ambas máquinas, que puede examinar con `netstat -rn`. Esta salida es de la puerta de enlace de la red #1.

```
# netstat -rn
Routing tables

Internet:
Destination      Gateway       Flags    Refs    Use    Netif  Expire
...
192.168.2.1      192.168.1.1   UH        0        0    gif0
...
```

Como el valor de "Flags" lo indica, esta es una ruta de equipo, lo que significa que cada puerta de enlace sabe como alcanzar la otra puerta de enlace, pero no saben cómo llegar al resto de sus respectivas redes. Ese problema se solucionará en breve.

Es posible que disponga de un cortafuegos en ambas máquinas, por lo que tendrá que buscar la forma de que el tráfico de la VPN pueda entrar y salir limpiamente. Puede permitir todo el tráfico de ambas redes, o puede que quiera incluir reglas en el cortafuegos para que protejan ambos extremos de la VPN uno del otro.

Las pruebas se simplifican enormemente si configura el cortafuegos para permitir todo el tráfico a través de la VPN. Siempre puede ajustar las cosas después. Si utiliza ipfw(8) en las puertas de enlace una orden similar a

```
ipfw add 1 allow ip from any to any via gif0
```

permitirá todo el tráfico entre los dos extremos de la VPN, sin afectar al resto de reglas del cortafuegos. Obviamente tendrá que ejecutar esta orden en ambas puertas de enlace.

Esto es suficiente para permitir a cada puerta de enlace hacer un ping entre ellas. En `192.168.1.1` deberí poder ejecutar

```
ping 192.168.2.1
```

y obtener una respuesta; es obvio que debería poder hacer los mismo en la otra puerte de enlace.

Aún no podrá acceder a las máquinas internas de las redes. El problema está en el encaminamiento: aunque las puertas de enlace saben cómo alcanzarse mútuamente no saben cómo llegar a la red que hay detrás de la otra.

Para resolver este problema debe añadir una ruta estática en cada puerta de enlace. La orden en la primera puerta de enlace podría ser:

```
route add 192.168.2.0 192.168.2.1 netmask 0xffffff00
```

Esto significa "Para alcanzar los equipos en la red `192.168.2.0` , envía los paquetes al equipo `192.168.2.1` ". Necesitará ejecutar una orden similar en la otra puerta de enlace, pero obviamente con las direcciones `192.168.1.x` .

El tráfico IP de equipos en una red no será capaz de alcanzar equipos en la otra red.

Ya tiene dos tercios de una VPN, puesto que ya es "virtual" y es una "red". Todavía no es privada. Puede comprobarlo con ping(8) y tcpdump(1). Abra una sesión en la puerta de enlace y ejecute

```
tcpdump dst host 192.168.2.1
```

En otra sesión en el mismo equipo ejecute

```
ping 192.168.2.1
```

Verá algo muy parecido a esto:

```
16:10:24.018080 192.168.1.1 > 192.168.2.1: icmp: echo request
16:10:24.018109 192.168.1.1 > 192.168.2.1: icmp: echo reply
16:10:25.018814 192.168.1.1 > 192.168.2.1: icmp: echo request
16:10:25.018847 192.168.1.1 > 192.168.2.1: icmp: echo reply
16:10:26.028896 192.168.1.1 > 192.168.2.1: icmp: echo request
16:10:26.029112 192.168.1.1 > 192.168.2.1: icmp: echo reply
```

Como puede ver los mensajes ICMP van y vienen sin cifrar. Si usa el parámetro `-s` en tcpdump(1) para tomar más bytes de datos de estos paquetes verá más información.

Obviamente esto es inaceptable. La siguiente sección explicará cómo asegurar el enlace entre las dos redes para que todo el tráfico se cifre automáticamente.

- Configure ambos kernel con "pseudo-device gif".
- Edite `/etc/rc.conf` en la puerta de enlace `#1` y añada las siguientes líneas (reemplazando las direcciones IP según sea necesario).

  ```
  gifconfig_gif0="A.B.C.D W.X.Y.Z"
  ifconfig_gif0="inet 192.168.1.1 192.168.2.1 netmask 0xffffffff"
  static_routes="vpn"
  route_vpn="192.168.2.0 192.168.2.1 netmask 0xffffff00"
  ```

- Edite la configuración de su cortafuegos (`/etc/rc.firewall` , o lo que corresponda) en ambos equipos y añada

```
ipfw add 1 allow ip from any to any via gif0
```

- Haga los cambios oportunos en el `/etc/rc.conf` de la puerta de enlace #2, invirtiendo el orden de las direcciones IP.

14.10.3.2. Paso 2: Asegurar el enlace

Para asegurar el enlace usaremos IPsec. IPsec ofrece un mecanismo para que dos equipos coincidan en una llave de cifrado, y usar esta llave para cifrar los datos entre los dos equipos.

Existen dos áreas de configuración a tener en cuenta:

1. Debe existir un mecanismo para que los dos equipos se pongan de acuerdo en el mecanismo de cifrado que van a utilizar. Una vez que los dos equipos se han puesto de acuerdo dice que existe una "asociación de seguridad" entre ellos.

2. Debe existir un mecanismo para especificar que tráfico debe ser cifrado. Obviamente, usted no querrá cifrar todo su tráfico saliente: solo querrá cifrar el tráfico que es parte de la VPN. Las reglas con las que determinará qué tráfico será cifrado se llaman "políticas de seguridad".

Tanto las asociaciones de seguridad como las políticas de seguridad son responsabilidad del kernel, pero pueden ser modificadas desde el espacio de usuario. Antes de poder hacerlo, tendrá que configurar el kernel para que incluya IPsec y el protocolo ESP (Encapsulated Security Payload). Incluya en el fichero de configuración de su kernel lo siguiente:

```
options IPSEC
options IPSEC_ESP
```

Recompile y resintale su kernel y reinicie. Como se dijo anteriormente, tendrá que hacer lo mismo en el kernel de las dos puertas de enlace.

Tiene dos opciones cuando se trata de configurar asociaciones de seguridad. Puede configurarlas a mano en los dos equipos, lo que significa elegir el algoritmo de cifrado, las llaves de cifrado, etc, o puede utilizar alguno de los dæmons que implementan el protocolo de intercambio de llaves de Internet (IKE, Internet Key Exchange).

Le recomiendo la segunda opción. Aparte de otras consideraciones es más fácil de configurar.

La edición y despliegue se efectúa con setkey(8). Todo esto se entiende mejor con una analogía. `setkey` es a las tablas de políticas de seguridad del kernel lo que route(8) es a las tablas de rutas del kernel. También puede usar `setkey` ver las asociaciones de seguridad en vigor, siguiendo con la analogía, igual que puede usar `netstat -r`.

Existen numerosos dæmons que pueden encargarse de la gestión de asociaciones de seguridad en FreeBSD. En este texto se muestra cómo usar uno de ellos, racoon (que puede instalar desde security/racoon en la colección de ports de FreeBSD.

El software security/racoon debe ejecutarse en las dos puertas de enlace. En cada equipo debe configurar la dirección IP del otro extremo de la VPN y una llave secreta (que usted puede y debe elegir, y debe ser la misma en ambas puertas de enlace).

Los dos dæmons entran en contacto uno con otro, y confirman que son quienes dicen ser (utilizando la llave secreta que usted configuró). Los dæmons generan una nueva llave secreta, y la utilizan para cifrar el tráfico que discurre a través de la VPN. Periódicamente cambian esta llave, para que incluso si un atacante comprometiera una de las llaves (lo cual es teóricamente cercano a imposible) no le serviriía de mucho: para cuando el atacante haya "crackeado" la llave los dæmons ya habrán escogido una nueva.

El fichero de configuración de racoon está en `${PREFIX}/etc/racoon`. No debería tener que hacer demasiados cambios a ese fichero. El otro componente de la configuración de racoon (que *sí* tendrá que modificar) es la "llave pre-compartida".

La configuración por defecto de racoon espera encontrarla en `${PREFIX}/etc/racoon/psk.txt`. Es importante saber que la llave precompartida *no* es la llave que se utilizará para cifrar el tráfico a través del enlace VPN; solamente es una muestra que permite a los dæmons que administran las llaves confiar el uno en el otro.

`psk.txt` contiene una línea por cada sitio remoto con el que esté tratando. En nuestro ejemplo, donde existen dos sitios, cada fichero `psk.txt` contendrá una línea (porque cada extremo de la VPN solo está tratando con un sitio en el otro extremo).

En la puerta de enlace #1 esta línea debería parecerse a esta:

```
W.X.Y.Z            secreto
```

Esto es, la dirección IP *pública* del extremo remoto, un espacio en blanco, y una cadena de texto que es el secreto en sí. en el extremo remoto, espacio en blanco, y un texto de cadena que proporcina el secreto. Obviamente, no debe utilizar "secret" como su llave; aplique aquí las reglas y recomendaciones habituales para la elección de contraseñas.

En la puerta de enlace #2 la línea se parecería a esta

```
A.B.C.D            secreto
```

Esto es, la dirección IP pública del extremo remoto, y la misma llave secreta. `psk.txt` debe tener modo `0600` (es decir, modo de solo lectura/escritura para `root`) antes de que ejecute racoon.

Debe ejecutar racoon en ambas puertas de enlace. También tendrá que añadir algunas reglas a su cortafuegos para permitir el tráfico IKE, que se transporta sobre UDP al puer-

El escenario: dos redes, conectadas por Internet, que queremos que se comporten como una sola

to ISAKMP (Internet Security Association Key Management Protocol). Esto debe estar al principio de las reglas de su cortafuegos.

```
ipfw add 1 allow udp from A.B.C.D to W.X.Y.Z isakmp
ipfw add 1 allow udp from W.X.Y.Z to A.B.C.D isakmp
```

Una vez que ejecute racoon puede tratar de hacer un ping a una puerta de enlace desde la otra. La conexión todavía no está cifrada porque aún no se han creado las asociaciones de seguridad entre los dos equipos: esto puede llevar un poco de tiempo; es posible que advierta un pequeño retraso antes de los ping empiecen responder.

Una vez creadas las asociaciones de seguridad puede verlas utilizando setkey(8). Ejecute

```
setkey -D
```

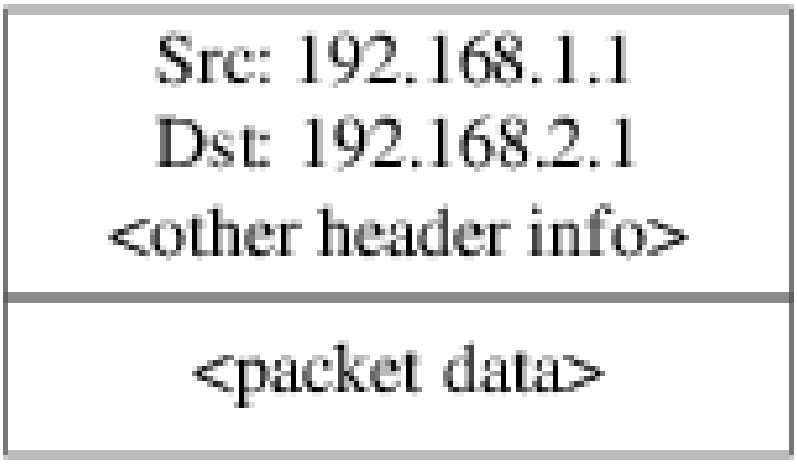
en cualquiera de los equipos para comprobar la información de la asociación de seguridad.

Ya está resuelta la mitad del problema. La otra mitad es configurar sus políticas de seguridad.

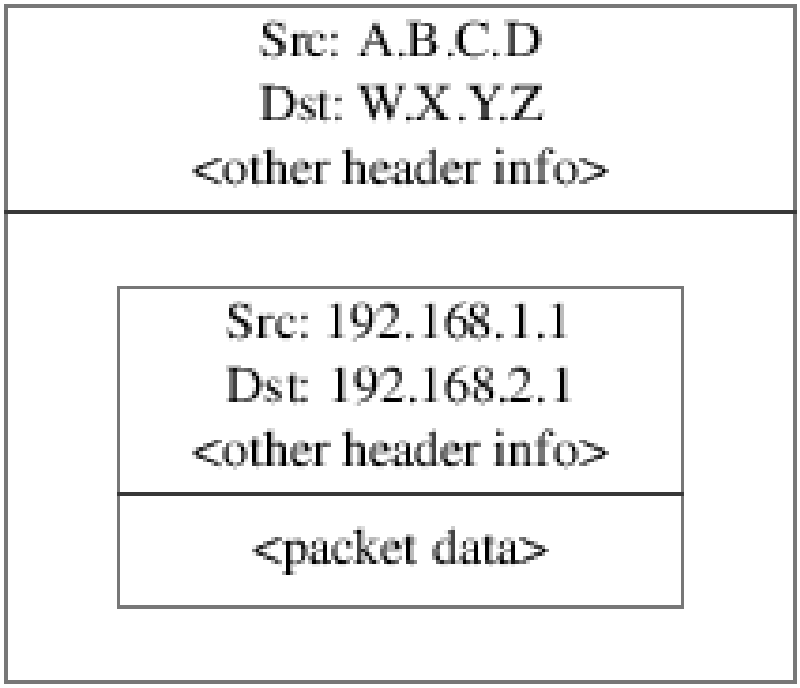
Queremos crear una política de seguridad sensata, así que vamos a revisar lo que tenemos configurado hasta el momento. Esta revisión abarca ambos extremos del enlace.

Cada paquete IP que usted manda tiene una cabecera que contiene datos acerca del paquete. La cabecera incluye la dirección IP de destino y del origen. Como ya sabemos, las direcciones IP privadas como el rango `192.168.x.y` no deberían aparezcan en Internet. Dado que es a través de Internet por donde los queremos transmitir los debemos encapsular dentro de otro paquete. Este paquete debe contener tanto la dirección IP de destino y origen públicas sustituidas por las direcciones privadas.

Así que si su paquete saliente empezó pareciendose a este:



tras el encapsulado se parecerá bastante a este:



El dispositivo `gif` se encarga del encapsulado. Como puede ver el paquete tiene una dirección IP real en el exterior, y nuestro paquete original ha sido envuelto como dato dentro del paquete que enviaremos a través de Internet.

Obviamente, queremos que todo el tráfico entre las VPN vaya cifrado. Pongamos esto último en palabras para comprenderlo mejor:

"Si un paquete sale desde `A.B.C.D`, y tiene como destino `W.X.Y.Z`, cífralo utilizando las asociaciones de seguridad necesarias."

"Si un paquete llega desde `W.X.Y.Z`, y tiene como destino `A.B.C.D`, descífralo utilizando las asociaciones de seguridad necesarias."

Este planteamiento se aproxima bastante, pero no es exactamente lo que queremos hacer. Si lo hiciera así todo el tráfico desde y hacia `W.X.Y.Z`, incluso el tráfico que no forma parte de la VPN, será cifrado; esto no es lo que queremos. La política correcta es la siguiente:

"Si un paquete sale desde `A.B.C.D`, y está encapsulando a otro paquete, y tiene como destino `W.X.Y.Z`, cífralo utilizando las asociaciones de seguridad necesarias."

"Si un paquete llega desde `W.X.Y.Z`, y está encapsulando a otro paquete, y tiene como destino `A.B.C.D`, descífralo utilizando las asociaciones de seguridad necesarias."

Las políticas de seguridad también se imponen utilizando setkey(8). setkey(8) proporciona un lenguaje de configuración para definir la política. Puede introducir las instrucciones de configuración a través de la entrada estándar (stdin), o puede usar la opción `-f` para especificar un fichero que contenga las instrucciones de configuración.

La configuración en la puerta de enlace #1 (que tiene la dirección IP pública `A.B.C.D`) para forzar que todo el tráfico saliente hacia `W.X.Y.Z` vaya cifrado es:

```
spdadd A.B.C.D/32 W.X.Y.Z/32 ipencap -P out ipsec esp/tunnel/A.B.C.D-
W.X.Y.Z/require;
```

Ponga estas órdenes en un fichero (por ejemplo `/etc/ipsec.conf`) y ejecute

```
# setkey -f /etc/ipsec.conf
```

`spdadd` le dice a setkey(8) que queremos añadir una regla a la base de datos de políticas de seguridad. El resto de la línea especifica qué paquetes se ajustarán a esta política. `A.B.C.D/32` y `W.X.Y.Z/32` son las direcciones IP y máscaras de red que identifican la red o equipos a los que se aplicará esta política. En nuestro caso queremos aplicarla al tráfico entre estos dos equipos. `-P out` dice que esta política se aplica a paquetes salientes, e `ipsec` hace que el paquete sea asegurado.

La segunda línea especifica cómo será cifrado este paquete. `esp` es el protocolo que se utilizará, mientras que `tunnel` indica que el paquete será después encapsulado en un paquete IPsec. El uso repetido de `A.B.C.D` y `W.X.Y.Z` se utiliza para seleccionar la asociación de seguridad a usar, y por último `require` exige que los paquetes deben cifrarse si concuerdan con esta regla.

Esta regla solo concuerda con paquetes salientes. Necesitará una regla similar para los paquetes entrantes.

```
spdadd W.X.Y.Z/32 A.B.C.D/32 ipencap -P in ipsec esp/tunnel/W.X.Y.Z-
A.B.C.D/require;
```

Observe el `in` en lugar del `out` en este caso, y la inversión necesaria de las direcciones IP.

La otra puerta de enlace (que tiene la dirección IP pública `W.X.Y.Z`) necesitará reglas similares.

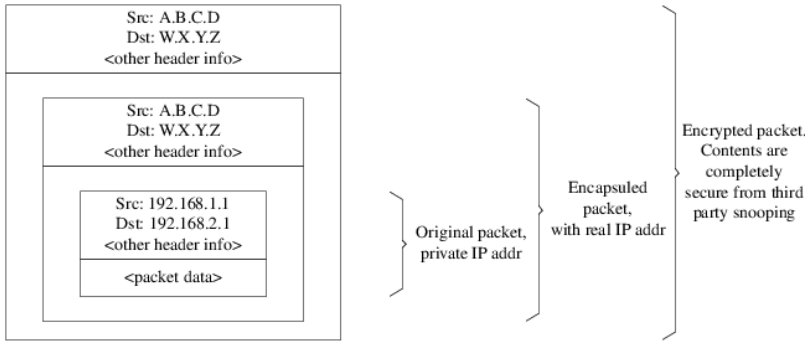
```
spdadd W.X.Y.Z/32 A.B.C.D/32 ipencap -P out ipsec esp/tunnel/W.X.Y.Z-
A.B.C.D/require;
spdadd A.B.C.D/32 W.X.Y.Z/32 ipencap -P in ipsec esp/tunnel/A.B.C.D-
W.X.Y.Z/require;
```

Finalmente, necesita añadir reglas a su cortafuegos para permitir la circulación de paquetes ESP e IPENCAP de ida y vuelta. Tendrá que añadir reglas como estas a ambos equipos.

```
ipfw add 1 allow esp from A.B.C.D to W.X.Y.Z
ipfw add 1 allow esp from W.X.Y.Z to A.B.C.D
ipfw add 1 allow ipencap from A.B.C.D to W.X.Y.Z
ipfw add 1 allow ipencap from W.X.Y.Z to A.B.C.D
```

Debido a que las reglas son simétricas puede utilizar las mismas reglas en ambas puertas de enlace.

Los paquetes salientes tendrán ahora este aspecto:



Cuando los paquetes llegan al otro extremo de la VPN serán descifrados (utilizando las asociaciones de seguridad que han sido negociadas por racoon). Después entrarán al interfaz `gif`, que desenvuelve la segunda capa, hasta que nos quedamos con paquete má interno, que puede entonces viajar a la red interna.

Puede revisar la seguridad utilizando la misma prueba de ping(8) anterior. Primero, inicie una sesión en la puerta de enlace `A.B.C.D` , y ejecute:

```
tcpdump dst host 192.168.2.1
```

En otra sesión en la misma máquina ejecute

```
ping 192.168.2.1
```

Debería ver algo similar a lo siguiente:

```
XXX tcpdump output
```

ahora, como puede ver, tcpdump(1) muestra los paquetes ESP. Si trata de examinarlos con la opción `-s` verá basura (aparentemente), debido al cifrado.

Felicidades. Acaba de configurar una VPN entre dos sitios remotos.

- Configure ambos kernel con:

  ```
  options IPSEC
  ```

```
options IPSEC_ESP
```

- Instale security/racoon. Edite `${PREFIX}/etc/racoon/psk.txt` en ambas puertas de enlace añadiendo una entrada para la dirección IP del equipo remoto y una llave secreta que ambos conozcan. Asegúrese de que este fichero esté en modo 0600.

- Añada las siguientes líneas a `/etc/rc.conf` en ambos equipos:

```
ipsec_enable="YES"
ipsec_file="/etc/ipsec.conf"
```

- Crée en ambos equipos un `/etc/ipsec.conf` que contenga las líneas spdadd necesarias. En la puerta de enlace #1 sería:

```
spdadd A.B.C.D/32 W.X.Y.Z/32 ipencap -P out ipsec
  esp/tunnel/A.B.C.D-W.X.Y.Z/require;
spdadd W.X.Y.Z/32 A.B.C.D/32 ipencap -P in ipsec
  esp/tunnel/W.X.Y.Z-A.B.C.D/require;
```

  En la puerta de enlace #2 sería:

```
spdadd W.X.Y.Z/32 A.B.C.D/32 ipencap -P out ipsec
  esp/tunnel/W.X.Y.Z-A.B.C.D/require;
spdadd A.B.C.D/32 W.X.Y.Z/32 ipencap -P in ipsec
  esp/tunnel/A.B.C.D-W.X.Y.Z/require;
```

- Añada a su(s) cortafuegos las reglas necesarias para que permita(n) el paso de tráfico IKE, ESP e IPENCAP en ambos equipos:

```
ipfw add 1 allow udp from A.B.C.D to W.X.Y.Z isakmp
ipfw add 1 allow udp from W.X.Y.Z to A.B.C.D isakmp
ipfw add 1 allow esp from A.B.C.D to W.X.Y.Z
ipfw add 1 allow esp from W.X.Y.Z to A.B.C.D
ipfw add 1 allow ipencap from A.B.C.D to W.X.Y.Z
ipfw add 1 allow ipencap from W.X.Y.Z to A.B.C.D
```

Los dos pasos previos deben bastar para levantar la VPN. Las máquinas en cada red seán capaces de dirigirse una a otra utilizando direcciones IP, y todo el tráfico a través del enlace será cifrado de forma automática y segura.

14.11. OpenSSH

Escrito por Chern Lee.

OpenSSH es un conjunto de herramientas de conectividad que se usan para acceder a sistemas remotos de forma segura. Puede usarse como sustituto directo de `rlogin`, `rsh`, `rcp` y `telnet`. Además cualquier otra conexión TCP/IP puede reenviarse o enviarse a través de un túnel a través de SSH. OpenSSH cifra todo el tráfico para eliminar de forma efectiva el espionaje, el secuestro de conexiones, y otros ataques en la capa de red.

OpenSSH está a cargo del proyecto OpenBSD, y está basado en SSH v1.2.12, con todos los errores recientes corregidos y todas las actualizaciones correspondientes. Es compatible con los protocolos SSH 1 y 2. OpenSSH forma parte del sistema base desde FreeBSD 4.0.

14.11.1. Ventajas de utilizar OpenSSH

Normalmente, al utilizar telnet(1) o rlogin(1) los datos se envían a través de la red en limpio, es decir, sin cifrar. Cualquier "sniffer" de red entre el cliente y el servidor puede robar la información de usuario/contraseña o los datos transferidos durante su sesión. OpenSSH ofrece diversos métodos de validación y cifrado para evitar que sucedan estas cosas.

14.11.2. Habilitar sshd

El dæmon sshd está habilitado por defecto FreeBSD 4.X y puede elegir habilitarlo o no durante la instalación en FreeBSD 5.X. Si quiere saber si está habilitado revise si la siguiente línea está en `rc.conf`:

```
sshd_enable="YES"
```

Esta línea cargará sshd(8), el programa dæmon de OpenSSH, en el arranque de su sistema. Puede ejecutar el dæmon sshd tecleando `sshd` en la línea de órdenes.

14.11.3. Cliente SSH

ssh(1) funciona de manera similar a rlogin(1).

```
# ssh user@example.com
Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)? yes
Host 'ejemplo.com' added to the list of known hosts.
usuario@ejemplo.com's password: *******
```

El login continuará como lo haría si fuera una sesión de `rlogin` o `telnet`. SSH utiliza un sistema de huellas de llaves para verificar la autenticidad del servidor cuando el cliente se conecta. Se le pide al usuario que introduzca `yes` solamente la primera vez que se conecta. Todos los intentos futuros de login se verifican contra la huella de la llave guardada la primera vez. El cliente SSH le alertará si la huella guardada difiere de la huella recibida en

futuros intentos de acceso al sistema. Las huellas se guardan en `~/.ssh/known_hosts` , y en `~/.ssh/known_hosts2` las huellas SSH v2.

Por defecto las versiones recientes de los servidores OpenSSH solamente aceptan conexiones SSH v2. El cliente utilizará la versión 2 si es posible y pasará como respaldo a la versión 1. El cliente puede también ser obligado a utilizar una u otra pasándole `-1` o `-2`, respectivamente para la versión 1 y la versión 2. Se mantiene la compatibilidad del cliente con la versión 1 para mantener la compatibilidad con versiones antiguas.

14.11.4. Copia segura

scp(1) funciona de manera muy similar a rcp(1); copia un fichero desde o hacia un sistema remoto, con la diferencia de que lo hace de una forma segura.

```
#  scp usuario@ejemplo.com:/COPYRIGHT COPYRIGHT
usuario@ejemplo.com's password: *******
COPYRIGHT            100% |****************************|  4735
00:00
#
```

Ya que la huella se guardó en este equipo durante el ejemplo anterior se verifica ahora al utilizar scp(1).

Los argumentos de scp(1) son similares a cp(1), con el fichero o ficheros como primer argumento, y el destino como segundo. Ya que el fichero se transfiere a través de la red, a través de SSH, uno o más argumentos tienen la estructura `user@host:<ruta_al_fichero_remoto>` .

14.11.5. Configuración

Los ficheros de configuración del sistema tanto para el dæmon OpenSSH como para el cliente están en `/etc/ssh` .

`ssh_config` contiene las opciones del cliente, mientras que `sshd_config` configura el dæmon.

Además las opciones `sshd_program` (`/usr/sbin/sshd` por defecto), y `sshd_flags` de `rc.conf` ofrecer más niveles de configuración.

14.11.6. ssh-keygen

ssh-keygen(1) le permite validar a un usuario sin pedirle la contraseña:

```
% ssh-keygen -t dsa
Generating public/private dsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_dsa):
Created directory '/home/user/.ssh'.
```

```
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_dsa.
Your public key has been saved in /home/user/.ssh/id_dsa.pub.
The key fingerprint is:
bb:48:db:f2:93:57:80:b6:aa:bc:f5:d5:ba:8f:79:17 usuario@host.↺
ejemplo.com
```

ssh-keygen(1) creará un par de llaves pública y privada para usar en la validación. La llave privada se guarda en `~/.ssh/id_dsa` o en `~/.ssh/id_rsa`, mientras que la llave pública se guarda en `~/.ssh/id_dsa.pub` o en `~/.ssh/id_rsa.pub`, respectivamente para llaves DSA y RSA. La llave pública debe guardarse en el `~/.ssh/authorized_keys` de la máquina remota para que la configuración funcione. Las llaves RSA versión 1 deben guardarse en `~/.ssh/authorized_keys`.

De este modo permitirá conexiones a la máquina remota mediante llaves SSH en lugar de contraseñas.

Si usa una contraseña al ejecutar ssh-keygen(1), se le pedirá al usuario una contraseña cada vez que quiera utilizar la llave privada. ssh-agent(1) puede evitar la molestia de introducir repetidamente frases largas. esto se explica má adelante, en la Sección 14.11.7, “ssh-agent y ssh-add”.

Aviso

Las opciones y ficheros pueden ser diferentes según la versión de OpenSSH que tenga en su sistema; para evitar problemas consulte la página de manual ssh-keygen(1).

14.11.7. ssh-agent y ssh-add

ssh-agent(1) y ssh-add(1) ofrecen métodos para que las llaves SSH se puedan cargar en memoria, permitiendo eliminar la necesidad de teclear la contraseña cada vez que haga falta.

ssh-agent(1) gestionará la validación utilizando la llave (o llaves) privada que le cargue. ssh-agent(1) se usa para lanzar otras aplicaciones. En el nivel más básico puede generar una shell o a un nivel más avanzado un gestor de ventanas.

Para usar ssh-agent(1) en una shell necesitará primero ser invocado como argumento por una shell. Segundo, añada la identidad ejecutando ssh-add(1) y facilitando la contraseña de la llave privada. Completados estos pasos el usuario puede hacer ssh(1) a cualquier equipo que tenga instalada la llave pública correspondiente. Por ejemplo:

```
% ssh-agent csh
```

```
% ssh-add
Enter passphrase for /home/user/.ssh/id_dsa:
Identity added: /home/user/.ssh/id_dsa (/home/user/.ssh/id_dsa)
%
```

Para utilizar ssh-agent(1) en X11 tendrá que incluir una llamada a ssh-agent(1) en `~/.xinitrc`. De este modo ofrecerá los servicios de ssh-agent(1) a todos los programas lanzados en X11. Veamos un ejemplo de `~/.xinitrc`:

```
exec ssh-agent startxfce4
```

Esto lanzaría ssh-agent(1), que a su vez lanzaría XFCE cada vez que inicie X11. Hecho esto y una vez reiniciado X11 para aplicar los cambios puede ejecutar ssh-add(1) para cargar todas sus llaves SSH.

14.11.8. Túneles SSH

OpenSSH permite crear un túnel en el que encapsular otro protocolo en una sesión cifrada.

La siguiente orden le dice a ssh(1) que cree un túnel para telnet:

```
% ssh -2 -N -f -L 5023:localhost:23 usuario@foo.ejemplo.com
%
```

Veamos las opciones que se le han suministrado a `ssh`:

`-2`
: Obliga a `ssh` a utilizar la versión 2 del protocolo. (No la use si está trabajando con servidores SSH antiguos)

`-N`
: Indica que no se ejecutará una orden remota, o solamente túnel. Si se omite, `ssh` iniciaría una sesión normal.

`-f`
: Obliga a `ssh` a ejecutarse en segundo plano.

`-L`
: Indica un túnel local según el esquema *`puerto local:equipo remoto:puerto remoto`*.

`usuario@foo.ejemplo.com`
: El servidor SSH remoto.

Un túnel SSH crea un socket que escucha en `localhost` en el puerto especificado. Luego reenvía cualquier conexión recibida en el puerto/equipo local vía la conexión SSH al puerto o equipo remoto especificado.

En el ejemplo el puerto *5023* en `localhost` se reenvía al puerto *23* del `localhost` de la máquina remota. Ya que *23* es telnet, esto crearía una sesión telnet segura a través de un túnel SSH.

Puede usar esto para encapsular cualquier otro protocolo TCP inseguro como SMTP, POP3, FTP, etc.

Ejemplo 14.1. Uso de SSH para crear un túnel seguro para SMTP

```
% ssh -2 -N -f -L 5025:localhost:25 usuario@correo.ejemplo.com
usuario@correo.ejemplo.com's password: *****
% telnet localhost 5025
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 correo.ejemplo.com ESMTP
```

Puede usar esta técnica junto con ssh-keygen(1) y cuentas adicionales de usuario para crear un entorno más transparente, esto es, más cómodo. Puede usar llaves en lugar de teclear contraseñas y puede ejecutar los túneles de varios usuarios.

14.11.8.1. Ejemplos prácticos de túneles SSH

14.11.8.1.1. Acceso seguro a un servidor POP3

En el trabajo hay un servidor SSH que acepta conexiones desde el exterior. En la misma red de la oficina reside un servidor de correo que ejecuta un servidor POP3. La red, o ruta de red entre su casa y oficina puede o no ser completamente de fiar. Debido a esto necesita revisar su correo electrónico de forma segura. La solución es crear una conexión SSH al servidor SSH de su oficina y llegar por un túnel al servidor de correo.

```
% ssh -2 -N -f -L 2110:correo.ejemplo.com:110 usuario@servidor-ssh.↺
ejemplo.com
usuario@servidor-ssh.ejemplo.com's password: ******
```

cuando el túnel esté funcionando haga que su cliente de correo envíe peticiones POP3 a `localhost` en el puerto 2110. La conexión será reenviada de forma totalmente segura a traveés del túnel a `correo.ejemplo.com` .

14.11.8.1.2. Saltarse un cortafuegos draconiano

Algunos administradores de red imponen reglas de cortafuegos extremadamente draconianas, filtrando no solo las conexiones entrantes, sino también las salientes. Tal vez solo se le otorgue acceso a máquinas remotas a través de los puertos 22 y 80 para ssh y navegar en web.

Tal vez quiera acceder a otros servicios (que tal vez ni siquiera estén relacionados con el trabajo), como un servidor Ogg Vorbis para escuchar música. Si ese servidor Ogg Vorbis transmite en un puerto que no sea el 22 o el 80 no podrá tener acceso a él.

La solución es crear una conexión SSH fuera del cortafuegos de su red y utilizarla para hacer un túnel al servidor Ogg Vorbis.

```
% ssh -2 -N -f -L 8888:musica.ejemplo.com:8000 usuario@sistema-no-
filtrado.ejemplo.org
usuario@sistema-no-filtrado.ejemplo.org's password: *******
```

Haga que el programa con el que suele escuchar música haga peticiones a `localhost` puerto 8888, que será reenviado a `musica.ejemplo.com` puerto 8000, evadiendo con éxito el cortafuegos.

14.11.9. La opción de usuarios `AllowUsers`

Limitar qué usuarios pueden entrar y desde dónde suele ser razonable. La opción `AllowUsers` le permite configurarlo, por ejemplo, para permitir entrar solamente al usuario `root` desde `192.168.1.32` . Puede hacerlo con algo parecido a esto en `/etc/ssh/sshd_config` :

```
AllowUsers root@192.168.1.32
```

Para permitir al usuario `admin` la entrada desde cualquier lugar, solamente introduzca el nombre de usuario:

```
AllowUsers admin
```

Puede listar múltiples usuarios en la misma línea:

```
AllowUsers root@192.168.1.32 admin
```

Nota

Es importante que incluya a cada usuario que necesite entrar a esta máquina o no podrán entrar.

Después de hacer los cambios a b `/etc/ssh/sshd_config` debe decirle a sshd(8) que cargue de nuevo sus ficheros de configuración ejecutando:

```
# /etc/rc.d/sshd reload
```

14.11.10. Lecturas complementarias

OpenSSH

ssh(1) scp(1) ssh-keygen(1) ssh-agent(1) ssh-add(1) ssh_config(5)

sshd(8) sftp-server(8) sshd_config(5)

14.12. Listas de control de acceso a sistemas de ficheros

Contribuido por Tom Rhodes.

Además de otras mejoras del sistema de ficheros como las instantáneas (“snapshots”), FreeBSD 5.0 y siguientes ofrecen las ACL (“Access Control Lists”, listas de control de acceso) como un elemento más de seguridad.

Las listas de control de acceso extienden el modelo de permisos estándar de UNIX® de una manera altamente compatible (POSIX®.1e). Esta opción permite al administrador usar con gran provecho un modelo de seguridad más sofisticado.

Para habilitar soporte de ACL en sistemas de ficheros UFS la siguiente opción:

```
options UFS_ACL
```

debe ser compilada en el kernel. Si esta opción no ha sido compilada, se mostrará un mensaje de advertencia si se intenta montar un sistema de ficheros que soporte ACL. Esta opción viene incluida en el kernel `GENERIC`. Las ACL dependen de los atributos extendidos habilitados en el sistema de ficheros. Los atributos extendidos están incluidos por defecto en la nueva generación de sistemas de ficheros UNIX® UFS2.

Nota

Los atributos extendidos pueden usarse también en UFS1 pero requieren una carga de trabajo mucho más elevada que en UFS2. El rendimiento de los atributos extendidos es, también, notablemente mayor en UFS2. Por todo esto si quiere usar ACL le recomendamos encarecidamente que use UFS2.

LasACL se habilitadan mediante una bandera administrativa durante el montaje, `acls`, en el fichero `/etc/fstab`. La bandera de montaje puede también activarse de forma permanente mediante tunefs(8) para modificar una bandera de superbloque ACLs en la cabecera del sistema de ficheros. En general es preferible usar la bandera de superbloque por varios motivos:

- La bandera de montaje ACL no puede cambiarse por un remontaje (mount(8) `-u`), sino con un completo umount(8) y un mount(8). Esto significa que no se pueden habilitar

las ACL en el sistema de ficheros raíz después del arranque. También significa que no se puede cambiar la disposición de un de ficheros una vez que se ha comenzado a usar.

- Activar la bandera de superbloque provocará que el sistema de ficheros se monte siempre con las ACL habilitadas incluso si no existe una entrada en `fstab` o si los dispositivos se reordenan. Esto es así para prevenir un montaje accidental del sistema de ficheros sin tener las ACL habilitadas, que podría resultar en que se impongan de forma inadecuada las ACL, y en consecuencia problema de seguridad.

Nota

Podemos cambiar el comportamiento de las ACL para permitirle a la bandera ser habilitada sin un mount(8) completo, pero puede salirle el tiro por la culata si activa las ACL, luego las desactiva, y después las vuelve a activar sin configurar desde cero las atributos extendidos. En general, una vez que se han deshabilitado las ACL en un sistema de ficheros no deben dehabilitarse, ya que la protección de ficheros resultante puede no ser compatible las que esperan los usuarios del sistema, y al volver a activar las ACL volver a asignar las ACL a ficheros cuyos permisos hubieran sido cambiados, lo que puede desenbocar en un escenario impredecible.

Los sistemas de ficheros con ACL habilitadas tienen un signo + (más) al visualizar sus configuraciones de permisos. Por ejemplo:

```
drwx------  2 robert  robert  512 Dec 27 11:54 private
drwxrwx---+ 2 robert  robert  512 Dec 23 10:57 directorio1
drwxrwx---+ 2 robert  robert  512 Dec 22 10:20 directorio2
drwxrwx---+ 2 robert  robert  512 Dec 27 11:57 directorio3
drwxr-xr-x  2 robert  robert  512 Nov 10 11:54 public_html
```

Aquí vemos que los directorios `directorio1`, `directorio2`, y `directorio3` están usando ACL. El directorio `public_html` no.

14.12.1. Uso de ACL

Las ACLs del sistema de ficheros pueden comprobarse con getfacl(1). Por ejemplo, para ver las configuraciones de ACL del fichero `test`, uno podría usar lo siguiente:

```
% getfacl test
 #file:test
 #owner:1001
 #group:1001
 user::rw-
 group::r--
```

```
other::r--
```

Para cambiar las configuraciones de las ACL en este fichero use setfacl(1). Observe:

```
% setfacl -k test
```

La bandera `-k` eliminará todas las ACLs definidas para un fichero o sistema ficheros. El método preferible sería utilizar `-b`, ya que deja los campos básicos imprescindibles para que las ACL sigan funcionando.

```
% setfacl -m u:trhodes:rwx,group:web:r--,o::--- test
```

La opción `-m` se usa para modificar las entradas por defecto de las ACL. Debido a que no había entradas predefinidas puesto que fueron eliminadas por la orden anterior, restauraremos las opciones por defecto y asignará las opciones listadas. Tenga en cuenta que si añade un nuevo usuario o grupo aparecerá el error Invalid argument en la salida estándar `stdout`.

14.13. Monitorización de fallos de seguridad de aplicaciones

Texto de Tom Rhodes.

En estos últimos años el mundo de la seguridad ha hecho grandes avances en cuanto a la gestión de las vulnerabilidades. La amenaza de asaltos a los sistemas se incrementa cuando se instalan y configuran aplicaciones de muy diversas procedencias en virtualmente cualquier sistema operativo disponible.

La evaluación de vulnerabilidades es un factor clave en la seguridad; aunque FreeBSD libere avisos de seguridad relacionados con el sistema base, llevar la gestión de vulnerabilidades hasta cada aplicación que se puede instalar en FreeBSD va mucho más allá de la capacidad del proyecto FreeBSD. A pesar de esto existe una forma de mitigar las vulnerabilidades de esas aplicaciones y advertir a los administradores sobre los problemas de seguridad a medida que se detectan. Portaudit existe para hacer ese trabajo.

El port security/portaudit consulta una base de datos, actualizada y mantenida por el equipo de seguridad y por los desarrolladores de FreeBSD en busca de incidentes de seguridad que hayan sido detectados.

Si quiere usar Portaudit instálelo desde la colección de ports:

```
# cd /usr/ports/security/portaudit && make install clean
```

Durante el proceso de instalación los ficheros de configuración de periodic(8) se actualizan haciendo que Portaudit aparezca en el mensaje sobre la seguridad del sistema que

diariamente Recuerde que ese correo (que se envia a la cuenta `root` es muy importante y debería leerlo. No hay ninguna configuración que deba modificar o crear.

Después de la instalación un administrador debe actualizar la base de datos alojada en local en `/var/db/portaudit` mediante:

```
# portaudit -F
```

Nota

La base de datos será actualizada automáticamente durante la ejecución de periodic(8); así que la orden anterior es totalmente opcional. Solo se necesita para los siguientes ejemplos.

Si quiere comprobar si entre las aplicaciones que haya instalado desde el árbol de ports en su sistema hay problemas de seguridad sólo tiene que ejecutar lo siguiente:

```
# portaudit -a
```

Este es un ejemplo de la salida:

```
Affected package: cups-base-1.1.22.0_1
Type of problem: cups-base -- HPGL buffer overflow vulnerability.
Reference: <http://www.FreeBSD.org/ports/
portaudit/40a3bca2-6809-11d9-a9e7-0001020eed82.html>

1 problem(s) in your installed packages found.

You are advised to update or deinstall the affected package(s) ↺
immediately.
```

El administrador del sistema obtendrá mucha más información sobre el problema de seguridad dirigiendo su navegador web a la URL que aparece en el mensaje. Esto incluye versiones afectadas (por versión de port de FreeBSD), junto con otros sitios web que contengan advertencias de seguridad.

En pocas palabras, Portaudit es un programa muy poderoso y extremadamente útil cuando se combina con el port Portupgrade.

14.14. FreeBSD Security Advisories

Texto de Tom Rhodes.

Como muchos sistemas operativos con calidad de producción, FreeBSD publica “Security Advisories” (advertencias de seguridad. Estas advertencias suelen enviarse por correo a

las listas de seguridad e incluidas en la Errata solamente después de que la versión apropiada haya sido corregida. Esta sección tiene como fin explicar en qué consiste una advertencia de seguridad, cómo entenderla y qué medidas hay que tomar para parchear el sistema.

14.14.1. ?Qué aspecto tiene una advertencia de seguridad?

Las advertencias de seguridad de FreeBSD tienen un aspecto similar a la que se muestra aquí. Fué enviada a la lista de correo freebsd-security-notifications.

```
=============================================================================
FreeBSD-SA-XX:XX.UTIL                                          Security ↺
Advisory
                                                               The ↺
FreeBSD Project

Topic:          denial of service due to some problem❶

Category:       core❷
Module:         sys❸
Announced:      2003-09-23❹
Credits:        Person@EMAIL-ADDRESS❺
Affects:        All releases of FreeBSD❻
                FreeBSD 4-STABLE prior to the correction date
Corrected:      2003-09-23 16:42:59 UTC (RELENG_4, 4.9-PRERELEASE)
                2003-09-23 20:08:42 UTC (RELENG_5_1, 5.1-RELEASE-p6)
                2003-09-23 20:07:06 UTC (RELENG_5_0, 5.0-RELEASE-p15)
                2003-09-23 16:44:58 UTC (RELENG_4_8, 4.8-RELEASE-p8)
                2003-09-23 16:47:34 UTC (RELENG_4_7, 4.7-RELEASE-p18)
                2003-09-23 16:49:46 UTC (RELENG_4_6, 4.6-RELEASE-p21)
                2003-09-23 16:51:24 UTC (RELENG_4_5, 4.5-RELEASE-p33)
                2003-09-23 16:52:45 UTC (RELENG_4_4, 4.4-RELEASE-p43)
                2003-09-23 16:54:39 UTC (RELENG_4_3, 4.3-RELEASE-
p39)❼
FreeBSD only:   NO❽

For general information regarding FreeBSD Security Advisories,
including descriptions of the fields above, security branches, and ↺
the
following sections, please visit
http://www.FreeBSD.org/security/.

I.   Background❾


II.  Problem Description❿


III. Impact
```

```
IV.  Workaround

V.   Solution

VI.  Correction details

VII. References
```

❶ El campo `Topic` indica cuál es exactamente el problema. Básicamente es la introducción de la advertencia de seguridad actual e indica el uso malintencionado que puede darse a la vulnerabilidad.

❷ `Category` se refiere a la parte afectada del sistema, que puede ser `core`, `contrib` o `ports`. La categoría `core` significa que la vulnerabilidad afecta a un componente central del sistema operativo FreeBSD. La categoría `contrib` significa que la vulnerabilidad afecta a software que no ha sido desarrollado por el proyecto FreeBSD, como sendmail. La categoría `ports` indica que la vulnerabilidad afecta a software incluido en la colección de ports.

❸ El campo `Module` se refiere a la ubicación del componente, por ejemplo `sys`. En este ejemplo vemos que está afectado el módulo `sys`; por lo tanto esta vulnerabilidad afecta a componentes utilizados dentro del kernel.

❹ El campo `Announced` refleja la fecha de publicación de la advertencia de seguridad fué publicada o anunciada al mundo. Esto significa que el equipo de seguridad ha verificado que el que el problema existe y que se ha incluido un parche que soluciona el problema en el repositorio de código fuente de FreeBSD.

❺ El campo `Credits` le da el crédito al individuo u organización que descubrió y reportó la vulnerabilidad.

❻ El campo `Affects` explica a qué versiones de FreeBSD afecta esta vulnerabilidad. En el caso del kernel una rápida revisión de la salida de `ident` en los ficheros afectados ayudará a determinar la versión. En el caso de de los ports el número de versión aparece después del nombre del port en `/var/db/pkg` . Si el sistema no se sincroniza con el repositorio CVS de FreeBSD y se reconstruye diariamente, existe la posibilidad de que esté afectado por el problema de seguridad.

❼ El campo `Corrected` indica la fecha, hora, zona horaria y versión de FreeBSD en que fué corregido.

❽ El campo `FreeBSD only` indica si la vulnerabilidad afecta solamente a FreeBSD o si afecta también a otros sistemas operativos.

❾ El campo `Background` informa acerca de qué es exactamente la aplicación afectada. La mayor parte de las veces se refiere a por qué la aplicación existe en FreeBSD, para qué se usa y un poco de información de cómo llegó llegó a ocupar el lugar que ocupa en el sistema o el árbol de ports.

❿ El campo `Problem Description` explica el problema de seguridad en profundidad. Puede incluir información del código erróneo, o incluso cómo puede usarse maliciosamente el error para abrir un agujero de seguridad.

El campo `Impact` describe el tipo de impacto que el problema pueda tener en un sistema. Por ejemplo, esto puede ser desde un ataque de denegación de servicio, hasta una escalada de privilegios de usuario, o incluso ofrecer al atacante acceso de superusuario.
El campo `Workaround` ofrece una solución temoral posible para los administradores de sistemas que tal vez no puedan actualizar el sistema. Esto puede deberse a la falta de tiempo, disponibilidad de de red, o a muchas otras razones. A pesar de todo la la seguridad no se debe tomar a la ligera y un sistema afectado debe parchearse al menos aplicar una solución temporal para el agujero de seguridad.
El campo `Solution` ofrece instrucciones para parchear el sistema afectado. Este es un método paso a paso, probado y verificado para parchear un sistema y que trabaje seguro.
El campo `Correction Details` despliega la rama del CVS o el nombre de la versión con los puntos cambiados a guiones bajos. También muestra el número de revisión de los ficheros afectados dentro de cada rama.
El campo `References` suele ofrecer fuentes adicionales de información: URL, libros, listas de correo y grupos de noticias.

14.15. Contabilidad de procesos

Texto de Tom Rhodes.

La contabilidad de procesos es un método de seguridad en el cual un administrador puede mantener un seguimiento de los recursos del sistema utilizados, su distribución entre los usuarios, ofrecer monitorización del sistema y seguir la pista mínimamente a las órdenes de usuario.

Esto en realidad tiene sus puntos positivos y negativos. Uno de los positivos es que una intrusión puede minimizarse en el momento de producirse. Uno negativo es la cantidad de logs generados por la contabilidad de procesos y el espacio de disco que requieren. Esta sección guiará al administrador a través de los fundamentos de la contabilidad de procesos.

14.15.1. Cómo habilitar y utilizar la contabilidad de procesos

Antes de poder usar la contabilidad de procesos tendrá que habilitarla. Ejecute la siguiente orden:

```
# touch /var/account/acct

# accton /var/account/acct

# echo 'accounting_enable="YES"' >> /etc/rc.conf
```

Una vez habilitada, la contabilidad de procesos empezará a seguir el rastro de estadísticas de la CPU, órdenes, etc. Todos los logs de contabilidad están en un formato ilegible para

humanos, pero accesibles para sa(8). Si se ejecuta sin opciones, `sa` imprimirá información sobre el número de llamadas por usuario, el tiempo total transcurrido expresado en minutos, el tiempo total de CPU y de usuario en minutos, el número medio de operaciones de E/S, etc.

Para ver información acerca de las órdenes que se están ejecutados puede usar la lastcomm(1). `lastcomm` imprime órdenes ejecutadas por los usuarios en ttys(5) específicas. Veamos un ejemplo:

```
# lastcomm ls
 trhodes ttyp1
```

Imprimiría todas las veces (conocidas) que el usuario `trhodes` ha usado `ls` en la terminal ttyp1.

Hay muchas más opciones que pueden serle muy útiles. Si quiere conocerlas consulte las páginas de manual lastcomm(1), acct(5) y sa(8).

Capítulo 15. Jaulas

Escrito por Matteo Riondato.

15.1. Sinopsis

En este capítulo se explica qué son las jaulas en FreeBSD y cómo usarlas. Las jaulas, citadas con frecuencia como la nueva generación de *entornos chroot*, son una herramienta muy poderosa que se ha puesto al servicio de los administradores de sistemas, aunque su uso más básico puede ser también de suma utilidad para usuarios avanzados.

Tras leer este capítulo sabrá usted:

- Qué es una jaula y para qué puede usarse en sistemas FreeBSD.
- Cómo generar, arrancar y parar una jaula.
- Cómo manejarse con los rudimentos de la administración de las jaulas, tanto desde dentro como desde fuera de la jaula.

Otras fuentes de información útil sobre las jaulas:

- La página de manual jail(8). Es la referencia completa de `jail`, la herramienta administrativa de FreeBSD con la que se arrancan, paran y controlan las jaulas.
- Las listas de correo y sus respectivos archivos. Los archivos de la lista de correo para preguntas generales sobre FreeBSD, entre otras listas de correo alojadas en el servidor de listas de FreeBSD contienen una enorme cantidad de información sobre jaulas. La ayuda que está buscando puede obtenerla, por tanto, de una búsqueda en los archivos de las listas o de enviar una pregunta que nadie haya hecho en la lista de correo freebsd-questions.

15.2. Términos relacionados con las jaulas

Para ayudar a comprender las partes de FreeBSD que intervienen en el funcionamiento de las jaulas, su funcionamiento interno y el modo en que interactuan con el resto de FreeBSD, durante el resto del capítulo se utilizarán los siguientes términos:

chroot(2) (comando)
: Es una llamada al sistema de FreeBSD que restringe el directorio raiz de un proceso y sus hijos.

chroot(2) (entorno)
: Es el entorno de procesos que se ejecutan en un "chroot". Esto incluye recursos como la parte visible del sistema de ficheros, los ID de usuario y grupo disponibles, interfaces de red u otros mecanismos IPC.

jail(8) (comando)
: La herramienta de administración que permite arrancar procesos dentro del entorno de una jaula.

servidor (sistema, proceso, usuario, etc)
: El sistema que controla una jaula. El servidor tiene acceso a todos los recursos de hardware y puede controlar procesos tanto dentro como fuera de la jaula. Una de las diferencias importantes entre el sistema que aloja la jaula y la jaula propiamente dicha: las limitaciones que afectan a los procesos que se ejecutan con privilegios de superusuario dentro de la jaula no dependen de los procesos del servidor que la aloja.

enjaulado (sistema, proceso, usuario, etc.)
: Un proceso, usuario u otra entidad, cuyo acceso a los recursos está restringido por una jaula de FreeBSD.

15.3. Introducción

Dado lo dificil y desconcertante de la tarea de administrar sistemas se han ido desarrollando poderosas herramientas con el fin de hacer la vida del administrador más sencilla. Dichas herramientas suelen facilitar cierto tipo de mejoras en la instalación, configuración o mantenimiento de los sistemas. Una de las tareas que se espera que cumpla un administrador de sistemas es la configuración adecuada de la seguridad, de modo que pueda dar el servicio para el que se ha destinado sin que pueda verse comprometido.

Una de las herramientas disponibles para mejorar los niveles de seguridad de un sistema FreeBSD es el uso de *jaulas*. Las jaulas fueron introducidas en FreeBSD 4.X por Poul-Henning Kamp, pero en FreeBSD 5.X sus capacidades fueron aumentadas hasta hacer de ellas un subsistema poderoso y flexible. Su desarrollo sigue avanzando, aumentando así su utilidad, rendimiento, fiabilidad y seguridad.

15.3.1. Qué es una jaula

Los sistemas tipo BSD disponen de chroot(2) desde la época de 4.2BSD. chroot(8) permite restringir el directorio raiz de un conjunto de procesos, creando un entorno seguro y separado del resto del sistema. Los procesos creados dentro de un entorno chroot no pueden acceder a ficheros o recursos ubicados fuera del mismo. Por esta razón, si un atacante logra comprometer un servicio que se ejecuta en un entorno chroot no debería automáticamente poder acceder al resto del sistema. chroot(8) es una buena herramienta para tareas sencillas que no requieran mucha flexibilidad o características complejas o muy avanzadas. Por desgracia, desde la invención de chroot se han ido encontrando muchas

formas de saltarse las barreras que chroot impone y, aunque estén corregidas en las versiones más modernas del kernel de FreeBSD, era evidente que chroot(2) no era la solución ideal para ejecutar servicios con seguridad. Había que implementar un nuevo subsistema.

Este es uno de los principales motivos por los que se crearon las *jaulas*.

Las jaulas llevan más allá en muchos sentidos el concepto tradicional de entorno chroot(2). En un entorno chroot(2) tradicional los procesos solo ven limitada la parte del sistema de ficheros a la que pueden acceder. El resto de recursos del sistema, es decir, el conjunto de usuarios del sistema, los procesos en ejecución o el subsistema de red están compartidos entre el sistema alojado y el servidor. Las jaulas extienden este modelo virtualizando no solamente el acceso al sistema de ficheros, sino al conjunto de usuarios, al subsistema de red del kernel de FreeBSD y unas cuantas cosas más. En la Sección 15.5, "Administración y personalización a fondo" se detallan diversas opciones de control exhaustivo para configurar el acceso a recursos de un entorno enjaulado.

Una jaula se caracteriza por disponer de cuatro elementos:

- Un "subárbol" de directorios: el punto desde el que se entra a una jaula. Una vez dentro de la jaula un proceso no puede escapar de dicho "subárbol". Los típicos problemas de seguridad que aparecín una y otra vez en el diseño del chroot(2) original no afectan a las jaulas de FreeBSD.
- Un nombre de máquina ("hostname"), que definirá a la jaula. Las jaulas se usan principalmente para albergar servicios de red, por lo que disponer de un nombre de máquina descriptivo ayuda enormemente al administrador de sistemas.
- Una dirección IP: debe asignarse a la jaula y no cambiarse durante el ciclo de vida de la jaula. La dirección IP de una jaula suele ser un alias de un interfaz de red, aunque no es imprescindible que así sea.
- Un comando: La ruta de un ejecutable ubicado dentro de la jaula. La ruta es relativa al directorio raiz de la jaula, por lo que puede ser muy diferentes según el entorno.

Además, las jaulas pueden tener sus propios usuarios e incluso su propio `root`. Es obvio que este usuario `root` tiene su poder para hacer circunscrito a la jaula y, desde el punto de vista del servidor, el usuario `root` de la jaula no es omnipotente. El usuario `root` de la jaula no puede ejecutar tareas críticas fuera de la jaula (jail(8)) a la que pertenece. Más adelante, en la Sección 15.5, "Administración y personalización a fondo", se dará más información sobre las restricciones del usuario `root`.

15.4. Creación y gestión de jaulas

Algunos administradores dividen las jaulas en dos tipos: jaulas "completas", que recrean un sistema FreeBSD real, y jaulas "de servicio", que son aquellas que están dedicadas a una sola aplicación o servicio, en muchos casos ejecutándose sin privilegios. Se trata de una

división exclusivamente conceptual, por lo que el proceso de generación de una jaula no se ve afectado por ella. La página de manual jail(8) explica claramente el procedimiento a seguir para generar una jaula:

```
# setenv D /aquí/está/la/jaula
# mkdir -p $D ❶
# cd /usr/src
# make world DESTDIR=$D ❷
# cd etc/ [1]
# make distribution DESTDIR=$D ❸
# mount_devfs devfs $D/dev ❹
```

❶ El mejor punto de partida es la elección del punto del sistema de ficheros del servidor donde estará físicamente ubicada la jaula. `/usr/jail/nombredelajaula` es un buen sitio. *`nombredelajaula`* es el nombre de máquina que identifica a la jaula. El sistema de ficheros `/usr/` suele tener espacio suficiente para albergar el sistema de ficheros de la jaula que, cuando se trata de jaulas "completas", es esencialmente lo necesario para alojar todos y cada uno de los sistemas de ficheros en una instalación del sistema base por omisión de FreeBSD.

❷ Este comando creará el contenido necesario (binarios, bibliotecas, páginas de manual, etc.) y lo copiará al "subárbol" elegido como ubicación física de la jaula. Todo se hace al típico estilo FreeBSD: se compila todo y luego se instala en la ruta de destino.

❸ Al pasar el "target" `distribution` a make se instalan todos los ficheros de configuración necesarios. En pocas palabras, instala cada fichero instalable que haya en `/usr/src/etc/` en el directorio `/etc` de la jaula, es decir, en `$D/etc/`.

❹ No es imprescindible montar el sistema de ficheros devfs(8) dentro de la jaula aunque por otra parte (casi) todas las aplicaciones necesitan acceso al menos a un dispositivo, dependiendo esto del propósito de la aplicación. Es muy importante el control del acceso a dispositivos desde la jaula, puesto que una configuración descuidada puede permitir que un atacante haga de las suyas. El control sobre devfs(8) se gestiona mediante reglas que se detallan en las páginas de manual devfs(8) y devfs.conf(5).

Una vez instalada la jaula puede arrancarla mediante jail(8). jail(8) usa los cuatro argumentos que se detallan en la Sección 15.3.1, "Qué es una jaula". Puede pasarle otros argumentos además de estos, por ejemplo para ejecutar procesos enjaulados bajo los permisos de un usuario específico. El argumento *`comando`* depende del tipo de jaula; si se trata de un *virtual system* `/etc/rc` es una buena elección, puesto que ejecutará la secuencia de arranque de un sistema FreeBSD real. Si se trata de una jaula *de servicio* depende del servicio o aplicación que se quiera ejecutar mediante la jaula.

Con frecuencia las jaulas se arrancan durante el arranque del servidor que las aloja; el sistema `rc` de FreeBSD permite hacerlo de un modo muy sencillo.

1. Puede crear una lista de jaulas que quiera arrancar en el inicio del sistema en el fichero rc.conf(5):

[1] Este paso no es necesario en FreeBSD 6.0 y versiones posteriores.

```
jail_enable="YES"   # Ponga NO si quiere desactivar el arranque ↺
de jaulas
jail_list="www"     # Lista de nombres de jaulas separados por ↺
espacios
```

2. Tendrá que añadir parámetros específicos para cada jaula al fichero rc.conf(5):

```
jail_www_rootdir="/usr/jail/www"     # directorio raiz de la jaula
jail_www_hostname="www.example.org"  # nombre de máquina de la ↺
jaula
jail_www_ip="192.168.0.10"           # dirección IP de la jaula
jail_www_devfs_enable="YES"          # montar devfs en la jaula
jail_www_devfs_ruleset="www_ruleset " # reglas a aplicar a devfs ↺
dentro de la jaula
```

   El arranque de jaulas por omisión que se configure en rc.conf(5) ejecutará el script `/etc/rc` de la jaula y asumirá que es un sistema virtual completo. Si se trata de una jaula de servicio el comando de arranque por omisión tendrá que cambiarse configurando la opción `jail_`*`nombredejaula`*`_exec_start` según convenga.

> **Nota**
>
> Si quiere consultar la lista completa de opciones consulte la página de manual rc.conf(5).

Puede arrancar o parar a mano una jaula mediante el script `/etc/rc.d/jail` siempre y cuando la jaula aparezca en `rc.conf`:

```
# /etc/rc.d/jail start www
# /etc/rc.d/jail stop www
```

De momento no hay una forma limpia de apagar una jaula (jail(8)) debido a que los comandos que se usan normalmente para producir un apagado limpio del sistema no pueden usarse dentro de una jaula. La mejor forma de parar una jaula es ejecutar el siguiente comando desde dentro de la propia jaula o bien mediante jexec(8) desde fuera:

```
# sh /etc/rc.shutdown
```

Para más información consulte la página de manual jail(8).

15.5. Administración y personalización a fondo

Hay diversas opciones que pueden usarse en las jaulas y varios tipos de formas de combinar un sistema FreeBSD servidor y las jaulas y poder disponer de aplicaciones de alto nivel. En esta sección se muestra lo siguiente:

- Algunas de las opciones disponibles para personalizar el comportamiento y las restricciones de seguridad que pueden aplicarse en una jaula.
- Algunas de las aplicaciones de alto nivel creadas para la administración de jaulas. Estas aplicaciones están en la colección de ports y pueden utilizarse en conjunto para implementar productos basados en jaulas.

15.5.1. Herramientas del sistema para la personalización de jaulas en FreeBSD

La personalización a fondo de las jaulas se hace en su mayor parte mediante la configuración de variables sysctl(8). Hay una subcategoría especial de sysctl para que sea más sencillo organizar las opciones más más importantes: se trata de las opciones de la jerarquía `security.jail.*` del kernel de FreeBSD. A continuación veremos una lista de las principales sysctl relacionadas con las jaulas y los valores que tienen por omisión. Los nombres deberían describir por sí mismos qué función tienen (N. del T.: En inglés, claro) pero si necesita más información sobre ellas consulte las páginas de manual jail(8) y sysctl(8).

- `security.jail.set_hostname_allowed: 1`
- `security.jail.socket_unixiproute_only: 1`
- `security.jail.sysvipc_allowed: 0`
- `security.jail.enforce_statfs: 2`
- `security.jail.allow_raw_sockets: 0`
- `security.jail.chflags_allowed: 0`
- `security.jail.jailed: 0`

El administrador del *servidor* puede usar estas variables para añadir o quitar limitaciones impuestas por omisión al usuario `root`. Tenga en cuenta que hay ciertas limitaciones que no pueden quitarse. El usuario `root` no puede montar o desmontar sistemas de ficheros desde su jaula. El usuario `root` no puede cargar o descargar reglas de devfs(8), configurar reglas de cortafuegos ni ejecutar muchas otras tareas administrativas que requieran modificaciones o acceso a datos internos del kernel, como cambiar el nivel de seguridad `securelevel` del kernel.

El sistema base de FreeBSD contiene un conjunto básico de herramientas que permiten el acceso a información sobre jaulas activas en el sistema, así como la conexión a una jaula

para ejecutar comandos administrativos. jls(8) y jexec(8) forman parte del sistema base de FreeBSD y permiten ejecutar las siguientes tareas:

- Mostrar una lista de jaulas activas y sus correspondientes identificadores de jaula (JID), dirección IP, nombre de máquina y ruta.

- Conectarse a una jaula en ejecució desde el servidor y ejecutar un comando dentro de la jaula o realizar tareas administrativas dentro de dicha jaula. Esto es muy útil cuando el usuario `root` quiere apagar la jaula de forma limpia. La herramienta jexec(8) permite también arrancar una shell dentro de la jaula para realizar tareas administrativas. Veamos un ejemplo:

```
# jexec 1 tcsh
```

15.5.2. Herramientas para tareas administrativas de alto nivel en la Colección de Ports

Entre las variadas aplicaciones ajenas al Proyecto FreeBSD que han ido apareciendo para administrar jaulas una de las más completas y útiles es sysutils/jailutils. Es un conjunto de pequeñas aplicaciones de mucha ayuda en la gestión de una jaula (jail(8)). Por favor, consulte su página web para más información.

15.6. Uso de las jaulas

15.6.1. Jaulas “de servicio”

Escrito por Daniel Gerzo.

Esta sección está basada en una idea que Simon L. B. Nielsen presentó por primera vez en `http://simon.nitro.dk/service-jails.html` y en un artículo con contenido adicional escrito por Ken Tom <`locals@gmail.com` >. En esta sección se detalla cómo configurar un sistema FreeBSD que añade una capa adicional de seguridad mediante el uso de jail(8). Para su verdadero aprovechamiento se asume que el sistema en el que se vaya a aplicar ejecuta al menos RELENG_6_0 y que la información que contienen las secciones previas de este capítulo se ha comprendido totalmente.

15.6.1.1. Diseño

Uno de los mayores problemas de las jaulas es la gestión de su proceso de actualización. Este proceso tiene a ser un problema porque cada jaula tiene que recompilarse íntegramente desde el código fuente cada vez que hay que actualizarla. Esto no es un gran problema si tenemos una sola jaula puesto que el proceso de actualización es bastante simple, pero si hay muchas jaulas será un trabajo largo y tedioso.

Aviso

: Esta configuración requiere mucha experiencia con FreeBSD y el uso de sus características. Si los pasos que se detallan a continuación le parecen demasiado complicados puede echar un vistazo a sistemas más sencillos como sysutils/ezjail, que le permitirá acceder a un método de administración de jaulas en FreeBSD más sencillo y no es tan sofisticado como el que le proponemos a continuación.

El origen de esta idea es resolver los problemas antes descritos compartiendo el máximo posible entre distintas jaulas, de un modo seguro (utilizando montajes using read-only mount_nullfs(8) mounts) para que la actualización sea más sencilla y el ubicar servicios aislados en jaulas sea más interesante. Además, se presenta una forma sencilla de añadir o borrar jaulas así como una forma de actualizarlas.

Nota

Los ejemplos de servicios en este contexto son: un servidor HTTP,un servidor DNS, un servidor SMTP, etc.

Los objetivos de la configuración descrita en esta sección son:

- Crear una estructura de jaulas simple y fácil de entender. Esto implica *no* tener que ejecutar un “installworld” completo en todas y cada una de las jaulas.
- Facilitar la creación de nuevas jaulas o el borrado de jaulas previamente existentes.
- Facilitar la actualización de jaulas ya existentes.
- Hacer posible el uso de una rama de FreeBSD personalizada.
- Ser paranoico en cuanto a seguridad, reduciendo todo lo posible la posibilidad de que los sistemas se vean comprometidos.
- Ahorrar todo el espacio e inodos que sea posible.

Como ya se ha dicho, este diseño se basa en gran medida en el disponer de una única plantilla en modo de sólo lectura (a la que llamaremos nullfs) montada en cada jaula y un dispositivo en modo lectura-escritura por cada jaula. El dispositivo puede ser otro disco físico adicional, una partición o un dispositivo md(4) basado en un vnode. En este ejemplo utilizaremos montajes nullfs en modo lectura-escritura.

La estructura del sistema de ficheros se detalla en la siguiente lista:

- Cada jaula se montará bajo `/home/j`.
- `/home/j/mroot` será la plantilla para cada jaula y la partición de sólo lectura para todas las jaulas.
- Se creará un directorio vacío para cada jaula bajo el directorio `/home/j`.
- Cada jaula tendrá un directorio `/s` que estará enlazado con la parte de lectura-escritura del sistema.
- Cada jaula tendrá su propio sistema en modo lectura-escritura basado en `/home/j/skel`.
- Cada parte de lectura-escritura correspondiente a cada jaula se creará en `/home/js`.

Nota

Se asume que las jaulas se instalarán bajo la partición `/home`. Por supuesto esto no es en absoluto obligatorio, pero hay que tener en cuenta que debe hacerse el mismo cambio en cada uno de los ejemplos que se muestran más adelante.

15.6.1.2. Creación de la plantilla

En esta sección se describen los pasos necesarios para crear la plantilla maestra que conformará la parte de sólo lectura que usarán las jaulas.

Siempre es recomendable actualizar el sistema FreeBSD a la última rama -RELEASE. Consulte el capítulo correspondiente de este libro si necesita más información. En caso de que la actualización no sea posible tendrá que usar "buidworld" para poder seguir adelante. También necesitará el paquete sysutils/cpdup. Usaremos portsnap(8) para descargar la Colección de Ports de FreeBSD. El capítulo sobre Portsnap es siempre una lectura muy recomendable para quienes no tengan experiencia con su funcionamiento.

1. Lo primero que haremos será crear una estructura de directorios para el sistema de ficheros de sólo lectura que contendrá los binarios de nuestras jaulas, luego iremos al directorio que contiene el árbol de código de FreeBSD e instalaremos el sistema de ficheros de sólo lectura en la plantilla de las jaulas:

```
# mkdir /home/j /home/j/mroot
# cd /usr/src
# make installworld DESTDIR=/home/j/mroot
```

2. Una vez hecho esto, prepararemos la Colección de Ports de FreeBSD para nuestras jaulas así como un árbol de código FreeBSD, necesario para usar mergemaster:

```
# cd /home/j/mroot
# mkdir usr/ports
# portsnap -p /home/j/mroot/usr/ports fetch extract
# cpdup /usr/src /home/j/mroot/usr/src
```

3. Crear la estructura de directorios necesaria para la parte de lectura-escritura del sistema:

```
# mkdir /home/j/skel /home/j/skel/home /home/j/skel/usr-X11R6 /
home/j/skel/distfiles
# mv etc /home/j/skel
# mv usr/local /home/j/skel/usr-local
# mv tmp /home/j/skel
# mv var /home/j/skel
# mv root /home/j/skel
```

4. Usamos mergemaster para instalar los ficheros de configuración que falten. Después nos libramos de los directorios adicionales que haya creado mergemaster:

```
# mergemaster -t /home/j/skel/var/tmp/temproot -D /home/j/skel -i
# cd /home/j/skel
# rm -R bin boot lib libexec mnt proc rescue sbin sys usr dev
```

5. Ahora enlazamos simbólicamente el sistema de ficheros de lectura-escritura con el sistema de ficheros de sólo lectura. Por favor, asegúrese de que los enlaces simbólicos se crean en las ubicaciones correctas: `s/`. Si se usan directorios reales o directorios erróneos la instalación no funcionará.

```
# cd /home/j/mroot
# mkdir s
# ln -s s/etc etc
# ln -s s/home home
# ln -s s/root root
# ln -s ../s/usr-local usr/local
# ln -s ../s/usr-X11R6 usr/X11R6
# ln -s ../../s/distfiles usr/ports/distfiles
# ln -s s/tmp tmp
# ln -s s/var var
```

6. Como último paso, cree un `/home/j/skel/etc/make.conf` genérico con el siguiente contenido:

```
WRKDIRPREFIX?=  /s/portbuild
```

   El tener `WRKDIRPREFIX` configurado de este modo hará posible compilar ports de FreeBSD dentro de cada jaula. Recuerde que el el directorio de los ports es de sólo lectura. La ruta personalizada por `WRKDIRPREFIX` permite ejecutar compilaciones en la parte de sólo lectura de cada jaula.

15.6.1.3. Creación de las jaulas

Ya tenemos una plantilla de jaulas de FreeBSD completa, así que podemos configurar nuestras jaulas en `/etc/rc.conf` . En este ejemplo crearemos 3 jaulas: “NS”, “MAIL” y “WWW”.

1. Introduzca las siguientes lineas en el fichero `/etc/fstab` ; con esto cada jaula tendrá acceso a la plantilla de sólo lectura y al espacio de lectura-escritura:

```
/home/j/mroot   /home/j/ns     nullfs  ro  0   0
/home/j/mroot   /home/j/mail   nullfs  ro  0   0
/home/j/mroot   /home/j/www    nullfs  ro  0   0
/home/js/ns     /home/j/ns/s   nullfs  rw  0   0
/home/js/mail   /home/j/mail/s nullfs  rw  0   0
/home/js/www    /home/j/www/s  nullfs  rw  0   0
```

Nota

Las particiones que tienen un 0 en la columna “pass” no serán revisadas por fsck(8) durante el arranque y las que tienen un 0 en la columna “dump” no serán copiadas por dump(8). No nos interesa que fsck compruebe la integridad de montajes nullfs ni que dump haga copias de seguridad de montajes nullfs de sólo lectura de las jaulas. Por esta razón el ejemplo de `fstab` tiene en las dos últimas columnas “0 0”.

2. Configure las jaulas en `/etc/rc.conf` :

```
jail_enable="YES"
jail_set_hostname_allow="NO"
jail_list="ns mail www"
jail_ns_hostname="ns.ejemplo.org"
jail_ns_ip="192.168.3.17"
jail_ns_rootdir="/usr/home/j/ns"
jail_ns_devfs_enable="YES"
jail_mail_hostname="mail.ejemplo.org"
jail_mail_ip="192.168.3.18"
jail_mail_rootdir="/usr/home/j/mail"
jail_mail_devfs_enable="YES"
jail_www_hostname="www.ejemplo.org"
jail_www_ip="62.123.43.14"
jail_www_rootdir="/usr/home/j/www"
jail_www_devfs_enable="YES"
```

Aviso

: La razón por la que `jail_nombre_rootdir` contiene `/usr/home` y no `/home` es que la ruta física del directorio `/home` en una instalación de FreeBSD por omisión es `/usr/home`. La variable `jail_nombre_rootdir` *no* debe apuntar a una ruta que contenga un enlace simbólico porque sería imposible arrancar las jaulas. Utilice la herramienta realpath(1) para asegurarse del valor exacto que debe asignar a la variable. Por favor, consulte el aviso de seguridad FreeBSD-SA-07:01.jail para más información.

3. Creamos los puntos de montaje de sistemas de ficheros de sólo lectura correspondientes a cada jaula:

```
# mkdir /home/j/ns /home/j/mail /home/j/www
```

4. Instalamos la plantilla de lectura-escritura dentro de cada jaula. Observe que utilizamos sysutils/cpdup para asegurarnos de que se hace una copia exacta de cada directorio:

```
# mkdir /home/js
# cpdup /home/j/skel /home/js/ns
# cpdup /home/j/skel /home/js/mail
# cpdup /home/j/skel /home/js/www
```

5. Llegados a este punto las jaulas están configuradas y listas para arrancar. Monte los sistemas de ficheros de cada jaula y luego arránquelas con el script `/etc/rc.d/jail` :

```
# mount -a
# /etc/rc.d/jail start
```

Las jaulas deberían haber arrancado. Asegúrese de ello con jls(8). La salida que verá debe parecerse a esta:

```
# jls
   JID  IP Address      Hostname                      Path
     3  192.168.3.17    ns.ejemplo.org                /home/j/ns
     2  192.168.3.18    mail.ejemplo.org              /home/j/mail
     1  62.123.43.14    www.ejemplo.org               /home/j/www
```

En este punto debería ser posible entrar a cada una de las jaulas, añadir nuevos usuarios o configurar dæmons. La columna `JID` indica el número de identificación de cada jaula

que esté funcionando en el sistema. Con el siguiente comando puede ejecutar tareas administrativas en la jaula cuyo `JID` sea 3:

```
# jexec 3 tcsh
```

15.6.1.4. Actualización

Llegará el momento en el que sea necesario actualizar el sistema, bien por seguridad o porque sea útil para las jaulas disponer de alguna nueva característica del sistema. El diseño de esta configuración facilita una forma fácil de actualizar sus jaulas. Además, minimiza la pérdida de servicio, puesto que las jaulas deben apagarse sólamente al final de todo el proceso. Se ofrece también la posibilidad de volver a la versión anterior en caso de que algo salga mal.

1. El primer paso es actualizar el servidor que aloja las jaulas de la forma habitual. Después creamos una plantilla de sólo lectura temporal en `/home/j/mroot2` .

   ```
   # mkdir /home/j/mroot2
   # cd /usr/src
   # make installworld DESTDIR=/home/j/mroot2
   # cd /home/j/mroot2
   # cpdup /usr/src usr/src
   # mkdir s
   ```

   La ejecución de `installworld` crea unos cuantos directorios innecesarios que debemos borrar:

   ```
   # chflags -R 0 var
   # rm -R etc var root usr/local tmp
   ```

2. Creamos de nuevo los enlaces simbólicos de lectura-escritura del sistema de ficheros principal:

   ```
   # ln -s s/etc etc
   # ln -s s/root root
   # ln -s s/home home
   # ln -s ../s/usr-local usr/local
   # ln -s ../s/usr-X11R6 usr/X11R6
   # ln -s s/tmp tmp
   # ln -s s/var var
   ```

3. Ha llegado el momento de parar las jaulas:

   ```
   # /etc/rc.d/jail stop
   ```

4. Desmontamos los sistemas de ficheros originales:

   ```
   # umount /home/j/ns/s
   # umount /home/j/ns
   # umount /home/j/mail/s
   # umount /home/j/mail
   ```

```
# umount /home/j/www/s
# umount /home/j/www
```

Nota

Los sistemas de ficheros de lectura-escritura cuelgan del sistema de sólo lectura `/s` y por tanto deben desmontarse antes.

5. Movemos el sistema de ficheros de sólo lectura viejo y lo reemplazamos por el nuevo. Nos servirá de copia de seguridad y como archivo en caso de que haya problemas. Para darle un nombre usamos la fecha en la que se creado una nueva copia del sistema de ficheros de sólo lectura. Movemos también la Colección de Ports de FreeBSD al sistema de ficheros nuevo para ahorrar un poco más de espacio e inodos:

```
# cd /home/j
# mv mroot mroot.20060601
# mv mroot2 mroot
# mv mroot.20060601/usr/ports mroot/usr
```

6. Una vez llegados a este punto la nueva plantilla de sólo lectura está lista, de manera que lo único que nos queda por hacer es montar los sistemas de ficheros y arrancar las jaulas:

```
# mount -a
# /etc/rc.d/jail start
```

Compruebe con jls(8) si las jaulas han arrancado sin contratiempos. No olvide ejecutar mergemaster en cada jaula. Tendrá que actualizar tanto los ficheros de configuración como los scripts rc.d.

Capítulo 16. Mandatory Access Control

Escrito por Tom Rhodes.

16.1. Sinopsis

Pendiente de Traducción

16.2. Términos clave en este capítulo

Pendiente de traducción

16.3. Explicación de MAC

Pendiente de traducción

16.4. Las etiquetas MAC

Pendiente de traducción

16.5. Configuración de módulos

Pendiente de traducción

16.6. El módulo MAC ifoff

Pendiente de traducción

16.7. El módulo MAC portacl

Pendiente de traducción

16.8. Políticas de etiquetas MAC

Pendiente de traducción

16.9. El módulo MAC partition

Pendiente de traducción

16.10. El módulo de seguridad multinivel MAC

Pendiente de traducción

16.11. El módulo MAC Biba

Pendiente de traducción

16.12. El módulo MAC LOMAC

Pendiente de traducción

16.13. Implementación de un entorno seguro con MAC

Pendiente de traducción

16.14. Otro ejemplo: Uso de MAC para restringir un servidor web

Pendiente de traducción

16.15. Depuración de errores en MAC

Pendiente de traducción

Capítulo 17. Auditoría de eventos de seguridad

Escrito por Tom Rhodes y Robert Watson.

17.1. *

Pendiente de traducción.

Capítulo 18. Almacenamiento

18.1. Sinopsis

Este capítulo trata sobre el uso de discos en FreeBSD. Esto incluye discos basados en memoria, discos conectados a través de la red, dispositivos de almacenamiento SCSI/IDE estándar y dispositivos que utilizan el interfaz USB.

Tras leer este capítulo:

- Conocerá la terminología que se usa en FreeBSD para describir la organización de datos en un disco físico (particiones y porciones).
- Sabrá cómo añadir discos duros a su sistema.
- Sabrá cómo configurar FreeBSD para utilizar dispositivos de almacenamiento USB.
- Sabrá cómo configurar sistemas virtuales de ficheros, como los discos de memoria.
- Sabrá cómo usar cuotas para limitar el uso del espacio en disco.
- Sabrá cómo cifrar discos para hacerlos más seguros ante un atacante.
- Sabrá cómo se crean y graban los CD y DVD en FreeBSD.
- Conocerá diversas opciones de almacenamiento de copias de seguridad.
- Sabrá cómo usar diversos programas de respaldo que pueden utilizarse en FreeBSD.
- Sabrá cómo hacer copias de seguridad utilizando disquetes (floppy).
- Sabrá en qué consiste una instantánea (“snapshot”) y cómo utilizarla de forma eficiente.

Antes de leer este capítulo:

- Debe saber cómo configurar e instalar un nuevo kernel en FreeBSD (Capítulo 8, *Configuración del kernel de FreeBSD*).

18.2. Nombres de dispositivo

A continuación le mostraremos una lista de dispositivos de físicos almacenamiento soportados por FreeBSD y los nombres de dispositivo asociados con ellos.

Tabla 18.1. Convenciones para nombrar discos físicos

Tipo de unidad	Nombre de dispositivo de la unidad
Discos duros IDE	`ad`
Unidades CDROM IDE	`acd`
Discos duros SCSI y dispositivos de almacenamiento masivo USB	`da`
Unidades CDROM SCSI	`cd`
Diferentes tipos de unidades CDROM no estándares	`mcd` para CD-ROM Mitsumi, `scd` para CD-ROM Sony, `matcd` para CD-ROM Matsushita/Panasonic
Unidades de disquete (floppy)	`fd`
Unidades de cinta SCSI	`sa`
Unidades de cinta IDE	`ast`
Unidades Flash	`fla` para dispositivos DiskOnChip®
Unidades RAID	`aacd` para Adaptec® AdvancedRAID, `mlxd` y `mlyd` para Mylex®, `amrd` para AMI MegaRAID®, `idad` para Compaq Smart RAID, `twed` para 3ware® RAID.

18.3. Añadir discos

Texto original de David O'Brien.

Digamos que queremos añadir un nuevo disco SCSI a una máquina que solo tiene un disco. Comience por apagar el sistema e instale el disco siguiendo las instrucciones del fabricante de la computadora, del disco y de la controladora. Debido a la gran variedad de procedimientos posibles los detalles están más allá del alcance de este texto.

Entre como usuario `root`. Una vez instalado el disco inspeccione `/var/run/dmesg.boot` para asegurarse de que el sistema encontró el nuevo disco. Continuando con nuestro ejemplo, el disco recién añadido será `da1` y queremos montarlo en `/1` (si está añadiendo un disco IDE, el nombre de dispositivo será `wd1` en sistemas anteriores a 4.0, y `ad1` en sistemas 4.X y 5.X).

FreeBSD funciona en computadoras IBM-PC y compatibles, por lo tanto tendrá en cuenta las particiones de la BIOS del PC, que son diferentes del tipo de partición que se ha venido usando en BSD. Un disco para PC puede contener hasta cuatro entradas de particiones BIOS. Si el disco va a utilizarse íntegramente con FreeBSD puede usar el modo *dedicado*. Si no, FreeBSD tendrá que instalarse dentro de una las particiones BIOS. En FreeBSD se llama slices (“porciones” o “rebanadas”) a las particiones de PC BIOS para no confundirlas con

las particiones BSD. También puede utilizar slices en un disco dedicado a FreeBSD pero que se está usando en un sistema que también tiene otro sistema operativo instalado. Esta es una buena manera de evitar confundir la versión de `fdisk` de otros sistemas operativos.

Desde el punto de vista de las slices el disco se añadirá como `/dev/da1s1e` . Se interpreta del siguiente modo: disco SCSI, unidad número 1 (segundo disco SCSI), slice 1 (partición 1 de PC BIOS), y partición BSD `e`. Si es un disco dedicado, el disco se añadirá como `/dev/da1e`.

Debido al uso de enteros de 32-bits para almacenar el número de sectores, bsdlabel(8) (llamado disklabel(8) en FreeBSD 4.X) está limitado a 2^32-1 sectores por disco ó 2TB (en la mayoría de los casos). El formato de fdisk(8) permite un sector de arranque de un máximo de más de 2^32-1 y no más de 2^32-1 de longitud, limitando las particiones a 2TB y los discos a 4TB (también en la mayoría de los casos). El formato sunlabel(8) tiene una limitación de 2^32-1 sectores por partición y 8 particiones en un espacio máximo de 16TB. Si va a usar discos mayores puede usar particiones gpt(8).

18.3.1. Uso de sysinstall(8)

1. Navegar en Sysinstall

   Puede utilizar `sysinstall` (`/stand/sysinstall` en versiones de FreeBSD anteriores a 5.2) para particionar y etiquetar un disco nuevo usando sus intuitivos menús. Entre como el usuario `root` o utilice `su`. Ejecute `sysinstall` y entre al menú `Configure`. Dentro de `FreeBSD Configuration Menu`, descienda y seleccione la opción `Fdisk`.

2. Editor de particiones fdisk

   Una vez dentro de fdisk, teclée **`A`** si quiere usar el disco entero con FreeBSD Cuando se le pregunte “remain cooperative with any future possible operating systems” [1], responda `YES`. Escriba los cambios al disco pulsando **`W`**. Salga del editor FDISK pulsando **`q`**. A continuación se le preguntará sobre el “Master Boot Record”. Debido a que está añadiendo un nuevo disco a un sistema que ya está instalado, tendrá que seleccionar `None`.

3. Editor de etiquetas de disco

   A continuación, debe salir de sysinstall e iniciarlo de nuevo. Siga las instrucciones arriba expuestas, pero esta vez elija la opción `Label`. De este modo accederá al `editor de etiquetas de disco`. En él creará las particiones BSD tradicionales. Un disco puede tener hasta ocho particiones, etiquetadas desde la `a` a la `h`. Algunas de las etiquetas de las particiones tienen usos especiales. La partición `a` se utiliza para la partición raíz (`/`), por lo tanto sólo su disco de sistema (esto es, el disco desde el cual arranca) tendrá una partición `a`. La partición `b` se usa como partición swap; puede

[1] Mantener el disco accesible a sistemas operativos que pudieran necesitar acceder al mismo en algún momento.

tener más de una partición swap y puede alojarlas en más de un disco. La partición `c` hace referencia al disco entero en modo dedicado, o a la slice de FreeBSD completa en modo slice. Las demás particiones son para el resto de los usos típicos.

El editor de etiquetas de sysinstall creará la partición `e` como partición "ni raíz, ni swap". En el editor de etiquetas crée un solo sistema de ficheros tecleando **`C`**. Cuando se le pregunte si debe etiquetarse como FS (sistema de ficheros) o swap, elija `FS` y teclée un punto de montaje (por ejemplo `/mnt`). Al añadir un disco en modo "post-instalación" sysinstall no creará automáticamente las entradas correspondientes en `/etc/fstab`, por lo que el punto de montaje que usted especifique no tiene importancia.

Ahora puede escribir la nueva etiqueta al disco y crear un sistema de ficheros en él tecleando **`W`**. Ignore cualquier error que pudiera generar sysinstall acerca de dificultades para montar la nueva partición. Salga del editor de etiquetas y de sysinstall.

4. Terminar

   El último paso es editar `/etc/fstab` y añadir una entrada para su nuevo disco.

18.3.2. Uso de utilidades de línea de comandos

18.3.2.1. Uso de slices

Esta configuración le permitirá a su disco convivir sin sobresaltos con otro sistema operativo que pueda estar instalado en su sistema y no confundirá a las utilidades `fdisk` de esos otros sistemas operativos. Se recomienda utilizar este método para instalar discos nuevos. *Utilice el modo dedicado sólamente si tiene un buen motivo para hacerlo.*

```
# dd if=/dev/zero of=/dev/da1 bs=1k count=1
# fdisk -BI da1  #Initialice el nuevo disco.
# disklabel -B -w -r da1s1 auto  #Etiquételo.
# disklabel -e da1s1  # Edite la etiqueta de disco que acaba de crear ↺
y añada particiones.
# mkdir -p /1
# newfs /dev/da1s1e  # Repita este paso por cada partición que crée.
# mount /dev/da1s1e /1  # Monte la partición o particiones.
# vi /etc/fstab  # Añada la/s entrada/s apropiadas en /etc/fstab .
```

Si tiene un disco IDE, sustituya `ad` por `da`. En sistemas anteriores a 4.X utilice `wd`.

18.3.2.2. Dedicado

Si no va a compartir el nuevo disco con otro sistema operativo puede utilizar el modo `dedicado`. Recuerde que este modo puede confundir a los sistemas operativos de Microsoft, aunque no podrán dañar por ello el disco o su contenido. Tenga en cuenta que FreeBSD (de IBM) se "apropiará" de cualquier partición que encuentre y no entienda.

```
# dd if=/dev/zero of=/dev/da1 bs=1k count=1
# disklabel -Brw da1 auto
# disklabel -e da1     # crear partición `e'
# newfs -d0 /dev/da1e
# mkdir -p /1
# vi /etc/fstab     # agregar una entrada para /dev/da1e
# mount /1
```

Una forma alternativa de hacerlo sería:

```
# dd if=/dev/zero of=/dev/da1 count=2
# disklabel /dev/da1 | disklabel -BrR da1 /dev/stdin
# newfs /dev/da1e
# mkdir -p /1
# vi /etc/fstab      # añadir una entrada para /dev/da1e
# mount /1
```

Nota

A partir de FreeBSD 5.1-RELEASE, la utilidad bsdlabel(8) reemplazó al antiguo programa disklabel(8). En bsdlabel(8) se han eliminado muchos parámetros y opciones obsoletas; en los ejemplos de arriba la opción `-r` debe eliminarse si se usa bsdlabel(8). Para más información diríjase al manual de bsdlabel(8).

18.4. RAID

18.4.1. Software RAID

18.4.1.1. Configuración de controlador de disco concatenado (CCD)

Texto original de Christopher Shumway.
Revisado por Jim Brown.

Al escoger una solución de almacenamiento masivo los factores más importantes a considerar son velocidad, fiabilidad y coste. Es raro tener los tres por igual; normalmente un dispositivo de almacenamiento masivo veloz y fiable es caro, y para recortar los costes suele sacrificarse la velocidad o la fiabilidad.

Al diseñar el sistema descrito más adelante se eligió el coste como el factor más importante, seguido de la velocidad, y luego la fiabilidad. La velocidad de transferencia de datos para este sistema está, en última instancia, limitada por la red. Y mientras que la confiabilidad es muy importante, el controlador CCD descrito más adelante sirve datos que están respaldados en CD-R y pueden ser reemplazados sin dificultad.

Al escoger una solución de almacenamiento masivo el primer paso es definir sus necesidades. Si prefiere velocidad o fiabilidad por encima del coste, el resultado será distinto del que vamos a describir en esta sección.

18.4.1.1.1. Instalación del hardware

Además del disco IDE, el núcleo del disco CCD está compuesto por tres discos IDE discos IDE Western Digital de 30GB y 5400 RPM, que ofrecen aproximadamente 90GB de almacenamiento. Lo ideal sería que cada disco IDE tuviera su propio cable y controlador, pero para minimizar costes no se utilizaron controladores IDE adicionales. En lugar de eso se configuraron los discos con "jumpers" para que cada controlador IDE tuviera un maestro y un esclavo.

Despues de reiniciar la BIOS se configuró para que detectara automáticamente los discos conectados. FreeBSD los detectó al reiniciar:

```
ad0: 19574MB <WDC WD205BA> [39770/16/63] at ata0-master UDMA33
ad1: 29333MB <WDC WD307AA> [59598/16/63] at ata0-slave UDMA33
ad2: 29333MB <WDC WD307AA> [59598/16/63] at ata1-master UDMA33
ad3: 29333MB <WDC WD307AA> [59598/16/63] at ata1-slave UDMA33
```

Nota

Si FreeBSD no detecta todos los discos asegúrese de que ha colocado correctamente los "jumpers". La mayoría de los discos IDE tienen un "jumper" "Cable Select". Este *no* es el "jumper" que define la relación maestro/esclavo. Consulte la documentación del disco para identificar el "jumper" correcto.

El siguiente paso es estudiar cómo conectarlos para que formen parte del sistema de ficheros. Investigue Debe investigar vinum(8) (Capítulo 20, *El Gestor de Volúmenes Vinum*) y ccd(4). Nosotros elegimos ccd(4) para nuestra configuración.

18.4.1.1.2. Configuración de CCD

El controlador ccd(4) le permite tomar varios discos idénticos y concatenarlos en un solo sistema lógico de ficheros. Para poder usar ccd(4) necesita un kernel compilado con soporte de ccd(4). Añada esta línea al fichero de configuración de su kernel, recompile y reinstale su kernel:

```
pseudo-device   ccd     4
```

En sistemas 5.X, use la siguiente línea:

```
device   ccd
```

Nota

En FreeBSD 5.X no es necesario especificar un número de dispositivos ccd(4), ya que el controlador de dispositivo ccd(4) es capaz de clonarse a sí mismo (se crearán nuevas instancias de dispositivo automáticamente según vayan haciendo falta).

El soporte de ccd(4) también puede cargarse como módulo en FreeBSD 3.0 y posteriores.

Para configurar ccd(4) tendrá que usar disklabel(8) para etiquetar los discos:

```
disklabel -r -w ad1 auto
disklabel -r -w ad2 auto
disklabel -r -w ad3 auto
```

Esto crea una etiqueta de disco para `ad1c`, `ad2c` y `ad3c` que abarcan el disco completo.

Nota

A partir de FreeBSD 5.1-RELEASE bsdlabel(8) reemplazó al antiguo programa disklabel(8). En bsdlabel(8) se eliminaron muchas opciones y parámetros obsoletos; en los ejemplos de arriba la opción `-r` deben obviarse. Para más información consulte bsdlabel(8).

El siguiente paso es cambiar el tipo de etiqueta de disco. Edite los discos con disklabel(8):

```
disklabel -e ad1
disklabel -e ad2
disklabel -e ad3
```

Esto abre la etiqueta de disco de cada disco con el editor declarado en la variable de entorno `EDITOR`, por defecto vi(1).

Esta es una etiqueta de disco sin modificar:

```
8 partitions:
#        size   offset    fstype   [fsize bsize bps/cpg]
  c: 60074784        0    unused        0     0     0   # (Cyl.    ↺
0 - 59597)
```

ccd(4) necesita que añada una nueva partición `e`. Puede copiarla desde la partición `c`, pero el tipo de sistema de ficheros (la opción `fstype`) debe ser **`4.2BSD`**. La etiqueta del disco debería tener este aspecto:

```
8 partitions:
#        size   offset    fstype   [fsize bsize bps/cpg]
  c: 60074784        0    unused        0     0     0   # (Cyl.    ↺
0 - 59597)
  e: 60074784        0    4.2BSD        0     0     0   # (Cyl.    ↺
0 - 59597)
```

18.4.1.1.3. Contrucción del sistema de ficheros

Puede que todavía no exista el nodo de dispositivo para `ccd0c`. Si es así, ejecute lo siguiente:

```
cd /dev
sh MAKEDEV ccd0
```

Nota

En FreeBSD 5.0 devfs(5) administrará automáticamente los nodos de dispositivos en `/dev`, así que no tendrá que usar `MAKEDEV`.

Una vez etiquetados todos los discos construya el ccd(4). Utilice ccdconfig(8) con opciones similares a las siguientes:

```
ccdconfig ccd0❶ 32❷ 0❸ /dev/ad1e❹ /dev/ad2e /dev/ad3e
```

El uso y el significado de cada una de las opciones se muestra más abajo:

❶ El primer argumento es el dispositivo a configurar, en este caso `/dev/ccd0c` . La parte `/dev/` es opcional.

❷ El intervalo para el sistema de ficheros. El intervalo define el tamaño de una banda en bloques de disco, normalmente 512 bytes. Por lo tanto, un intervalo de 32 equivaldría 16.384 bytes.

❸ Banderas para ccdconfig(8). Si desea disponer sus discos en espejo use aquí una bandera. Esta configuración no necesita discos en espejo, por lo que está dispuesta a 0 (cero).

❹ Los últimos argumentos de ccdconfig(8) son los dispositivos a colocar en el array. Utilice la ruta completa para cada dispositivo.

Despues de ejecutar ccdconfig(8) el ccd(4) estará configurado y podrá instalar un sistema de ficheros. Consulte las opciones de newfs(8) y ejecute:

```
newfs /dev/ccd0c
```

18.4.1.1.4. Automatización

Seguramente querrá que ccd(4) esté dispuesto tras cada reinicio. Para ello, debe configurarlo. Guarde su configuración en `/etc/ccd.conf` mediante lo siguiente:

```
ccdconfig -g > /etc/ccd.conf
```

Durante el reinicio, el "script" `/etc/rc` ejecuta `ccdconfig -C` si encuentra el fichero `/etc/ccd.conf`. De este modo ccd(4) queda configurado automáticamente para que pueda montarse.

Nota

Si ha arrancando en modo mono usuario necesita ejecutar el siguiente comando antes de que pueda montar el ccd(4) para configurar el array:

```
ccdconfig -C
```

Para montar automaticamente el ccd(4) coloque una entrada para ccd(4) en `/etc/fstab` para que se monte durante el arranque:

```
/dev/ccd0c              /media      ufs     rw      2       2
```

18.4.1.2. El administrador de volúmenes Vinum

El administrador de volúmenes Vinum es un controlador de dispositivos de bloque que implementa unidades de disco virtuales. Aísla los discos hardware de la interfaz de dispositivos de bloque y mapea datos de modo que revierta en un incremento de flexibilidad, rendimiento y fiabilidad comparados con el sistema de slices de almacenamiento de disco tradicional. vinum(8) implementa los modelos RAID-0, RAID-1 y RAID-5, individualmente o combinados.

Consulte el Capítulo 20, *El Gestor de Volúmenes Vinum* para mayor información sobre vinum(8).

18.4.2. RAID por Hardware

FreeBSD admite una gran variedad de controladores RAID por hardware. Estos dispositivos controlan un subsistema RAID sin necesidad de software específico para FreeBSD que administre el array.

Puede controlar la mayoría de las operaciones de disco con una tarjeta que incorpore BIOS. El siguiente texto es una breve descripción de configuración utilizando una controladora Promise RAID IDE. Cuando se instala esta tarjeta e inicia el sistema despliega un "prompt" pidiendo información. Siga las instrucciones para entrar a la pantalla de configuración de la tarjeta. Ahí tendrá posibilidad de combinar todos los discos que haya conectado. Hecho esto el disco (o discos) aparecerán como una sola unidad en FreeBSD. Pueden configurarse otros niveles de RAID.

18.4.3. Reconstrucción de arrays ATA RAID1

FreeBSD le permite reemplazar en caliente un disco dañado. Esto requiere que lo intercepte antes de reiniciar.

Probablemente vea algo como lo siguiente en `/var/log/messages` o en la salida de dmesg(8):

```
ad6 on monster1 suffered a hard error.
ad6: READ command timeout tag=0 serv=0 - resetting
ad6: trying fallback to PIO mode
ata3: resetting devices .. done
ad6: hard error reading fsbn 1116119 of 0-7 (ad6 bn 1116119; cn ↺
1107 tn 4 sn 11)\\
status=59 error=40
ar0: WARNING - mirror lost
```

Consulte atacontrol(8) para más información:

```
# atacontrol list
ATA channel 0:
 Master:      no device present
 Slave:   acd0 <HL-DT-ST CD-ROM GCR-8520B/1.00> ATA/ATAPI rev 0

ATA channel 1:
 Master:      no device present
 Slave:       no device present

ATA channel 2:
 Master:  ad4 <MAXTOR 6L080J4/A93.0500> ATA/ATAPI rev 5
 Slave:       no device present

ATA channel 3:
 Master:  ad6 <MAXTOR 6L080J4/A93.0500> ATA/ATAPI rev 5
 Slave:       no device present

# atacontrol status ar0
ar0: ATA RAID1 subdisks: ad4 ad6 status: DEGRADED
```

1. Primero debe desconectar el disco del array para que pueda retirarlo con seguridad:

   ```
   # atacontrol detach 3
   ```

2. Reemplace el disco.

3. Conecte el disco de repuesto:

   ```
   # atacontrol attach 3
   Master:  ad6 <MAXTOR 6L080J4/A93.0500> ATA/ATAPI rev 5
   Slave:   no device present
   ```

4. Reconstruya el array:

```
# atacontrol rebuild ar0
```

5. El comando de reconstrucción no responderá hasta que termine la tarea. Puede abrir otra terminal (mediante Alt+F***n***) y revisar el progreso ejecutando lo siguiente:

```
# dmesg | tail -10
[texto eliminado]
ad6: removed from configuration
ad6: deleted from ar0 disk1
ad6: inserted into ar0 disk1 as spare

# atacontrol status ar0
ar0: ATA RAID1 subdisks: ad4 ad6 status: REBUILDING 0% completed
```

6. Espere hasta que termine la operación.

18.5. Dispositivos de almacenamiento USB

Texto de Marc Fonvieille.

Hoy día hay una enorme cantidad de soluciones de almacenamiento externoque usan el bus serie universal (USB): discos duros, "mecheros" (o "lápices") USB, grabadoras de CD-R, etc. FreeBSD puede usar estos dispositivos.

18.5.1. Configuración

El controlador de dispositivos de almacenamiento masivo USB, umass(4), ofrece soporte para dispositivos de almacenamiento USB. Si usa el kernel `GENERIC` no necesita cambiar nada en su configuración. Si utiliza un kernel personalizado asegúrese de que su fichero de configuración del kernel contiene las siguientes líneas:

```
device scbus
device da
device pass
device uhci
device ohci
device usb
device umass
```

El controlador umass(4) usa el subsistema SCSI para acceder a los dispositivos de almacenamiento USB y su dispositivo USB aparecerá en el sistema como dispositivo SCSI. Dependiendo del chipset USB de su placa base sólamente necesitará `device uhci` o `device ohci`; en cualquier caso tener los dos en el fichero de configuración del kernel no provocará ningún daño. No olvide compilar e instalar el nuevo kernel si hizo alguna modificación.

Nota

Si su dispositivo USB es una grabadora CD-R o DVD el controlador SCSI CD-ROM, cd(4), debe ser añadirse al kernel mediante la siguiente línea:

```
device cd
```

Dado que la grabadora aparece como una unidad SCSI no tiene que usar el controlador atapicam(4) en la configuración del kernel.

En FreeBSD 5.X y en la rama 4.X desde FreeBSD 4.10-RELEASE el soporte para controladores USB 2.0 se incorpora al sistema del siguiente modo:

```
device ehci
```

Tenga en cuenta que uhci(4) y ohci(4) siguen siendo necesarios si quiere disponer de soporte para USB 1.X.

Nota

En FreeBSD 4.X, El dæmon USB (usbd(8)) debe ejecutarse para poder ver ciertos tipos de dispositivo USB. Para habilitarlo, añada `usbd_enable="YES"` en `/etc/rc.conf` y reinicie la máquina.

18.5.2. Prueba de la configuración

La configuración está lista para probarse: conecte su dispositivo USB; en el búfer de mensajes del sistema (dmesg(8)), la unidad debe aparecer como algo similar a esto:

```
umass0: USB Solid state disk, rev 1.10/1.00, addr 2
GEOM: create disk da0 dp=0xc2d74850
da0 at umass-sim0 bus 0 target 0 lun 0
da0: <Generic Traveling Disk 1.11> Removable Direct Access SCSI-2 ↺
device
da0: 1.000MB/s transfers
da0: 126MB (258048 512 byte sectors: 64H 32S/T 126C)
```

Obviamente la marca, el nodo de dispositivo (`da0`) y otros detalles pueden diferir dependiendo de su hardware.

Ya que el dispositivo USB aparece como uno SCSI, puede usar `camcontrol` para ver una lista de dispositivos USB conectados al sistema:

```
# camcontrol devlist
<Generic Traveling Disk 1.11>      at scbus0 target 0 lun 0 ↺
(da0,pass0)
```

Si la unidad tiene un sistema de ficheros puede montarla. La Sección 18.3, “Añadir discos” contiene información que le resultará muy útil para formatear y crear particiones en el disco USB en caso de necesitarlo.

Si desconecta el dispositivo (el disco debe desmontarse previamente), debería ver en el búfer de mensajes del sistema algo parecido a esto:

```
umass0: at uhub0 port 1 (addr 2) disconnected
(da0:umass-sim0:0:0:0): lost device
(da0:umass-sim0:0:0:0): removing device entry
GEOM: destroy disk da0 dp=0xc2d74850
umass0: detached
```

18.5.3. Lecturas recomendadas

Ademas de las secciones Cómo añadir discos y Montado y desmontado de sistemas ficheros, consulte las siguientes páginas man: umass(4), camcontrol(8) y usbdevs(8).

18.6. Creación y uso de medios ópticos (CD)

Texto de Mike Meyer.

18.6.1. Introducción

Los CD tienen muchas opciones que los hacen distintos de los discos convencionales. Al principio los usuarios no podían escribirlos. Su diseño permite que leamos en ellos sin el retardo del movimiento de una cabeza lectora de una pista a otra. También son mucho más fáciles de transportar de un sistema a otro que muchos otros soportes de información.

Los CD tienen pistas, pero son una sección de los que permiten lectura contínua, no una propiedad física del disco. Para crear un CD en FreeBSD debe preparar los ficheros de datos que van a constituir las pistas del CD y luego escribir las pistas al CD.

El sistema de ficheros ISO 9660 se diseñó para gestionar estas diferencias. Por desgracia implementa límites de sistema de ficheros que eran comunes en la época en que se diseñó. Por suerte también proporciona un mecanismo de extensiones que permite que CD escritos excediendo dichos límites funcionen en sistemas que no soportan esas extensiones.

El port sysutils/cdrtools incluye mkisofs(8), un programa que le permitirá crear un fichero de datos que contenga un sistema de ficheros ISO 9660. Incorpora opciones que soportan varias extensiones. Se describe más adelante.

Qué herramienta usar para grabar el CD depende de si su grabadora es ATAPI o no. Las grabadoras de CD ATAPI usan el programa `burncd`, que forma parte del sistema base. Las grabadoras SCSI y USB usan `cdrecord`, del port sysutils/cdrtools.

`burncd` no soporta cualquier unidad de grabación. Para saber si una unidad está soportada consulte la siguiente lista de unidades CD-R/RW soportadas.

Nota

Si utiliza FreeBSD 5.X, FreeBSD 4.8-RELEASE o posteriores, puede utilizar `cdrecord` y otras herramientas para unidades SCSI en hardware ATAPI con el módulo ATAPI/CAM.

Si quiere usar un interfaz gráfico con su software de grabación de CD quizás le guste X-CD-Roast o K3b. Puede instalar estas herramientas como paquetes o desde los ports sysutils/xcdroast y sysutils/k3b, respectivamente. X-CD-Roast y K3b requieren el módulo ATAPI/CAM si usa hardware ATAPI.

18.6.2. mkisofs

El programa mkisofs(8) (que forma parte del port sysutils/cdrtools) genera un sistema de ficheros ISO 9660 que es una imagen de un árbol de directorios en el espacio de nombres del sistema de ficheros UNIX®. Esta es la forma más simple de usarlo:

```
# mkisofs -o ficherodeimagen.iso /ruta/del/árbol
```

Este comando creará un *`ficherodeimagen.iso`* que contenga un sistema de ficheros ISO 9660 que es una copia del árbol ubicado en *`/ruta/al/árbol`*. En el proceso, mapeará los nombres de fichero a nombres que se ajusten a las limitaciones del estándar del sistema de ficheros ISO 9660, y excluirá ficheros que posean nombres no característicos de sistemas de ficheros ISO.

Existe gran cantidad de opciones que permiten superar esas restricciones. En particular, `-R` habilita las extensiones Rock Ridge comunes para sistemas UNIX®, `-J` habilita las extensiones Joliet usadas por sistemas Microsoft y `-hfs` puede usarse para crear sistemas de ficheros utilizados por Mac OS®.

Puede utilizar `-U` para deshabilitar todas las restricciones de nombres de fichero si quiere crear un CD que se vaya a usar exclusivamente en sistemas FreeBSD. Cuando se usa con `-R` produce una imagen de sistema de ficheros que es idéntica al árbol FreeBSD origen, aunque puede violar el estándar ISO 9660 de múltiples formas.

La última opción de uso general es `-b`. Se usa para configurar la ubicación de la imagen de arranque que se usará al crear un CD arrancable “El Torito”. Esta opción usa como

argumento la ruta a la imagen de arranque desde la raíz del árbol de directorios que se va a escribir en el CD. Por defecto mkisofs(8) crea una imagen ISO en un modo llamado "de emulación de disquete (floppy)", y por lo tanto espera que la imagen de arranque sea exactamente de 1.200, 1.440 o 2880 KB de tamaño. Algunos cargadores de arranque, como el que se usa en los discos de la distribución FreeBSD, no utilizan modo de emulación: se usa la opción `-no-emul-boot` . Por tanto, si `/tmp/miarranque` tiene un sistema FreeBSD arrancable con la imagen de arranque en `/tmp/miarranque/boot/cdboot` podría crear la imagen en un sistema de ficheros ISO 9660 en `/tmp/arrancable.iso` de la siguiente manera:

```
# mkisofs -R -no-emul-boot -b boot/cdboot -o /tmp/arrancable.iso /tmp/
miarranque
```

Hecho esto, si tiene vn (FreeBSD 4.X), o `md` (FreeBSD 5.X) configurado en su kernel, puede montar el sistema de ficheros del siguiente modo:

```
# vnconfig -e vn0c /tmp/arrancable.iso
# mount -t cd9660 /dev/vn0c /mnt
```

En FreeBSD 4.X y FreeBSD 5.X proceda del siguiente modo:

```
# mdconfig -a -t vnode -f /tmp/arrancable.iso -u 0
# mount -t cd9660 /dev/md0 /mnt
```

Ahora puede verificar que `/mnt` y `/tmp/miarranque` sean idénticos.

Existen muchas otras opciones que puede usar para depurar el comportamiento de mkisofs(8), sobre todo en lo que se refiere al esquema ISO 9660 y la creación de discos Joliet y HFS. Consulte el manual de mkisofs(8).

18.6.3. burncd

Si tiene una grabadora ATAPI puede usar `burncd` para grabar una imagen ISO en un CD. `burncd` forma parte del sistema base, y está en `/usr/sbin/burncd` . Su uso es muy sencillo, ya que tiene pocas opciones:

```
# burncd -f unidaddecd data ficheroimagen.iso fixate
```

Esto grabará una copia de *`ficheroimagen.iso`* en *`unidadcd`*. El dispositivo por defecto es `/dev/acd0` (o `/dev/acd0c` en FreeBSD 4.X). Consulte burncd(8) para ver las opciones de configuración de velocidad de escritura, expulsión de CD una vez grabado, y escritura de datos de audio.

18.6.4. cdrecord

Si no dispone de una grabadora ATAPI de CD, tendrá que usar `cdrecord` para grabar sus CD. `cdrecord` no forma parte del sistema base; instálelo desde el port sysutils/cdrtools o

como paquete. Los cambios en el sistema base pueden hacer que las versiones binarias del programa fallen. Tendrá que actualizar el port cuando actualice su sistema o, si está siguiendo la rama -STABLE, actualizar el port cuando haya una nueva versión disponible.

Aunque `cdrecord` tiene muchas opciones, el uso básico es incluso más simple que el de `burncd`. Así se graba una imagen ISO 9660:

```
# cdrecord dev=dispositivo ficheroimagen.iso
```

La parte complicada de utilizar `cdrecord` es encontrar qué `dev` usar. Utilice la bandera `-scanbus` para dar con la configuración apropiada. La salida será parecida a la siguiente:

```
# cdrecord -scanbus
Cdrecord 1.9 (i386-unknown-freebsd4.2) Copyright (C) 1995-2000 Jörg ↺
Schilling
Using libscg version 'schily-0.1'
scsibus0:
        0,0,0     0) 'SEAGATE ' 'ST39236LW       ' '0004' Disk
        0,1,0     1) 'SEAGATE ' 'ST39173W        ' '5958' Disk
        0,2,0     2) *
        0,3,0     3) 'iomega  ' 'jaz 1GB         ' 'J.86' Removable ↺
Disk
        0,4,0     4) 'NEC     ' 'CD-ROM DRIVE:466' '1.26' Removable ↺
CD-ROM
        0,5,0     5) *
        0,6,0     6) *
        0,7,0     7) *
scsibus1:
        1,0,0   100) *
        1,1,0   101) *
        1,2,0   102) *
        1,3,0   103) *
        1,4,0   104) *
        1,5,0   105) 'YAMAHA  ' 'CRW4260         ' '1.0q' Removable ↺
CD-ROM
        1,6,0   106) 'ARTEC   ' 'AM12S           ' '1.06' Scanner
        1,7,0   107) *
```

Esta lista muestra los valores `dev` apropiados para los dispositivos de la lista. Localice su grabadora de CD y utilice los tres números separados por comas como valor para `dev`. En este caso, el dispositivo CDW es 1,5,0 y por tanto la entrada apropiada sería `dev=1,5,0`. Hay modos más fáciles de especificar este valor; consulte cdrecord(1) para más detalles. También es el lugar donde buscar información sobre la escritura de pistas de audio, controlar la velocidad de escritura y muchas más cosas.

18.6.5. Copiar CD de audio

Puede duplicar un CD de audio extrayendo los datos de audio del CD a ficheros y escribir estos ficheros en un CD virgen. El proceso es ligeramente diferente en unidades ATAPI y SCSI.

Procedimiento 18.1. Unidades SCSI

1. Use `cdda2wav` para extraer el audio.

```
% cdda2wav -v255 -D2,0 -B -Owav
```

2. Use `cdrecord` para escribir los ficheros `.wav`.

```
% cdrecord -v dev=2,0 -dao -useinfo  *.wav
```

Asegúrese de que *`2,0`* este configurado apropiadamente, como se describe en la Sección 18.6.4, “cdrecord”.

Procedimiento 18.2. Unidades ATAPI

1. El controlador de CD ATAPI hace que cada pista sea accesible como `/dev/acddtnn` , donde *d* es el número de unidad y *nn* es el número de pista expresado con dos dígitos decimales, precedido por un cero si es necesario. La primera pista del primer disco es `/dev/acd0t01` , la segunda es `/dev/acd0t02` , la tercera es `/dev/acd0t03` y así sucesivamente.

   Asegúrese de que existen los ficheros apropiados en `/dev`.

```
# cd /dev
# sh MAKEDEV acd0t99
```

> **Nota**
>
> En FreeBSD 5.0 devfs(5) creará y gestionará automáticamente las entradas necesarias en `/dev`, así que no será necesario usar `MAKEDEV`.

2. Extraer cada pista con dd(1). También deberá declarar un tamaño específico de bloque al extraer los ficheros.

```
# dd if=/dev/acd0t01 of=pista1.cdr bs=2352
# dd if=/dev/acd0t02 of=pista2.cdr bs=2352
...
```

3. Grabar los ficheros extraídos a disco con `burncd`. Debe declarar que son ficheros de audio y que `burncd` debe cerrar (“fixate”) el disco al terminar la grabación.

```
# burncd -f /dev/acd0 audio pista1.cdr pista2.cdr ... fixate
```

18.6.6. Duplicar CDs de datos

Puede copiar un CD de datos a un fichero de imagen que será funcionalmente equivalente al fichero de imagen creado con mkisofs(8), y puede usarlo para duplicar cualquier CD de datos. El ejemplo dado aquí asume que su dispositivo CDROM es `acd0`. Sustitúyalo por el dispositivo CDROM correcto para su configuración. Bajo FreeBSD 4.X, se debe añadir una `c` al final del nombre del dispositivo para indicar la partición entera o, en el caso de los CDROM, el disco entero.

```
# dd if=/dev/acd0 of=fichero.iso bs=2048
```

Hecha la imagen puede garbarla en un CD como se describió anteriormente.

18.6.7. Uso de CD de datos

Ahora que ha creado un CDROM de datos estándar tal vez quiera montarlo y leer los datos que contiene. Por defecto mount(8) asume que los sistemas de ficheros son de tipo `ufs`. Si trata de hacer algo como

```
# mount /dev/cd0 /mnt
```

recibirá un error como este: Incorrect super block y no se montará. Un CDROM no es un sistema de ficheros `UFS` así que los intentos de montarlo como tal fallarán. Tendrá que decirle a mount(8) que el sistema de ficheros es de tipo `ISO9660` y funcionará. Puede hacerlo mediante la opción `-t cd9660`. Por ejemplo, si quiere montar el dispositivo CDROM `/dev/cd0` en `/mnt` ejecute:

```
# mount -t cd9660 /dev/cd0 /mnt
```

Tenga en cuenta que el nombre de su dispositivo (`/dev/cd0` en este ejemplo) puede ser diferente, dependiendo de la interfaz que su CDROM utilice. Además la opción `-t cd9660` sólo ejecuta mount_cd9660(8). El ejemplo de arriba puede resumirse del siguiente modo:

```
# mount_cd9660 /dev/cd0 /mnt
```

En general puede usar CDROM de datos de cualquier fabricante, aunque los discos con ciertas extensiones ISO 9660 pueden mostrar un comportamiento extraño. Por ejemplo, los discos Joliet almacenan todos los nombres de fichero en caracteres unicode de dos-bytes. El kernel de FreeBSD no comprende unicode (*todavía*) así que los caracteres que no están en inglés aparecen como signos de interrogación. (Si utiliza FreeBSD 4.3 o alguna versión posterior, el controlador CD9660 incluye unas estructuras llamadas “ganchos”, que le permitirán cargar una tabla de conversión unicode apropiada cuando haga falta. Hay módulos para algunas de las codificaciones más comunes en el port sysutils/cd9660_unicode.)

Es posible que reciba un error Device not configured al tratar de montar un CDROM. Generalmente esto significa que la unidad de CDROM piensa que no hay disco en la bandeja,

o que la unidad no es visible en el bus. Puede llevar un par de segundos el que una unidad de CDROM se dé cuenta de que ha sido alimentada, por lo tanto sea paciente.

Algunas veces un CDROM SCSI puede "perdido" debido a que no tuvo tiempo suficiente para responder al reset del bus. Si tiene un CDROM SCSI añada la siguiente opción a su fichero de configuración del kernel y recompile su kernel.

```
options SCSI_DELAY=15000
```

Esto le indica a su bus SCSI que haga una pausa de 15 segundos durante el arranque para darle ocasión a su unidad de CDROM de responder al reset del bus.

18.6.8. Grabar CD de datos "crudos" (Raw)

Puede guardar un fichero directamente a CD sin crear un sistema de ficheros ISO 9660. Algunas personas hacen esto al crear respaldos. Es un proceso más rápido que grabar un CD estándar:

```
# burncd -f /dev/acd1 -s 12 data fichero.tar.gz fixate
```

Para recuperar los datos guardardados de este modo en un CD, debe leer los datos desde el nodo de dispositivo "crudo":

```
# tar xzvf /dev/acd1
```

No puede montar este disco como lo haría con un CDROM normal. Estos CDROM no pueden leerse en ningún sistema operativo que no sea FreeBSD. Si quiere montar el CD o compartir los datos con otro sistema operativo debe utilizar mkisofs(8) como se describió previamente.

18.6.9. Uso del controlador ATAPI/CAM

Original de Marc Fonvieille.

Este controlador permite que dispositivos ATAPI (CD-ROM, CD-RW, unidades DVD, etc) sean accesibles a través del subsistema SCSI y por lo tanto permite el uso de aplicaciones como sysutils/cdrdao o cdrecord(1).

Para usar este controlador necesitará añadir la siguiente línea al fichero de configuración de su kernel:

```
device atapicam
```

Es posible que necesite también las siguientes líneas en el fichero de configuración de su kernel:

```
device ata
device scbus
```

```
device cd
device pass
```

(que, por otra parte, ya deberín estar presentes).

Recompile, instale su nuevo kernel y reinicie su máquina. Durante el proceso de arranque su grabadora debe ser detectada; veamos un ejemplo:

```
acd0: CD-RW <MATSHITA CD-RW/DVD-ROM UJDA740> at ata1-master PIO4
cd0 at ata1 bus 0 target 0 lun 0
cd0: <MATSHITA CDRW/DVD UJDA740 1.00> Removable CD-ROM SCSI-0 device
cd0: 16.000MB/s transfers
cd0: Attempt to query device size failed: NOT READY, Medium not ↺
present - tray closed
```

Puede acceder a la unidad a través del del nombre de dispositivo `/dev/cd0` ; por ejemplo, para montar un CDROM en `/mnt`, teclée lo siguiente:

```
# mount -t cd9660 /dev/cd0 /mnt
```

Como `root`, puede ejecutar el siguiente comando para obtener las direcciones SCSI del dispositivo:

```
# camcontrol devlist
<MATSHITA CDRW/DVD UJDA740 1.00>   at scbus1 target 0 lun 0 ↺
(pass0,cd0)
```

Según esto, `1,0,0` será la dirección SCSI a utilizar con cdrecord(1) y otras aplicaciones SCSI.

Para mayor información sobre sistemas ATAPI/CAM y SCSI, diríjase a las páginas de manual atapicam(4) y cam(4).

18.7. Crear y utilizar medios ópticos (DVDs)

Texto de Marc Fonvieille.
Con colaboraciones de Andy Polyakov.

18.7.1. Introducción

Comparado con el CD, el DVD es la nueva generación de tecnología de almacenamiento en medios ópticos. El DVD puede almacenar más datos que cualquier CD y hoy día es el estándar para publicación de vídeo.

Se pueden definir cinco formatos de grabación para lo que llamamos un DVD grabable:

- DVD-R: Este fué el primer formato de grabación de DVD. El DVD-R estándar fué definido por el DVD Forum. Este formato es de una sola escritura.

- DVD-RW: Esta es la versión reescribible del DVD-R estándar. Un DVD-RW puede reescribirse unas 1.000 veces.
- DVD-RAM: Este es también un formato reescribible soportado por el DVD Forum. Un DVD-RAM puede verse como un disco duro extraíble. Este medio no es compatible con la mayoría de las unidades DVD-ROM y reproductores de video DVD; hay muy pocas grabadoras de DVD que soporten el formato DVD-RAM.
- DVD+RW: Este es un formato reescribible definido por la DVD+RW Alliance. Un DVD +RW puede reescribirse unas 1000 veces.
- DVD+R: Este un formato es la versión de una sola escritura del formato DVD+RW.

Un DVD grabable de una capa puede almacenar hasta 4.700.000.000 bytes, es decir, 4'38 GB o 4485 MB (1 kilobyte son 1.024 bytes).

Nota

Debemos hacer una distinción entre medio físico y aplicación. Un DVD de vídeo es una estructura de fichero específica que puede escribirse en cualquier medio físico consistente en un DVD grabable: DVD-R, DVD+R, DVD-RW, etc. Antes de elegir el tipo de medio, debe asegurarse que la grabadora y el reproductor de DVD de vídeo (un reproductor independiente o una unidad DVD-ROM en una computadora) son compatibles con el medio que pretende utilizar.

18.7.2. Configuración

Utilice growisofs(1) para grabar el DVD. Forma parte de las herramientas dvd+rw-tools (sysutils/dvd+rw-tools). Las dvd+rw-tools permiten usar todos los tipos de DVD.

Estas herramientas utilizan el subsistema SCSI para acceder a los dispositivos, por lo tanto el soporte ATAPI/CAM debe estar presente en su kernel. Si su grabadora usa el interfaz USB no tendrá que hacerlo, pero tendrá que leer la Sección 18.5, “Dispositivos de almacenamiento USB” para más información sobre la configuración de dispositivos USB.

También debe que habilitar el acceso DMA para dispositivos ATAPI. Para ello añada la siguiente línea a `/boot/loader.conf` :

```
hw.ata.atapi_dma="1"
```

Antes de intentar utilizar dvd+rw-tools debe consultar las notas de compatibilidad de hardware de dvd+rw-tools por si apareciera cualquier información relacionada con su grabadora de DVD.

Nota

Si desea un interfaz gráfico debería echar un vistazo a K3b (sysutils/k3b), que ofrece un interfaz de usuario amigable para growisofs(1) y muchas otras herramientas de grabación.

18.7.3. Quemado de DVD de datos

growisofs(1) es un "frontend" de mkisofs, invocará a mkisofs(8) para crear una estructura de sistema de ficheros y realizará la escritura del DVD. Esto significa que no necesita crear una imagen de los datos antes del proceso de escritura.

La grabación en DVD+R o DVD-R de los datos del directorio `/ruta/a/los/datos` , se hace del siguiente modo:

```
# growisofs -dvd-compat -Z /dev/cd0 -J -R /ruta/a/los/datos
```

Las opciones `-J -R` se suministran a mkisofs(8) para la creación del sistema de ficheros (en este caso: un sistema de ficheros ISO 9660 con extensiones Joliet y Rock Ridge). Consulte la página de manual mkisofs(8) para más detalles.

La opción `-Z` se usa la sesión inicial de grabación en todos los casos, sesiones múltiples o no. El dispositivo DVD del ejemplo, */dev/cd0* , debe ajustarse de acuerdo a la configuración de su sistema. El parámero `-dvd-compat` cerrar´ el disco (no se podrá añadir nada a la grabación). Por contra, esto le brindará una mejor compatibilidad del medio con unidades DVD-ROM.

También es posible grabar una imagen pre-masterizada, por ejemplo para guardar la imagen *ficheroimagen.iso* :

```
# growisofs -dvd-compat -Z /dev/cd0=ficheroimagen.iso
```

La velocidad de escritura se detecta y configura automáticamente según el medio y la unidad que se esté utilizando. Si quiere forzar la velocidad de escritura utilice el parámetro `-speed=` . Para más información consulte la página de manual growisofs(1).

18.7.4. Grabación de un DVD de vídeo

Un DVD de vídeo es una estructura de ficheros específica basada en las especificiones ISO 9660 y micro-UDF (M-UDF). El DVD de vídeo también dispone de una jerarquía de estructura de datos específica; por esta razón es necesario un programa especializado para crear tal DVD: multimedia/dvdauthor.

Si ya tiene una imagen de un sistema de ficheros de DVD de vídeo grábelo de la misma manera que cualquier otra imagen; consulte la sección previa para ver un ejemplo. Si ha creado el DVD y el resultado está en, por ejemplo, el directorio `/ruta/al/vídeo` , use el siguiente comando para grabar el DVD de vídeo:

```
# growisofs -Z /dev/cd0 -dvd-video /ruta/al/vídeo
```

La opción `-dvd-video` de mkisofs(8) hará posible la creación de una estructura de sistema de ficheros de DVD de vídeo. Además, la opción `-dvd-video` implica la opción `-dvd-compat` de growisofs(1).

18.7.5. Uso de un DVD+RW

A diferencia de un CD-RW, un DVD+RW virgen necesita ser formateado antes de usarse por primera vez. El programa growisofs(1) se encargará de ello automáticamente cuando sea necesario, lo cual es el método *recomendado*. De todas formas puede usted usar el comando `dvd+rw-format` para formatear el DVD+RW:

```
# dvd+rw-format /dev/cd0
```

Necesita ejecutar esta operación solamente una vez, recuerde que sólo los DVD+RW vírgenes necesitan ser formateados. Hecho eso ya puede usar el DVD+RW de la forma expuesta en las secciones previas.

Si desea guardar nuevos datos (grabar un sistema de ficheros totalmente nuevo, no añadir más datos) en un DVD+RW no necesita borrarlo, sólo tiene que escribir sobre la grabación anterior (realizando una nueva sesión inicial):

```
# growisofs -Z /dev/cd0 -J -R /ruta/alos/datosnuevos
```

El formato DVD+RW ofrece la posibilidad de añadir datos fácilmente a una grabación previa. La operación consiste en fusionar una nueva sesión a la existente, no es escritura multisesión; growisofs(1) *hará crecer* el sistema de ficheros ISO 9660 presente en el medio.

Si, por ejemplo, añadir datos al DVD+RW del ejemplo anterior tenemos que usar lo siguiente:

```
# growisofs -M /dev/cd0 -J -R /ruta/alos/datosnuevos
```

Las mismas opciones de mkisofs(8) que utilizamos para quemar la sesión inicial pueden usarse en ulteriores escritura.

Nota

Puede usar la opción `-dvd-compat` si desea mejor la compatibilidad de medios con unidades DVD-ROM. Si la usa en un DVD+RW no evitará que pueda añadir más datos.

Si por alguna razón desea borrar el contenido del medio, haga lo siguiente:

```
# growisofs -Z /dev/cd0=/dev/zero
```

18.7.6. Uso de un DVD-RW

Un DVD-RW acepta dos formatos de disco: el incremental secuencial y el de sobreescritura restringida. Por defecto los discos DVD-RW están en formato secuencial.

Un DVD-RW virgen puede utilizarse directamente sin necesidad de formateo, sin embargo un DVD-RW no virgen en formato secuencial necesita ser borrado antes de poder guardar una nueva sesión inicial.

Para borrar un DVD-RW en modo secuencial, ejecute:

```
# dvd+rw-format -blank=full /dev/cd0
```

Nota

Un borrado total (`-blank=full`) tardará aproximadamente una hora en un medio 1x. Un borrado rápido puede realizarse con la opción `-blank` si el DVD-RW fué grabado en modo Disk-At-Once (DAO). Para grabar el DVD-RW en modo DAO use el comando:

```
# growisofs -use-the-force-luke=dao -Z /dev/
cd0=ficheroimagen.iso
```

La opción `-use-the-force-luke=dao` no es imprescindible, ya que growisofs(1) trata de detectar el medio (borrado rápido) y entrar en escritura DAO.

Debería usarse el modo de reescritura restringida en los DVD-RW, pues este formato es más flexible que el formato de incremento secuencial, el formato por defecto.

Para escribir datos en un DVD-RW secuencial proceda del mismo modo que con los demás formatos de DVD:

```
# growisofs -Z /dev/cd0 -J -R /ruta/alos/datos
```

Si desea añadir datos a una grabación previa tendrá que usar la opción `-M` de growisofs(1). si añade datos a un DVD-RW en modo incremental secuencial se creará en el disco una nueva sesión y el resultado será un disco multisesión.

Un DVD-RW en formato de sobreescritura restringido no necesita ser borrado antes de una nueva sesión inicial, sólo tiene que sobreescribir el disco con la opción `-Z`. esto es similar al caso DVD+RW. También es posible ampliar un sistema de ficheros ISO 9660 ya existente y escrito en el disco del mismo modo que para un DVD+RW con la opción `-M`. El resultado será un DVD de una sesión.

Para poner un DVD-RW en el formato de sobreescritura restringido haga lo siguiente:

```
# dvd+rw-format /dev/cd0
```

Para devolverlo al formato secuencial use:

```
# dvd+rw-format -blank=full /dev/cd0
```

18.7.7. Multisesión

Muy pocas unidades DVD-ROM soportan DVDs multisesión. La mayoría de las veces (y si tiene suerte) solamente leerán la primera sesión. Los DVD+R, DVD-R y DVD-RW en formato secuencial pueden aceptar multisesiones. El concepto de multisesión no existe en los formatos de sobreescritura restringida de DVD+RW y DVD-RW.

Usando el siguiente comando despues de una sesión inicial (no-cerrada) en un DVD+R, DVD-R o DVD-RW en formato secuencial añadirá una nueva sesión al disco:

```
# growisofs -M /dev/cd0 -J -R /ruta/alos/nuevosdatos
```

Usando este comando con un DVD+RW o un DVD-RW en modo de sobreescritura restringida, agregará datos fusionando la nueva sesión a la ya existente. El resultado será un disco de una sola sesión. Este es el procedimiento habitual para añadir datos tras la escritura inicial.

Nota

Una cierta cantidad de espacio en el medio se usa en cada sesión al finalizar e iniciar sesiones; por tanto, se deben añadir sesiones con grandes cantidades de datos para optimizar el espacio del DVD. El

número de sesiones está limitado a 154 para un DVD+R, aproximadamente 2.000 para un DVD-R y 127 para un DVD+R de doble capa.

18.7.8. Para mayor información

Para más información sobre un DVD,puede ejecutar el comando `dvd+rw-mediainfo /dev/cd0` con el disco en la unidad.

Tiene más información sobre dvd+rw-tools en la manual growisofs(1), en el sitio web de dvd+rw-tools y en los archivos de la lista de correos de cdwrite.

Nota

Si va a enviar un informe de problemas es imperativo que adjunte la salida que `dvd+rw-mediainfo` produjo al grabar (o no grabar) el medio. Sin esta salida será prácticamente imposible ayudarle.

18.8. Creación y uso de disquetes (floppies)

Texto original de Julio Merino.
Reescrito por Martin Karlsson.

Poder almacenar datos en discos flexibles es útil algunas veces, por ejemplo cuando no se tiene cualquier otro medio de almacenamiento extraible o cuando se necesita transferir una cantidad pequeña de datos a otro sistema.

Esta sección explicará cómo usar disquetes en FreeBSD. Cubrirá principalmente el formateo y utilización de disquetes DOS de 3.5 pulgadas, pero los conceptos son similares en otros formatos de disquete.

18.8.1. Formateo de disquetes

18.8.1.1. El dispositivo

El acceso a los disquetes se efectúa a través de entradas en `/dev`, igual que en otros dispositivos. Para acceder al disquete "crudo" (raw) en versiones 4.X y anteriores, se usa `/dev/fdN`, donde *N* representa el número de unidad, generalmente 0, o `/dev/fdNX`, donde *X* representa una letra.

En versiones 5.0 o posteriores, simplemente use `/dev/fdN`.

18.8.1.1.1. El tamaño de disco en versiones 4.X y anteriores

También existen dispositivos `/dev/fdN.tamaño` , donde *`tamaño`* es el tamaño del disquete en kilobytes. Estas entradas se usan durante el formateo a bajo nivel para determinar el tamaño del disco. En los siguientes ejemplos se usará el tamaño de 1440kB.

Algunas veces las entradas bajo `/dev` tendrán que ser (re)creadas. Para ello, ejecute:

```
# cd /dev && ./MAKEDEV "fd*"
```

18.8.1.1.2. El tamaño de disco en versiones 5.0 y posteriores

En 5.0, devfs(5) administrará automáticamente los nodos de dispositivo en `/dev`, así que el uso de `MAKEDEV` no es necesario.

El tamaño de disco deseado se pasa a fdformat(1) mediante la bandera `-f`. Los tamaños soportados aparecen en fdcontrol(8), pero tenga muy en cuenta que 1440kB es el que funciona mejor.

18.8.1.2. Formatear

Un disquete necesita ser formateado a bajo nivel antes de poder usarse. Esto suele hacerlo el fabricante, pero el formateo es una buena manera de revisar la integridad del medio. Aunque es posible forzar tamaños de disco más grandes (o pequeños), 1440kB es para lo que la mayoría de los disquetes están diseñados.

Para formatear un disquete a bajo nivel debe usar fdformat(1). Esta utilidad espera el nombre del dispositivo como argumento.

Tome nota de cualquier mensaje de error, ya que éstos pueden ayudarle a determinar si el disco está bien o mal.

18.8.1.2.1. Formateo en versiones 4.X y anteriores

Use el dispositivo `/dev/fdN.tamaño` para formatear el disquete. Inserte un disco de 3'5 pulgadas en su unidad y ejecute:

```
# /usr/sbin/fdformat /dev/fd0.1440
```

18.8.1.2.2. Formateo en versiones 5.0 y posteriores

Use el dispositivo `/dev/fdN` para formatear el disquete. Inserte un disco de 3'5 pulgadas en su unidad y ejecute:

```
# /usr/sbin/fdformat -f 1440 /dev/fd0
```

18.8.2. La etiqueta de disco

Tras un formato del disco a bajo nivel necesitará colocar una etiqueta de disco en él. Esta etiqueta de disco será destruida más tarde, pero es necesaria para que el sistema determine el tamaño del disco y su geometría.

La nueva etiqueta de disco ocupará todo el disco, y contendrá toda la información apropiada sobre la geometría del disquete. Los valores de geometría para la etiqueta de disco están en `/etc/disktab`.

Ejecute disklabel(8) así:

```
# /sbin/disklabel -B -r -w /dev/fd0 fd1440
```

Nota

Desde FreeBSD 5.1-RELEASE bsdlabel(8) reemplazó al viejo programa disklabel(8). En bsdlabel(8) se eliminaron muchas opciones y parámetros obsoletos; en el ejemplo de arriba la opción `-r` no debe usarse. Para mayor información consulte la página de manual de bsdlabel(8).

18.8.3. El sistema de ficheros

Ahora el disquete está listo para ser formateado a alto nivel. Esto colocará un sistema de ficheros nuevo en el disco y permitirá a FreeBSD leer y escribir en el disco. Después de crear el sistema de ficheros se destruye la etiqueta de disco, así que si desea reformatearlo, tendrá que recrear la etiqueta de disco.

El sistema de ficheros del disquete puede ser UFS o o FAT. FAT suele ser una mejor opción para disquetes.

Para formatear un disquete ejecute:

```
# /sbin/newfs_msdos /dev/fd0
```

El disco está para su uso.

18.8.4. Uso de un disquete

Para usar el disquete móntelo con mount_msdos(8) (en versiones 4.X y anteriores) o con mount_msdosfs(8) (en versiones 5.X o posteriores). También se puede usar emulators/mtools.

18.9. Creación y uso de cintas de datos

Los principales medios de cinta son 4mm, 8mm, QIC, mini-cartridge y DLT.

18.9.1. 4mm (DDS: Digital Data Storage)

Las cintas de 4mm están reemplazando a las QIC como los medios de respaldo más frecuentes en estaciones de trabajo. Esta tendencia se aceleró en gran medida cuando Conner adquirió Archive, un fabricante líder de unidades QIC, y abandonó la producción de unidades QIC. Las unidades de 4mm son pequeñas y silenciosas pero no tienen la reputación de fiabilidad de la que disfrutan las unidades de 8mm. Los cartuchos son más baratos y más pequeños (3 x 2 x 0.5 pulgadas, 76 x 51 x 12 mm) que los cartuchos de 8mm. En el caso de las cintas de 4mm, igual que las de 8mm, tienen un cabezal con una vida comparativamente más corta. Ambos utilizan el escaneado en espiral.

El ancho de datos de estas unidades comienza por aprox. 150 kB/s, con un pico de aprox. ~500 kB/s. La capacidad de datos va de los 1'3 GB a 2'0 GB. La compresión por hardware, disponible con la mayoría de estas unidades, dobla aproximadamente la capacidad. Existen unidades de biblioteca de cinta multi-unidad con 6 unidades en un solo armario y cambio de cinta automático. Las capacidades de estas bibliotecas alcanzan los 240 GB.

El estándar DDS-3 soporta capacidades de cinta de hasta 12 GB (o 24 GB con compresión).

Las unidades de 4mm, igual que las unidades de 8mm, utilizan escaneo en espiral. Tanto unas como otras tienen las mismas ventajas y desventajas.

Las cintas deben renovarse por otras después de 2,000 pasadas ó 100 respaldos completos.

18.9.2. 8mm (Exabyte)

Las cintas de 8mm son las unidades de cinta SCSI más comunes; son la mejor opción de cintas reemplazables y eso hace que las unidades de cinta Exabyte 8mm de 2 GB sean casi omnipresentes. Las unidades de 8mm son fiables, prácticas y silenciosas. Los cartuchos son baratos y bastante pequeños (4.8 x 3.3 x 0.6 pulgadas; 122 x 84 x 15 mm). Una desventaja de las cintas de 8mm es la vida relativamente corta del cabezal y de la propia cinta debido a la alta tasa de movimiento relativo de la cinta por los cabezales.

El ancho de datos varía de aprox. 250 kB/s hasta los 500 kB/s. La capacidad va desde los 300 MB hasta los 7 GB. La compresión por hardware, disponible con la mayoría de estas unidades, dobla aproximadamente la capacidad. Estas unidades están disponibles como unidades solas o como unidades de biblioteca de cinta multi-unidad con 6 unidades y 120 cintas en un solo armario. Las cintas las cambia automáticamente la unidad. La capacidad de dichas bibliotecas alcanza los 840+ GB.

El modelo Exabyte “Mammoth” soporta 12 GB en una cinta (24 GB con compresión) y cuesta aproximadamente el doble que las unidades de cinta convencionales.

Los datos se graban en cinta utilizando escaneo en espiral. Las cabezas se posicionan en ángulo en relación al medio (6 grados aproximadamente). La cinta se envuelve cerca de 270 grados en el cilindro que soporta las cabezas. El cilindro gira mientras la cinta se des-

liza sobre el cilindro. El resultado es una alta densidad de datos y pistas almacenadas muy pegadas, dispuestas en ángulo a través de la cinta de un extremo al otro.

18.9.3. QIC

Las cintas y unidades QIC-150 son, quizás, las unidades y medios de cinta más comunes. Las unidades de cinta QIC son las unidades de respaldo "serias" menos caras. La desventaja es el coste del medio. Las cintas QIC son caras comparadas con las cintas de 8mm o de 4mm, hasta 5 veces el precio de almacenamiento de datos por GB. No obstante, si sus necesidades pueden satisfacerse con media docena de cintas, QIC tal vez sea la decisión correcta. QIC es la unidad de cinta *más* común. Casi en todas partes tienen una unidad QIC de una u otra densidad. Y ese es el problema, QIC ofrece un enorme número de densidades en cintas físicamente similares (algunas veces idénticas). Las unidades QIC son cualquier cosa menos silenciosas. Hacen bastante ruido antes de iniciar la grabación de datos y son claramente audibles siempre que leen, escriben o hacen una búsqueda. Las cintas QIC miden 6 x 4 x 0.7 pulgadas; 152 x 102 x 17 mm.

El ancho de datos varía de aprox. 150 kB/s a aprox. 500 kB/s. La capacidad de datos varía de 40 MB a 15 GB. La compresión por hardware existe en muchas de las nuevas unidades QIC. Las unidades QIC se ven con menos frecuencia y además están siendo suplantadas por unidades DAT.

Los datos se graban en la cinta en pistas. Las pistas discurren a lo largo del extenso eje de la cinta de un extremo al otro. El número de pistas, y por lo tanto el ancho de una pista varía según la capacidad de la cinta. La mayoría, si no todas las unidades nuevas, ofrecen compatibilidad con modelos anteriores al menos para lectura (y también en muchos casos de escritura). QIC tiene buena reputación en cuanto a seguridad de los datos (las piezas mecánicas son más simples y más robustas que en las unidades de búsqueda en espiral).

Las cintas deben ser sustituirse por otras después de 5,000 respaldos.

18.9.4. DLT

DLT tiene la tasa de transferencia de datos más rápida de todos los tipos de unidades mostradas aquí. La cinta de 1/2" (12'5mm) está alojada en un cartucho de un solo cilindro (4 x 4 x 1 pulgadas; 100 x 100 x 25 mm). El cartucho tiene una puerta giratoria a lo largo de todo un lado del cartucho. El mecanismo de la unidad abre esta puerta para extraer el "líder". El "lider" de la cinta tiene un agujero oval que la unidad utiliza para "enganchar" la cinta. El cilindro de levantamiento está dentro de la unidad de cinta. Los demás cartuchos descritos en este texto (los cartuchos de 9 pistas son la única excepción) tienen el cilindro proveedor alojados dentro del propio cartucho de cinta.

El ancho de datos es aproximadamente de 1'5 MB/s, tres veces el ancho de unidades de cinta de 4mm, de 8mm o QIC. Las capacidades de datos varían entre 10 GB y 20 GB en una sola unidad. Hay unidades multicinta y con cargadores multi-cinta, y bibliotecas multi-

unidad que pueden albergar de 5 a 900 cintas con una a 20 unidades, con lo que pueden alcanzar desde 50 GB hasta 9 TB de almacenamiento.

Con compresión, el formato DLT Type IV soporta hasta 70 GB de capacidad.

Los datos se almacenan en cinta en pistas paralelas a la dirección del movimiento de la cinta (como en las cintas QIC). Se escriben dos pistas al mismo tiempo. El tiempo de vida de lectura/escritura es relativamente largo. Una vez que la cinta no hay movimiento relativo entre las cabezas y la cinta.

18.9.5. AIT

AIT es un nuevo formato de Sony, y puede almacenar hasta 50 GB (con compresión) por cinta. Las cintas contienen chips de memoria que retienen un índice de los contenidos de la cinta. Este índice puede ser leido rápidamente para determinar la posición de los ficheros en la cinta, en lugar de los varios minutos que requeriría el proceso con otras cintas. SAMS:Alexandria puede gestionar más de 40 bibliotecas de cinta AIT, comunicándose directamente con el chip de memoria de la cinta para desplegar el contenido en pantalla, determinar qué ficheros fueron respaldados a qué cinta, ubicar la cinta correcta, cargarla y restaurar los datos desde la cinta.

Las bibliotecas como ésta cuestan alrededor de 20.000 dólares, lo que las aleja bastante del alcance de los aficionados.

18.9.6. Estreno de una cinta

La primera vez que trate de leer o escribir una cinta nueva, completamente en blanco, la operación fallará. El mensaje de la consola se parecerá al siguiente:

```
sa0(ncr1:4:0): NOT READY asc:4,1
sa0(ncr1:4:0):  Logical unit is in process of becoming ready
```

La cinta no contiene un bloque identificador (bloque número 0). Todas las unidades de cinta QIC desde la adopción del estándar QIC-525 escriben un bloque identificador en la cinta. Existen dos soluciones:

- `mt fsf 1` hace que la unidad de cinta escriba un bloque identificador a la cinta.
- Use el botón del panel frontal para expulsar la cinta.

  Inserte nuevamente la cinta y haga un `dump` de datos a la cinta.

  `dump` devolverá DUMP: End of tape detected y la consola mostrará HARDWARE FAILURE info:280 asc:80,96.

  Rebobine la cinta usando: `mt rewind`.

A partir de ese momento podrá utilizar la cinta.

18.10. Respaldos en disquetes

18.10.1. ?Puedo utilizar disquetes para respaldar mis datos?

Los disquetes no son realmente el medio ideal para hacer respaldos debido a que:

- El medio no es fiable, especialmente después de largos periodos de tiempo.
- El respaldo y la restauración es muy lento.
- Tienen una capacidad muy limitada (los dís de respaldar un disco duro entero en una docena de disquetes pasaron hace mucho).

De todas maneras, si no tiene otro método para respaldar sus datos los disquetes son una mejor solución que no tener ningún respaldo.

Si tiene que utilizar disquetes asegúrese de usar discos de buena calidad. Los disquetes que han estado almacenados en la oficina durante un par de años son una mala elección. Lo mejor sería que utilizara discos nuevos de un fabricante respetado.

18.10.2. ?Cómo respaldo mis datos a discos flexibles?

La mejor manera de respaldar a un disquete es usar tar(1) con la opción `-M` (multi volumen), que permite que el respaldo se guarde en varios disquetes.

Para respaldar todos los ficheros en el directorio actual y sus subdirectorios use esto (como `root`):

```
# tar Mcvf /dev/fd0 *
```

Cuando el primer disquete esté lleno tar(1) le solicitará que inserte el siguiente volumen (debido a que tar(1) es independiente del medio se refiere a volúmenes; en éste contexto se refiere a disquetes).

```
Prepare volume #2 for /dev/fd0 and hit return:
```

Esto se repite (con el número de volumen incrementando) hasta que todos los ficheros especificados hayan sido archivados.

18.10.3. ?Puedo comprimir mis respaldos?

Desafortunadamente, tar(1) no permite el uso de la opción `-z` para archivos multi-volumen. Puede, por supuesto, hacer un gzip(1) a todos los ficheros, mandarlos con tar(1) a los disquetes, *y después hacer gunzip(1) a los archivos*

18.10.4. ?Cómo recupero mis respaldos?

Para restaurar el archivo completo use:

```
# tar Mxvf /dev/fd0
```

Hay dos maneras que puede usar para restaurar ficheros específicos. La primera, puede comenzar por el primer disco flexible y usar:

```
# tar Mxvf /dev/fd0 nombredefichero
```

La utilidad tar(1) le pedirá que inserte el resto de disquetes hasta que encuentre el fichero requerido.

La segunda manera es: si sabe en qué disco se encuentra el fichero puede insertar ese disco y usar el comando expuesto arriba. Tenga en cuenta que si el primer fichero en el disquete es la continuación del anterior tar(1) le advertirá que no puede restaurarlo *incluso si no se lo ha solicitado*

18.11. Bases para respaldos

Los tres principales programas para respaldos son dump(8), tar(1) y cpio(1).

18.11.1. Dump y Restore

Los programas UNIX® que se han usado durante muchos años para hacer copias de seguridad son `dump` y `restore`. Operan en las unidades como una colección de bloques de disco, bajo la abstracción de ficheros, los enlaces y directorios creados por el sistema de ficheros. `dump` respalda un sistema de ficheros completo en un dispositivo. No es capaz de respaldar solamente parte de un sistema de ficheros o un árbol de directorios que se extienda por más de un sistema de ficheros. `dump` no escribe ficheros y directorios a cinta, escribe los bloques de datos "crudos" (raw) que conforman los ficheros y directorios.

Nota

Si utiliza `dump` en su directorio raíz, no respaldará `/home`, `/usr` ni muchos otros directorios, ya que suelen ser puntos de montaje de otros sistemas de ficheros o enlaces simbólicos hacia dichos sistemas de ficheros.

`dump` tiene peculiaridades que se mantienen desde sus primeros días en la Version 6 de AT&T UNIX (alrededor de 1975). Los parámetros por defecto son los adecuados para cintas

de 9 pistas (6250 bpi), pero no para los medios de alta densidad disponibles hoy en día (hasta 62,182 ftpi). Estos valores por defecto deben obviarse en la línea de comandos para aprovechar la capacidad de las unidades de cinta actuales.

También es posible respaldar datos a través de la red a una unidad de cinta conectada a otra computadora con `rdump` y `rrestore`. Ambos programas dependen de rcmd(3) y ruserok(3) para acceder a la unidad de cinta remota. Por consiguiente, el usuario que realiza el respaldo debe estar listado en el fichero `.rhosts` de la computadora remota. Los argumentos para `rdump` y `rrestore` deben ser adecuados para usarse en la computadora remota. Cuando realice un `rdump` desde FreeBSD a una unidad de cinta Exabyte conectada a una Sun llamada `komodo`, use:

```
# /sbin/rdump 0dsbfu 54000 13000 126 komodo:/dev/nsa8 /dev/da0a 2>&1
```

Advertencia: existen implicaciones de seguridad al permitir autentificación mediante `.rhosts`. Le recomendamos que evalúe la situación cuidadosamente.

También es posible usar `dump` y `restore` de una forma más segura a través de `ssh`.

Ejemplo 18.1. Utilizando `dump` a través de ssh

```
# /sbin/dump -0uan -f - /usr | gzip -2 | ssh -c blowfish \
          usuario@maquinaobjetivo.ejemplo.com dd of=/
misficherosgrandes/dump-usr-l0.gz
```

Uso del método integrado de `dump`, configurando la variable de ambiente `RSH`:

Ejemplo 18.2. Uso de `dump` a través de ssh con `RSH` configurada

```
# RSH=/usr/bin/ssh /sbin/dump -0uan -f usuario@maquinaobjetivo.↺
ejemplo.com:/dev/sa0 /usr
```

18.11.2. `tar`

tar(1) también es de la época de la Version 6 de AT&T UNIX (alrededor de 1975). `tar` trabaja en cooperación con el sistema de ficheros; escribe ficheros y directorios a cinta. `tar`

no soporta el rango completo de opciones que ofrece cpio(1), pero no requiere el inusual comando de pipeline que utiliza `cpio`.

En FreeBSD 5.3 y posteriores, tiene a su disposición GNU `tar` y el comando por defecto `bsdtar`. La versión GNU puede ser invocada mediante `gtar`. Soporta dispositivos remotos mediante la misma sintaxis que `rdump`. Para hacer un `tar` a una unidad de cinta conectada a una Sun llamada `komodo`, use:

```
# /usr/bin/gtar cf komodo:/dev/nsa8 . 2>&1
```

Puede hacer lo mismo con o con `bsdtar` usando un "pipe" y `rsh` para mandar los datos a una unidad de cinta remota.

```
# tar cf - . | rsh nombredemaquina dd of=dispositivo-de-cinta obs=20b
```

Si le preocupa la seguridad del proceso de hacer un respaldo a través de una red debe usar `ssh` en lugar de `rsh`.

18.11.3. `cpio`

cpio(1) es el programa de intercambio de archivos de cinta para medios magnéticos. `cpio` tiene opciones (entre muchas otras) para realizar intercambio de bytes, escribir un número diferente de formatos de archivo y hacer "pipe" de datos hacia otros programas. Esta última opción hace de `cpio` una elección excelente para medios de instalación. `cpio` no sabe cómo recorrer el árbol de directorios, así que debe facilitarle una lista de directorios a través de `stdin`.

`cpio` no permite respaldos a través de la red. Puede usar un pipe y `rsh` para mandar los datos a una unidad de cinta remota.

```
# for f in lista_directorios; do
find $f << backup.list
done
# cpio -v -o --format=newc < backup.list | ssh usuario@máquina "cat > ↺
dispositivo_de_respaldo"
```

Donde *`lista_directorios`* es la lista de directorios que desea respaldar, *`usuario`*`@`*`máquina`* es la combinación usuario/nombre de equipo que realizará el respaldo y *`dispositivo_de_respaldo`* es donde el respaldo se escribirá efectivamente (por ejemplo `/dev/nsa0`).

18.11.4. `pax`

pax(1) es la respuesta IEEE/POSIX® a `tar` y `cpio`. A través de los años las diversas versiones de `tar` y `cpio` se han vuelto ligeramente incompatibles, así que en lugar de pelear por hacerlo completamente estándar, POSIX® creó una nueva utilidad de archivado. `pax`

trata de leer y escribir muchos de los diversos formatos de `cpio` y `tar`, además de nuevos formatos propios. Su conjunto de comandos se parece más a `cpio` que a `tar`.

18.11.5. Amanda

Amanda (Advanced Maryland Network Disk Archiver) es un sistema de respaldos cliente/servidor, en lugar de un solo programa. Un servidor Amanda respaldará a una sola unidad de cinta cualquier cantidad de computadoras que tengan clientes Amanda y una conexión de red al servidor Amanda. Un problema común en sitios con gran cantidad de discos grandes es que la cantidad de tiempo requerida para respaldar los datos directamente a cinta excede la cantidad de tiempo disponible para la tarea. Amanda resuelve este problema. Amanda puede usar un "disco intermedio" para respaldar varios sistemas de ficheros al mismo tiempo. Amanda crea "conjuntos de archivo", esto es, un grupo de cintas usadas durante un periodo de tiempo para crear respaldos completos de todos los sistemas de ficheros listados en el fichero de configuración de Amanda. El "conjunto de archivo" también contiene respaldos incrementales nocturnos (o diferenciales) de todos los sistemas de ficheros. Para restaurar un sistema de ficheros dañado hace falta el respaldo completo más reciente y los respaldos incrementales.

El fichero de configuración ofrece un control exhaustivo de los respaldos y del tráfico de red que Amanda genera. Amanda usará cualquiera de los programas de respaldo mencionados arriba para escribir los datos a cinta. Puede instalar Amanda como paquete y como port. No forma parte del sistema base.

18.11.6. No hacer nada

"No hacer nada" no es un programa, pero es la estrategia de respaldo más extendida. No tiene coste. No hay un calendario de respaldos a seguir. Simplemente hay que decir *que no*. Si algo le sucediera a sus datos *sonría y acostúmbrese a su nueva situación.*

Si su tiempo y sus datos valen poco o nada, entonces "no hacer nada" es el programa de respaldo más adecuado para usted. Pero cuidado, UNIX® es una herramienta muy poderosa y puede suceder que dentro de seis meses tenga un montón de ficheros que sean valiosos para usted.

"No hacer nada" es el método correcto de respaldo para `/usr/obj` y otros árboles de directorios que pueden ser fácilmente recreados por su computadora. Un ejemplo son los archivos que forman la versión HTML o PostScript® de este manual. Estos documentos han sido generados desde ficheros SGML. Crear respaldos de los archivos HTML o PostScript® no es necesario dado que los ficheros SGML se respaldan regularmente.

18.11.7. ?Cuál es el mejor programa de respaldos?

dump(8). *Y no hay más que hablar.* Elizabeth D. Zwicky realizó pruebas de estrés a a todos los programas de copia de seguridad aquí expuestos. La elección clarísima para preservar

todos sus datos y todas las peculiaridades de sus sistemas de ficheros UNIX® es `dump`. Elizabeth creó sistemas de ficheros conteniendo una gran variedad de condiciones inusuales (y algunos no tan inusuales) y probó cada programa haciendo un respaldo y restaurando esos sistemas de ficheros. Esas peculiaridades incluían: ficheros con y un bloque nulo, ficheros con caracteres extraños en sus nombres, ficheros que no se podían leer ni escribir, dispositivos, ficheros que cambiaban de tamaño durante el respaldo, ficheros que eran creados/borrados durante el respaldo y cosas así. Elizabeth presentó los resultados en LISA V en octubre de 1991. Consulte torture-testing Backup and Archive Programs.

18.11.8. Procedimiento de restauración de emergencia

18.11.8.1. Antes del desastre

Solamente existen cuatro pasos que debe realizar en preparación de cualquier desastre que pudiera ocurrir.

Primero, imprima la etiqueta de disco de cada uno de sus discos (`disklabel da0 | lpr`), su tabla de sistemas de ficheros (`/etc/fstab`) y todos los mensajes de arranque, dos copias de cada uno.

Segundo, asegúrese que los disquetes de rescate (`boot.flp` y `fixit.flp`) tienen todos sus dispositivos. La manera más fácil de revisarlo es reiniciar su máquina con el disquete en la unidad y revisar los mensajes de arranque. Si todos sus dispositivos aparecen en la lista y funcionan, pase al tercer paso.

Si ha habido algún problema tiene que crear dos disquetes de arranque personalizados, que deben tener un kernel que pueda montar todos sus discos y acceder a su unidad de cinta. Estos discos deben contener: `fdisk`, `disklabel`, `newfs`, `mount` y cualquier programa de respaldo que utilice. Estos programas deben estar enlazados estáticamente. Si usa `dump`, el disquete debe contener `restore`.

Tercero, use cintas de respaldo regularmente. Cualquier cambio que haga después de su último respaldo puede perderse irremediablemente. Proteja contra escritura las cintas de respaldo.

Cuarto, pruebe los disquetes (ya sea `boot.flp` y `fixit.flp` o los dos discos personalizados que creó en el segundo paso) y las cintas de respaldo. Documente el procedimiento. Almacene estas notas con los discos de arranque, las impresiones y las cintas de respaldo. Estará tan perturbado cuando restaure su sistema que las notas pueden pueden evitar que destruya sus cintas de respaldo. (?Como? en lugar de `tar xvf /dev/sa0` , puede teclear accidentalmente `tar cvf /dev/sa0` y sobreescribir su cinta).

Como medida adicional de seguridad haga discos de inicio y dos cintas de respaldo cada vez. Almacene una de cada en una ubicación remota. Una ubicación remota *NO* es el sótano del mismo edificio. Muchas firmas alojadas en el World Trade Center aprendieron esta lección de la manera más difícil. Esa ubicación remota debe estar separada físicamente de sus computadoras y unidades de disco por una distancia significativa.

Ejemplo 18.3. Un “script” para la creación de discos flexibles de arranque

```
#!/bin/sh
#
# create a restore floppy
#
# format the floppy
#
PATH=/bin:/sbin:/usr/sbin:/usr/bin

fdformat -q fd0
if [ $? -ne 0 -]
then
  echo "Bad floppy, please use a new one"
  exit 1
fi

# place boot blocks on the floppy
#
disklabel -w -B /dev/fd0c fd1440

#
# newfs the one and only partition
#
newfs -t 2 -u 18 -l 1 -c 40 -i 5120 -m 5 -o space /dev/fd0a

#
# mount the new floppy
#
mount /dev/fd0a /mnt

#
# create required directories
#
mkdir /mnt/dev
mkdir /mnt/bin
mkdir /mnt/sbin
mkdir /mnt/etc
mkdir /mnt/root
mkdir /mnt/mnt   # for the root partition
mkdir /mnt/tmp
mkdir /mnt/var

#
# populate the directories
#
if [ ! -x /sys/compile/MINI/kernel -]
then
```

```
  cat &lt;&lt; EOM
The MINI kernel does not exist, please create one.
Here is an example config file:
#
# MINI - A kernel to get FreeBSD onto a disk.
#
machine         "i386"
cpu             "I486_CPU"
ident           MINI
maxusers        5

options         INET                    # needed for _tcp ↺
_icmpstat _ipstat
                                        #            _udpstat ↺
_tcpstat _udb
options         FFS                     #Berkeley Fast File ↺
System
options         FAT_CURSOR              #block cursor in ↺
syscons or pccons
options         SCSI_DELAY=15           #Be pessimistic about ↺
Joe SCSI device
options         NCONS=2                 #1 virtual consoles
options         USERCONFIG              #Allow user ↺
configuration with -c XXX

config          kernel root on da0 swap on da0 and da1 dumps ↺
on da0

device          isa0
device          pci0

device          fdc0 at isa? port "IO_FD1" bio irq 6 drq 2 ↺
vector fdintr
device          fd0 at fdc0 drive 0

device          ncr0

device          scbus0

device          sc0 at isa? port "IO_KBD" tty irq 1 vector ↺
scintr
device          npx0 at isa? port "IO_NPX" irq 13 vector ↺
npxintr

device          da0
device          da1
device          da2

device          sa0

pseudo-device   loop            # required by INET
pseudo-device   gzip            # Exec gzipped a.out's
EOM
```

```
  exit 1
fi

cp -f /sys/compile/MINI/kernel /mnt

gzip -c -best /sbin/init > /mnt/sbin/init
gzip -c -best /sbin/fsck > /mnt/sbin/fsck
gzip -c -best /sbin/mount > /mnt/sbin/mount
gzip -c -best /sbin/halt > /mnt/sbin/halt
gzip -c -best /sbin/restore > /mnt/sbin/restore

gzip -c -best /bin/sh > /mnt/bin/sh
gzip -c -best /bin/sync > /mnt/bin/sync

cp /root/.profile /mnt/root

cp -f /dev/MAKEDEV /mnt/dev
chmod 755 /mnt/dev/MAKEDEV

chmod 500 /mnt/sbin/init
chmod 555 /mnt/sbin/fsck /mnt/sbin/mount /mnt/sbin/halt
chmod 555 /mnt/bin/sh /mnt/bin/sync
chmod 6555 /mnt/sbin/restore

#
# create the devices nodes
#
cd /mnt/dev
./MAKEDEV std
./MAKEDEV da0
./MAKEDEV da1
./MAKEDEV da2
./MAKEDEV sa0
./MAKEDEV pty0
cd /

#
# create minimum file system table
#
cat > /mnt/etc/fstab &lt;&lt;EOM
/dev/fd0a    /    ufs    rw  1  1
EOM

#
# create minimum passwd file
#
cat > /mnt/etc/passwd &lt;&lt;EOM
root:*:0:0:Charlie &amp;:/root:/bin/sh
EOM

cat > /mnt/etc/master.passwd &lt;&lt;EOM
root::0:0::0:0:Charlie &amp;:/root:/bin/sh
EOM
```

```
chmod 600 /mnt/etc/master.passwd
chmod 644 /mnt/etc/passwd
/usr/sbin/pwd_mkdb -d/mnt/etc /mnt/etc/master.passwd

#
# umount the floppy and inform the user
#
/sbin/umount /mnt
echo "The floppy has been unmounted and is now ready."
```

18.11.8.2. Después del desastre

La pregunta clave es: ?sobrevivió su hardware? Ha estado haciendo respaldos regularmente, así que no hay necesidad de preocuparse por el software.

Si el hardware ha sufrido daños los componentes deben reemplazarse antes de intentar de usar su sistema.

Si su hardware está bien revise sus discos de arranque. Si usa disquetes de arranque personalizados arranque en modo monousuario (teclée `-s` en el en el "prompt" de arranque `boot:`). Sáltese el siguiente párrafo.

Si utiliza usando los discos `boot.flp` y `fixit.flp` , siga leyendo. Inserte el disco `boot.flp` en la primera unidad de disquete y arranque la máquina. El menú de instalación original se desplegará en pantalla. Seleccione la opción `Fixit--Repair mode with CDROM or floppy.` . Inserte el disco `fixit.flp` cuando se le pida. Tanto `restore` como los demás programas que necesitará están en `/mnt2/rescue` (`/mnt2/stand` para versiones de FreeBSD anteriores a 5.2).

Recupere cada sistema de ficheros por separado.

Trate de montar (por ejemplo `mount /dev/da0a /mnt`) la partición raíz de su primer disco. Si la etiqueta del disco ha sufrido daños use `disklabel` para reparticionar y etiquetar el disco de forma que coincida con la etiqueta que imprimió y guardó previamente. Use `newfs` para crear de nuevo sus sistemas de ficheros. Monte de nuevo la partición raíz del disquete en modo lectura/escritura (`mount -u -o rw /mnt`). Ejecute su programa de respaldo y utilice las cintas de respaldo para restaurar sus datos en este sistema de ficheros (`restore vrf /dev/sa0`). Desmonte el sistema de ficheros (`umount /mnt`). Repita el proceso con cada sistema de ficheros que sufrió daños.

Una vez que su sistema esté en marcha respalde sus datos en cintas nuevas. Cualquiera que haya sido la causa de la caída o pérdida de datos puede suceder de nuevo. Una hora más que gaste ahora puede ahorrarle mucho sufrimiento más adelante.

18.12. Sistemas de ficheros en red, memoria y respaldados en fichero

Reorganizado y mejorado por Marc Fonvieille.

Además de los discos que conecta físicamente en su máquina (discos flexibles, CDs, discos duros, etc.) FreeBSD permite usar otro tipo de discos: *los discos virtuales*.

Esto incluye sistemas de ficheros en red como NFS y Coda, sistemas de ficheros basados en memoria y sistemas de ficheros basados en fichero.

Según la versión de FreeBSD que utilice tendrá que utilizar diferentes herramientas para la creación y uso de sistemas de ficheros en memoria y sistemas de ficheros basados en fichero.

Nota

Los usuarios de FreeBSD 4.X tendrán que usar MAKEDEV(8) para crear los dispositivos requeridos. FreeBSD 5.0 y posteriores usan devfs(5) para gestionar los nodos de dispositivo correspondientes de forma transparente para el usuario.

18.12.1. Sistema de ficheros basado en fichero en FreeBSD 4.X

La utilidad vnconfig(8) configura y habilita vnodes de dispositivos de pseudodisco. Un *vnode* es una representación de un fichero y es el enfoque de la actividad de fichero. Esto significa que vnconfig(8) utiliza ficheros para crear y operar un sistema de ficheros. Un uso posible es el montaje de imágenes de disquetes o CD almacenadas como ficheros.

Para poder usar vnconfig(8) necesitará tener vn(4) en el fichero de configuración de su kernel:

```
pseudo-device vn
```

Para montar una imagen de un sistema de ficheros:

Ejemplo 18.4. Uso de vnconfig para montar una imagen de un sistema de ficheros bajo FreeBSD 4.X

```
# vnconfig vn0 imagendedisco
```

```
# mount /dev/vn0c /mnt
```

Para crear una nueva imagen de un sistema de ficheros con vnconfig(8):

Ejemplo 18.5. Creación de una imagen nueva de un sistema de ficheros respaldado en un archivo con **vnconfig**

```
# dd if=/dev/zero of=nuevaimagen bs=1k count=5k
5120+0 records in
5120+0 records out
# vnconfig -s labels -c vn0 nuevaimagen
# disklabel -r -w vn0 auto
# newfs vn0c
Warning: 2048 sector(s) in last cylinder unallocated
/dev/vn0c:     10240 sectors in 3 cylinders of 1 tracks, 4096 ↺
sectors
        5.0MB in 1 cyl groups (16 c/g, 32.00MB/g, 1280 i/g)
super-block backups (for fsck -b #) at:
 32
# mount /dev/vn0c /mnt
# df /mnt
Filesystem  1K-blocks     Used    Avail Capacity  Mounted on
/dev/vn0c        4927        1     4532     0%    /mnt
```

18.12.2. Sistemas de ficheros basados en fichero en FreeBSD 5.X

mdconfig(8) se usa para configurar y habilitar discos habilitar discos de memoria, md(4), en FreeBSD 5.X. Para usar mdconfig(8), tendrá que cargar el módulo md(4) o añadir soporte para el mismo el el fichero de configuración del kernel:

```
device md
```

mdconfig(8) soporta tres tipos de discos virtuales en memoria: discos de memoria asignados mediante malloc(9), discos de memoria usando un fichero o espacio de swap como respaldo. Un uso posible es montar imágenes de disquetes o CD archivadas.

Para montar una imagen de un sistema de ficheros:

Ejemplo 18.6. Uso de `mdconfig` para montar una imagen de un sistema de ficheros en FreeBSD 5.X

```
# mdconfig -a -t vnode -f imagendedisco -u 0
# mount /dev/md0 /mnt
```

Para crear una imagen nueva de un sistema de ficheros con mdconfig(8):

Ejemplo 18.7. Creación de un disco respaldado en fichero con `mdconfig`

```
# dd if=/dev/zero of=nuevaimagen bs=1k count=5k
5120+0 records in
5120+0 records out
# mdconfig -a -t vnode -f nuevaimagen -u 0
# disklabel -r -w md0 auto
# newfs md0c
/dev/md0c: 5.0MB (10240 sectors) block size 16384, fragment ↺
size 2048
 using 4 cylinder groups of 1.27MB, 81 blks, 256 inodes.
super-block backups (for fsck -b #) at:
 32, 2624, 5216, 7808
# mount /dev/md0c /mnt
# df /mnt
Filesystem  1K-blocks     Used    Avail Capacity  Mounted on
/dev/md0c        4846        2     4458     0%    /mnt
```

Si no especifica el número de unidad con la opción `-u` mdconfig(8) usará la designación automática de md(4) para seleccionar un dispositivo sin usar. El nombre de la unidad designada se enviará a la salida esándar como `md4`. Para más información sobre mdconfig(8) consulte su página de manual.

Nota

A partir de FreeBSD 5.1-RELEASE bsdlabel(8) reemplazó a disklabel(8). En bsdlabel(8) se eliminaron muchas opciones y parámetros

obsoletos. En el ejemplo de arriba ignore la opción `-r`. Para más información consulte la página de manual de bsdlabel(8).

mdconfig(8) es muy útil, aunque requiere muchas líneas de comando para crear un sistema de ficheros basado en un fichero. FreeBSD 5.0 incorpora mdmfs(8), que configura un disco md(4) utilizando mdconfig(8), pone un sistema de ficheros UFS en él mediante newfs(8) y lo monta usando mount(8). Por ejemplo, si desea crear y montar la misma imagen de sistema de ficheros de arriba, simplemente teclée lo siguiente:

Ejemplo 18.8. Configurar y montar un disco basado en un fichero con `mdmfs`

```
# dd if=/dev/zero of=nuevaimagen bs=1k count=5k
5120+0 records in
5120+0 records out
# mdmfs -F newimage -s 5m md0 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md0        4846    2  4458     0%    /mnt
```

Si utiliza la opción `md` sin número de unidad, mdmfs(8) usará la opción de auto unidad de md(4) para selecionar automáticamente un dispositivo sin usar. Para más información sobre mdmfs(8) diríjase a la página de manual.

18.12.3. Sistemas de ficheros basados en memoria en FreeBSD 4.X

El controlador md(4) es un modo sencillo y eficiente de crear sistemas de ficheros basados en memoria en FreeBSD 4.X. malloc(9) se usa para ubicar la memoria.

Sencillamete toma un sistema de ficheros que usted ha preparado con, por ejemplo, vnconfig(8), y:

Ejemplo 18.9. Disco de memoria md en FreeBSD 4.X

```
# dd if=nuevaimagen of=/dev/md0
5120+0 records in
5120+0 records out
# mount /dev/md0c /mnt
# df /mnt
```

```
Filesystem  1K-blocks     Used    Avail Capacity  Mounted on
/dev/md0c        4927        1     4532     0%    /mnt
```

Para más información por favor consulte la página de manual de md(4).

18.12.4. sistemas de ficheros basados en memoria en FreeBSD 5.X

Se usan las mismas herramientas para tratar con sistemas de ficheros basados en memoria o en ficheros: mdconfig(8) o mdmfs(8). El almacenamiento de sistemas de ficheros basados en memoria requiere el uso de malloc(9).

Ejemplo 18.10. Creación de un nuevo disco basado en memoria con **mdconfig**

```
# mdconfig -a -t malloc -s 5m -u 1
# newfs -U md1
/dev/md1: 5.0MB (10240 sectors) block size 16384, fragment ↺
size 2048
 using 4 cylinder groups of 1.27MB, 81 blks, 256 inodes.
 with soft updates
super-block backups (for fsck -b #) at:
 32, 2624, 5216, 7808
# mount /dev/md1 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md1        4846    2  4458     0%    /mnt
```

Ejemplo 18.11. Creación de un nuevo disco basado en memoria con **mdmfs**

```
# mdmfs -M -s 5m md2 /mnt
# df /mnt
Filesystem 1K-blocks Used Avail Capacity  Mounted on
/dev/md2        4846    2  4458     0%    /mnt
```

En lugar de usar un sistema de ficheros respaldado en malloc(9), es posible utilizar swap; lo único que debe hacer es sustituir `malloc` por `swap` en la línea de comando de mdcon-

fig(8). mdmfs(8) por defecto (sin `-M`) crea un disco basado en swap). Para más información, consulte las páginas de manual de mdconfig(8) y de mdmfs(8).

18.12.5. Desconexión del sistema de un disco de memoria

Cuando un sistema de ficheros basado en memoria o basado en fichero no se usa puede liberar recursos del sistema. Lo primero es desmontar el sistema de ficheros: use mdconfig(8) para desconectar el disco del sistema y liberar dichos recursos.

Por ejemplo, para desconectar y liberar todos los recursos usados por `/dev/md4` :

```
# mdconfig -d -u 4
```

Es posible listar información sobre dispositivos md(4) configurados en el sistema mediante `mdconfig -l`.

En FreeBSD 4.X se usa vnconfig(8) para desconectar el dispositivo. Por ejemplo, para desconectar y liberar todos los recursos usados por `/dev/vn4` :

```
# vnconfig -u vn4
```

18.13. Instantáneas (“snapshots”) de sistemas de ficheros

Escrito por Tom Rhodes.

FreeBSD 5.0 ofrece una característica relacionada con Soft Updates: las instantáneas del sistema de ficheros.

Las instantáneas permiten a un usuario crear imágenes de uno o más sistemas de ficheros dados, y tratarlas como un fichero. Los ficheros de instantánea deben crearse en el sistema de ficheros en el que se realiza la acción, y un usuario puede crear hasta 20 (veinte) instantáneas por sistema de ficheros. Las instantáneas activas se graban en el superbloque, lo que hace que sigan ahí independientemente de montajes, remontajes y reinicios del sistema. Cuando ya no necesite una instantánea puede borrarla con rm(1). Las instantáneas pueden borrarse en cualquier orden pero puede que no pueda recuperar todo el espacio debido a que otra instantánea puede reclamar algunos bloques liberados.

La bandera inalterable de fichero `snapshot` se activa con mksnap_ffs(8) después de la creación inicial de un fichero de instantánea. unlink(1) hace una excepción con los ficheros de instantánea, ya que permite que se les borre.

Las instantáneas se crean con mount(8). Veamos un ejemplo. Vamos a colocar una instantánea de `/var` en `/var/snapshot/snap` :

```
# mount -u -o snapshot /var/snapshot/snap /var
```

También puede usar mksnap_ffs(8) para crear una instantánea:

```
# mksnap_ffs /var /var/snapshot/snap
```

Si busca ficheros de instantánea en un sistema de de ficheros (por ejemplo `/var`) puede usar find(1):

```
# find /var -flags snapshot
```

Una instantánea tiene distintos usos:

- Algunos administradores usan un fichero de instantánea como respaldo, puesto que la instantánea puede guardarse en CD o cinta.
- Integridad de ficheros; fsck(8) puede ejecutarse en una instantánea. Asumiendo que el sistema de ficheros estuviera limpio cuando se montó se debe obtener un resultado limpio (e intacto). En esencia el proceso fsck(8) hace esto mismo en segundo plano.
- Ejecución de dump(8) en la instantánea. Se obtendrá un dump consistente con el sistema de ficheros y los sellos de hora de la instantánea. dump(8) también puede leer una instantánea, crear una imagen dump y eliminar la instantánea en un comando usando la opción `-L`.
- Ejecutar un mount(8) contra la instantánea como una imagen congelada del sistema de ficheros. Para montar la instantánea `/var/snapshot/snap` ejecute:

```
# mdconfig -a -t vnode -f /var/snapshot/snap -u 4
# mount -r /dev/md4 /mnt
```

Podrá recorrer la jerarquía de su sistema de ficheros `/var` congelado montado en `/mnt`. Todo estará en el mismo estado en el que estaba cuando creó la instantánea. La única excepción es que cualquier instantánea anterior aparecerá como un fichero de longitud cero. Cuando haya acabado de usar una instantánea puede desmontarla con:

```
# umount /mnt
# mdconfig -d -u 4
```

Para más información sobre `softupdates` e instantáneas de sistemas ficheros, incluyendo textos técnicos, visite el sitio web de Marshall Kirk McKusick: `http://www.mckusick.com/` .

18.14. Cuotas en sistemas de ficheros

Las cuotas son una opción del sistema operativo que le permite limitar la cantidad de espacio en disco y/o el número de fichero que un usuario o miembros de un grupo pueden crear en el sistema, pudiendo además hacerlo de forma independiente en cada sistema de ficheros. Suele usarse principalmente en sistemas de tiempo compartido, donde se busca

limitar la cantidad de recursos que cualquier usuario o grupo pueden utilizar. Esto evitará que un usuario o un grupo de usuarios consuma todos el espacio disponible en disco.

18.14.1. Configuración del sistema para habilitar cuotas de disco

Antes de intentar configurar el uso de cuotas de disco hay que asegurarse de que las cuotas están activadas en el kernel. La siguiente línea debe estar en el fichero de de configuración del kernel:

```
options QUOTA
```

El kernel `GENERIC` no lo tiene activado por defecto, así que tendrá que configurar, compilar e instalar un kernel personalizado para poder usar cuotas de disco. Por favor, consulte el Capítulo 8, *Configuración del kernel de FreeBSD* para más información sobre la configuración del kernel.

A continuación tendrá que habilitar las cuotas de disco en `/etc/rc.conf` . Añadale la siguiente línea:

```
enable_quotas="YES"
```

Hay una variable que le permitirá efectuar un control más exhaustivo sobre el arranque de cuotas. Normalmente se revisa la integridad de cuotas de cada sistema de ficheros en el arranque; el responsable es quotacheck(8). quotacheck(8) se asegura de que los datos que hay en su base de datos de cuotas reflejen realmente los datos del sistema de ficheros. Es un proceso que lleva mucho tiempo y que afectará significativamente al tiempo que tardará su sistema en arrancar. Si desea saltarse ese paso puede usar una variable al efecto en `/etc/rc.conf` :

```
check_quotas="NO"
```

Para concluir tendrá que editar `/etc/fstab` para habilitar las cuotas de disco para cada sistema de ficheros. Es aquí donde podrá habilitar cuotas por usuario, por grupo, o ambos en todos sus sistemas de ficheros.

Para habilitar cuotas por usuario en un sistema de ficheros añada la opción `userquota` al campo de opciones en la entrada de `/etc/fstab` que corresponda al sistema de ficheros en el que quiere habilitar las cuotas. Veamos un ejemplo:

```
/dev/da1s2g   /home    ufs rw,userquota 1 2
```

En el caso de las cuotas de grupo es muy similar. Use la opción `groupquota` en lugar de `userquota`. Para habilitar cuotas por usuario y por grupo modifique la entrada de este modo:

```
/dev/da1s2g    /home    ufs rw,userquota,groupquota 1 2
```

Por defecto los ficheros de cuota se guardan en el directorio raíz del sistema de ficheros con los nombres `quota.user` y `quota.group` para cuotas de usuario y grupo respectiva-

mente. Consulte fstab(5) para más información. Aunque la página de manual de fstab(5) diga que puede especificar otra ubicación para los ficheros de cuota, no se recomienda hacerlo debido a que las diversas herramientas de gestió cuotas no parecen sobrellevar esto adecuadamente.

Hecho todo esto puede reiniciar su sistema con el nuevo kernel. `/etc/rc` ejecutará automáticamente los comandos apropiados para crear los ficheros de cuota iniciales que requieran todas las entradas en en `/etc/fstab` , así que no hay necesidad de crear ficheros de cuota de longitud cero.

En el curso normal de operaciones no se le debería pedir que ejecute quotacheck(8), quotaon(8) o quotaoff(8) manualmente. Sin embargo, tal vez quiera leer sus páginas de manual para familiarizarse con su funcionamiento.

18.14.2. Configuración de límites de cuota

Una vez que tenga configurado su sistema para usar cuotas verifique que en realidad estén habilitadas. Una manera sencilla de hacer esto es ejecutar:

```
# quota -v
```

Debe ver un resumen de una sola línea de uso del disco y los límites de cuota actuales para cada sistema de ficheros donde estén habilitadas las cuotas.

Ahora puede iniciar la asignación de límites de cuota con edquota(8).

Tiene varias opciones para imponer límites en el espacio de disco que un usuario o grupo puede ocupar, y cuántos ficheros pueden crear. Puede limitar el uso de disco basándose en el espacio en disco (cuotas de bloque) o en el número de ficheros (cuotas de inodo) o una combinación de ambas. Cada uno de estos límites a su vez se divide en dos categorías: límites duros y suaves.

Un límite duro no puede ser excedido. Una vez que un usuario alcanza su límite duro no puede realizar más ubicaciones en el sistema de ficheros en cuestión. Por ejemplo, si el usuario tiene un límite duro de 500 kbytes en un sistema de ficheros y está utilizando 490 kbytes, el usuario solo puede ocupar otros 10 kbytes. Un intento de ocupar 11 kbytes más fallará.

Los límites suaves pueden excederse por un periodo Este periodo de tiempo recibe el nombre de periodo de gracia, que por defecto es una semana. Si un usuario sobrepasa su periodo de gracia el límite suave se convertirá en un límite duro y no se permitir´n usos de disco adicionales. Cuando el usuario devuelve su cuota de uso de recursos a un punto por debajo de su límite suave el periodo de gracia se reinicia.

Veamos un ejemplo de uso de edquota(8). Si se usa edquota(8) se entra en el el editor declarado en la variable de entorno `EDITOR`, o en el editor vi si no ha modificado el valor por defecto de la variable `EDITOR`, para que pueda editar los límites de cuota.

```
# edquota -u test
```

```
Quotas for user test:
/usr: kbytes in use: 65, limits (soft = 50, hard = 75)
        inodes in use: 7, limits (soft = 50, hard = 60)
/usr/var: kbytes in use: 0, limits (soft = 50, hard = 75)
        inodes in use: 0, limits (soft = 50, hard = 60)
```

Debería ver dos líneas por cada sistema de ficheros que tenga habilitadas las cuotas. Una línea para los límites de bloque y una línea para límites de inodo. Por ejemplo, para elevar los límites de este usuario de un límite suave de 50 y un límite duro de 75 a un límite suave de 500 y un límite duro de 600, cambie:

```
/usr: kbytes in use: 65, limits (soft = 50, hard = 75)
```

por:

```
/usr: kbytes in use: 65, limits (soft = 500, hard = 600)
```

Los nuevos límites de cuota se aplicarán en cuanto salga del editor.

Algunas veces se quieren activar límites de cuota en un rango de UIDs. Esto puede realizarse con la opción `-p` de edquota(8). Primero asigne el límite de cuota deseado a un usuario y luego ejecute `edquota -p protouser startuid-enduid`. Por ejemplo, si el usuario `test` tiene el límite de cuota deseado, el siguiente comando puede usarse para duplicar esos límites de cuota para los UIDs de 10,000 hasta 19,999:

```
# edquota -p test 10000-19999
```

Para más información consulte la página de manual edquota(8).

18.14.3. Revisión de los límites de cuota y uso de disco

Puede usar quota(1) o repquota(8) para revisar los límites de cuota y uso del disco. El comando quota(1) le permitirá revisar cuotas individuales de usuario o grupo y uso del disco. Un usuario puede sólamente examinar su propia cuota y la cuota de un grupo al que pertenezca. Sólamente el superusuario puede ver las cuotas de todos los usuarios y grupos. repquota(8) permite obtener un resumen de todas las cuotas y uso del disco de todos los sistemas de ficheros con cuotas habilitadas.

En el siguiente ejemplo vemos la salida de `quota -v` para un usuario que tiene límites de cuota en dos sistemas de ficheros.

```
Disk quotas for user test (uid 1002):
     Filesystem  usage    quota   limit   grace   files   quota   ↺
limit   grace
          /usr      65*     50      75   5days       7      50
  60
```

```
       /usr/var        0      50      75              0      50
  60
```

En el sistema de ficheros `/usr` del ejemplo este usuario está actualmente 15 kbytes sobre su límite suave de 50 kbytes y le quedan 5 días de su periodo de gracia. Observe el asterisco, * que indica que el usuario está actualmente por encima de su límite de cuota.

Normalmente los sistemas de ficheros en los que el usuario no esté utilizando espacio en disco no se mostrarán en la salida del comando quota(1), incluso si tiene un límite de cuota asignado para esos sistemas de fichero. La opción `-v` desplegará esos sistemas de ficheros, en nuestro ejemplo el sistema de ficheros `/usr/var`.

18.14.4. Cuotas en NFS

Las cuotas son impuestas por el subsistema de cuotas en el servidor NFS. El dæmon rpc.rquotad(8) facilita la información a quota(1) en los clientes NFS, permitiéndoles a los usuarios de esas máquinas ver sus estadísticas de cuota.

Habilite `rpc.rquotad` en `/etc/inetd.conf` del siguiente modo:

```
rquotad/1      dgram rpc/udp wait root /usr/libexec/rpc.rquotad rpc.↺
rquotad
```

Y reinicie `inetd`:

```
# kill -HUP `cat /var/run/inetd.pid`
```

18.15. Cifrado de particiones de disco

Escrito por Lucky Green.

FreeBSD ofrece un alto grado de protección contra el acceso no autorizado a los datos. Los Permisos de fichero y MAC (Mandatory Access Control, controles de acceso obligatorio, consulte el Capítulo 16, *Mandatory Access Control*) ayudan a evitar que otros tengan acceso no autorizado a los datos mientras el sistema operativo está funcionando y la computadora está encendida. Sin embargo los permisos impuestos por el sistema operativo son irrelevantes si un atacante tiene acceso físico al sistema y puede simplemente mover el disco duro de la computadora a otro sistema para copiar y analizar datos sensibles.

Independientemente de cómo un atacante pueda conseguir acceso a un disco duro a a un sistema apagado, el cifrado de disco basado en GEOM (GEOM Based Disk Encryption, gbde) puede proteger los datos de los sistemas de ficheros del sistema incluso contra atacantes muy decididos y con recursos adecuados a su disposición. A diferencia de otros métodos de cifrado más difíciles de usar, que cifran únicamente ficheos individuales, gbde cifra sistemas de ficheros completos de forma transparente. Ni un solo texto en limpio llega a tocar el disco duro.

18.15.1. Habilitar gbde en el kernel

1. Conviértase en **root**

   La configuración de gbde requiere privilegios de superusuario.

   ```
   % su -
   Password:
   ```

2. Verifique la versión del sistema operativo

   gbde(4) requiere FreeBSD 5.0 o posterior.

   ```
   # uname -r
   5.0-RELEASE
   ```

3. Añada soporte de gbde(4) al fichero de configuración de su kernel

   Añada la siguiente línea al fichero de configuración de su kernel con el editor que prefiera:

   ```
   options GEOM_BDE
   ```

   Configure, recompile e instale el kernel de FreeBSD. Este proceso se detalla en el Capítulo 8, *Configuración del kernel de FreeBSD*.

   Reinicie con el nuevo kernel.

18.15.2. Preparación del disco duro cifrado

El siguiente ejemplo asume que añade a su sistema un disco duro nuevo que contendrá una sola partición cifrada. Esta partición se montará como `/private`. gbde puede usarse también para cifrar `/home` y `/var/mail` , pero esto requeriría instrucciones más complejas que las que se pretenden dar en esta introducción.

1. Añada el nuevo disco

   Instale el nuevo disco en el sistema como se explicó en la Sección 18.3, "Añadir discos". En nuestro ejemplo hemos añadido una nueva partición de disco como `/dev/ad4s1c`. Los dispositivos `/dev/ad0s1*` representan particiones FreeBSD estándar que i existían previamente en el sistema.

   ```
   # ls /dev/ad*
   /dev/ad0        /dev/ad0s1b     /dev/ad0s1e     /dev/ad4s1
   /dev/ad0s1      /dev/ad0s1c     /dev/ad0s1f     /dev/ad4s1c
   /dev/ad0s1a     /dev/ad0s1d     /dev/ad4
   ```

2. Crée un directorio para los ficheros "lock" de gbde

   ```
   # mkdir /etc/gbde
   ```

Los ficheros “lock” de gbde contienen información que gbde requiere para acceder a las particiones cifradas. Sin el acceso a los ficheros “lock” gbde no podrá descifrar los datos alojados en la partición cifrada sin una cantidad significativa de trabajo, tarea para la que además no le resultará de ayuda este software. Cada partición cifrada utiliza un fichero “lock” separado.

3. Inicialice la partición gbde

   Una partición gbde debe inicializarse antes de que pueda utilizarse. Esta inicialización sólo debe hacerse una vez:

```
# gbde init /dev/ad4s1c -i -L /etc/gbde/ad4s1c
```

   gbde(8) abrirá su editor para que pueda configurar las opciones de configuración que se le presentarán en una plantilla. Para utilizar UFS1 o UFS2, ponga el sector_size a 2048:

```
$FreeBSD: src/sbin/gbde/template.txt,v 1.1 2002/10/20 11:16:13 ↺
phk Exp $
#
# El tamaño de sector (sector size) es la unidad de datos más
# pequeña que podrá leer o escribir.  Si la elige demasiado
# pequeña reducirá el rendimiento y la cantidad de espacio
# útil.  Si la elige demasiado grande puede hacer que los sistemas
# de ficheros no funcionen.  512 es el tamaño mínimo y
# siempre funciona.  Si va a usar UFS utilice
#
sector_size     =       2048
[...-]
```

   gbde(8) le pedirá dos veces que escriba la contraseña que debe usarse para asegurar los datos. La contraseña debe ser la misma las dos veces. La capacidad de gbde de proteger sus datos depende íntegramente de la calidad de la contraseña que elija. [2]

   El fichero `gbde init` crea un fichero “lock” para su partición gbde, que en nuestro ejemplo está en `/etc/gbde/ad4s1c` .

Atención

Es imprescindible que los ficheros “lock” de gbde *deben* respaldarse junto con el contenido de cualquier partición cifrada. Aunque la sola acció de borrar un fichero “lock” no puede evitar que un atacante motivado descifre una partición gbde sin

[2] Si quiere ayuda para seleccionar una contraseña segura que además sea fácil de recordar visite el sitio web Diceware Passphrase.

el fichero “lock”, el propietario legítimo no podrá acceder a los datos en la partición cifrada sin una cantidad notable de trabajo, que es necesario señalar que no entra dentro de las funciones de gbde(8) ni de su diseñador.

4. Conecte al kernel la partición cifrada

```
# gbde attach /dev/ad4s1c -l /etc/gbde/ad4s1c
```

Se le pedirá la contraseña que elijió al inicializar la partición cifrada. El nuevo dispositivo cifrado aparecerá en `/dev` como `/dev/nombre_de_dispositivo.bde` :

```
# ls /dev/ad*
/dev/ad0        /dev/ad0s1b     /dev/ad0s1e     /dev/ad4s1
/dev/ad0s1      /dev/ad0s1c     /dev/ad0s1f     /dev/ad4s1c
/dev/ad0s1a     /dev/ad0s1d     /dev/ad4        /dev/ad4s1c.bde
```

5. Crée un sistema de ficheros en el dispositivo cifrado

Una vez el dispositivo cifrado está conectado al kernel puede crear un sistema de ficheros en el dispositivo con newfs(8). Dado que es más rápido inicializar un sistema de ficheros del nuevo UFS2 que un sistema de ficheros del tradicional UFS1, le recomendamos encarecidamente usar newfs(8) con la opción `-O2`.

Nota

La opción `-O2` es el valor por defecto en FreeBSD 5.1-RELEASE y siguientes.

```
# newfs -U -O2 /dev/ad4s1c.bde
```

Nota

newfs(8) debe ejecutarse en una partición gbde conectada, que podrá identificar por la extensión `*.bde` del nombre del dispositivo.

6. Montar la partición cifrada

   Crée un punto de montaje para el sistema cifrado de ficheros.

   ```
   # mkdir /private
   ```

   Montar el sistema cifrado de ficheros.

   ```
   # mount /dev/ad4s1c.bde /private
   ```

7. Verificar que el sistema cifrado de ficheros esté disponible

   el sistema cifrado de ficheros debería ser visible para df(1) y estar listo para su uso.

   ```
   % df -H
   Filesystem        Size   Used  Avail Capacity  Mounted on
   /dev/ad0s1a      1037M    72M   883M     8%    /
   /devfs            1.0K   1.0K     0B   100%    /dev
   /dev/ad0s1f       8.1G    55K   7.5G     0%    /home
   /dev/ad0s1e      1037M   1.1M   953M     0%    /tmp
   /dev/ad0s1d       6.1G   1.9G   3.7G    35%    /usr
   /dev/ad4s1c.bde   150G   4.1K   138G     0%    /private
   ```

18.15.3. Montaje de sistemas cifrados de ficheros

Todos los sistemas cifrados de ficheros deben reconectarse al kernel después de cada arranque. Además, antes de poder utilizarlo debe revisarlo por si contuviera errores y montarlo. Todo el proceso debe ser ejecutado por el usuario `root`.

1. Conectar la partición gbde al kernel

   ```
   # gbde attach /dev/ad4s1c -l /etc/gbde/ad4s1c
   ```

   Se le pedirá la contraseña que elijió en la inicialización de la partición cifrada gbde.

2. Revisión de errores en el sistema de ficheros

   Como que los sistemas cifrados de ficheros no pueden aparecer en `/etc/fstab` (lo que haría que fueran montados automáticamente), los sistemas de ficheros deben revisarse manualmente mediante fsck(8) antes de montarlos.

   ```
   # fsck -p -t ffs /dev/ad4s1c.bde
   ```

3. Montar los sistemas cifrados de ficheros

   ```
   # mount /dev/ad4s1c.bde /private
   ```

   El sistema cifrado de ficheros está listo para su uso.

18.15.3.1. Montar automáticamente particiones cifradas

Es posible usar un “script” para automatizar la conexión, revisión y el montaje de una partición cifrada, pero por razones de seguridad el “script” no debe contener la contraseña de gbde(8). Se recomienda ejecutar esos “scripts” se ejecuten de forma manual proporcionando la contraseña vía consola o ssh(1).

18.15.4. Protección criptográfica que usa gbde

gbde(8) cifra el XXX sector payload usando AES de 128 bits en modo CBC. Cada sector en el disco se cifra con una clave AES diferente. Para más información sobre el diseño criptográfico de gbde, incluyendo cómo se derivan las claves de sector a partir de la contraseña consulte gbde(4).

18.15.5. Problemas de compatibilidad

sysinstall(8) es incompatible con dispositivos gbde cifrados. Todos los dispositivos `*.bde` deben desconectarse del kernel antes de iniciar sysinstall(8) o se “congelará” durante la prueba inicial de dispositivos. Para desconectar el el dispositivo cifrado de nuestro ejemplo haga lo siguiente:

```
# gbde detach /dev/ad4s1c
```

Tenga en cuenta también que, como vinum(4) no utiliza el subsistema geom(4), no es posible usar gbde en volúmenes vinum.

Capítulo 19. GEOM: Marco de trabajo modular de transformación de discos

Escrito por Tom Rhodes.

Traducción de Enrique Matías Sánchez <cronopios@gmail.com >.

19.1. Sinopsis

Este capítulo explica el uso de discos bajo el marco de trabajo GEOM en FreeBSD. Esto incluye las principales utilidades de control de RAID que usan el marco de trabajo para su configuración. Este capítulo no se adentrará en un examen en profundidad de como GEOM maneja o controla la E/S, el subsistema subyacente, o el código. Esta información se proporciona en la página de manual geom(4) y sus diversas referencias VEA TAMBIÉN. Este capítulo tampoco es una guía definitiva de configuraciones RAID. Sólo se examinan las clasificaciones de RAID que puede usar GEOM.

Tras leer este capítulo, sabrá:

- Que tipo de soporte para RAID está disponible a través de GEOM.
- Como utilizar las utilidades base para configurar, mantener, y manipular los diversos niveles de RAID.
- Como replicar, unir, cifrar, y conectar remotamente dispositivos de disco por medio de GEOM.
- Como solucionar problemas con los discos adscritos al marco de trabajo GEOM.

Antes de leer este capítulo, debería:

- Entender como trata FreeBSD a los dispositivos de disco (Capítulo 18, *Almacenamiento*).
- Saber como configurar e instalar un nuevo núcleo de FreeBSD (Capítulo 8, *Configuración del kernel de FreeBSD*).

19.2. Introducción a GEOM

GEOM permite el acceso y control de clases —sectores de arranque maestros (MBR), etiquetas BSD, etc— por medio del uso de proveedores, o de los ficheros especiales de `/dev`.

Capaz de trabajar con varias configuraciones de RAID por software, GEOM proporcionará transparentemente acceso al sistema operativo y las utilidades del mismo.

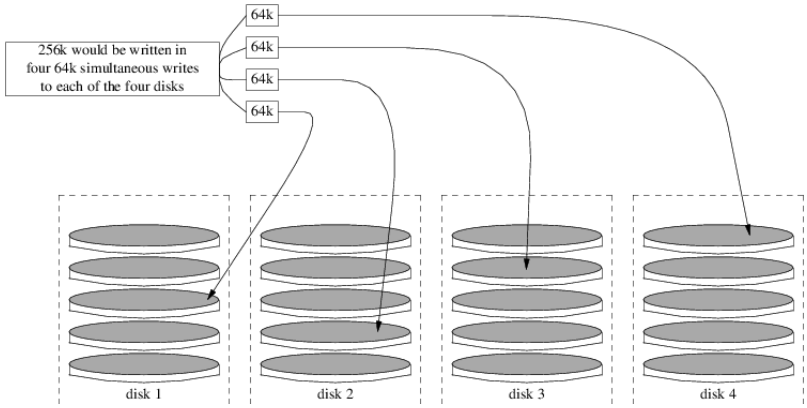
19.3. RAID0 - Distribución por bandas

Escrito por Tom Rhodes y Murray Stokely.

La distribución por bandas (striping) es un método que se usa para combinar varias unidades de disco en un único volumen. En muchos casos, esto se hace usando controladoras por hardware. El subsistema de discos GEOM proporciona soporte por software para RAID0, también conocido como discos configurados en bandas.

En un sistema RAID0, los datos se dividen en bloques que son escritos por todas las unidades de la agrupación. En lugar de tener que esperar a que el sistema escriba 256 kB en un disco, un sistema RAID0 puede escribir simultáneamente 64 kB en cada uno de los cuatro discos, ofreciendo un superior rendimiento de E/S. Este rendimiento se puede mejorar aún más usando varias controladoras de disco.

Cada uno de los discos de una banda RAID0 debe ser del mismo tamaño, pues las peticiones de E/S están intercaladas para leer o escribir en varios discos en paralelo.



Procedimiento 19.1. Creación de una banda de discos ATA sin formatear

1. Cargue el módulo `geom_stripe` :

```
# kldload geom_stripe.ko
```

2. Asegúrese de que existe un punto de montaje adecuado. Si este volumen se convertirá en una partición raíz, utilice temporalmente otro punto de montaje, como `/mnt`.

```
# mkdir /mnt
```

3. Determine los nombres de dispositivo de los discos que serán configurados en bandas, y cree el nuevo dispositivo de banda. Por ejemplo, podría utilizar la siguiente orden para configurar en bandas dos discos ATA sin usar ni particionar: `/dev/ad2` y `/dev/ad3`.

```
# gstripe label -v st0 /dev/ad2 /dev/ad3
```

4. Si se va a usar este volumen como dispositivo raíz para arrancar el sistema, debe ejecutar la siguiente orden antes de crear el sistema de ficheros:

```
# fdisk -vBI /dev/stripe/st0
```

5. Se debe crear una tabla de particiones en el nuevo volumen con la siguiente orden:

```
# bsdlabel -wB /dev/stripe/st0
```

6. Aademás del dispositivo `st0`, este proceso debería haber creado otros dos dispositivos en el directorio `/dev/stripe`, incluyendo `st0a` y `st0c`. Ahora se debe crear un sistema de ficheros en el dispositivo `st0a` usando la siguiente orden `newfs`:

```
# newfs -U /dev/stripe/st0a
```

   Por la pantalla se deslizarán muchos números, y al cabo de unos pocos segundos, el proceso habrá finalizado. El volumen ha sido creado y está preparado para ser montado:

Se puede usar la siguiente orden para montar manualmente una banda de discos recién creada:

```
# mount /dev/stripe/st0a /mnt
```

Para montar automáticamente este sistema de ficheros distribuido por bandas durante el proceso de arranque, ponga la información del volumen en el fichero `/etc/fstab`:

```
# echo "/dev/stripe/st0a /mnt ufs rw 2 2" \
    >> /etc/fstab
```

También se debe cargar automáticamente durante la inicialización del sistema el módulo geom, añadiendo una línea a `/boot/loader.conf` :

```
# echo 'geom_stripe_load="YES"' >> /boot/loader.conf
```

19.4. RAID1 - Replicación

La replicación es una tecnologín que usan muchas empresas y usuarios para hacer copias de respaldo de sus datos sin interrupciones. Cuando hay una réplica, simplemente signi-

fica que el discoB replica al discoA. O, quizá el discoC+D replica al discoA+B. Al margen de la configuración de los discos, lo importante es que la información de un disco o partición está siendo replicada. Más adelante se podría restaurar esa información más fácilmente, hacerse una copia de respaldo sin provocar intrrupciones de servicio o acceso, e incluso almacenarla físicamente en una caja fuerte para datos.

Para empezar, asegúrese de que el sistema tiene dos unidades de disco del mismo tamaño, eb este ejercicio se supone que son discos SCSI de acceso directo (da(4)).

Comience por instalar FreeBSD en el primer disco con sólo dos particiones. Una debería ser una partición de intercambio, de dos veces el tamaño de la RAM, y todo el espacio restante se dedicará al sistema de ficheros raíz (`/`). Es posible tener particiones aparte para otros puntos de montajes; sin embargo, esto multiplicará por diez el nivel de dificultad, debido a la alteración manual de las opciones de bsdlabel(8) y fdisk(8).

Reinicie y espere a que el sistema se inicie por completo. Una vez haya finalizado este proceso, ingrese como usuario `root`.

Cree el dispositivo `/dev/mirror/gm` y enlácelo a `/dev/da1`:

```
# gmirror label -vnb round-robin gm0 /dev/da1
```

El sistema debería responder con:

```
Metadata value stored on /dev/da1.
Done.
```

Inicialice GEOM, esto cargará el módulo del núcleo `/boot/kernel/geom_mirror.ko` :

```
# gmirror load
```

Nota

Esta orden debería haber creado en el directorio `/dev/mirror` los nodos de dispositivo `gm0`, `gm0s1`, `gm0s1a`, y `gm0s1c`.

Instale una etiqueta genérica `fdisk` y el código de arranque en el recién creado dispositivo `gm0`:

```
# fdisk -vBI /dev/mirror/gm0
```

Ahora instale la información `bsdlabel` genérica:

```
# bsdlabel -wB /dev/mirror/gm0s1
```

Nota

Si hay varias slices (rodajas) y particiones, necesitará modificar las opciones de las dos órdenes anteriores. Deben coincidir con la slice y tamaño de partición del otro disco.

Utilice la utilidad newfs(8) para crear un sistema de ficheros predefinido en nodo de dispositivo `gm0s1a`:

```
# newfs -U /dev/mirror/gm0s1a
```

Esto debería haber hecho que el sistema mostrara alguna información y un puñado de números. Esto es bueno. Examine la pantalla por si hay algún mensaje de error y monte el dispositivo en el punto de montaje `/mnt`:

```
# mount /dev/mirror/gm0s1a /mnt
```

Ahora mueva todos los datos del disco de arranque a este nuevo sistema de ficheros. Este ejemplo usa las órdenes dump(8) y restore(8); aunque, dd(1) también debería funcionar en este escenario. Evitamos utilizar tar(1) porque no copiará el código de arranque. De ese modo, el fallo estaría garantizado.

```
# dump -L -0 -f- / |(cd /mnt && restore -r -v -f-)
```

Se debe hacer esto para cada sistema de ficheros. Simplemente ponga el sistema de ficheros adecuado en la ubicación correcta al ejecutar la orden mencionada.

Ahora edite el fichero replicado `/mnt/etc/fstab` y elimine o comente el fichero swap [1]. Cambie la información del otro sistema de ficheros para que utilice el nuevo disco. Vea el siguiente ejemplo:

```
# Device                Mountpoint      FStype  Options         Dump
   Pass#
#/dev/da0s2b             none            swap    sw              0
    0
/dev/mirror/gm0s1a       /               ufs     rw              1
    1
```

Ahora cree un fichero `boot.conf` tanto en la partición actual como en la nueva partición raíz. Este fichero "ayudará" al BIOS del sistema a arrancar la unidad correcta:

[1] Debe advertirse que comentar la entrada del fichero swap en `fstab` probablemente le obligará a reestablecer una manera diferente de habilitar el espacio de intercambio. Consulte Sección 11.4, "Añadir espacio swap" para más información.

```
# echo "1:da(1,a)/boot/loader" > /boot.config
```

```
# echo "1:da(1,a)/boot/loader" > /mnt/boot.config
```

Nota

Lo hemos colcoado en ambas particiones raíz para asegurar un arranque correcto. Si por alguna razón el sistema no pudiera leer en la nueva partición raíz, está disponible un arranque a prueba de fallos.

Ahora agregue la siguiente línea al nuevo `/boot/loader.conf` :

```
# echo 'geom_mirror_load="YES"' >> /mnt/boot/loader.conf
```

Esto le dice a la utilidad loader(8) que cargue el `geom_mirror.ko` durante la inicialización del sistema.

Reinicie el sistema:

```
# shutdown -r now
```

Si todo ha ido bien, el sistema debería haber arrancado desde el dispositivo `gm0s1a`, y un prompt `login` debería estar a la espera. Si algo fue mal, consulte la sección posterior de resolución de problemas. Ahora agregue el disco `da0` al dispositivo `gm0`:

```
# gmirror configure -a gm0
# gmirror insert gm0 /dev/da0
```

La opción `-a` le dice a gmirror(8) que use sincronización automática; por ejemplo, que replique las escrituras en disco automáticamente. La página de manual explica como reconstruir y reemplazar discos, aunque utiliza `data` en vez de `gm0`.

19.4.1. Resolución de problemas

19.4.1.1. El sistema se niega a arrancar

Si el sistema arranca hasta un prompt similar a:

```
ffs_mountroot: can't find rootvp
Root mount failed: 6
mountroot>
```

Reinicie la máquina utilizando el botón de encendido o el de reset. En el menú de arranque, seleccione la opción seis (6). Esto llevará al sistema a un prompt de loader(8). Cargue el módulo del núcleo manualmente:

```
OK? load geom_mirror.ko
OK? boot
```

Si esto funciona, es que por alguna razón el módulo no se cargaba correctamente. Ponga:

```
options GEOM_MIRROR
```

en el fichero de configuración del núcleo, recompile y reinstale. Esto debería solucionar el problema.

Capítulo 20. El Gestor de Volúmenes Vinum

20.1. Sinopsis

20.2. Los Discos son Demasiado Pequeños

Texto Original de Greg Lehey.

20.3. Cuellos de Botella en el Acceso

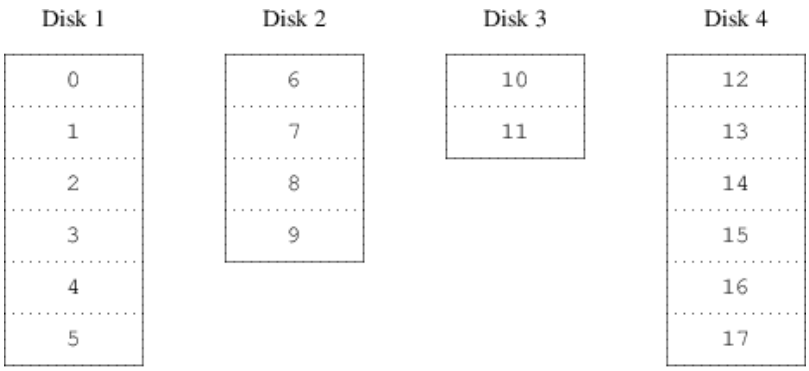


Figura 20.1. Organización Concatenada

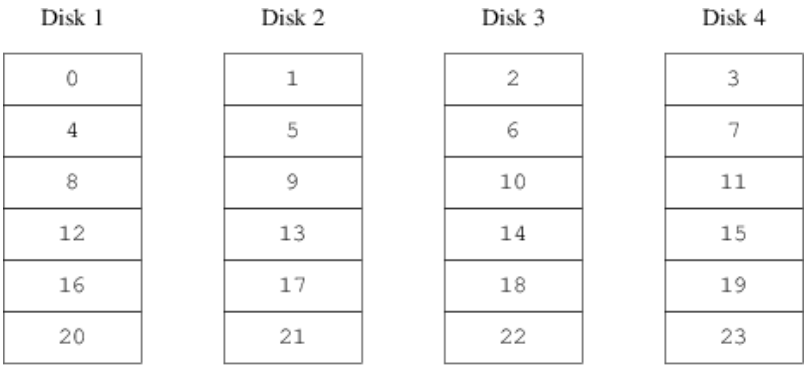


Figura 20.2. Organización con “Striping”

20.4. Integridad de Datos

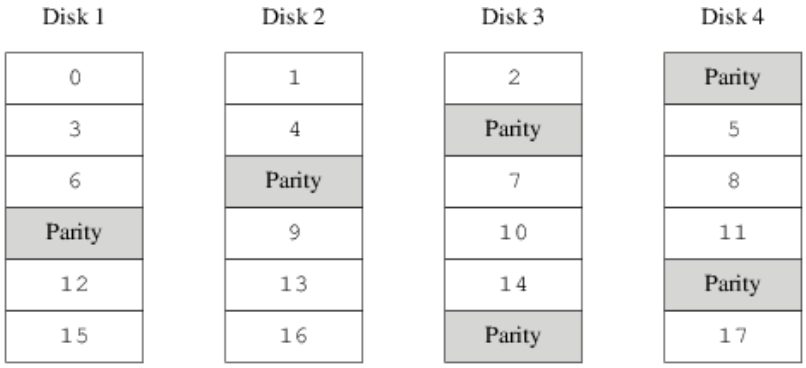


Figura 20.3. Organización en RAID-5

20.5. Objetos Vinum

20.5.1. Consideraciones sobre el Tamaño de los Volúmenes

20.5.2. Almacenamiento Redundante de Datos

20.5.3. Cuestiones Relacionadas con el Rendimiento

20.5.4. Which Plex Organization?

20.5.5. Vinum Plex Organizations

20.6. Ejemplos

20.6.1. El Fichero de Configuración

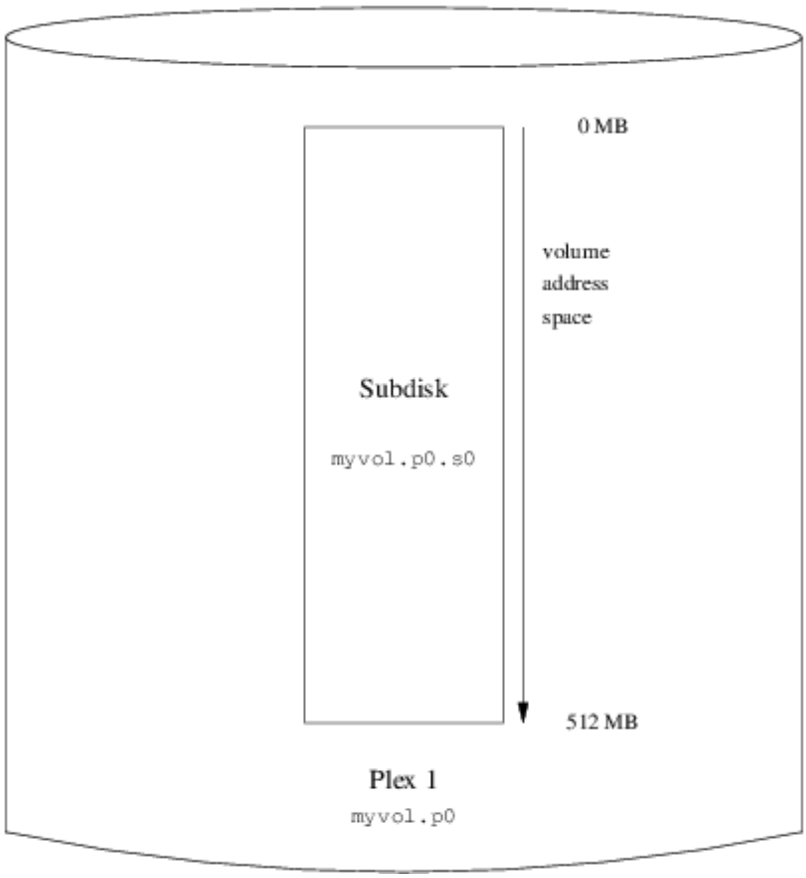


Figura 20.4. Un Volumen Vinum Sencillo

20.6.2. Increased Resilience: Mirroring

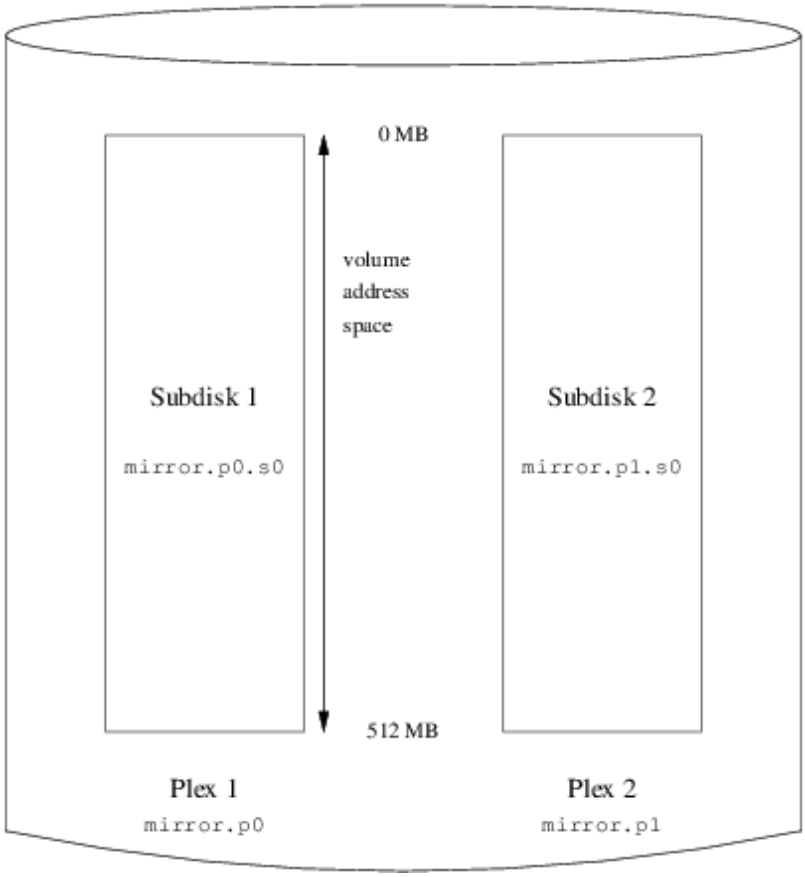


Figura 20.5. Un Volumen Vinum Replicado

20.6.3. Optimización del Rendimiento

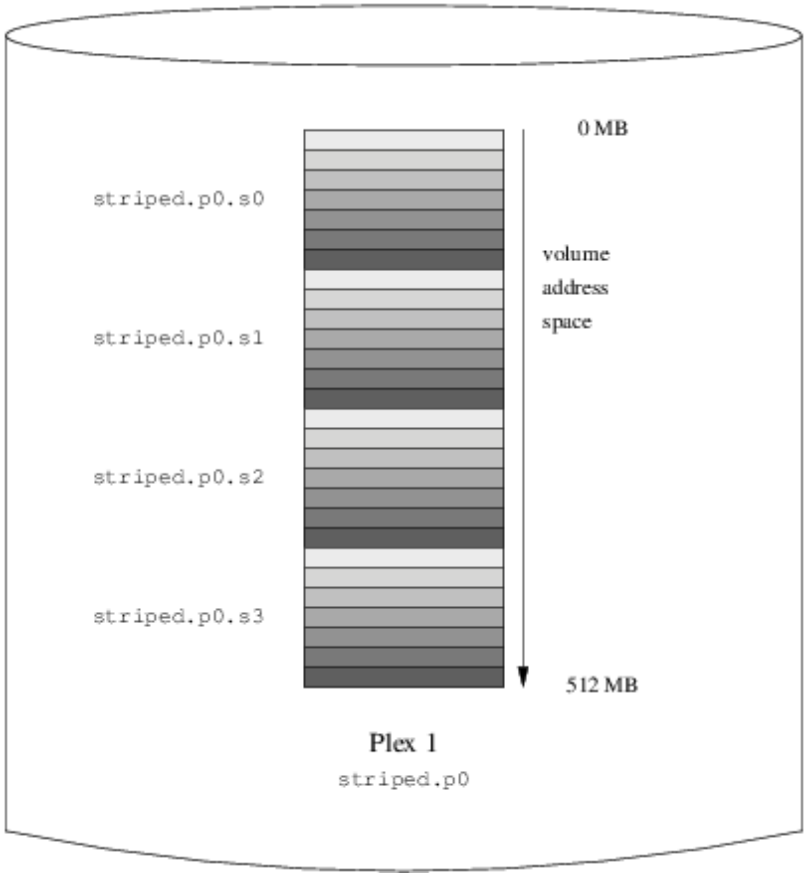


Figura 20.6. Un Volumen Vinum en “Striping”

20.6.4. Resilience and Performance

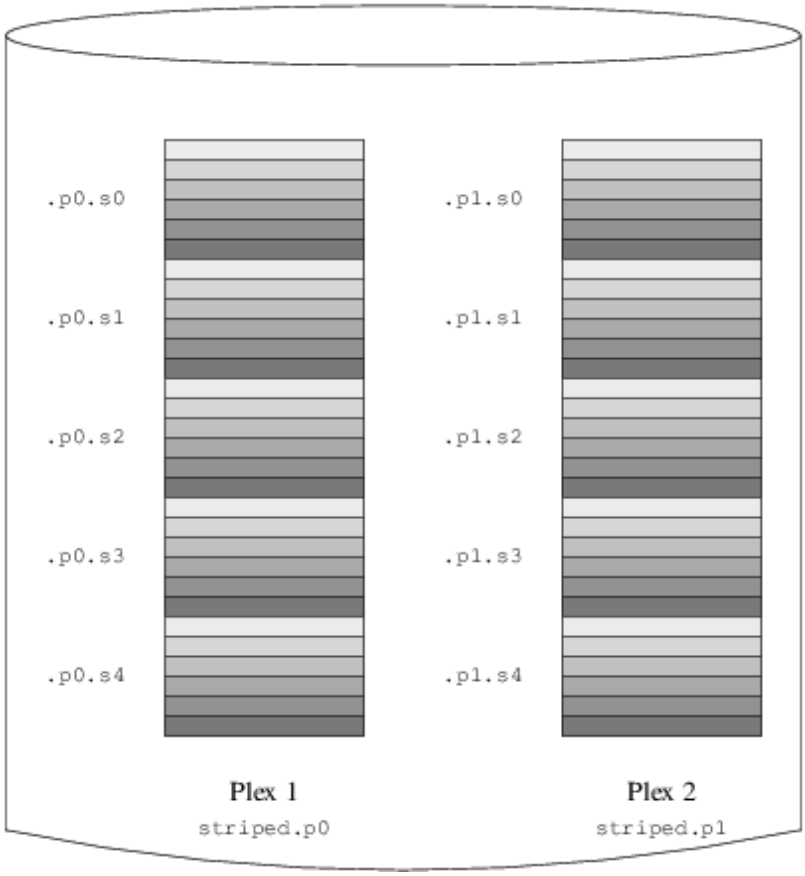


Figura 20.7. Un Volumen Vinum Replicado y en “Striping”

20.7. Esquema de Nombres de los Objetos

- The control devices `/dev/vinum/control` and `/dev/vinum/controld`, which are used by vinum(8) and the Vinum daemon respectively.

20.7.1. Creación de un Sistema de Ficheros

20.8. Configuración de Vinum

20.8.1. Arranque

20.8.1.1. Arranque Automático

20.9. Uso de Vinum en el Sistema de Ficheros Raíz

20.9.1. Inicio de Vinum en el Arranque para que Incluya el Sistema de Ficheros Raíz

20.9.2. Configuración de un Volumen Raíz en Vinum Accesible Para la Secuencia de Arranque

20.9.3. Ejemplo de Raíz basado en Vinum

20.9.4. Solución de Problemas

20.9.4.1. La Secuencia de Arranque Carga Pero el Sistema no Arranca

20.9.4.2. Sólo se Carga la Secuencia Primaria de Arranca

20.9.4.3. No Hay Arranque, la Secuencia de Arranque da un Pánico

20.9.5. Particularidades de Vinum en FreeBSD 4.x

Capítulo 21. Virtualización

Contributed by Murray Stokely.

21.1. *

Pendiente de traducción.

Capítulo 22. Localización - Uso y configuración de I18N/L10N

Contribuido por Andrey Chernov.
Reescrito por Michael C. Wu.

Traducción de Enrique Matías Sánchez <cronopios@gmail.com >.

22.1. Sinopsis

FreeBSD es un proyecto muy distribuido, que cuenta con usuarios y contribuidores por todo el mundo. Este capítulo examina las funcionalidades de internacionalización y localización de FreeBSD que permiten llevar a cabo su trabajo a los usuarios no angloparlantes. Hay muchos aspectos en la implementación de la i18n, tanto en los niveles de sistema como de aplicación, así que cuando sea conveniente dirigiremos al lector a fuentes de documentación más específicas.

Tras leer este capítulo, sabrá:

- Cómo se codifican los distintos idiomas y “locales” en los sistemas operativos modernos.
- Cómo configurar las “locales ” para su intérprete de órdenes.
- Cómo configurar la consola para idiomas distintos al inglés.
- Cómo emplear efizcamente el sistema de ventanas X en distintos idiomas.
- Dónde encontrar más información sobre como escribir aplicaciones que sigan la i18n.

Antes de leer este capítulo, debería:

- Saber como instalar aplicaciones adicionales de terceras partes (Capítulo 4, *Instalación de aplicaciones: «packages» y ports*).

22.2. Lo básico

22.2.1. ?Qué es I18N/L10N?

Los desarrolladores acortaron la palabra internacionalización al término I18N, contando el número de letras entre la primera y la última letra. L10N sigue el mismo esquema, y

procede de “localización”. Combinados, los métodos, protocolos y aplicaciones de I18N/L10N permiten a los usuarios usar el idioma de su elección.

Las aplicaciones I18N se programan usando herramientas de I18N de bibliotecas. Éstas permiten a los desarrolladores escribir un fichero sencillo y traducir los menús y textos contenidos a cada idioma. Animamos fervientemente a los programadores a que sigan esta convención.

22.2.2. ?Por qué debería usar I18N/L10N?

I18N/L10N se usa siempre que quiera ver, introducir o procesar datos en idiomas distintos al inglés.

22.2.3. ?Qué idiomas están soportados en el proyecto de I18N?

La I18N y L10N no son específicos de FreeBSD. En la actualidad, uno puede elegir entre la mayoría de los principales idiomas del mundo, incluyendo pero sin limitarse a ellos: chino, alemán, japonés, coreano, francés, ruso, vietnamita y otros.

22.3. Uso de la localización

En todo su esplendor, la I18N no es específica de FreeBSD, y es una convención. Le animamos a que ayude a FreeBSD siguiendo esta convención.

Las opciones de localización se basan en tres términos principales: código de idioma, código de país y codificación. Los nombres de las “locales” se construyen a partir de estas tres partes como sigue:

```
CódigoDelIdioma
      _CódigoDelPaís .
      Codificación
```

22.3.1. Códigos de idioma y país

Para localizar un sistema FreeBSD a un idioma concreto (o cualquier otro sistema de tipo UNIX® que soporte I18N), los usuarios necesitar averiguar los códigos del país e idioma concreto (los códigos de país le dicen a las aplicaciones que variedad del idioma dado deben usar). Además, los navegadores web, servidores SMTP/POP, servidores web, etc toman decisiones basándose en ellos. Los siguientes son ejemplos de códigos de idioma/país:

Código de idioma/país	Descripción
en_US	Inglés - Estados Unidos de América
ru_RU	Ruso de Rusia

Código de idioma/país	Descripción
zh_TW	Chino tradicional de Taiwán

22.3.2. Codificaciones

Algunos idiomas usan codificaciones distintas al ASCII que son de 8 bits, caracteres anchos o multibyte; consulte multibyte(3) para conocer más detalles. Algunas aplicaciones más antiguas no los reconocen y los confunden con caracteres de control. Las aplicaciones modernas normalmente reconocen los caracteres de 8 bits. Dependiendo de la implementación, los usuarios pueden necesitar compilar una aplicación con soporte para caracteres anchos o multibyte, o configurarlo correctamente. Para poder introducir y procesar caracteres anchos o multibyte, la Colección de "Ports" de FreeBSD proporciona diferentes programas a cada idioma. Diríjase a la documentación de I18N del " port" de FreeBSD correspondiente.

Específicamente, los usuarios necesitan mirar la documentación de la aplicación para decidir como configurarla correctamente o pasar valores correctos al configure/Makefile/compilador.

Algunas cosas a tener presentes son:

- Los juegos de caracteres de tipo char de C específicos para el idioma (consulte multibyte(3)), v.g. ISO-8859-1, ISO-8859-15, KOI8-R, CP437.
- Las codificaciones anchas o multibyte, v.g. EUC, Big5 .

Puede comprobar la lista activa de juegos de caracteres en el Registro IANA.

Nota

Las versiones 4.5 y posteriores de FreeBSD usan en su lugar codificaciones de la "locale" compatibles con X11.

22.3.3. Aplicaciones I18N

En el sistema de paquetes y ports de FreeBSD, las aplicaciones I18N se han denominado con `I18N` en su nombre para una fácil identificación. Sin embargo, no siempre soportan el idioma necesitado.

22.3.4. Configuración de las "locales"

Normalmente basta con exportar el valor del nombre de la "locale" como `LANG` en el intérprete de órdenes de la sesión. Esto se podría hacer en el fichero `~/.login_conf` del usua-

rio o en el fichero de inicio del intérprete de órdenes del usuario (`~/.profile`, `~/.bashrc` , `~/.cshrc`). No es necesario configurar las otras variables de localización como `LC_CTYPE` o `LC_CTIME` . Diríjase a la documentación de FreeBSD específica de su idioma para más información.

Debería configurar las siguientes dos variables de entorno en sus ficheros de configuración:

- `LANG` para las funciones de la familia setlocale(3) de POSIX®
- `MM_CHARSET` para el juego de caracteres MIME de las aplicaciones

Esto comprende la configuración del intérprete de órdenes del usuario, la configuración específica de la aplicación y la configuración de X11.

22.3.4.1. Métodos de configuración de las “ locales”

Hay dos métodos para configurar las “ locales”, que se describen aquí abajo. El primero (que es el recomendado) es asignar las variables de entorno en una clase de sesión, y el segundo es añadir las asignaciones de las variables de entorno al fichero de inicio del intérprete de órdenes del sistema.

22.3.4.1.1. Método de las clases de sesión

Este método permite asignar las variables de entorno necesarias para el nombre de las “locales ” y el juego de caracteres MIME de una sola vez para todos los posibles intérpretes de órdenes, en vez de añadir asignaciones específicas en los ficheros de inicio de cada uno de los intérpretes de órdenes. La configuración a nivel de usuario la puede realizar el propio usuario, mientras que la configuración a nivel de administrador precisa de permisos de superusuario.

22.3.4.1.1.1. Configuración a nivel de usuario

Esto es un ejemplo minimalista de un fichero `.login_conf` de la carpeta de inicio de un usuario, que contiene las dos variables configuradas para la codificación Latin-1:

```
me:\
 :charset=ISO-8859-1:\
 :lang=de_DE.ISO8859-1:
```

Esto es un ejemplo de un `.login_conf` que configura las variables para el chino tradicional en la codificación BIG-5. Observe que se configuran muchas más variables porque algunos programas no respetan correctamente las variables de las “locales” para el chino, el japonés y el coreano.

```
#Users who do not wish to use monetary units or time formats
```

```
#of Taiwan can manually change each variable
me:\
 :lang=zh_TW.Big5:\
 :lc_all=zh_TW.Big:\
 :lc_collate=zh_TW.Big5:\
 :lc_ctype=zh_TW.Big5:\
 :lc_messages=zh_TW.Big5:\
 :lc_monetary=zh_TW.Big5:\
 :lc_numeric=zh_TW.Big5:\
 :lc_time=zh_TW.Big5:\
 :charset=big5:\
 :xmodifiers="@im=xcin": #Setting the XIM Input Server
```

Consulte la configuración a nivel de administrador y login.conf(5) para conocer más detalles.

22.3.4.1.1.2. Configuración a nivel de administrador

Compruebe que la clase de sesión en `/etc/login.conf` establece el idioma adecuado. Asegúrese de que estas opciones aparecen en `/etc/login.conf` :

```
nombre_del_idioma :título_cuentas :\
 :charset=juego_de_caracteres_MIME :\
 :lang=nombre de la locale :\
 :tc=default:
```

Así que, si seguimos con nuestro ejemplo anterior que usaba Latin-1, tendría este aspecto:

```
german:German Users Accounts:\
 :charset=ISO-8859-1:\
 :lang=de_DE.ISO8859-1:\
 :tc=default:
```

Modificación de las clases de sesión con vipw(8)

Utilice `vipw` para añadir nuevos usuarios, y haga que la entrada tenga este aspecto:

```
usuario:contraseña:1111:11:idioma:0:0:Nombre de usuario:/home/
usuario:/bin/sh
```

Modificación de las clases de sesión con with adduser(8)

Utilice `adduser` para añadir nuevos usuarios, y haga lo siguiente:

- Establezca `defaultclass = idioma` en `/etc/adduser.conf` . Recuerde que en este caso debe introducir una clase `default` (por omisión) para todos los usuarios de otros idiomas.
- Una variante alternativa es contestar el idioma indicado cada vez que adduser(8) muestre

  ```
  Enter login class: default
  ```

```
    []:
```

- Otra alternativa es utilizar lo siguiente para cada usuario de un idioma diferente al que desee añadir:

```
# adduser -class
    idioma
```

Modificación de las clases de sesión con pw(8)

Si utiliza pw(8) para añadir nuevos usuarios, llámelo de esta manera:

```
# pw useradd
       nombre_usuario -L
       idioma
```

22.3.4.1.2. Método de los ficheros de inicio de los intérpretes de órdenes

Nota

No se recomienda este método porque precisa de una configuración diferente para cada intérprete de órdenes que se pueda elegir. Utilice en su lugar el método de las clases de sesión.

Para añadir el nombre de la “locale ” y el juego de caracteres MIME, simplemente establezca las dos variables de entorno mostradas abajo en los ficheros de inicio del intérprete de órdenes `/etc/profile` y/o `/etc/csh.login` . Aquí abajo usaremos el idioma alemán como ejemplo:

En `/etc/profile`:

```
LANG=de_DE.ISO8859-1; export LANG
MM_CHARSET=ISO-8859-1; export MM_CHARSET
```

O en `/etc/csh.login` :

```
setenv LANG de_DE.ISO8859-1
setenv MM_CHARSET ISO-8859-1
```

Como alternativa, puede añadir las instrucciones anteriores a `/usr/share/skel/dot.profile` (similar a lo que se utilizó antes en `/etc/profile`), o `/usr/share/skel/dot.login` (similar a lo que se utilizó antes en `/etc/csh.login`).

Para X11:

En `$HOME/.xinitrc` :

```
LANG=de_DE.ISO8859-1; export LANG
```

O:

```
setenv LANG de_DE.ISO8859-1
```

En función de su intérprete de órdenes (vea más arriba).

22.3.5. Configuración de la consola

Para todos los juegos de caracteres representables con el tipo char en C, establezca los tipos de letra para la consola adecuados para el idioma en cuestión en `/etc/rc.conf` con:

```
font8x16=nombre_del_tipo_de_letra

font8x14=nombre_del_tipo_de_letra
font8x8=nombre_del_tipo_de_letra
```

Aquí, el *`nombre_del_tipo_de_letra`* se toma del directorio `/usr/share/sys-cons/fonts` , sin el sufijo `.fnt` .

Asegúrese también de configurar los mapas de teclado y pantalla correctos para su juego de caracteres C por medio de `sysinstall` (`/stand/sysinstall` en versiones de FreeBSD anteriores a la 5.2). Una vez dentro de sysinstall , seleccione Configure , y después Console . Como alternativa, puede añadir lo siguiente en `/etc/rc.conf` :

```
scrnmap=nombre_del_mapa_de_pantalla
keymap=nombre_del_mapa_de_teclado
keychange="secuencia número_tecla_de_función  "
```

Aquí, el *`nombre_del_mapa_de_pantalla`* se toma del directorio `/usr/share/sys-cons/scrnmaps`, sin el sufijo `.scm` . Normalmente es necesario un mapa de pantalla (“screenmap”) con un tipo de letra correspondiente para poder extender de 8 a 9 bits la matriz de caracteres de una tarjeta VGA en un área pseudográfica, es decir, desplazar las letras fuera de ese área si el tipo de letra de pantalla usa una columna de 8 bits.

Si tiene habilitado el dæmon moused por configurar esto en su `/etc/rc.conf` :

```
moused_enable="YES"
```

entonces estudie la información sobre el cursor del ratón del siguiente párrafo.

Por omisión, el cursor del ratón del controlador syscons(4) ocupa el intervalo 0xd0-0xd3 del juego de caracteres. Si su idioma usa este intervalo, necesita desplazar el intervalo del cursor fuera de él. En versiones de FreeBSD anteriores a la 5.0, introduzca la siguiente línea en la configuración del núcleo:

```
options  SC_MOUSE_CHAR=0x03
```

En FreeBSD 4.4 y posteriores, introduzca la siguiente línea en `/etc/rc.conf` :

```
mousechar_start=3
```

Aquí, el *`nombre_del_mapa_de_teclado`* se toma del directorio `/usr/share/sys-cons/keymaps`, sin el sufijo `.kbd` . Si no está seguro de cual mapa de teclado usar, puede usar kbdmap(1) para probar los mapas de teclado sin reiniciar.

Normalmente se necesita el `keychange` para programar las teclas de función para que coincidan con el tipo de terminal seleccionado, porque las secuencias de las teclas de función no se pueden definir en el mapa de teclado.

Asegúrese también de configurar el tipo de terminal consola correcto en `/etc/ttys` para todas las entradas `ttyv*`. Las correspondencias predefinidas actualmente son:

Juego de caracteres	Tipo de terminal
ISO-8859-1 o ISO-8859-15	`cons25l1`
ISO-8859-2	`cons25l2`
ISO-8859-7	`cons25l7`
KOI8-R	`cons25r`
KOI8-U	`cons25u`
CP437 (predeterminada en VGA)	`cons25`
US-ASCII	`cons25w`

Para los idiomas en caracteres anchos o multibyte, utilice el port correcto de FreeBSD en su directorio `/usr/ports/idioma` . Algunos ports aparecen como consola mientras que el sistema los ve como una vtty serie, por lo tanto debe reservar suficientes vtty tanto para X11 como la consola pseudoserie. Aquí tiene una lista parcial de aplicaciones para usar otros idiomas en la consola:

Idioma	Ubicación
Chino tradicional (BIG-5)	chinese/big5con
Japonés	japanese/kon2-16dot o japanese/mule-freewnn
Coreano	korean/han

22.3.6. Configuración de X11

Aunque X11 no es parte del Proyecto FreeBSD, hemos incluido aquí algo de información para usuarios de FreeBSD. Para más detalles, acuda al sitio web de Xorg o del servidor X11 que utilice.

En `~/.Xresources`, puede afinar más las opciones de I18N específicas de la aplicación (v.g., tipos de letra, menús, etc).

22.3.6.1. Visualización de los tipos de letra

Instale el servidor Xorg (x11-servers/xorg-server o el servidor XFree86™ (x11-servers/XFree86-4-Server), y despúes instale los tipos de letra TrueType® del idioma. La configuración de la "locale" correcta para el idioma debería permitirle ver el idioma seleccionado en menús y similares.

22.3.6.2. Introducción de caracteres no ingleses

El protocolo Método de Introducción X11 (XIM) es un nuevo estándar para todos los clientes X11. Todas las aplicaciones X11 deberían estar escritas como clientes XIM que reciben entradas de servidores XIM. Hay varios servidores XIM disponibles para distintos idiomas.

22.3.7. Configuración de la impresora

Hay algunos juegos de caracteres de tipo char de C que están normalmente codificados por hardware en las impresoras. Los juegos de caracteres anchos o multibyte precisan de una configuración especial y recomendamos el uso de apsfilter. También puede convertir el documento a los formatos PostScript® o PDF usando conversores específicos del idioma.

22.3.8. El núcleo y los sistemas de ficheros

El sistema de ficheros rápido (FFS) de FreeBSD funciona bien a 8 bits, así que se puede usar con cualquier juego de caracteres de tipo char de C (vea multibyte(3)), pero no hay almacenado ningún nombre de juego de caracteres en el sistema de ficheros; es decir, son 8 bits en bruto y no sabe nada acerca del orden de codificación. Oficialmente, FFS no soporta todavía ninguna forma de juegos de caracteres anchos o multibyte. Sin embargo, algunos juegos de caracteres anchos o multibyte tienen parches independientes para habilitar dicho soporte en FFS. Son solamente soluciones temporales no portables o "hacks", y hemos decicido no incluirlas en el árbol de código fuente. Diríjase a los sitios web de los respectivos idiomas para encontrar más información y los parches.

El sistema de ficheros MS-DOS®de FreeBSD tiene la capacidad configurable de convertir entre los juegos de caracteres MS-DOS®, Unicode y los juegos de caracteres seleccionados del sistema de ficheros de FreeBSD. Vea mount_msdos(8) para más detalles.

22.4. Compilación de programas con soporte para I18N

Muchos ports de FreeBSD han sido portados con soporte para I18N. Algunos de ellos están marcados con -I18N en el nombre del port. Éstos y muchos otros programas tienen incorporado soporte para I18N y no necesitan ninguna consideración especial.

Sin embargo, algunas aplicaciones como MySQL necesitan tener el `Makefile` configurado con el juego de caracteres específico. Esto se hace normalmente en el `Makefile` o pasando un valor a configure en el código fuente.

22.5. Localización de FreeBSD a idiomas específicos

22.5.1. Idioma ruso (codificación KOI8-R)

Contribuido originalmente por Andrey Chernov.

Para más información sobre la codificación KOI8-R, vea las Referencias KOI8-R (Juego de caracteres rusos para la red).

22.5.1.1. Configuración de la "locale"

Ponga las siguientes líneas en su fichero `~/.login_conf` :

```
me:My account:\
  :charset=KOI8-R:\
  :lang=ru_RU.KOI8-R:
```

Vea anteriormente en este mismo capítulo ejemplos de configuración de las locales.

22.5.1.2. Configuración de la consola

- En versiones de FreeBSD anteriores a la 5.0, añada la liguiente línea en el fichero de configuración del núcleo:

  ```
  options  SC_MOUSE_CHAR=0x03
  ```

  En FreeBSD 4.4 y posteriores introduzca la siguiente línea en `/etc/rc.conf` :

  ```
  mousechar_start=3
  ```

- Utilice las siguientes opciones en `/etc/rc.conf` :

  ```
  keymap="ru.koi8-r"
  scrnmap="koi8-r2cp866"
  font8x16="cp866b-8x16"
  font8x14="cp866-8x14"
  font8x8="cp866-8x8"
  ```

- Para cada entrada `ttyv*` en `/etc/ttys` , utilice `cons25r` como el tipo de terminal.

Vea anteriormente en este mismo capítulo ejemplos de configuración de la consola.

22.5.1.3. Configuración de la impresora

Dado que la mayoría de las impresoras con caracteres rusos tienen un código de página CP866 en hardware, es necesario un filtro de salida especial para convertir de KOI8-R a CP866. Tal filtro es instalado por omisión como `/usr/libexec/lpr/ru/koi2alt`. Una entrada de impresora rusa `/etc/printcap` debería tener este aspecto:

```
lp|Russian local line printer:\
 :sh:of=/usr/libexec/lpr/ru/koi2alt:\
 :lp=/dev/lpt0:sd=/var/spool/output/lpd:lf=/var/log/lpd-errs:
```

Consulte printcap(5) para una explicación detallada.

22.5.1.4. Sistema de ficheros MS-DOS® y nombres de ficheros en ruso

La siguiente entrada fstab(5) de ejemplo habilita el soporte para nombres de fichero en ruso en los sistemas de ficheros MS-DOS® montados:

```
/dev/ad0s2      /dos/c  msdos   rw,-Wkoi2dos,-Lru_RU.KOI8-R 0 0
```

La opción `-L` selecciona el nombre de la "locale" usada, y `-W` establece la tabla de conversión de caracteres. Para usar la opción `-W`, asegúrese de montar `/usr` antes que la partición MS-DOS®, porque las tablas de conversión se ubican en `/usr/libdata/msdosfs`. Para más información, vea la página de manual mount_msdos(8).

22.5.1.5. Configuración de X11

1. Antes haga la configuración de las "locales" para la consola como se ha explicado.

> **Nota**
>
> La "locale " rusa KOI8-R puede no funcionar con versiones antiguas (anteriores a la 3.3) de XFree86™. Xorg es ahora la versión predefinida del sistema de ventanas X en FreeBSD. Esto no debería ser un problema salvo que esté usando una versión antigua de FreeBSD.

2. Si utiliza Xorg, instale el paquete x11-fonts/xorg-fonts-cyrillic.

   Compruebe la sección `"Files"` de su fichero `/etc/X11/xorg.conf`. Se deben añadir las siguientes líneas *antes* de ninguna otra entrada `FontPath`:

```
FontPath   "/usr/X11R6/lib/X11/fonts/cyrillic/misc"
```

```
FontPath   "/usr/X11R6/lib/X11/fonts/cyrillic/75dpi"
FontPath   "/usr/X11R6/lib/X11/fonts/cyrillic/100dpi"
```

Si utiliza un modo de vídeo de alta resolución, intercambie las líneas 75 dpi y 100 dpi.

3. Para activar un teclado ruso, añada lo siguiente a la sección `"Keyboard"` de su fichero `XF86Config` .

   Para XFree86™ 3.X:

   ```
   XkbLayout  "ru"
   XkbOptions "grp:caps_toggle"
   ```

   Para Xorg (o XFree86™ 4.X):

   ```
   Option "XkbLayout"   "us,ru"
   Option "XkbOptions"  "grp:toggle"
   ```

   Asegúrese también de que `XkbDisable` esté desactivado (comentado) allí.

   Para `grp:caps_toggle` la tecla para pasar entre la codificación rusa y la latina será BloqMayús . La antigua función de BloqMayús todavía está disponible a través de May+BloqMayús (sólamente en modo latino). Para `grp:toggle` la tecla para pasar entre la codificación rusa y la latina será Alt derecha. `grp:caps_toggle` no funciona en Xorg por alguna razón desconocida.

   Si en su teclado dispone de teclas “Windows® ”, y observa que algunas teclas no alfanuméricas están mapeadas incorrectamente en el modo ruso, añada la siguiente línea en su fichero `XF86Config` .

   Para XFree86™ 3.X:

   ```
   XkbVariant "winkeys"
   ```

   Para Xorg (o XFree86™ 4.X):

   ```
   Option "XkbVariant" ",winkeys"
   ```

Nota

El teclado XKB ruso puede no funcionar con versiones antiguas de XFree86™ , vea la nota anterior para más información. El teclado XKB ruso puede no funcionar tampoco con aplicaciones no localizadas.

Nota

Las aplicaciaones mínimamente localizadas deberían llamar una función `XtSetLanguageProc (NULL, NULL, NULL);` al principio del programa.

Vea KOI8-R para X Window para más instrucciones sobre la localización de aplicaciones X11.

22.5.2. Localización al chino tradicional para Taiwán

El proyecto FreeBSD-Taiwán tiene una Guía de chino para FreeBSD en `http://netlab.cse.yzu.edu.tw/~statue/freebsd/zh-tut/` que utiliza muchos ports chinos. El editor actual del `FreeBSD Chinese HOWTO` es Shen Chuan-Hsing <`statue@freebsd.sinica.edu.tw`>.

Chuan-Hsing Shen <`statue@freebsd.sinica.edu.tw`> ha creado la Colección de FreeBSD en chino (CFC) usando `zh-L10N-tut` de FreeBSD-Taiwán. Los paquetes y los guiones están disponibles en `ftp://freebsd.csie.nctu.edu.tw/pub/taiwan/CFC/` .

22.5.3. Localización al idioma alemán (para todos los idiomas ISO 8859-1)

Slaven Rezic <`eserte@cs.tu-berlin.de`> ha escrito un tutorial de como usar las diéresis en una máquina FreeBSD. El tutorial está disponible en `http://www.de.FreeBSD.org/de/umlaute/`.

22.5.4. Localización a los idiomas japonés y coreano

Para el japonés, diríjase a `http://www.jp.FreeBSD.org/` , y para el coreano, acuda a `http://www.kr.FreeBSD.org/` .

22.5.5. Documentación sobre FreeBSD en otros idiomas aparte del inglés

Algunos contribuidores de FreeBSD han traducido partes de FreeBSD a otros idiomas. Están disponibles a través de enlaces en el sitio principal o en `/usr/share/doc` .

Capítulo 23. Lo último de lo último

Restructured, reorganized, and parts updated by Jim Mock.
Original work by Jordan Hubbard, Poul-Henning Kamp, John Polstra y Nik Clayton.

23.1. Sinopsis

Pendiente de Traducción

23.2. FreeBSD-CURRENT vs. FreeBSD-STABLE

23.2.1. Current

Pendiente de Traducción

23.2.2. Stable

Pendiente de Traducción

23.3. Sincronización de su código fuente

Pendiente de Traducción

23.4. Uso de `make world`

Pendiente de Traducción

23.5. Redes pequeñas

Pendiente de Traducción

Parte IV. Comunicaciones en red

FreeBSD es uno de los sistemas operativos más utilizados como servidores en red de alto rendimiento. Los siguientes capítulos cubren:

- Comunicaciones serie
- PPP y PPP sobre Ethernet
- Correo electrónico
- Servidores de red
- Otros aspectos avanzados de red

Estos capítulos están diseñados para leerse según van haciendo falta. No tienen por qué leerse en un determinado orden, ni tiene por qué leerlos todos antes de poder usar FreeBSD en un entorno de red.

Tabla de contenidos

24. Comunicaciones serie 533
    24.1. Sinopsis 533
    24.2. Introducción 533
    24.3. Terminales 539
    24.4. Servicio dial-in 545
    24.5. Servicio dial-out 553
    24.6. Configurando la consola serie 557
25. PPP y SLIP 569
    25.1. Sinopsis 569
    25.2. Uso de User PPP 570
    25.3. Uso de Kernel PPP 586
    25.4. Uso de PPP sobre Ethernet (PPPoE) 594
    25.5. Uso de PPP sobre ATM (PPPoA) 596
    25.6. Uso de SLIP 599
26. Cortafuegos 613
    26.1. * 613
27. Correo Electrónico 615
    27.1. Sinopsis 615
    27.2. Utilización del correo electrónico 616
    27.3. Configuración de sendmail 619
    27.4. Sustitución del Agente de Transferencia de Correo 622
    27.5. Depuración de Problemas 626
    27.6. Conceptos Avanzados 630
    27.7. SMTP con UUCP 633
    27.8. Configuración para sólamente enviar correo 636
    27.9. Utilización del correo con una conexión mediante módem analógico (dial-up) 637
    27.10. Autentificación utilizando SMTP 638
    27.11. Agente de Correo de Usuario 640
    27.12. Manejo de fetchmail 648
    27.13. Uso de procmail 649
28. Servidores de red 651
    28.1. * 651
29. Networking avanzado 653
    29.1. Resumen 653
    29.2. Pasarelas y “routers” 654
    29.3. Redes sin cables (“wireless”) 661
    29.4. Bluetooth 668
    29.5. Puenteado 678
    29.6. NFS 681
    29.7. Ejecución sin disco duro 689
    29.8. RDSI 700
    29.9. NIS/YP 705
    29.10. DHCP 727

29.11. DNS 733
29.12. NTP 748
29.13. Traducción de direcciones de red 751
29.14. El “Superservidor” inetd 756
29.15. Línea IP paralela (PLIP) 761
29.16. IPv6 763
29.17. ATM en FreeBSD 5.X 769

Capítulo 24. Comunicaciones serie

24.1. Sinopsis

UNIX® siempre ha tenido soporte para comunicación serie. De hecho, las primeras máquinas UNIX® dependían de líneas serie para tener interacción con el usuario. Las cosas han cambiado mucho desde esos días cuando la "terminal" promedio consistía de una terminal serie de 10-caracteres-por-segundo, impresora y teclado. Este capítulo cubrirá algunas de las maneras en las que FreeBSD utiliza comunicaciones serie.

Despues de leer este capítulo, usted entenderá:

- Como conectar terminales a su sistema FreeBSD.
- Como utilizar un modem para marcar a equipos remotos.
- Como permitir a usuarios remotos entrar a su sistema con un modem.
- Como arrancar su sistema desde una consola serie.

Antes de leer este capítulo usted debería:

- Saber como configurar e instalar un nuevo kernel (Capítulo 8, *Configuración del kernel de FreeBSD*).
- Entender procesos y permisos UNIX® (Capítulo 3, *Conceptos básicos de Unix*).
- Tener acceso al manual técnico para el hardware serie (modem o tarjeta multipuerto) que le gustaría utilizar con FreeBSD.

24.2. Introducción

24.2.1. Terminología

bps

Bits por segundo — la tasa a la cual los datos son transmitidos

DTE

Data Terminal Equipment (Equipo terminal de datos) — por ejemplo, su computadora

DCE

Data Communications Equipment (Equipo de comunicación de datos) — su modem

RS-232

Estándar EIA para hardware de comunicación serie

Cuando se habla de tasa de comunicación de datos, ésta sección no usa el término "baud". Baud se refiere al número de estados de transición eléctricos que pueden realizarse en un periodo de tiempo, mientras que "bps" (bits por segundo) es el término *correcto* a usar (al menos parece no molestar demasiado a los más aferrados).

24.2.2. Cables y puertos

Para conectar un modem o terminal a su sistema FreeBSD necesitará un puerto serie en su computadora y el cable apropiado para conectar a su dispositivo serie. Si ya está familiarizado con su hardware y el cable que requiere, puede saltarse esta sección.

24.2.2.1. Cables

Existen diferentes tipos de cables serie. Los dos tipos más comunes para nuestros propósitos son cables null-modem y cables RS-232 estándar ("normal"). La documentación de su hardware debería describir el tipo de cable requerido.

24.2.2.1.1. Cables null-modem

Un cable null-modem pasa algunas señales, como "tierra", normalmente, pero cambia otras señales. Por ejemplo, el pin "envío de datos" en un extremo va al pin "recepción de datos" en el otro.

Si le gusta fabricar sus propios cables, puede construir un cable null-modem para utilizar con terminales. Esta tabla muestra los nombres de señal y números de pin en un conector DB-25.

Señal	Pin #		Pin #	Señal
SG	7	conecta a	7	SG
TD	2	conecta a	3	RD
RD	3	conecta to	2	TD
RTS	4	conecta a	5	CTS
CTS	5	conecta a	4	RTS
DTR	20	conecta a	6	DSR

Señal	Pin #		Pin #	Señal
DCD	8		6	DSR
DSR	6	conecta a	20	DTR

Nota

Conectar “Data Set Ready” (DSR) y “Data Carrier Detect” (DCD) internamente en la capucha del conector, y entonces a “Data Terminal Ready” (DTR) en la capucha remota.

24.2.2.1.2. Cables estándard RS-232C

Un cable serie estándar pasa todas las señales RS-232C normalmente. Esto es, el pin “envío de datos” en un extremo va al pin “envío de datos” en el otro extremo. Este es el tipo de cable a utilizar para conectar un modem a su sistema FreeBSD, y también es apropiado para algunas terminales.

24.2.2.2. Puertos

Puertos serie son los dispositivos a través de los cuales los datos son transferidos entre una computadora FreeBSD y la terminal. Esta sección describe los tipos de puertos que existen y como son referidos en FreeBSD.

24.2.2.2.1. Tipos de puertos

Existen varios tipos de puertos serie. antes de comprar o construir un cable, necesita asegurarse que coincida con los puertos en su terminal y en su sistema FreeBSD.

La mayoría de las terminales tienen puertos DB25. Computadoras personales, incluyendo PCs corriendo FreeBSD, tienen puertos DB25 o DB9. Si tiene una tarjeta multipuertos serie para su PC, tal vez tenga puertos RJ-12 o RJ-45.

Vea la documentación que acompaña su hardware para las especificaciones sobre el tipo de puerto en uso. Una inspección visual del puerto también funciona en la mayoría de los casos.

24.2.2.2.2. Nombres de puerto

En FreeBSD, se accesa cada puerto serie a través de una entrada en el directorio `/dev`. Existen dos tipos de entradas:

- Puertos de llamada-entrante son llamados `/dev/ttydN` donde *N* es el número de puerto, iniciando desde cero. Generalmente, los puertos de llamada-entrante se utilizan para

terminales. Los puertos de llamada-entrante requieren que la línea serie especifique la señal data carrier detect (DCD) para funcionar correctamente.

- Puertos de llamada-saliente son llamados `/dev/cuaaN` . Usualmente no se utilizan los puertos de llamada-saliente para terminales, solo para modems. Puede utilizar el puerto de llamada-saliente si el cable serie o la terminal no soporta la señal de carrier detect.

Si tiene conectada una terminal al primer puerto serie (`COM1` en MS-DOS®), entonces usará `/dev/ttyd0` para referirse a la terminal. Si la terminal está en el segundo puerto serie (también conocido como `COM2`), utilice `/dev/ttyd1` , y así sucesivamente.

24.2.3. Configuración del kernel

FreeBSD soporta cuatro puertos serie por omisión. en el mundo MS-DOS® éstos son conocidos como `COM1`, `COM2`, `COM3`, y `COM4`. FreeBSD actualmente soporta tarjetas de interfaz serie "tontas", como la BocaBoard 1008 y 2016, así como tarjetas multipuerto más inteligentes como las fabricadas por Digiboard y Stallion Technologies. De cualquier manera, el kernel por omisión solo busca por los puertos COM estándares.

Para ver si su kernel reconoce cualquiera de sus puertos serie, mire los mensajes mientras el kernel esta arrancando, o utilice el comando `/sbin/dmesg` para repetir los mensajes de arranque del kernel. En particular busque por mensajes que inicien con el caracter `sio`.

Sugerencia

Para ver solamente los mensajes que contienen la palabra `sio`, use el comando:

```
# /sbin/dmesg | grep 'sio'
```

Por ejemplo, en un sistema con cuatro puertos serie, éstos son los mensajes de arranque del kernel específicos de puerto serie:

```
sio0 at 0x3f8-0x3ff irq 4 on isa
sio0: type 16550A
sio1 at 0x2f8-0x2ff irq 3 on isa
sio1: type 16550A
sio2 at 0x3e8-0x3ef irq 5 on isa
sio2: type 16550A
sio3 at 0x2e8-0x2ef irq 9 on isa
sio3: type 16550A
```

Si su kernel no reconoce todos sus puertos serie, probablemente necesitará configurar un kernel FreeBSD personalizado para su sistema. Para información detallada sobre configurar su kernel, por favor vea Capítulo 8, *Configuración del kernel de FreeBSD*.

Las líneas de dispositivo relevantes para su fichero de configuración del kernel podrín verse de esta manera, para FreeBSD 4.X:

```
device  sio0 at isa? port IO_COM1 irq 4
device  sio1 at isa? port IO_COM2 irq 3
device  sio2 at isa? port IO_COM3 irq 5
device  sio3 at isa? port IO_COM4 irq 9
```

y de esta manera, para FreeBSD 5.X:

```
device  sio
```

Puede comentar o remover completamente líneas de dispositivos que no tenga en el caso de FreeBSD 4.X; para FreeBSD 5.X tiene que editar su fichero `/boot/device.hints` para configurar sus puertos serie. Por favor diríjase a la página de manual sio(4) para mayor información sobre puertos serie y configuración de tarjetas multipuertos. Tenga cuidado si está utilizando un fichero de configuración que fué usado anteriomente para una versión diferente de FreeBSD porque las banderas de dispositivo y la sintaxis han cambiado entre versiones.

Nota

`port IO_COM1` es una substitución para `port 0x3f8`, `IO_COM2` es `0x2f8`, `IO_COM3` es `0x3e8`, y `IO_COM4` es `0x2e8`, las cuales son direcciones de puerto comunes para sus respectivos puertos serie; interrupciones 4,3,5 y 9 son peticiones comunes de líneas de interrupción. Note también que puertos serie regulares *no pueden* compartir interrupciones en PCs con bus ISA (las tarjetas multipuerto tienen electrónicos que les permiten a los 16550A's compartir una o dos líneas de peticiones de interrupción).

24.2.4. Archivos especiales de dispositivo

La mayoría de dispositivos en el kernel son accesados a través de “ficheros especiales de dispositivo”, los cuales están localizados en el directorio `/dev`. Los dispositivos `sio` son accesados a través de los dispositivos `/dev/ttydN` (dial-in) y `/dev/cuaaN` (call-out). FreeBSD provee también dispositivos de inicialización (`/dev/ttyidN` y `/dev/cuaiaN`) y dispositivos de bloqueo (`/dev/ttyldN` y `/dev/cualaN`). Los dispotivos de inicialización son utilizados para inicializar los parámetros de comunicación de puerto cada vez que un puerto es abierto, como `crtscts` para modems que utilizan señalización `RTS/CTS` para control de flujo. Los dispositivos de bloqueo son utilizados para bloquear banderas en puertos y prevenir que usuarios o programas cambien ciertos parámetros; vea las páginas de manual termios(4), sio(4), y stty(1) para información de las propiedades de terminales, bloqueo e inicialización de dispositivos y aplicación de opciones de terminal, respectivamente.

24.2.4.1. Creando ficheros de dispositivo especiales

Nota

FreeBSD 5.0 incluye el sistema de ficheros devfs(5) el cual crea automáticamente nodos de dispositivos según se necesiten. si está corriendo una versión de FreeBSD con `devfs` habilitado entonces puede saltarse esta sección.

Un script de shell llamado `MAKEDEV` en el directorio `/dev` administra los ficheros especiales de dispositivo. Para utilizar `MAKEDEV` para crear un fichero especial de dispositivo dial-up para `COM1` (port 0), `cd` a `/dev` y ejecute el comando `MAKEDEV ttyd0`. De la misma manera, para crear ficheros especiales de dispositivo para `COM2` (port 1), utilice `MAKEDEV ttyd1`.

`MAKEDEV` no crea solamente el fichero especial de dispositivo `/dev/ttydN` también crea los nodos `/dev/cuaaN` , `/dev/cuaiaN` , `/dev/cualaN`, `/dev/ttyldN` , y `/dev/ttyidN`.

Despues de crear ficheros especiales de dispositivo nuevos, asegúrese de revisar los permisos en los ficheros (especialmente los ficheros `/dev/cua*`) para asegurarse que solamente los usuarios que deben tener acceso a esos ficheros especiales de dispositivo puedan leer y escribir en ellos — probablemente no desee permitir al usuario promedio utilizar sus modems para marcar al exterior. Los permisos por omisión en los ficheros `/dev/cua*` deberían ser suficientes:

```
crw-rw----    1 uucp     dialer    28, 129 Feb 15 14:38 /dev/cuaa1
crw-rw----    1 uucp     dialer    28, 161 Feb 15 14:38 /dev/cuaia1
crw-rw----    1 uucp     dialer    28, 193 Feb 15 14:38 /dev/cuala1
```

Estos permisos permiten al usuario `uucp` y usuarios en el grupo `dialer` utilizar dispositivos call-out.

24.2.5. Configuración de puerto serie

El dispositivo `ttydN` (o `cuaaN`) es el dispositivo regular que usted deseará abrir para sus aplicaciones. Cuando un proceso abre el dispositivo, tendrá un conjunto por omisión de propiedades de terminal E/S. Puede ver estas propiedades con el comando

```
# stty -a -f /dev/ttyd1
```

Cuando cambia las propiedades de este dispositivo, las propiedades son efectivas hasta que el dispositivo es cerrado. Cuando es reabierto regresa a las propiedades por omisión. Para realizar cambios al conjunto por omisión, usted puede abrir y ajustar las propiedades del dispositivo de “estado inicial” . Por ejemplo, para activar el modo `CLOCAL`, comunicación de 8 bits y control de flujo `XON/XOFF` por omisión para `ttyd5`, teclée:

```
# stty -f /dev/ttyid5 clocal cs8 ixon ixoff
```

La inicialización para todo el sistema de los dispositivos serie es controlada en `/etc/rc.serial`. Este fichero afecta las propiedades por omisión de dispositivos serie.

Para prevenir que ciertas propiedades sean cambiadas por una aplicación, haga ajustes al dispositivo "bloquear estado". Por ejemplo, para confinar la velocidad en `ttyd5` a 57600 bps, teclée:

```
# stty -f /dev/ttyld5 57600
```

Ahora, una aplicación que abra `ttyd5` y trate de cambiar la velocidad del puerto se mantendrá con 57600 bps.

Naturalmente, debería crear los dispositivos de estado inicial y bloqueo de estado escribible únicamente para la cuenta `root`.

24.3. Terminales

Contribuido porSean Kelly.

Las terminales proveen una manera conveniente y de bajo coste de accesar su sistema FreeBSD cuando no se encuentra en la consola de la computadora o en una red conectada. Esta sección describe como utilizar terminales con FreeBSD.

24.3.1. Usos y tipos de terminales

Los sistemas originales UNIX® no tenín consolas. En su lugar la gente se firmaba y corría programas a través de terminales conectadas a los puertos serie de la computadora. Es bastante similar a usar un modem y un programa de terminal para marcar hacia un sistema remoto para hacer trabajo en modo texto.

Las PCs actuales tienen consolas con gráficos de alta calidad, pero la habilidad para establecer una sesión en un puerto serie todavía existe en casi cualquier sistema operativo UNIX® al día de hoy; FreeBSD no es la excepción. Utilizando una terminal conectada a un puerto serie libre, usted puede accesar y correr cualquier programa de texto que podría correr normalmente en la consola o en una ventana `xterm` en el sistema X Window.

Para el usuario corporativo, se pueden conectar muchas terminales a un sistema FreeBSD y ponerlas en los escritorios de sus empleados. Para un usuario casero, una computadora de reserva, como una IBM PC más antigua o una Macintosh®, puede ser una terminal cableada a una computadora más poderosa corriendo FreeBSD. Puede convertir lo que de otra manera sería una computadora de un solo usuario en un poderoso sistema de usuarios múltiples.

Para FreeBSD, existen tres clases de terminales:

- Terminales tontas
- PCs actuando como terminales
- Terminales X

Las subsecciones siguientes describen cada tipo.

24.3.1.1. Terminales tontas

Terminales tontas son piezas de hardware especializadas que le permiten conectar a computadoras a través de líneas serie. Son llamadas “tontas” porque solo tienen poder computacional suficiente para desplegar, enviar y recibir texto. No puede ejecutar ningún programa en ellas. Es la computadora a la cual se conectan la que tiene todo el poder para correr editores de texto, compiladores, correo electrónico, juegos, y demás.

Existen cientos de tipos de terminales tontas hechas por muchos fabricantes, incluyendo VT-100 de Digital Equipment Corporation y WY-75 de Wyse. Cualquier tipo funcionará con FreeBSD. Algunas terminales superiores pueden incluso desplegar gráficos, pero solo ciertos paquetes de software pueden tomar ventaja de estas funciones avanzadas.

Las terminales tontas son populares en ambientes de trabajo donde los trabajadores no necesitan acceso a aplicaciones gráficas como las que provee el sistema X Window.

24.3.1.2. PCs actuando como terminales

Si una terminal tonta tiene apenas la habilidad para desplegar, enviar y recibir texto, entonces ciertamente cualquier computadora personal de reserva puede ser una terminal tonta. Todo lo que necesita es el cable apropiado y algún software de *emulación de terminal* para correr en la computadora.

Tal configuración es popular en hogares. Por ejemplo, si su consorte se encuentra ocupado trabajando en la consola de su sistema FreeBSD, usted puede realizar algún trabajo en modo texto al mismo tiempo desde una computadora personal menos poderosa conectada como una terminal al sistema FreeBSD.

24.3.1.3. Terminales X

Las terminales X son el tipo más sofisticado de terminal disponible. En lugar de conectar a un puerto serie, usualmente se conectan a una red como Ethernet. En lugar de ser relegadas a aplicaciones de modo texto pueden desplegar aplicaciones X.

Hemos introducido terminales X solo por complementar. Sin embargo, este capítulo *no* cubre instalación, configuración o uso de terminales X.

24.3.2. Configuración

Esta sección describe lo que necesita para configurar en su sistema FreeBSD y permitirle habilitar sesiones de entrada en una terminal. Asume que ya tiene configurado su kernel para soportar el puerto serie al cual la terminal está conectada—y que la tiene conectada.

Recuerde del Capítulo 12, *El proceso de arranque en FreeBSD* que el proceso `init` es responsable del control e inicialización de todos los procesos al inicio del sistema. Una de las tareas ejecutadas por `init` es leer el fichero `/etc/ttys` e iniciar un proceso `getty` en las terminales disponibles. El proceso `getty` es responsable de leer un nombre de entrada e iniciar el programa `login`.

Así, para configurar terminales para su sistema FreeBSD los siguientes pasos deben hacerse como `root`:

1. Agregue una línea a `/etc/ttys` para la entrada en el directorio `/dev` para el puerto serie si todavía no se encuentra ahí.

2. Especifique que `/usr/libexec/getty` sea ejecutado en el puerto, y especifique el tipo apropiado de *`getty`* desde el fichero `/etc/gettytab`.

3. Especifique el tipo de terminal por omisión.

4. Ponga el puerto a "on."

5. Especifique si el puerto debe ser o no "seguro."

6. Obligue a `init` a releer el fichero `/etc/ttys`.

Como un paso opcional, tal vez desee crear un tipo *`getty`* personalizado para utilizar en el paso 2 mediante una entrada en `/etc/gettytab`. Este capítulo no explica como realizarlo; por lo que se le exhorta a leer las páginas de manual gettytab(5) y getty(8) para mayor información.

24.3.2.1. Agregando una entrada a `/etc/ttys`

El fichero `/etc/ttys` lista todos los puertos en su sistema FreeBSD donde quiere permitir logins. Por ejemplo, la primera consola virtual `ttyv0` tiene una entrada en este fichero. Puede firmarse en la consola utilizando esta entrada. Este fichero también contiene entradas para las otras consolas virtuales, puertos serie y pseudo-ttys. Para una terminal conectada por cable, solo liste la entrada `/dev` del puerto serie sin la parte de `/dev` (por ejemplo, `/dev/ttyv0` debería estar listado como `ttyv0`).

Una instalación por omisión de FreeBSD incluye un fichero `/etc/ttys` con soporte para los primeros cuatro puertos serie: `ttyd0` hasta `ttyd3`. Si está conectando una terminal a uno de esos puertos, no necesita agregar otra entrada.

Ejemplo 24.1. Agregando entradas de terminal a `/etc/ttys`

Suponga que quisiéramos conectar dos terminales al sistema: una Wyse-50 y una vieja IBM PC 286 corriendo el software de terminal Procomm emulando una terminal VT-100. Conectamos la Wyse al segundo puerto serie y la 286 al sexto puerto serie (un puerto en una tarjeta multipuerto serie). Las entradas correspondiente en el fichero `/etc/ttys` se verían como esto:

```
ttyd1❶  "/usr/libexec/getty std.38400"❷  wy50❸  on❹  ↺
insecure❺
ttyd5   "/usr/libexec/getty std.19200"  vt100  on  insecure
```

❶ El primer campo normalmente especifica el nombre de fichero especial de la terminal como es hallado en `/dev`.

❷ El segundo campo es el comando a ejecutar por esta línea, el cual es usualmente getty(8). `getty` inicializa y abre la línea, establece la velocidad, pregunta por un nombre de usuario y entonces ejecuta el programa login(1) .

El programa `getty` acepta un parámetro (opcional) en su línea de comando, el tipo *`getty`*. Un tipo *`getty`* configura características en la línea de terminal, como tasa de bps y paridad. El programa `getty` lee estas características desde el fichero `/etc/gettytab` .

El fichero `/etc/gettytab` contiene muchas entradas para líneas de terminal viejas y nuevas. En la mayoría de los casos, las entradas que empiezan con el texto `std` funcionarán para terminales conectadas físicamente. Estas entradas ignoran la paridad. Existe una entrada `std` por cada tasa de bps de 110 a 115200. Por supuesto puede agregar sus propias entradas a este fichero. La página de manual gettytab(5) provee mayor información.

Al establecer el tipo *`getty`* en el fichero `/etc/ttys` , asegúrese que las propiedades de comunicaciones en la terminal concuerden.

Para nuestro ejemplo, la Wyse-50 no usa paridad y conecta a 38400 bps. La 286 PC no usa paridad y conecta a 19200 bps.

❸ El tercer campo es el tipo de terminal usualmente conectado a esa línea tty. Para puertos dial-up, `unknown` o `dialup` son usados típicamente en este campo puesto que los usuarios pueden marcar prácticamente con cualquier tipo de terminal o de software. Para terminales conectadas físicamente, el tipo de terminal no cambia, así que puede poner un tipo de terminal real del fichero de base de datos termcap(5) en este campo.

Para nuestro ejemplo, la Wyse-50 utiliza el tipo de terminal real mientras que la PC 286 corriendo Procomm será puesta a emular una VT-100.

❹ El cuarto campo especifica si el puerto debe habilitarse. Poniendo `on` aquí provocará que el proceso `init` inicie el programa en el segundo campo, `getty`. Si pone `off` en este campo, no habrá `getty`, y por consecuencia ningun login en el puerto.

❺ El último campo es utilizado para especificar si el puerto es seguro. Marcar un puerto como seguro significa que se confía en él lo suficiente para permitir que la cuenta `root` (o cualquier cuenta con un ID de usuario 0) se firme desde ese puerto. Los puertos inseguros no permiten entradas de `root`. En un puerto inseguro, los usuarios deben firmarse desde cuentas sin privilegios y entonces utilizar su(1) o un mecanismo similar para acceder a privilegios de superusuario.

Es altamente recomendable que utilize "insecure" incluso para terminales que se encuentran detrás de puertas con llave. Es muy sencillo entrar y usar `su` si necesita privilegios de superusuario.

24.3.2.2. Forzar `init` a que relea `/etc/ttys`

Despues de realizar los cambios necesarios al fichero `/etc/ttys` debería mandar una señal SIGHUP (hangup) al proceso `init` para forzarlo a releer su fichero de configuración. Por ejemplo:

```
# kill -HUP 1
```

Nota

`init` siempre es el primer proceso que corre en un sistema, por lo tanto siempre tendrá el PID 1.

Si todo está puesto correctamente, todos los cables en su lugar, y las terminales están encendidas, entonces un proceso `getty` debe estar corriendo en cada terminal y debería ver prompts de entrada en sus terminales en este punto.

24.3.3. Determinando errores en su conexión

Incluso con la más meticulosa etención al detalle, algo puede salir mal mientras se configura una terminal. Esta es una lista de síntomas y algunos arreglos sugeridos.

24.3.3.1. No aparece prompt de login

Asegúrese que la terminal está conectada y encendida, asegúrese que se encuentra ejecutando un software de emulación de terminal en el puerto serie correcto.

Asegúrese que el cable está conectado firmemente tanto a la terminal como a la computadora FreeBSD. Asegúrese que es el tipo correcto de cable.

Asegúrese que la terminal y FreeBSD concuerdan en la tasa de bps y propiedades de paridad. Si tiene una terminal de desplegado de video, asegúrese que los controles de contraste y brillo estén encendidos. Si es una terminal de impresión, asegúrese que papel y tinta se encuentren en forma.

Asegúrese que un proceso `getty` esté corriendo y sirviendo la terminal. Por ejemplo, para obtener una lista de procesos `getty` con `ps`, teclée:

```
# ps -axww|grep getty
```

Debería ver una entrada para la terminal. Por ejemplo, el siguiente desplegado muestra que un `getty` está corriendo en el segundo puerto serie `ttyd1` y está utilizando la entrada `std.38400` en `/etc/gettytab`:

```
22189  d1  Is+    0:00.03 /usr/libexec/getty std.38400 ttyd1
```

Si no hay un proceso `getty` corriendo, asegúrese que tiene habilitado el puerto en `/etc/ttys`. Recuerde también ejecutar `kill -HUP 1` despues de modificar el fichero `ttys`.

Si el proceso `getty` está corriendo pero la terminal todavía no despliega un prompt de login, o si despliega un prompt pero no le permite escribir, su terminal o cable tal vez no soporte inicialización por hardware. Trate cambiar la entrada en `/etc/ttys` de `std.38400` a `3wire.38400` recuerde correr `kill -HUP 1` despues de modificar `/etc/ttys`. La entrada `3wire` es similar a `std`, pero ignora la inicialización por harware. Tal vez necesite reducir la tasa de baudios o habilitar control de flujo por software cuando utilice `3wire` para prevenir desbordamientos de buffer.

24.3.3.2. Si aparece basura en lugar de un prompt de login

Asegúrese que la terminal y FreeBSD concuerdan en la tasa de bps y propiedades de paridad. Revise los procesos `getty` para asegurarse que el tipo correcto de *`getty`* está en uso. Si no es así, edite `/etc/ttys` y ejecute `kill -HUP 1`.

24.3.3.3. Los caracteres aparecen doble; las contraseñas aparecen cuando se escriben

Cambie la terminal (o el software de emulación de terminal) de “half duplex” o “local echo” a “full duplex.”

24.4. Servicio dial-in

Contribuido por Guy Helmer.
Adiciones por Sean Kelly.

Configurar su sistema FreeBSD para servicio dial-in es muy similar a conectar terminales excepto que en lugar de lidiar con terminales se hace con modems.

24.4.1. Modems externos vs. internos

Los modems externos parecen ser más convenientes para dial-up, debido a que los modems externos con frecuencia pueden ser configurados semi permanentemente vía parámetros almacenados en RAM no volatil y usualmente proveen indicadores luminosos que despliegan el estado de señales importantes RS-232. Luces parpadeantes impresionan a los visitantes, pero las luces son también útiles para ver si un modem se encuentra operando adecuádamente.

Los modems internos usualmente carecen de RAM no volatil, entonces su configuración puede estar limitada a especificar DIP switches. Si su modem interno cuenta con algún indicador luminoso de señales, es probablemente dificil observar las luces cuando el sistema está cubierto y en su lugar.

24.4.1.1. Modems y cables

Si se encuentra utilizando un modem externo, entonces necesitará por supuesto un cable adecuado. Un cable serie estándar RS-232C debe ser suficiente mientras todas las señales normales sean cableadas:

- Transmitted Data (TD)
- Received Data (RD)
- Request to Send (RTS)
- Clear to Send (CTS)
- Data Set Ready (DSR)
- Data Terminal Ready (DTR)
- Carrier Detect (CD)
- Signal Ground (SG)

FreeBSD necesita las señales RTS y CTS para control de flujo a velocidades mayores a 2400 bps, la señal CD para detectar cuando una llamada ha sido respondida o la línea ha sido colgada, y la señal DTR para reiniciar el modem despues de completar una sesión. Algunos cables son hechos sin incluir todas las señales necesarias, así que si tiene proble-

mas, como cuando una sesión no finaliza cuando la línea es colgada, tal vez el problema se deba al cable.

Como otros sistemas operativos tipo UNIX®, FreeBSD utiliza las señales de hardware para saber cuando una llamada ha sido contestada o una línea ha sido colgada y poder colgar y reiniciar el modem despues de una llamada. FreeBSD evita enviar comandos al modem o esperar por reportes de estado del modem. Si está familiarizado con la conexión de modems una PC funcionando como BBS, tal ves esto parezca extraño.

24.4.2. Consideraciones de interfaces serie

FreeBSD soporta interfaces de comunicación NS8250-, NS16450-, NS16550-, y NS16550A-basado en EIA RS-232C (CCITT V.24). Los dispositivos 8250 y 16450 tienen buffers de un solo caracter. El dispositivo 16550 brinda un buffer de 16 caracteres, el cual permite un mejor desempeño del sistema. (Errores en 16550 simple impiden el uso del buffer de 16 caracteres, así que utilice 16550A si es posible). Debido a que los dispositivos de buffer de un solo caracter requieren más trabajo del sistema operativo que los dispotivos de buffer de 16 caracteres, las tarjetas de interfaz serie basadas en 16550A son mayormente preferidas. Si el sistema tiene muchos puertos serie activos o tendrá una carga elevada, las tarjetas basadas en 16550A son mejores para comunicaciones con baja tasa de error.

24.4.3. Revisión rápida

Como con las terminales, `init` engendra un proceso `getty` para cada puerto serie configurado para conexiones dial-in. Por ejemplo, si un modem está conectado a `/dev/ttyd0` , el comando `ps ax` podría mostrar esto:

```
 4850 ??  I      0:00.09 /usr/libexec/getty V19200 ttyd0
```

Cuando un usuario marca la línea del modem y el modem conecta, la línea CD (Carrier Detect) es reportada por el modem. El kernel nota que se ha detectado una portadora y completa la apertura de `getty` del puerto. `getty` manda un prompt `login:` a la velocidad inicial de línea especificada. `getty` observa si se reciben caracteres válidos, y, en una configuración típica, si encuentra basura (probablemente debido a que la velocidad de conexión del modem es diferente a la velocidad de `getty`), `getty` trata de ajustar la velocidad de la línea hasta que recibe caracteres razonables.

Despues que el usuario entra su nombre de login, `getty` ejecuta `/usr/bin/login` , que completa la entrada preguntando por la contraseña del usuario y entonces inicia el shell del usuario.

24.4.4. Archivos de configuración

Existen tres ficheros de configuración del sistema en el directorio `/etc` que probablemente necesitará editar para permitir acceso de dial-up a su sistema FreeBSD. El primero, `/etc/gettytab` , contiene información de configuración para el daemon `/usr/libe-`

xec/getty. El segundo, /etc/ttys contiene información que le dice a /sbin/init que dispositivos tty deben tener procesos getty corriendo. Por último, puede incluir comandos de inicialización de puerto en el script /etc/rc.serial .

Existen dos escuelas de pensamiento en relación a modems dial-up en UNIX®. Un grupo gusta de configurar sus modems y sistemas para que sin importar a que velocidad un usuario remoto marque, la interfaz local RS-232 computadora-a-modem corra a una velocidad fija. El beneficio de esta configuración es que el usuario remoto siempre obtiene un prompt de login del sistema inmediátamente. La desventaja es que el sistema no sabe cual es la tasa de datos verdadera del usuario, así que programas a pantalla completa como Emacs no ajustarán sus métodos de dibujado de pantalla para mejorar sus respuestas en conexiones más lentas.

La otra escuela configura sus modems de interfaz RS-232 para variar su velocidad basado en la velocidad de conexión del usuario remoto. Por ejemplo, conexiones V.32bis (14.4 Kbps) al modem podrían hacer al modem correr su interfaz RS-232 a 19.2 Kbps, mientras que conexiones 2400 bps hacen correr la interfaz RS-232 del modem a 2400 bps. Debido a que getty no entiende el reporte de velocidad de conexión de cualquier modem, getty brinda un mensaje login: a una velocidad inicial y observa los caracteres que regresan en respuesta. Si el usuario recibe basura, se asume que sabe que debe presionar la tecla Enter hasta que reciba un prompt reconocible. Si la tasa de datos no concuerda, getty trata todo lo que el usuario escriba como "basura", trata yendo a la siguiente velocidad y brinda el prompt login: de nuevo. Este procedimiento puede continuar hasta el cansancio, pero normalmente solo toma un teclazo o dos antes que el usuario reciba un prompt correcto. Obviamente, esta secuencia de login no parece tan limpia como el anterior método de "velocidad fija", pero un usuario en una conexión de velocidad baja podría recibir una respuesta interactiva mejor desde programas a pantalla completa.

Esta sección tratará de dar información de configuración balanceada, pero está cargada hacia tener la tasa de datos del modem siguiendo la tasa de conexión.

24.4.4.1. /etc/gettytab

/etc/gettytab es un fichero tipo termcap(5) de información de configuración para getty(8). Por favor vea la página de manual gettytab(5) para información completa del formato del fichero y la lista de capacidades.

24.4.4.1.1. Configuración de velocidad fija

Si está fijando la tasa de comunicación de datos de su modem a una velocidad particular, probablemente no necesitará ningún cambio a /etc/gettytab.

24.4.4.1.2. Configuración de velocidad concordante

Necesitará crear una entrada en /etc/gettytab para darle información a getty acerca de las velocidades que desea usar para su modem. Si tiene un modem 2400 bps, puede probablemente utilizar la entrada existente D2400.

```
#
# Fast dialup terminals, 2400/1200/300 rotary (can start either way)
#
D2400|d2400|Fast-Dial-2400:\
        :nx=D1200:tc=2400-baud:
3|D1200|Fast-Dial-1200:\
        :nx=D300:tc=1200-baud:
5|D300|Fast-Dial-300:\
        :nx=D2400:tc=300-baud:
```

Si tiene un modem de mayor velocidad, probablemente necesite agregar una entrada en `/etc/gettytab`; aquí está una entrada que puede utilizar para un modem 14.4 Kbps con una velociad de interfaz máxima de 19.2 Kbps:

```
#
# Additions for a V.32bis Modem
#
um|V300|High Speed Modem at 300,8-bit:\
        :nx=V19200:tc=std.300:
un|V1200|High Speed Modem at 1200,8-bit:\
        :nx=V300:tc=std.1200:
uo|V2400|High Speed Modem at 2400,8-bit:\
        :nx=V1200:tc=std.2400:
up|V9600|High Speed Modem at 9600,8-bit:\
        :nx=V2400:tc=std.9600:
uq|V19200|High Speed Modem at 19200,8-bit:\
        :nx=V9600:tc=std.19200:
```

esto resultará en una conexión de 8 bits, sin paridad.

El ejemplo de arriba inicia la tasa de comunicaciones a 19.2 Kbps (para conexiones V.32bis), entonces cicla a través de 9600 bps (para V.32), 2400 bps, 1200 bps, 300 bps, y de vuelta a 19.2 Kbps. El ciclado de la tasa de comunicaciones es implementado con la capacidad `nx=` ("siguiente tabla"). Cada una de las líneas usa una entrada `tc=` ("continuación de tabla") para recoger el resto de las propiedades "estándar" para una tasa de datos en particular.

Si tiene un modem 28.8 Kbps y/o quiere tomar ventaja de la compresión en un modem 14.4 Kbps, necesita utilizar una tasa de comunicaciones mayor a 19.2 Kbps. Aquí hay un ejemplo de una entrada `gettytab` iniciando a 57.6 Kbps:

```
#
# Additions for a V.32bis or V.34 Modem
# Starting at 57.6 Kbps
#
vm|VH300|Very High Speed Modem at 300,8-bit:\
        :nx=VH57600:tc=std.300:
vn|VH1200|Very High Speed Modem at 1200,8-bit:\
        :nx=VH300:tc=std.1200:
vo|VH2400|Very High Speed Modem at 2400,8-bit:\
        :nx=VH1200:tc=std.2400:
```

```
vp|VH9600|Very High Speed Modem at 9600,8-bit:\
        :nx=VH2400:tc=std.9600:
vq|VH57600|Very High Speed Modem at 57600,8-bit:\
        :nx=VH9600:tc=std.57600:
```

Si tiene un CPU lento o un sistema muy cargado y no tiene puertos serie basados en 16550A, tal vez reciba errores en sio "silo" a 57.6 Kbps.

24.4.4.2. `/etc/ttys`

La configuración del fichero `/etc/ttys` fué cubierto en Ejemplo 24.1, "Agregando entradas de terminal a `/etc/ttys`". La configuración para modems es similar pero debemos pasar un argumento diferente a `getty` y especificar un tipo diferente de terminal. El formato general tanto para configuración de velocidad fija y velocidad concordante es:

```
ttyd0   "/usr/libexec/getty xxx"   dialup on
```

El primer componente de la línea de arriba es el fichero de dispositivo especial para esta entrada — `ttyd0` significa que `/dev/ttyd0` es el fichero que `getty` estará vigilando. El segundo componente `"/usr/libexec/getty xxx"` (*xxx* será reemplazado por la capacidad inicial de `gettytab`) es el proceso que `init` ejecutará en el dispositivo. El tercer componente, `dialup`, es el tipo de terminal por omisión. El cuarto parámetro, `on`, le indica a `init` que la línea es operacional. Puede existir un quinto parámetro, `secure`, pero solo debería ser utilizado para terminales que estén físicamente seguras (como la consola del sistema).

El tipo de terminal por omisión (`dialup` en el ejemplo de arriba) puede depender de preferencias locales. `dialup` es el tipo de terminal tradicional por omisión en líneas dial-up para que los usuarios puedan personalizar sus scripts de login para reconocer cuando la terminal es `dialup` y ajustar sus tipos de terminal automáticamente. De toda maneras, el autor encuentra más sencillo especificar en su sitio `vt102` como el tipo de terminal por omisión, puesto que los usuarios solo utilizan emulación VT102 en sus sistemas remotos.

Despues de realizar los cambios a `/etc/ttys` , puede enviar al proceso `init` una señal HUP para que relea el fichero. Puede utilizar el comando

```
# kill -HUP 1
```

para mandar la señal. Si esta es su primera vez instalando el sistema, tal vez quiera esperar hasta que su(s) modem(s) estén configurados y conectados correctamente antes de señalizar a `init`.

24.4.4.2.1. Configuración de velocidad fija

Para una configuración de velocidad fija, su entrada `ttys` necesita tener una entrada de velocidad fija provista en `getty`. Para un modem cuya velocidad de puerto está fijada en 19.2 Kbps, la entrada `ttys` podría verse así:

```
ttyd0   "/usr/libexec/getty std.19200"   dialup on
```

Si su modem está fijado a un velocidad de datos diferente, sustituya el valor apropiado por `std.velocidad` en lugar de `std.19200`. Asegúrese de usar un tipo válido listado en `/etc/gettytab`.

24.4.4.2.2. Configuración de velocidad concordante

En una configuración de velocidad concordante su entrada `ttys` necesita referenciar el inicio de la entrada "auto-baud" (sic) en `/etc/gettytab`. Por ejemplo, si agregó la entrada sugerida arriba para un modem con velocidad concordante que inicia a 19.2 Kbps (la entrada `gettytab` conteniendo el punto de inicio `V19200`), su entrada `ttys` podría verse como esta:

```
ttyd0   "/usr/libexec/getty V19200"   dialup on
```

24.4.4.3. `/etc/rc.serial`

Modems de alta velocidad, como V.32, V.32bis, y V.34, necesitan usar control de flujo por hardware (`RTS/CTS`). Puede agregar comandos `stty` a `/etc/rc.serial` para activar la bandera de control de flujo por hardware en el kernel de FreeBSD para los puertos del modem.

Por ejemplo para activar la bandera `termios crtscts` de dispositivos de inicialización dial-in y dial-out en el puerto serie `#1` (`COM2`), las siguientes líneas pueden agregarse a `/etc/rc.serial`:

```
# Serial port initial configuration
stty -f /dev/ttyid1 crtscts
stty -f /dev/cuaia1 crtscts
```

24.4.5. Propiedades del modem

Si tiene un modem cuyos parámetros pueden ser activados permanentemente en RAM no volatil, necesitará utilizar un programa de terminal (como Telix en MS-DOS® o `tip` en FreeBSD) para activar los parámetros. Conecte al modem usando la misma velocidad de comunicación como velocidad inicial que `getty` usará y configure la RAM no volatil del modem para que concuerde con estos requerimientos:

- CD activado cuando esté conectado
- DTR activado para operación; tirar DTR cuelga la línea y reinicia el modem
- CTS control de flujo de datos transmitidos
- Deshabilitar control de flujo XON/XOFF
- RTS control de flujo de datos recibidos

- Modo silencioso (sin códigos resultantes)
- Sin eco de comandos

Por favor lea la documentación de su modem para saber que comandos y/o switches DIP necesita proporcionarle.

Pro ejemplo, para activar los parámetros de arriba en un modem U.S. Robotics® Sportster® 14,400 externo, uno podría dar estos comandos al modem:

```
ATZ
AT&C1&D2&H1&I0&R2&W
```

Tal vez quiera también tomar esta oportunidad para ajustar otras propiedades en el modem, como si utilizará compresión V.42bis y/o MNP5.

El modem U.S. Robotics® Sportster® 14,400 externo también posee switches DIP que necesitan activarse; para otros modems, tal vez pueda utilizar estas propiedades como un ejemplo:

- Switch 1: ARRIBA — DTR Normal
- Switch 2: N/A (Códigos resultantes verbales/Códigos resultantes numéricos)
- Switch 3: ARRIBA — Suprimir códigos resultantes
- Switch 4: ABAJO — No eco, comandos offline
- Switch 5: ARRIBA — Auto respuesta
- Switch 6: ARRIBA — Detección de señal Normal
- Switch 7: ARRIBA — Cargar valores NVRAM por omisión
- Switch 8: N/A (Modo inteligente/Modo tonto)

Códigos resultantes deberían ser deshabilitados o suprimidos para modems dial-up para evitar problemas que pueden ocurrir si `getty` erroneamente ofrece un prompt `login:` a un modem que se encuentra en modo de comandos y el modem hace eco del comando o regresa un código resultante. Esta secuencia puede resultar en una conversación larga y tonta entre `getty` y el modem.

24.4.5.1. Configuración de velocidad fija

Para una configuración de velocidad fija necesitará configurar el modem para mantener una tasa de datos constante modem-a-computadora independiente de la tasa de comunicaciones. En un modem U.S. Robotics® Sportster® 14,400 externo estos comandos fijarán la tasa de datos modem-a-computadora a la velocidad utilizada para pasar los comandos:

```
ATZ
AT&B1&W
```

24.4.5.2. Configuración de velocidad concordante

Para una configuración de velocidad variable necesitará configurar su modem para ajustar la tasa de datos de su puerto serie para que coincida con la tasa de llamada entrante. En un modem U.S. Robotics® Sportster® 14,400 externo estos comandos fijarán la tasa de corrección de errores de datos a la velocidad usada para pasar los comandos, pero le permite a la tasa del puerto serie variar para conexiones que no corrigen errores:

```
ATZ
AT&B2&W
```

24.4.5.3. Revisando la configuración del modem

La mayoría de los modems de alta velocidad brindan comandos para ver los parámetros actuales de operación del modem en un modo entendible para humanos. En el modem U.S. Robotics® Sportster® 14,400 externo, el comando `ATI5` despliega los parámetros que están almacenados en la RAM no volatil. Para ver los parámetros reales de operación del modem (influenciado por los parámetros de los switches DIP del modem), utilice el comando `ATZ` y entonces `ATI4`.

Si tiene una marca diferente de modem, revise el manual de su modem para ver como checar doblemente los parámetros de configuración de su modem.

24.4.6. Determinando errores

Aquí hay unos cuantos pasos que puede seguir para revisar en sus sistema el modem dialup.

24.4.6.1. Revisando el sistema FreeBSD

Conecte su modem a su sistema FreeBSD, arranque el sistema, y, si su modem tiene luces de indicación de estado, mire si el indicador DTR del modem enciende cuando el prompt `login:` aparece en la consola del sistema — si enciende, eso debería significar que FreeBSD ha iniciado un proceso `getty` en el puerto de comunicaciones apropiado y está esperando a que el modem acepte la llamada.

Si el indicador DTR no enciende, entre al sistema FreeBSD a través de la consola y ponga un `ps ax` para ver si FreeBSD está tratando de correr un proceso `getty` en el puerto correcto. Debería ver líneas como estas entre los procesos desplegados:

```
  114 ??  I      0:00.10 /usr/libexec/getty V19200 ttyd0
  115 ??  I      0:00.10 /usr/libexec/getty V19200 ttyd1
```

Si ve algo diferente, como esto:

```
  114 d0  I      0:00.10 /usr/libexec/getty V19200 ttyd0
```

y el modem no ha aceptado una llamada todavía, esto significa que `getty` ha completado su apertura en el puerto de comunicaciones. Esto puede indicar un problema con el cableado o un modem mal configurado, debido a que `getty` no podría abrir el puerto de comunicaciones hasta que un CD (detección de señal) sea afirmado por el modem.

si no ve ningún proceso `getty` esperando para abrir el puerto `ttydN` deseado, revise de nuevo sus entradas en `/etc/ttys` para ver si existe algún error ahí. También revise el fichero de log `/var/log/messages` para ver si existe algún mensaje de `init` o de `getty` relacionados a cualquier problema. Si existe cualquier mensaje, revise nuevamente los ficheros de configuración `/etc/ttys` y `/etc/gettytab`, así como los ficheros especiales de dispositivo `/dev/ttydN`, por cualquier error, entradas faltantes, o ficheros especiales de dispositivo faltantes.

24.4.6.2. Trate de llamar

Trate de llamar al sistema; asegúrese de usar 8 bits, sin paridad, y 1 bit de parada en el sistema remoto. Si no obtiene un prompt inmediatamente, o recibe basura, trate presionando Enter una vez por segundo. Si continua sin ver un prompt de `login:` despues de un tiempo, trate enviando un `BREAK`. Si está usando un modem de alta velocidad para realizar la marcación, trate marcando de nuevo despues de fijar la velociad de interfaz del modem (por medio de `AT&B1` en un modem U.S. Robotics® Sportster®, por ejemplo).

Si todavía no puede obtener un prompt de `login:`, revise `/etc/gettytab` de nuevo y revise nuevamente que

- El nombre de capacidad inicial especificado en `/etc/ttys` para la línea coincida con un nombre de una capacidad en `/etc/gettytab`
- Cada entrada `nx=` coincida con otro nombre de capacidad de `gettytab`
- Cada entrada `tc=` coincida con otro nombre de capacidad de `gettytab`

Si marca pero el modem en el sistema FreeBSD no contesta, asegúrese que el modem está configurado para contestar el teléfono cuando DTR sea detectado. Si el modem parece estar configurado correctamente, verifique que DTR sea detectado revisando las luces indicadoras del modem (si tiene alguna).

Si ha revisado todo varias veces y todavía no funciona, tome un descanso y regrese a eso depues. Si todavía continua sin funcionar, tal vez puede mandar un correo electrónico a lista de correo para preguntas generales sobre FreeBSD describiendo su modem y su problema, y las buenas personas en la lista tratarán de ayudarle.

24.5. Servicio dial-out

Los siguientes son tips para que su equipo pueda conectarse a otra computadora mediante el modem. Esto es apropiado para establecer una sesión de terminal con un equipo remoto.

Esto es útil para entrar a una BBS.

Este tipo de conexión puede ser extremadamente útil para obtener un fichero del Internet si tiene problemas con PPP. Si necesita mandar por FTP algo y PPP no funciona, utilice la sesión de terminal para mandarlo por FTP. Entonces use zmodem para transferirlo a su máquina.

24.5.1. ¿Mi modem Hayes no está soportado, que puedo hacer?

En realidad, la página de manual para `tip` no está actualizada. Existe un marcador Hayes genérico incluido. Solo utilice `at=hayes` en su fichero `/etc/remote`.

El controlador Hayes no es lo suficientemente inteligente para reconocer algunas de las funciones avanzadas de nuevos modems como `BUSY`, `NO DIALTONE`, o `CONNECT 115200` y solamente se confundirá. Debería apagar esos mensajes cuando utilice `tip` (usando `ATX0&W`).

También, la pausa de marcado para `tip` es de 60 segundos. Su modem debe utilizar un poco menos, o de otra manera tip pensará que existe un problema de comunicación. Trate con `ATS7=45&W`.

Nota

Así como se envía, `tip` todavía no soporta modems Hayes completamente. La solución es editar el fichero `tipconf.h` en el directorio `/usr/src/usr.bin/tip/tip`. Obviamente necesita las fuentes de la distribución para hacer esto.

Edite la línea `#define HAYES 0` a `#define HAYES 1`. Entonces haga un `make` y `make install`. Todo funciona bien despues de eso.

24.5.2. ¿Como se espera que yo entre estos comandos AT?

Haga lo que se llama una entrada "directa" en su fichero `/etc/remote`. Por ejemplo, si su modem está conectado al primer puerto serie, `/dev/cuaa0`, entonces ponga la siguiente línea:

```
cuaa0:dv=/dev/cuaa0:br#19200:pa=none
```

Utilice la tasa más alta de bps que su modem soporte en la capacidad br. Entonces, escriba `tip cuaa0` y estará conectado a su modem.

Si no existe un fichero `/dev/cuaa0` en su sistema, haga esto:

```
# cd /dev
# sh MAKEDEV cuaa0
```

O use cu como `root` con el siguiente comando:

```
# cu -lline -sspeed
```

`line` es el puerto serie (ejem.`/dev/cuaa0`) y *`speed`* es la velocidad (ejem.`57600`). Cuando termine de meter los comandos AT presione ~. para salir.

24.5.3. ¡El signo `@` para la capacidad pn no funciona!

El signo `@` en la capacidad número de teléfono le dice a tip que busque en `/etc/phones` por un número de teléfono. Pero el signo `@` también es un caracter especial en ficheros de capacidad como `/etc/remote`. Escápelo con una diagonal invertida:

```
pn=\@
```

24.5.4. Como puede marcar un número de teléfono. en la línea de comando?

Ponga lo que se llama una entrada "generica" en su fichero `/etc/remote`. Por ejemplo:

```
tip115200|Dial any phone number at 115200 bps:\
        :dv=/dev/cuaa0:br#115200:at=hayes:pa=none:du:
tip57600|Dial any phone number at 57600 bps:\
        :dv=/dev/cuaa0:br#57600:at=hayes:pa=none:du:
```

Entonces puede hacer cosas como:

```
# tip -115200 5551234
```

Si prefiere cu y no `tip`, use una entrada cu:

```
cu115200|Use cu to dial any number at 115200bps:\
        :dv=/dev/cuaa1:br#57600:at=hayes:pa=none:du:
```

y escriba:

```
# cu 5551234 -s 115200
```

24.5.5. ¿Tengo que teclear la tasa de bps cada vez que haga eso?

Ponga una entrada para `tip1200` o `cu1200`, pero utilice cualquier tasa bps que sea apropiada con la capacidad br. `tip` piensa que una buena opción por omisión es 1200 bps es por eso que busca una entrada `tip1200`. Aunque no tiene que usar 1200 bps.

24.5.6. Acceso a un número de equipos a través de un servidor de terminales

en lugar de esperar hasta que esté conectado y teclear `CONNECT <host>` cada vez, use la capacidad de tip `cm`. Por ejemplo, estas entradas en `/etc/remote`:

```
pain|pain.deep13.com|Forrester's machine:\
        :cm=CONNECT pain\n:tc=deep13:
muffin|muffin.deep13.com|Frank's machine:\
        :cm=CONNECT muffin\n:tc=deep13:
deep13:Gizmonics Institute terminal server:\
        :dv=/dev/cuaa2:br#38400:at=hayes:du:pa=none:pn=5551234:
```

Le permitirá teclear `tip pain` o `tip muffin` para conectar a los equipos pain o muffin, y `tip deep13` para accesar al servidor de terminales.

24.5.7. ¿Puede Tip tratar más de una línea para cada sitio?

Eso es a menudo un problema donde una universidad tiene varias líneas de modems y varios miles de estudiantes tratando de usarlas.

Haga una entrada para su universidad en `/etc/remote` y use `@` para la capacidad `pn`:

```
big-university:\
        :pn=\@:tc=dialout
dialout:\
        :dv=/dev/cuaa3:br#9600:at=courier:du:pa=none:
```

Entonces liste los números de teléfonos para la universidad en `/etc/phones`:

```
big-university 5551111
big-university 5551112
big-university 5551113
big-university 5551114
```

`tip` tratará cada uno en el orden listado, entonces se rendirá. Si quiere seguir tratando, ejecute `tip` en un ciclo while.

24.5.8. ¿Porqué tengo que presionar Ctrl+P dos veces para mandar un Ctrl+P ?

Ctrl+P es el caracter de "forzado" por omisión, usado para decirle a `tip` que el siguiente caracter es un dato literal. Puede establecer el caracter de forzado a cualquier otro caracter con el escape `~s`, el cual significa "establecer una variable."

Escriba `~sforce=single-char` seguido por una nueva línea. *`single-char`* es cualquier caracter. Si no especifica *`single-char`*, entonces el caracter de forzado es el caracter nulo, el cual puede obtener tecleando Ctrl+2 o Ctrl+Espacio. Un buén valor para *`single-char`* es Shift+Ctrl+6, es cual solamente es usado en algunos servidores de terminales.

Puede hacer que el caracter de forzado sea cualquiera que usted quiera especificando lo siguiente en su fichero `$HOME/.tiprc` :

```
force=<single-char>
```

24.5.9. ¿¿De repente todo lo que escribo está en mayúsculas??

Debe haber presionado Ctrl+A, el"caracter de mayúsculas" de `tip` especialmente diseñado para personas con teclas caps-lock dañadas. Use `~s` como se ve arriba y establezca la variable `raisechar` a algo razonable. De hecho, puede establecerla a la misma del caracter de forzado, si nunca espera utilizar ninguna de estas funciones.

Aquí hay un ejemplo de fichero .tiprc perfecto para usuarios de Emacs que necesitan teclear Ctrl+2 y Ctrl+A con frecuencia:

```
force=^^
raisechar=^^
```

El ^^ es Shift+Ctrl+6.

24.5.10. ¿Como puedo realizar transferencias de ficheros con `tip`?

Si esta hablando con otro sistema UNIX®, puede mandar y recibir ficheros con `~p` (put) y `~t` (take). Estos comandos ejecutan `cat` y `echo` en el sistema remoto para aceptar y envíar ficheros. La sintaxis es:

`~p` fichero-local [fichero-remoto]

`~t` fichero-remoto [fichero-local]

No existe revisión de errores, así que probablemente debería usar otro protocolo, como zmodem.

24.5.11. ¿Como puedo ejecutar zmodem con `tip`?

Para recibir ficheros, inicie el programa de envío en el extremo remoto. Entonces escriba `~C rz` para empezar a recibirlos localmente.

Para enviar ficheros, inicie el programa de recepción el extremo remoto. Entonces escriba `~C sz files` para enviarlos al sistema remoto.

24.6. Configurando la consola serie

Contribuido por Kazutaka YOKOTA.
Basado en un documento de Bill Paul.

24.6.1. Introducción

FreeBSD tiene la habilidad de arrancar en un sistema con solo una terminal tonta como consola. Tal configuración podría ser útil para dos clases de gente: administradores de sistema que quieran instalar FreeBSD en máquinas que no tienen teclado o monitor conectado, y desarrolladores que quieran corregir errores en el kernel o controladores de dispositivos.

Como se describe en Capítulo 12, *El proceso de arranque en FreeBSD*, FreeBSD emplea un sistema de arranque de tres estados. Los primeros dos estados se encuentran en el código del bloque de arranque el cual es almacenado al principio del slice en el disco de arranque. El bloque de arranque entonces cargará y ejecutará el cargador de arranque (`/boot/loader`) como la tercera etapa de código.

Para poder configurar la consola serie debe configurar el código del bloque de arranque, el código del cargador de arranque y el kernel.

24.6.2. Configuracióm de consola serie, versión breve

Esta sección asume que está usando la configuración por omisión y solo quiere una rápida revisión de la configuración de la consola serie.

1. Conecte el cable serie a COM1 y la terminal controladora.

2. Para ver todos los mensajes de arranque en la consola serie escriba el siguiente comando mientras está firmado como superusuario:

   ```
   # echo 'console="comconsole"' >> /boot/loader.conf
   ```

3. Edite `/etc/ttys` y cambie `dialup` a `vt100` para la entrada `ttyd0`. De otra manera una contraseña no será requerida para conectar por medio de la consola serie, resultando en un agujero de seguridad potencial.

4. Reinicie el sistema para ver si los cambios tuvieron efecto.

Si una configuración diferente es requerida, una explicación mas detallada existe en Sección 24.6.3, “Configuración de la consola serie”.

24.6.3. Configuración de la consola serie

1. Prepare un cable serie.

   Necesitará ya sea un cable null-modem o un cable serie estándar y un adaptador null-modem. Vea Sección 24.2.2, “Cables y puertos” para una discusión sobre cables serie.

2. Desconecte su teclado.

La mayoría de sistemas PC buscan el teclado durante la autoprueba de encendido (POST) y generarán un error si el teclado no es detectado. Algunas máquinas se quejan fuerte sobre la falta de un teclado y no continuarán arrancando hasta que este contectado.

Si su computadora se queja con este error, pero arranca de todos modos, entonces no tiene que hacer nada especial. (Algunas máquinas con BIOS Phoenix instalado solo mostrarán Keyboard failed y continuarán arrancando normalmente.)

Si su computadora se niega a arrancar sin un teclado conectado, entonces tendrá que configurar el BIOS para que ignore este error (si es posible). Consulte el manual de su tarjeta madre para los detalles de como hacer esto.

Sugerencia

Poniendo el teclado como "No instalado" en el BIOS *no* significa que no podá usar su teclado. Todo lo que hace es decirle al BIOS que no busque un teclado al momento de encender, asi no se quejará si el teclado no se encuentra conectado. Puede dejar el teclado conectado incluso si esta bandera está puesta a "No instalado" y el teclado todaía funcionará.

Nota

Si su sistema tiene un ratón PS/2® es muy probable que también tenga que desconectar su ratón junto con el teclado. Esto se debe a que los ratones PS/2® comparten algún hardware con el teclado y dejandolo conectado puede ocasionar que el sistema piense que el teclado sigue conectado. Se dice que un sistema Gateway 2000 Pentium 90 MHz con un AMI BIOS se comporta de esta manera. En general, esto no representa un problema puesto que el ratón no es muy útil sin el teclado de todas maneras.

3. Conecte una terminal tonta a `COM1` (`sio0`).

   Si no tiene una terminal tonta, puede utilizar una PC/XT vieja con un programa de modem, o el puerto serie en otro equipo UNIX®. Si no tiene un `COM1` (`sio0`), consiga uno. En este momento, no existe manera de seleccionar un puerto diferente a `COM1`

para los bloques de arranque sin recompilar los bloques de arranque. Si ya está utilizando `COM1` para otro dispositivo, necesitará remover temporalmente ese dispositivo e instalar un nuevo bloque de arranque y kernel una vez que tenga a FreeBSD arriba y funcionando. (Se asume que `COM1` estará disponible en un fichero/computadora/servidor de terminales de todas maneras; si realmente necesita `COM1` para algo más (y no puede cambiar ese algo a `COM2` (`sio1`)), entonces probablemente no debería de molestarse con todo esto en primer lugar.)

4. Asegúrese de que el fichero de configuración de su kernel tenga las banderas apropiadas activadas para `COM1` (`sio0`).

   Las banderas relevantes son:

   `0x10`
   : Habilita el soporte de consola para esta unidad. Las otras banderas de consola son ignoradas a menos que ésta está activada. Actualmente, al menos una unidad puede tener soporte de consola; la primera (en orden de configuración) con esta bandera activada es preferida. Esta opción por si sola no hará del puerto serie una consola. Active la siguiente bandera o utilice la opción `-h` descrita abajo, junto con esta bandera.

   `0x20`
   : Obliga a esta unidad a ser la consola (a menos que exista otra consola de mayor prioridad), sin importar la opción `-h` discutida abajo. Esta bandera reemplaza la opción `COMCONSOLE` en las versiones 2.*X* de FreeBSD. La bandera `0x20` debe ser utilizada junto con la bandera `0x10`.

   `0x40`
   : Reserva esta unidad (en conjunto con `0x10`) y hace esta unidad no disponible para acceso normal. No debería activar esta bandera en la unidad de puerto serie la cual desee utilizar como la consola serie. El único uso de esta bandera es designar la unidad para corección de errore remota del kernel. Revise “El manual del desarrollador” para mayor información sobre corrección de errores remotamente.

   Nota

   En FreeBSD 4.0 o posterior la semántica de la bandera `0x40` es ligeramente diferente y existe otra bandera para especificar un puerto serie para corrección de errores remotamente.

   Ejemplo:

```
device sio0 at isa? port IO_COM1 flags 0x10 irq 4
```

Vea la página de manual sio(4) para más detalles.

Si las banderas no fueron activadas, necesita correr UserConfig (en una consola diferente) o recompilar el kernel.

5. Cree `boot.config` en el directorio raíz de la partición `a` del disco de arranque.

   Este fichero instruirá al código del bloque de arranque como le gustaría arrancar el sistema. Para activar la consola serie, necesita una o más de las siguientes opciones—si quiere opciones múltiples inclúyalas todas en la misma línea:

   `-h`
   : Cambia entre consola interna y serie. Puede usar esto para cambiar los dispositivos de consola. Por ejemplo, si arranca desde la consola interna (video), puede utilizar `-h` para dirigir el cargador de arranque y el kernel a que usen el puerto serie como su dispositivo de consola. Alternativamente, si arranca desde el puerto serie, puede utilizar `-h` para decirle al cargador de arranque y al kernel que usen el video como consola.

   `-D`
   : Cambia entre configuración de consola simple y dual. En la configuración simple la consola será ya sea la consola interna (video) o el puerto serie, dependiendo del estado de la opción `-h` de arriba. En la configuración de consola dual, el video y el puerto serie se convertirán en la consola al mismo tiempo, sin importar del estado de la opción `-h`. De todas maneras, note que la configuración de consola dual toma efecto solamente mientras el bloque de arranca está corriendo. Una vez que el cargador de arranque toma el control, la consola especificada por la opción `-h` se convierte en la única consola.

   `-P`
   : Hace que el bloque de arranque busque el teclado. Si no se encuentra un teclado, la opción `-D` y `-h` son activadas automátivamente.

   ### Nota

   Debido a problemas de espacio en la versión actual del bloque de arranque, la opción `-P` es capaz de detectar unicamente teclados extendidos. Teclados con menos de 101 teclas (y carentes de teclas F11 y F12) no pueden ser detectados. Algunos teclados en laptops puede que no sean correctamente encontrados debido a esta limitación. Si este es el caso con su sistema, debe abandonar el uso de la

opción `-P`. Desafortunádamente no hay una solución para este problema.

Utilice ya sea la opción `-P` para seleccionar la consola automáticamente, o la opción `-h` para activar la consola serie.

Puede incluir otras opciones descritas en boot(8) también.

Las opciones, excepto por `-P`, serán pasadas al cargador de arranque (`/boot/loader`). El cargador de arranque determinará si el video interno o el puerto serie debería convertirse en la consola examinando el estado de la opción `-h` solamente. Esto significa que si especifica la opción `-D` pero no la opción `-h` en `/boot.config` , puede utilizar el puerto serie como consola solamente durante el bloque de arranque; el cargador de arranque usará el video interno como consola.

6. Arranque la máquina.

   Cuando inicia su equipo FreeBSD, los bloques de arranque mostrarán los contenidos de `/boot.config` a la consola. Por ejemplo:

```
/boot.config: -P
Keyboard: no
```

   La segunda línea aparece solamente si pone `-P` en `/boot.config` e indica la presencia/ausencia del teclado. Estos mensajes van a la consola serie o a la interna, o a ambas, dependiendo de la opción en `/boot.config` .

Opciones	Mensaje va a
ninguna	consola interna
`-h`	consola serie
`-D`	consola serie e interna
`-Dh`	consola serie e interna
`-P`, teclado presente	consola interna
`-P`, teclado ausente	consola serie

   Despues de los mensajes de arriba, existirá una pausa pequeña antes que los bloques de arranque continuen cargando el cargador de arranque y antes de que cualquier mensaje posterior sea impreso en la consola. Bajo ciertas circunstancias, no necesita interrumpir los bloques de arranque, pero tal vez quiera hacerlo para asegurarse que las cosas están configuradas corréctamente.

Presione cualquier tecla, diferente a Enter, en la consola para interrumpir el proceso de arranque. Los bloques de arranque entonces esperarán una entrada para determinar como continuar. Debe ver algo como esto:

```
>> FreeBSD/i386 BOOT
Default: 0:ad(0,a)/boot/loader
boot:
```

Verifique que el mensaje de arriba aparece en la consola serie o en la interna o en ambas, de acuerdo a las opciones que puso en `/boot.config` . Si el mensaje aparece en la consola correcta, presione Enter para continuar el proceso de arranque.

Si quiere usar la consola serie pero no ve el prompt en la terminal serie, algo está mal con su configuración. Mientras tanto, entre `-h` y presione Enter/Return (si es posible) para decirle al bloque de arranque (y entonces al cargador de arranque y al kernel) que elija el puerto serie como consola. Una vez que el sistema arranque, regrese y revise que fué lo que estuvo mal.

Despues que el cargador de arranque ha cargado y usted se encuentra en la tercera etapa del proceso de arranque todavía puede cambiar entre la consola interna y la consola serie activando las variables de entorno apropiadas en el cargador de arranque. Vea Sección 24.6.6, “Cambiando la consola desde el cargador de arranque”.

24.6.4. Resumen

aquí está el resumen de varias configuraciones discutidas en esta sección y la consola seleccionada eventualmente.

24.6.4.1. Caso 1: Activó las banderas a 0x10 para `sio0`

```
device sio0 at isa? port IO_COM1 flags 0x10 irq 4
```

Opciones en /boot.config	Consola durante bloques de arranque	Consola durante cargador de arranque	Consola en kernel
ninguna	interna	interna	interna
`-h`	serie	serie	serie
`-D`	serie e interna	interna	interna
`-Dh`	serie e interna	serie	serie
`-P`, teclado presente	interna	interna	interna
`-P`, teclado ausente	serie e interna	serie	serie

24.6.4.2. Caso 2: Activó las banderas a 0x30 para sio0

```
device sio0 at isa? port IO_COM1 flags 0x30 irq 4
```

Opciones en /boot.config	Consola durante bloques de arranque	Consola durante cargador de arranque	Consola en kernel
ninguna	interna	interna	serie
`-h`	serie	serie	serie
`-D`	serie e interna	interna	serie
`-Dh`	serie e interna	serie	serie
`-P`, teclado presente	interna	interna	serie
`-P`, teclado ausente	serie e interna	serie	serie

24.6.5. Consejos para la consola serie

24.6.5.1. Configurando un velocidad de puerto serie más rápida

Por omisión, la configuración del puerto serie es: 9600 baud, 8 bits, sin paridad, y 1 bit de parada. Si desea cambiar la velocidad, necesita recompilar al menos los bloques de arranque. Agregue la siguiente línea a `/etc/make.conf` y compile nuevos bloques de arranque:

```
BOOT_COMCONSOLE_SPEED=19200
```

Vea Sección 24.6.5.2, “Usando puertos serie para consola diferentes a `sio0`” para instrucciones detalladas sonre construir e instalar nuevos bloques de arranque.

Si la consola serie está configurada de alguna otra manera que arrancando con `-h`, o si la consola serie usada por el kernel es diferente de la usada por los bloques de arranque, entonces también debe agregar la siguiente opción al fichero de configuración del kernel y compilar un nuevo kernel:

```
options CONSPEED=19200
```

24.6.5.2. Usando puertos serie para consola diferentes a `sio0`

Utilizar un puerto serie diferente a `sio0` como consola requiere cierta recompilación. Si quiere usar otro puerto serie por la razón que sea, recompile los bloques de arranque, el cargador de arranque y el kernel como sigue.

1. Consiga las fuentes del kernel. (Vea Capítulo 23, *Lo último de lo último*)

2. Edite `/etc/make.conf` y ponga `BOOT_COMCONSOLE_PORT` a la dirección del puerto que quiera usar (0x3F8, 0x2F8, 0x3E8 o 0x2E8). Solamente de `sio0` hasta `sio3` (`COM1` hasta `COM4`) pueden usarse; tarjetas multipuertos serie no funcionarán. No se necesita especificar interrupción.

3. Cree un fichero personalizado de configuración de kernel y agregue las banderas apropiadas para el puerto serie que desee utilizar. Por ejemplo, si desea hacer de `sio1` (`COM2`) la consola:

```
device sio1 at isa? port IO_COM2 flags 0x10 irq 3
```

o

```
device sio1 at isa? port IO_COM2 flags 0x30 irq 3
```

Las banderas de consola para otros puertos serie no deben activarse.

4. Recompile e instale los bloques de arranque y el cargador de arranque:

```
# cd /sys/boot
# make clean
# make
# make install
```

5. Reconstruya e instale el kernel.

6. Escriba los bloques de arranque al disco de arranque con disklabel(8) y arranque desde el nuevo kernel.

24.6.5.3. Accesando DDB Debugger desde la línea serie

Si desea entrar al modo kernel debugger desde una consola serie (útil para diagnósticos remotos, !pero también peligroso si genera un BREAK ilegítimo en el puerto serie!) entonces debe compilar con las siguientes opciones:

```
options BREAK_TO_DEBUGGER
options DDB
```

24.6.5.4. Obteniendo un prompt de login en la consola serie

Aunque esto no es requerido, tal vez quiera obtener un prompt de *login* a través de una línea serie, ahora que puede ver los mensajes de arranque y puede entrar a una sesión en modo kernel debug a través de la consola serie. Aquí está como hacerlo.

Abra el fichero `/etc/ttys` con un editor y localice las líneas:

```
ttyd0 "/usr/libexec/getty std.9600" unknown off secure
ttyd1 "/usr/libexec/getty std.9600" unknown off secure
ttyd2 "/usr/libexec/getty std.9600" unknown off secure
ttyd3 "/usr/libexec/getty std.9600" unknown off secure
```

`ttyd0` hasta `ttyd3` corresponde a `COM1` hasta `COM4`. Cambie `off` a `on` para el puerto deseado. Si ha cambiado la velocidad del puerto serie, necesita cambiar `std.9600` para que concuerde con los parámetros actuales, ej. `std.19200` .

Tal vez tambien desee cambiar el tipo de terminal de `unknown` al tipo actual de su terminal serie.

Despues de editar el fichero, debe hacer un `kill -HUP 1` para que este cambio surta efecto.

24.6.6. Cambiando la consola desde el cargador de arranque

Secciones anteriores describieron como instalar la consola serie manipulando el bloque de arranque. Esta sección muestra que puede especificar la consola especificando algunos comandos y variables de entorno en el cargador de arranque. Como el cargador de arranque es invocado en la tercera etapa del proceso de arranque, despues del bloque de arranque, las propiedades en el cargador de arranque sobreescribirán las del bloque de arranque.

24.6.6.1. Instalando la consola serie

Puede facilmente especificarle al cargador de arranque y al kernel que utilicen la consola serie escribiendo solamente una línea en `/boot/loader.rc` :

```
set console="comconsole"
```

Esto tendrá efecto sin importar las opciones del bloque de arranque discutidas en la sección previa.

Es mejor que ponga la línea de arriba como la primera línea en `/boot/loader.rc` para ver los mensajes de arranque en la consola serie tan pronto como sea posible.

De igual manera, puede especificar la consola interna como:

```
set console="vidconsole"
```

Si no activa la variable de entorno `console`, el cargador de arranque, y por consecuencia el kernel, utilizarán cualquier consola que esté indicada por la opción `-h` en el bloque de arranque.

En versiones 3.2 o posteriores, puede especificar la consola en `/boot/loader.conf.local` o `/boot/loader.conf` , en lugar de `/boot/loader.rc` . En este método su `/boot/loader.rc` debe verse como:

```
include /boot/loader.4th
start
```

Entonces, puede crear `/boot/loader.conf.local` y ponerle la siguiente línea.

```
console=comconsole
```

o

```
console=vidconsole
```

Vea loader.conf(5) para mayor información.

Nota

Hasta el momento, el cargador de arranque no tiene una opción equivalente a la opción `-P` del bloque de arranque, y no existe una manera de seleccionar automáticamente la consola interna y la consola serie basandose en la presencia del teclado.

24.6.6.2. Utilizando un puerto serie para la consola diferente a `sio0`

Necesita recompilar el cargador de arranque para usar un puerto serie diferente a `sio0` para la consola serie. Siga el procedimiento descrito en Sección 24.6.5.2, “Usando puertos serie para consola diferentes a `sio0`”.

24.6.7. Advertencias

La idea aquí es permitir a las personas configurar servidores dedicados que no requieran hardware de gráficos o teclados conectados. Desafortunádamente, mientras la mayoría de los sistemas le permitirán arrancar sin un teclado, existen bastantes que no le permitirán arrancar sin un adaptador de gráficos. Máquinas con BIOS AMI pueden configurarse para arrancar sin adaptadores de gráficos instalados cambiando simplemente la opción “graphics adapter” en la configuración del CMOS a “Not installed.”

De cualquier manera, muchas máquinas no soportan esta opción y se negarán a arrancar si no tiene algún hardware de gráficos instalado en el sistema. Con estas máquinas, debe dejar algún tipo de tarjeta gráfica instalada, (incluso si solamente es una tarjeta mono barata) aunque no tendrá que conectarle un monitor. También puede tratar instalando un BIOS AMI.

Capítulo 25. PPP y SLIP

Reestructurado, reorganizado, y actualizado por Jim Mock.

25.1. Sinopsis

FreeBSD cuenta con un gran numero de formas para conectar una computadora a otra. Para establecer una red o una conección a Internet por medio de un módem, o bien, permitir a otras computadoras conectarse por medio de este, se requiere del uso de PPP o SLIP. Este capítulo describe en detalle como configurar los servicios de comunicación para llevar esto a cabo.

Una vez que haya leido este capítulo, usted sabrá:

- Como configurar User PPP.
- Como configurar Kernel PPP.
- Como configurar PPPoE (PPP over Ethernet*).
- Como configurar PPPoA (PPP over ATM*).
- Como instalar y configurar un cliente y servidor SLIP.

N. de T.: En estricto sentido esto se refiere a contar con la conexión por medio de un dispositivo Ethernet, o bien ATM, pero debido a que usted encontrará estos metodos en su sistema, como PPPoE o bien PPPoA, se han dejado los conceptos “literales” del documento original. Espero que no sea un problema.

Antes de leer este capítulo, usted debiese:

- Estar familiarizado con la terminología básica de redes.
- Comprender lo básico y el propósito de una conección por módem SLIP y/o PPP.

Puede ser que usted se pregunte cual es la principal diferencia entre User PPP y kernel PPP. La respuesta es sencilla; el método User PPP procesa la entrada y salida de datos en userland (ver nota siguiente) en lugar de hacerlo en el kernel. Esto es algo desgastante, en términos del manejo de datos entre userland y el kernel, pero permite, por mucho, un mejor desempeño e implementación de PPP. User PPP utiliza el dispositivo `tun` para comunicarse con el mundo exterior, mientras que kernel—ppp, utiliza el dispositivo `ppp`.

En el desarrollo de este capítulo, se hará referencia a User PPP, simplemente como *ppp*, a menos de que sea necesaria hacer una distinción entre este y otro software de PPP, como es el caso de `pppd`. Así mismo, si en el desarrollo del capítulo no se señala lo contrario, todos los comandos explicados, deberán ser ejecutados como `root`.

N. de T. : Cuando se habla de "userland" se hace referencia a todo aquello que *no* forma parte del kernel y que en el caso de código de programa, se ejecuta en modo usuario, ya que el código del kernel se ejecuta en modo kernel, supervisor, o bien en modo privilegiado de ejecución. En lo sucesivo este término será utilizado tal cual.

25.2. Uso de User PPP

Originalmente contribuido por Brian Somers.
Con apoyo de Nik Clayton, Dirk Frömberg, Peter Childs y Tom Rhodes.

25.2.1. User PPP

25.2.1.1. Aclaraciones

Este documento asume que usted cuenta con lo siguiente:

•

- Adicionalmente, un módem o algún otro dispositivo, conectado a su sistema, y configurado correctamente, que le permite realizar la conexión con su ISP.
- El número telefónico de su proveedor.

•

•
La dirección IP de uno o más servidores de nombres (DNS). Normalmente, estos serán provistos por su proveedor de Internet. Si su proveedor no le ha dado esta información, puede utilizar la opción `enable dns` en su fichero `ppp.conf` , para indicarle a ppp que configure el DNS por usted. Esta característica depende del sistema de negociación de DNS que mantenga su proveedor de Internet.

La siguiente información puede ser que haya sido provista por su proveedor de servicios de internet, pero no es completamente necesaria:

- La dirección IP del gateway (pasarela de salida) de su PSI. El gateway es la máquina a la cual usted se conectará y será la *ruta por default*. Si usted no cuenta con esta información, puede inventar uno y al intentar conectarse, el servidor de su PSI, este nos indicará cual es el valor correcto.

  Esta dirección IP, es referida por ppp como `HISADDR` .

- La mascara de red (netmask) que debe utilizar. Si su PSI no le ha provisto de una, puede utilizar sin problema `255.255.255.255` .

•
Si su PSI, le ha provisto de una dirección de IP estática y un nombre de host, puede capturarla. De otra forma podemos dejar que el servidor asigne cualquier IP que corresponda.

Si usted no cuenta con alguna de la información que hemos comentado, le recomendamos contactar con su PSI para requerirla.

Nota

En el transcurso de la presente sección, algunos ejemplos muestran el contenido de archivos de configuración los cuales presentan una numeración. Estos números sirven como ayuda y referencia a cada línea, pero estos no deben de estar presentes en el archivo original. Una sangría adecuada, así como espacios adecuados, también son de suma importancia.

25.2.1.2. Preparando el Kernel

Como se comento anteriormente, la aplicación ppp utiliza el dispositivo `tun`. Si este dispositivo no ha sido compilado dentro del kernel, ppp lo cargará como módulo cuando sea requerido. El dispositivo tun es dinámico, de tal forma que se generara de acuerdo a la demanda que tenga (usted no esta limitado por el kernel).

Nota

Vale la pena hacer notar que el controlador tun, crea los dispositivos de acuerdo a sus necesidades, por lo que el comando `ifconfig -a`, no necesariamente mostrará los dispositivos `tun`.

25.2.1.3. Verificando el dispositivo `tun`

Bajo circunstancias normales, la mayoría de los usuarios sólo utilizaran un dispositivo `tun` (`/dev/tun0`). En lo sucesivo podemos hacer referencia a `tun0` con la expresión `tunN` donde `N` es el número que corresponde en su sistema.

Para instalaciones de FreeBSD que no tienen el habilitado el DEVFS la existencia de `tun0` debe ser verificada (esto no es necesario si se cuenta habilitada la opción DEVFS ya que los nodos de dispositivos seán creados en función a las necesidades).

La forma más sencilla de verificar si el dispositivo `tun0` se encuentra configurado correctamente, es la de rehacer el dispositivo. Para hacer esto simplemente siga los siguientes pasos:

```
# cd /dev
# sh MAKEDEV tun0
```

Si usted necesita 16 dispositivos tun en su kernel, deberá crearlos. Esto puede hacerse de la siguiente manera:

```
# cd /dev
# sh MAKEDEV tun15
```

25.2.1.4. Configuración de la Resolución de Nombres

La resolución es la parte del sistema que busca una dirección IP en los nombres de servidores (host) y viceversa. Puede ser configurado para que busque en "mapas" que describen la IP del servidor en uno de dos lugares, el primero es un archivo llamado `/etc/hosts` . Lea hosts(5) para más información al respecto. El segundo es el Servicio de Nombres de Dominio de Internet (DNS-Internet Domain Name Service), el cual es una base de datos de distribución. Para mayor información con respecto a los DNS, referirse a dns.

La resolución de nombres es un sistema que por medio de llamadas, realiza el mapeo de nombres, pero es necesario inidicarle donde debe buscar la información. Para versiones de FreeBSD anteriores a la 5.0, esto es hecho al editar el archivo `/etc/host.conf` . La versión 5.0 de FreeBSD utiliza el archivo `/etc/nsswitch.conf` .

25.2.1.4.1. Edición del archivo `/etc/host.conf`

Para versiones de FreeBSD anteriores a la 5.0, este archivo debe contener las siguientes dos líneas (en este orden):

```
hosts
bind
```

Esto le indica a la resolución que busque en primer término en el archivo `/etc/hosts` , y posteriormente en el DNS, si el nombre no fué localizado

25.2.1.4.2. Editando el archivo `/etc/nsswitch.conf`

Para versiones de FreeBSD 5.0 y posteriores, este archivo debe contener, al menos, la siguiente línea:

```
hosts: files, dns
```

Esto le indica a la resolución de nombres, que busque en primer lugar en el archivo `/etc/hosts` , y en caso de que el nombre no haya sido localizado, busque en el DNS.

25.2.1.4.3. Editando el archivo `/etc/hosts`

Este archivo puede contener direcciones IP, así como el nombre de las máquinas de su red local. Como mínimo debe contar con la información de la máquina que correrá ppp. Asumiendo que su ordenador se llama `foo.bar.com` con la dirección IP `10.0.0.1` , el archivo `/etc/hosts` debiese contener:

```
127.0.0.1 localhost.bar.com localhost
::1 localhost.bar.com localhost
```

```
10.0.0.1 foo.bar.com  foo
```

Las primeras dos líneas definen el alias del `localhost`, como sinónimo de la maquina actual. Independientemente de su propia dirección IP, la dirección IP en estas líneas siempre debe ser `127.0.0.1` y `::1`. La última línea especifica el nombre `foo.bar.com` (asi como `foo` para acortarlo), para la dirección `10.0.0.1`.

Nota

La dirección `127.0.0.1` y el nombre `localhost` son conocidos como direcciones "loopback" las cuales hacen un "loopback" (salto de regreso) a la maquina local.

Si su proveedor de Internet, le asigna una dirección IP fija, asín como un nombre, y usted no lo utiliza como nombre del host, añada esto también al archivo `/etc/hosts`.

25.2.1.4.4. Editando el archivo `/etc/resolv.conf`

El archivo `/etc/resolv.conf`, le indica a la resolución de nombres, como comportarse. Normalmente deberá de incluir la(s) siguiente(s) línea(s):

```
domain ejemplo.com
nameserver x.x.x.x
nameserver y.y.y.y
```

Donde *`x.x.x.x`* y *`y.y.y.y`* deben reemplazarse con las direcciones IP de los servidores DNS, de su ISP. Puede ser que esta información se la hayan entregado al suscribirse o no, pero una rápida llamada a su ISP debe resolver esto.

Tambié puede configurar su sistema, de tal forma que syslog(3) provee de un login para su conección por PPP. Sólo añada:

```
!ppp
*.*         /var/log/ppp.log
```

al fichero `/etc/syslog.conf`. En la mayoría de los casos esto funciona bien.

25.2.1.5. Configuración Automática de PPP

Ambos, `ppp` así como `pppd` (la implementación del kernel para PPP), utilizan la configuración de los archivos localizados en el directorio `/etc/ppp`. Ejemplos para ppp, pueden encontrarse en: `/usr/share/examples/ppp/`.

Para efecto de configurar correctamente `ppp`, es necesario editar varios ficheros, dependiendo de sus necesidades. La manera en que edite dichos archivos, depende en la forma

que utilice su PSI (Proveedor de Servicios de Internet) para brindarle conexión, ya sea por medio de una dirección IP estática o bien una IP dinámica (ya sea que cada vez que se conecta obtiene una nueva dirección).

25.2.1.5.1. PPP y direcciones de IP estáticas (fijas)

Será necesario editar el archivo de configuración; `/etc/ppp/ppp.conf` . Y deberá quedar de una manera similar al ejemplo que se describe a continuación.

Nota

Las líneas que terminan con `:`, deben comenzar en la primer columna del archivo — el resto de las líneas deberán utilizar sangría como se muestra, utilizando espacios o bien el tabulador. La mayor parte de la información que requiere ingresar aqui, se mostro en el marcado manual anterior.

```
1     default:
2       set log Phase Chat LCP IPCP CCP tun command
3       ident user-ppp VERSION (built COMPILATIONDATE)
4       set device /dev/cuaa0
5       set speed 115200
6       set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 5 \
7                 \"\" AT OK-AT-OK ATE1Q0 OK \\dATDT\\T TIMEOUT 40 ↺
CONNECT"
8       set timeout 180
9       enable dns
10
11    provider:
12      set phone "(123) 456 7890"
13      set authname foo
14      set authkey bar
15      set login "TIMEOUT 10 \"\" \"\" gin:--gin: \\U word: \\P ↺
col: ppp"
16      set timeout 300
17      set ifaddr x.x.x.x y.y.y.y 255.255.255.255 0.0.0.0
18      add default HISADDR
```

Línea 1:

Identifica la entrada por omisión a utilizar. Los comandos descritos en esta parte, serán ejecutados de manera automática cuando se ejecute ppp.

Línea 2:

Habilita los par´metros de acceso. Cuando la configuración trabaja sin problemas, esta línea deberá quedar de la siguiente forma:

```
set log phase tun
```

para efecto de evitar avisos masivos del sistema (logs).

Línea 3:
Esta línea le indica a PPP como identificarse ante el puerto. PPP se identifica, si tiene algun problema para efecto de establecer la conexión, en esta identificación, PPP provee de cierta información que puede resultar util para detectar el probelma.

Línea 4:
Le indica a PPP cual es el dispositivo a utilizar para realizar la conexión, o bien al que esta conectado el módem. El dispositivo `COM1` es `/dev/cuaa0` y `COM2` es `/dev/cuaa1` .

Línea 5:
Establece la velocidad a utilizar en la conexión. Si la velocidad de 115200 no trabaja correctamente (la cual deberia con cualquier módem normal), intente con una velocidad inferior, como puede ser 38400.

Líneas 6 y 7:

La cadena de inicialización. El modo User PPP, utiliza y espera enviar-recibir, la información utilizando una sintaxis similar a la descrita en el programa chat(8). Favor de consultar la página de ayuda para conocer las opciones de este lenguaje.

Nota: Este comando continua en la siguiente línea, para facilitar su lectura. Cualquier comando en el archivo `ppp.conf` puede utilizar este formato, siempre y cuando el último caracter de la línea sea una diagonal invertida "\".

Línea 8:
Establece el tiempo de espera que debe tratar de realizar la conexión. Por omisión se establecen 180 segundos, por lo que esta línea se deja por pura estética.

Línea 9:
Esta línea le indica a PPP, que solicite confirmación al puerto, sobre la configuración de la resolución local. Si usted esta corriendo un servidor local de nombres, deberá comentar o eliminar esta línea.

Línea 10:
Una línea en blanco, para facilitar la lectura. Las líneas en blanco son ignoradas por PPP.

Línea 11:
Identifica el incio de datos para un “proveedor” determinado, de servicios de internet. Este podrá ser cambiado por el nombre de su ISP, de tal forma que en lo sucesivo utilice la opción `load ISP` , para iniciar una sesión.

Línea 12:
Indica el numero telefónico del proveedor. Pueden indicarse varios numeros a utilizar, utilizando el signo de dos puntos (`:`) o bien la barra (`|`) como separador. La di-

ferencia entre estos dos separadores, es detallada en el ppp(8). Pero en resumen, se puede decir que si se desean utilizar varios numeros de manera aleatoria se debe utilizar los dos puntos, pero si se desea siempre utilizar el primer numero y en caso de falla el siguiente y así sucesivamente, se debe utilizar la barra. Es importante que todo lo que se refiere a numeros telefonicos, este entre comillas como se muestra. Es importante que si piensa usar espacios en los numeros, haga uso de estas comillas ("). La falta de estas pueden ocasionar un simple error.

Líneas 13 y 14:

Identifica el nombre de usuario y su contraseña. Cuando uno se conecta utilizando un login de tipo Unix, estos valores hacen referencia al comando `set login`, utilizando las variables \U y \P. Cuando la conexión es utilizando algún metodo como PAP o CHAP, estos valores, son utilizados al momento de la autentificación.

Línea 15:

Si usted esta utilizando el metodo PAP o CHAP, no habrá un login en este punto, y esta línea deberá ser comentada (utilizando el símbolo # al principio de la línea) o bien eliminada por completo. Vea la parte Autentificación con PAP y CHAP para más detalles.

La cadena de acceso (login), utiliza la misma sintáxis que se utiliza en la cadena de marcado. En este ejemplo, la cadena sirve para un servicio, en el cual el inicio de sesión se ve algo así como lo siguiente:

```
Proveedor de servicios X
login: foo
password: bar
protocol: ppp
```

Es recomendable editar el script, para que se ajuste a sus propias necesidades. Cuando cree este script por primera vez, asegurese de haber habilitado la parte que se refiere a al acceso por medio de “chat”, para efecto de poder dar seguimiento al curso de la conexión y la resolución de la misma.

Línea 16:

Establece el tiempo por defecto en el que se perderá la conexión (en segundos). En este caso la conexión será cortada de forma automática, después de 300 segundos de inactividad. Si no desea habilitar esta función establezca este valor en cero o bien utilice el comando en línea `-ddial`.

Línea 17:

Indica la dirección de la interfaz. La cadena que aparece como *`x.x.x.x.`* , debe se cambiada por la dirección asignada por su PSI. La línea que aparece como *`y.y.y.y.`* , debe se substituida por la direcció IP especificada por su PSI, como servidor de salida

o pasarela (gateway)(la maquina a la cual se va a conectar). Si su PSI no le ha indicado una dirección de este tipo, puede utilizar `10.0.0.2/0` . Si usted necesita utilizar una dirección "aleatoria", asegurese de crear el fichero `/etc/ppp/ppp.linkup` , siguiendo las instrucciones de PPP y las direcciones de IP Dinámicas, para su llenado. Si esta línea es omitida, `ppp`, no podrá ejecutarse en el modo `-auto`.

Línea 18:

Añade una ruta por omisión al servidor de salida de su PSI. La palabra especial `HISADDR` se reemplaza con la dirección del gateway indicado por su PSI, que esta en la línea 9, de otra forma `HISADDR` no será inicializado.

Si no desea ejecutar `ppp` en modo `-auto`, esta línea deberá pasar al archivo `ppp.linkup` .

No hay necesidad de editar el archivo `ppp.linkup` si usted cuenta con una dirección IP estática y se esta ejecutando ppp en modo `-auto`, en virtud de que para efecto de realizar la conexión sus mapas de ruteo debe estar correctas. De cualquier forma puede ser que usted desee ejecutar algun programa/comando, posterior a la conexión. Este es explicado con más detalle posteriormente, cuando se vea el ejemplo de sendmail.

Ejemplo de los archivos de configuración, se pueden encontrar en el directorio; `/usr/share/examples/ppp`.

25.2.1.5.2. PPP y direcciones de IP Dinámicas (Variables)

Si su proveedor de servicios, no le asigna una dirección de IP fija, será necesario configurar a `ppp`, de tal forma que al momento de realizar la conexión, negocie tanto la dirección local, como la remota. Esto se lleva a cabo al "adivinar" una dirección IP y permitiendo a `ppp` que la establezca correctamente, usando el Protocolo de Configuración de IP (IPCP), una vez que se ha conectado. La configuración que debe tener el archivo `ppp.conf` , es la misma que la utilizada en PPP y direcciones de IP fijas, salvo el siguiente cambio:

```
17      set ifaddr 10.0.0.1/0 10.0.0.2/0 255.255.255.255
```

Una vez más, no debe incluir el numero de línea, este sólo es una referencia. Así mismo deberá existir sangrado, de cuando menos 1 espacio.

Línea 17:

El numero siguiente a la diagonal (`/`), es el numero de bits de la dirección en la cual ppp insistirá en conectarse. Puede ser que usted desee utilizar numeros de IP que sean más apropiados, para ajustar a sus necesiadades, pero el ejemplo descrito anteriormente siempre podrá utilizarse.

El último argumento (`0.0.0.0`), le indica a PPP, que inicie las negociaciones, utilizando como dirección `0.0.0.0` , en lugar de que utilice `10.0.0.1` , lo cual es necesario con algunos proveedores. No utilice la dirección `0.0.0.0` como el primer argumen-

to, para el comando `set ifaddr` , ya que impide que PPP configure de forma correcta el sistema, cuando se utiliza en modo `-auto`.

Si usted no esta ejecutando PPP en modo `-auto`, deberá editar su archivo `/etc/ppp/ppp.linkup` . El archivo `ppp.linkup` , es utilizado una vez que se ha realizado la conexión. En este punto, `ppp` habrá negociado una dirección de interfaz, y será posible ahora, añadir las entradas para la las tablas de ruteo:

```
1     provider:
2      add default HISADDR
```

Línea 1:

Al establecer (`ppp`) una conexión, buscará en `ppp.linkup` una entrada, de acuerdo a las siguientes reglas. Primero, tratar de encontrar una entrada que sea igual a la utilizada en el archivo `ppp.conf` . Si esto falla, buscar una IP con la dirección de nuestro gateway. Esta entrada es una etiqueta de tipo IP, de cuatro-octetos. Si aun después de esto no se ha detectado la entrada correcta, buscar la entrada `MYADDR`.

Línea 2:

Esta línea le indica a `ppp` que añada una ruta por omisión, que este dirigida hacia `HISADDR`. `HISADDR` será reemplazada, con la IP del gateway, como se negocio por IPCP.

Para ver un detalle más preciso de esto, puede consultar la entrada de `pmdemand` en los archivos de ejemplo `/usr/share/examples/ppp/ppp.conf.sample` así como `/usr/share/examples/ppp/ppp.linkup.sample` .

25.2.1.5.3. Recibiendo Llamadas Externas

Cuando se configure ppp, para recibir llamadas externas, en una maquina conectada a un LAN (Red de Area Local), debe decidir si se va a permitir el envío de paquetes a la LAN. Si es así, debe asignar un numero de IP de su red local y utilizar el comando `enable proxy` en el archivo de configuracion `/etc/ppp/ppp.conf` . También deberá asegurarse que en su archivo `/etc/rc.conf` cuente con la línea:

```
gateway_enable="YES"
```

25.2.1.5.3.1. ?Qué getty utilizar?

El enlace Configurando FreeBSD para Servicios de Marcado provee de una buena descripció, sobre la configuración de estos servicios, basado en getty(8).

Una alternativa para el comando `getty` es mgetty , el cual es una versión más inteligente de getty diseñada para servicios de marcado telefonico.

Una de las principales ventajas de `mgetty` es que, de hecho *platica* con los modems, esto es, significativo, ya que si el puerto esta desactivado en su `/etc/ttys` el modem no responderá el llamado.

Las últimas versiones de `mgetty` (de la 0.99beta y sucesivas), también cuentan con soporte para la detección automática de llamados de PPP, permitiendo el acceso a servidores de una manera más sencilla (sin uso de tanto scripts).

Puede referirse a Mgetty y AutoPPP para más información con respecto al comando `mgetty`.

25.2.1.5.3.2. Permisos de PPP

El comando `ppp` normalmente debe ser ejecutado por root (superusuario). Si de cualquier forma, usted desea permitir que `ppp` pueda ser ejecutado en modo servidor, por un usuario regular, como se describe a continuación, deberá otorgar los permisos necesarios a ese usuario al añadirlo al grupo `network`, en el fichero `/etc/groups` .

También será necesario darle acceso a una o más partes del archivo de configuración, haciendo uso del comando `allow`, como se ve a continuación:

```
allow users fred mary
```

Si el comando es utilizado en la sección `default`, esto le dará a el(los) usuario(s) especificado(s), acceso a todo.

25.2.1.5.3.3. Shells de PPP para Usuarios de IP Dinámica

Cree un fichero llamado: `/etc/ppp/ppp-shell` y que contenga lo siguiente:

```
#!/bin/sh
IDENT=`echo $0 | sed -e 's/^.*-\(.*\)$/\1/'`
CALLEDAS="$IDENT"
TTY=`tty`

if [ x$IDENT = xdialup -]; then
        IDENT=`basename $TTY`
fi

echo "PPP for $CALLEDAS on $TTY"
echo "Starting PPP for $IDENT"

exec /usr/sbin/ppp -direct $IDENT
```

Este script deberá ser ejecutable. Ahora cree un enláce simbólico llamado `ppp-dialup` a este script, utilizando los siguientes comandos:

```
# ln -s ppp-shell /etc/ppp/ppp-dialup
```

Deberá utilizar este script como *shell* para todos los usuarios que realicen conexión. Este es un ejemplo del fichero `/etc/password` para un usuario con acceso a PPP, con nombre de usuario `pchilds` (recuerde no editar directamente el fichero password, utilice `vipw`).

```
pchilds:*:1011:300:Peter Childs PPP:/home/ppp:/etc/ppp/ppp-dialup
```

Cree un directorio llamado `/home/ppp` que contenga los siguentes archivos de `0` bytes:

```
-r--r--r--   1 root     wheel           0 May 27 02:23 .hushlogin
-r--r--r--   1 root     wheel           0 May 27 02:22 .rhosts
```

los cuales impiden que `/etc/motd` sea desplegado.

25.2.1.5.3.4. Shells de PPP para Usuarios de IP Estática

Cree el fichero `ppp-shell` al igual que el mencionado con anterioridad, y por cada cuenta donde se tenga asignada una IP estática, cree un enlace simbólico al fichero `ppp-shell`.

Por ejemplo, si usted cuenta con tres usuarios que utilicen este servicio; `fred`, `sam` y `mary`, los cuales redirecciona a una red de clase `C`, habria que hacer lo siguiente:

```
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-fred
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-sam
# ln -s /etc/ppp/ppp-shell /etc/ppp/ppp-mary
```

Cada uno de los usuarios señalados, deberán de contar con el enlace a su shell-script como se indicó (por ejemplo, el usuario `mary`, debe contar con su enlace al fichero `/etc/ppp/ppp-mary`).

25.2.1.5.3.5. Configurando `ppp.conf` para Usuarios de IP-Dinámica

El archivo `/etc/ppp/ppp.conf` deberá contener algo similar a lo siguiente:

```
default:
  set debug phase lcp chat
  set timeout 0

ttyd0:
  set ifaddr 203.14.100.1 203.14.100.20 255.255.255.255
  enable proxy

ttyd1:
  set ifaddr 203.14.100.1 203.14.100.21 255.255.255.255
  enable proxy
```

Nota

Tomar en cuenta el sangrado, ya que es importante.

La sección `default:` es cargada para cada sesión. Para cada línea que exista y habilite el marcado, en el fichero `/etc/ttys`, se deberá crear una entrada similar a la línea `ttyd0:` mencionada arriba. Cada línea debera contar con su propia dirección IP, de sus direcciones IP disponibles para asignar dinámicamente.

25.2.1.5.3.6. Configurando ppp.conf para Usuarios de IP Estática

Junto con el contenido del fichero de ejemplo /usr/share/examples/ppp/ppp.conf mencionado anteriormente, deberá agregar una sección para cada usuario asignado estáticamente. Continuaremos con nuestro ejemplo con los usuarios fred, sam y mary.

```
fred:
  set ifaddr 203.14.100.1 203.14.101.1 255.255.255.255

sam:
  set ifaddr 203.14.100.1 203.14.102.1 255.255.255.255

mary:
  set ifaddr 203.14.100.1 203.14.103.1 255.255.255.255
```

El archivo /etc/ppp/ppp.linkup deberá de contener también información del ruteo, para cada IP estática, si es necesario. Las líneas a continuación añadirán una ruta a la dirección 203.14.101.0 de clase C, por medio del ppp link del cliente.

```
fred:
  add 203.14.101.0 netmask 255.255.255.0 HISADDR

sam:
  add 203.14.102.0 netmask 255.255.255.0 HISADDR

mary:
  add 203.14.103.0 netmask 255.255.255.0 HISADDR
```

25.2.1.5.4. Algo más de mgetty, AutoPPP, y Extensiones MS

25.2.1.5.4.1. mgetty y AutoPPP

Configurando y compilando mgetty con la opción AUTO_PPP habilitada, permite a mgetty detectar la fase LCP de conexiones PPP y automáticamente enviarlo a un shel de ppp. Aun con esto, y debido a que no se ingresa el nombre de usuario y contraseña, es necesario autntificarse por medio de PAP o CHAP.

Esta sección asume que el usuaio ha configurado, compilado e instalado correctamente una versión de mgetty, con la opción Auto_PPP (v0.99beta o posterior).

Asegurese de que su fichero /usr/local/etc/mgetty+sendfax/login.conf contiene la siguiente línea en él:

```
/AutoPPP/ -     -         /etc/ppp/ppp-pap-dialup
```

Esto le indicará a mgetty que ejecute el script ppp-pap-dialup , para efecto de detectar conexiones de tipo PPP.

Cree un fichero llamado /etc/ppp/ppp-pap-dialup que contenga las siguientes líneas (el fichero deberá ser ejecutable):

```
#!/bin/sh
exec /usr/sbin/ppp -direct pap$IDENT
```

Para cada línea de marcado habilitada en `/etc/ttys` , cree la entrada correspondiente en `/etc/ppp/ppp.conf` . Esto co-existirá pacíficamente con las definiciaones que se hayan hecho, de acuerdo a lo mostrado en la parte de arriba.

```
pap:
  enable pap
  set ifaddr 203.14.100.1 203.14.100.20-203.14.100.40
  enable proxy
```

Cada usuario que ingrese al sistema utilizando este metodo, deberá de contar con su clave de usuario, así como su contraseña, en el archivo `/etc/ppp/ppp.secret` , o bien agregue la siguiente opción, para efecto de que se pueda realizar la autentificación por medio de PAP, directamente del fichero `/etc/password`.

```
enable passwdauth
```

Si desea asignar una dirección IP fija a algunos usuarios, puede especificar el número como un tercer argumento en el fichero `/etc/ppp/ppp.secrets` . Vea el archivo `/usr/share/examples/ppp/ppp.secret.sample` para obtener ejemplos más detallados de esto.

25.2.1.5.4.2. Extensiones de MS

Es posible configurar PPP, para efecto de que brinde a DNS y a NetBIOS, direcciones de servidores de nombres de forma automática.

Para efecto de habilitar estas extensiones con PPP versión 1.x, las siguientes líneas deberán añadirse a la sección relevante de `/etc/ppp/ppp.conf` .

```
enable msext
set ns 203.14.100.1 203.14.100.2
set nbns 203.14.100.5
```

Y para versiones de PPP 2 y posteriores:

```
accept dns
set dns 203.14.100.1 203.14.100.2
set nbns 203.14.100.5
```

Esto le indicará a los clientes, las direcciones del servidor primario y secundario y el servidor-host para NetBIOS.

Si la línea `set dns`, es omitida en versiones 2 y posteriores, PPP utilizará los valores que encuentre en `/etc/resolv.conf` .

25.2.1.5.5. Autentificación por medio de PAP y CHAP

Algunos proveedores de internet tienen su sistema configurado para que cada usuario al conecatrse sean autentificados por medio de PAP o CHAP. Si este es el caso, al momento de realizar la conexion, no aparecerá un `login:`, sino que comenzará a comunicarse PPP inmediatamente.

El metodo PAP es menos seguro que CHAP, pero la seguridad normalmente no se toma mucho en cuenta en este tipo de conexiones, en funcion de que al enviarse la información de contraseña en texto plano, por medio de una línea serial, no deja mucho espacio para que los crackers "husmeen".

Haciendo referencia a lo que vimos de PPP y Direcciones de IP Fijas o bien PPP y Direcciones de IP Dinámicas, habría que aplicar los siguientes cambios:

```
7       set login
…
12      set authname MiNombreDeUsuario
13      set authkey MiContraseña
```

Línea 7:
: Su PSI normalmente requerirá que usted ingrese al sistema, cuando se utiliza PAP o CHAP. Por esta razón debemos deshabilitar la línea que corresponde a "set login".

Línea 12:
: Esta línea especifíca a PAP/CHAP su nombre de usuario. Usted deberá cambiar el valor a quedar el nombre correcto en el campo; *`MiNombreDeUsuario`*.

Línea 13:

: Esta línea especifíca su contraseña de PAP/CHAP. Es necesario que usted cambie el valor a quedar el dato correcto, en el campo; *`MiContraseña`*. Quizás sea recomendable que añada una línea a quedar:

```
15      accept PAP
```

o

```
15      accept CHAP
```

la intención de esto es para hacerlo obvio, aunque en realidad PAP y CHAP son aceptadas por omisión.

25.2.1.5.6. Cambiando la configuración de ppp sobre la marcha (al vuelo)

Es posible hablar con el programa ppp mientras se esta ejecutando en segundo plano, pero sólo si se ha habilitado un puerto de diagnóstico. Para hacer esto, añada lo siguiente a su configuración:

```
set server /var/run/ppp-tun%d DiagnosticPassword 0177
```

Esto le indicará a PPP que preste atención al socket del dominio-Unix, solicitando a los usuarios su contraseña, antes de permitir el acceso. La variable `%d` deberá ser reemplazada por el numero de dispositivo `tun` que este utilizando (ej. `tun0`).

Una vez que se a configurado el socket, se puede utilizar pppctl(8) en scripts que deseen manipular el programa.

25.2.1.6. Configuración Final del Sistema

Ahora usted cuenta con un `ppp` configurado, pero es necesario hacer algunas cosas, antes de que este disponible para trabajar. Todas ellas giran entorno a la edición del fichero `/etc/rc.conf` .

En primer lugar es importante que se asegure que ha asignado un nombre a su maquina. Esto se hace asignandolo en la línea de `hostname=`, por ejemplo:

```
hostname="foo.ejemplo.com"
```

Si su Proveedor de Servicios de Internet (PSI), le ha provisto de una dirección fija y un nombre de host, es recomendable que utilice este como su `hostname`.

Localice la línea que se refiera a sus dispositivos de red, la cual es `network_interfaces` .Si desea configurar su sistema para marcar a su PSI a petición, asegurese de que el dispositivo `tun0` este en la lista, de otra forma eliminelo.

```
network_interfaces="lo0 tun0" ifconfig_tun0=
```

Nota

La variable `ifconfig_tun0` debe permanecer en blanco (vacia), y deberá crearse un fichero llamado `/etc/start_if.tun0` que contenga la siguiente línea:

```
ppp -auto MiSistema
```

Este script se ejecuta cuando se esta configurando la red, inicializando el demonio de ppp de modo automático. Si usted cuenta con una LAN (red de área local), de la cual esta maquina sea la pasarela (gateway), es tambien recomendable que utilice la opción `-alias`. Referirse a la página de ayuda (man) para mayores detalles.

Especifique el programa router a `NO`, con la siguiente línea en su fichero `/etc/rc.conf` :

```
router_enable="NO"
```

Es importante que el demonio `routed` no se inicialice por default, en virtud de que `routed` tiende a eliminar las variables creadas por `ppp`.

Probablemente valga la pena asegurarse de que la línea `sendmail_flags` , no incluya la opción `-q`, ya que de ser así `sendmail` intentará localizar los parámetros de la red de vex en cuando, ocasionando que realice llamados al exterior. Puede intentar esto:

```
sendmail_flags="-bd"
```

La parte negativa de esta conifguración es que tiene que forzar a `sendmail` a re-examinar los llamados del servidor de correo, cada vez que `ppp` realiza una conexión, con el siguiente comando:

```
# /usr/sbin/sendmail -q
```

Puede utilizar el comando `!bg` en el fichero `ppp.linkup` para hacer esto de manera automática:

```
1     provider:
2       delete ALL
3       add 0 0 HISADDR
4       !bg sendmail -bd -q30m
```

Si usted no desea hacer esto, es posible establecer un "dfilter" (filtro), para bloquear el tráfico al servidor de salida de correo (SMTP). Favor de referirse a los archivos de ejemplos para mayor detalle al respecto.

Ahora lo único que queda pendiente de hacerse es reiniciar el equipo. Una vez reiniciado el equipo, puede teclear:

```
# ppp
```

y posteriormente `dial proveedor` para iniciar la sesión, o bien si desea que `ppp` inicie la sesión automáticamente, cuando haya una petición de salida (y no haya creado el fichero `start_if.tun0`), puede teclear:

```
# ppp -auto proveedor
```

25.2.1.7. Summario

A manera de recapitulación, podemos decir que para configurar ppp por primera ocasión, debemos:

Por parte del Cliente:

1. Asegurese de que existe el dispositivo `tun` dentro de su kernel.

2. Asegurses de que el dispositivo `tunX`, se encuentra disponible, bajo el directorio `/dev`.

3. Cree una entrada en su fichero `/etc/ppp/ppp.conf` . Con el fichero de ejemplo `pm-demand` debe ser suficiente para la mayoria de proveedores.

4. Si cuenta con una dirección de IP dinámica, cree una entrada en el fichero `/etc/ppp/ppp.linkup` .

5. Actualice su fichero `/etc/rc.conf` .

6. Cree un archivo script llamado `start_if.tun0` si requiere servicio de conexión a solicitud.

Por parte del Servidor:

1. Asegurese de que dentro de su kernel exista el dispositivo `tun`.

2. Asegurses de que el dispositivo `tunX`, se encuentra disponible, bajo el directorio `/dev`.

3. Cree una entrada en el fichero `/etc/passwd` (usando el programa vipw(8)).

4. Cree un perfil en el directorio home de este usuario, que ejecute `ppp -direct direct-server` o algo similar.

5. Cree una entrada en el fichero `/etc/ppp/ppp.conf` . El fichero de ejemplo `direct-server` debe ser suficiente para darse una idea.

6. Cree una entrada en el fichero `/etc/ppp/ppp.linkup` .

7. Actualice su fichero `/etc/rc.conf` .

25.3. Uso de Kernel PPP

Partes originalmente desarrolladas por Gennady B. Sorokopud y Robert Huff.

25.3.1. Configurando Kernel PPP

Antes de comenzar a configurar PPP en su maquina, asegurese de `pppd` se localiza en `/usr/sbin` y de que existe el directorio `/etc/ppp` .

`pppd` puede trabajar de dos maneras

1. Como un “cliente” — cuando desea conectar su maquina al mundo exterior utilizando PPP, por medio de una conexión serial o bien una línea de modem.

2. como un “servidor” — cuando su maquina esta conectada a una red y es utilizada para que otras maquinas se conecten utilizando ppp.

En ambos casos, será necesario configurar un fichero de opciones (`/etc/ppp/options` o bien `~/.ppprc` si se cuenta con más de un usuario que utilizará ppp en la misma maquina.

También deberá de contar con un software para hacer la conexión por medio de módem (de preferencia kermit), de manera que pueda hacer la conexión con un host remoto.

25.3.2. Uso de `pppd` como Cliente

Basedo en información provista por Trev Roydhouse.

El siguiente archivo de configuración `/etc/ppp/options` puede utilizarse para realizar la conexión a una terminal CISCO, por medio de PPP.

```
crtscts         # habilita el flujo de controls de hardware
modem           # línea de control del modem
noipdefault     # el servidor PPP remoto asignará la dirección IP
                # si el servidor no envia una dirección IP durante ↺
IPCP
                # remueva esta opción.
passive         # espere por los paquetes LCP
domain ppp.foo.com      # escriba su nombre de dominio aqui

:<remote_ip>    # escriba la IP del host remoto aqui
                # este será utilizado para el ruteo de paquetes por ↺
medio
                # de PPP, si no especifica esta opción, cambie la
                # línea a quedar  <local_ip>:<remote_ip>

defaultroute    # establezca esta opción si el servidor su ruteador
                # por default
```

Para conectarse:

1. Realice el llamado al host remoto, utilizando kermit (o cualquier otra aplicación de este tipo), ingrese su nombre de usuario y contraseña (o cualquier info que sea necesaria para habilitar PPP en el host remoto).

2. Salga de kermit (sin colgar la línea).

3. Ingrese lo siguiente:

   ```
   # /usr/src/usr.sbin/pppd.new/pppd /dev/tty01 19200
   ```

   Asegurese de utilizar el dispositivo y la velocidad adecuados.

Ahora su computadora esta conectada por medio de PPP. Si la conexión falla, puede añadir la opción `debug` en el fichero `/etc/ppp/options` de tal forma que pueda verificar la que esta ocurriendo y pueda resolver el problema.

El siguiente script; `/etc/ppp/pppup` realizará los 3 pasos de forma automática:

```
#!/bin/sh
ps ax |grep pppd |grep -v grep
pid=`ps ax |grep pppd |grep -v grep|awk '{print $1;}'`
if [ "X${pid}" != "X" -] -; then
        echo 'killing pppd, PID=' ${pid}
        kill ${pid}
fi
ps ax |grep kermit |grep -v grep
pid=`ps ax |grep kermit |grep -v grep|awk '{print $1;}'`
if [ "X${pid}" != "X" -] -; then
        echo 'killing kermit, PID=' ${pid}
        kill -9 ${pid}
fi

ifconfig ppp0 down
ifconfig ppp0 delete

kermit -y /etc/ppp/kermit.dial
pppd /dev/tty01 19200
```

El fichero `/etc/ppp/kermit.dial` es un script de kermit, uqe realiza el marcado y negocia la autorización necesaria con el host remoto (un ejemplo de este script se encuentra al final de este documento).

Utilice el siguiente script, llamado `/etc/ppp/pppdown` para desconectar la línea PPP:

```
#!/bin/sh
pid=`ps ax |grep pppd |grep -v grep|awk '{print $1;}'`
if [ X${pid} != "X" -] -; then
        echo 'killing pppd, PID=' ${pid}
        kill -TERM ${pid}
fi

ps ax |grep kermit |grep -v grep
pid=`ps ax |grep kermit |grep -v grep|awk '{print $1;}'`
if [ "X${pid}" != "X" -] -; then
        echo 'killing kermit, PID=' ${pid}
        kill -9 ${pid}
fi

/sbin/ifconfig ppp0 down
/sbin/ifconfig ppp0 delete
kermit -y /etc/ppp/kermit.hup
/etc/ppp/ppptest
```

Verifique que su PPP aun se esta ejecutando, por medio de `/usr/etc/ppp/ppptest` , que deberá verse algo similar a esto:

```
#!/bin/sh
pid=`ps ax| grep pppd |grep -v grep|awk '{print $1;}'`
if [ X${pid} != "X" -] -; then
```

```
        echo 'pppd running: PID=' ${pid-NONE}
else
        echo 'No pppd running.'
fi
set -x
netstat -n -I ppp0
ifconfig ppp0
```

Para colgar el módem, ejecute `/etc/ppp/kermit.hup` , que deberá contener:

```
set line /dev/tty01 ; aqui va el dispositivo del modem
set speed 19200
set file type binary
set file names literal
set win 8
set rec pack 1024
set send pack 1024
set block 3
set term bytesize 8
set command bytesize 8
set flow none

pau 1
out +++
inp 5 OK
out ATH0\13
echo \13
exit
```

He aqui un metodo alterno, donde se utiliza `chat` en lugar de utilizar `kermit`.

Los siguientes dos ficheros deben ser suficiente, para realizar una conexión por medio de `pppd`.

```
/dev/cuaa1 115200

crtscts  # habilita el control de flujo por medio de hardware
modem  # línea de control del módem
connect "/usr/bin/chat -f /etc/ppp/login.chat.script"
noipdefault # el servidor remoto debe asignar la dirección IP.
         # si el servidor no asigna una IP durante la negociación
                # IPCP , remueva esta línea y espere por los
passive         # paquetes LCP
domain <your.domain> # aqui va su dominio

:  # escriba la IP del host remoto aqui
                # si no ha especificado la opción noipdefault
                # cambie esta línea a quedar <local_ip>:<remote_ip>

defaultroute # escriba esto, si desea que el servidor PPP sea su
             # router por default
```

`/etc/ppp/login.chat.script` :

Nota

Lo siguiente debe ir en una sola línea.

```
ABORT BUSY ABORT 'NO CARRIER' "" AT OK ATDT<numero.de.telefono>
  CONNECT "" TIMEOUT 10 ogin:-\\r-ogin: <nombre.usuario>
  TIMEOUT 5 sword: <contraseña>
```

Una vez que estos ficheros han sido modificados correctamente e instalados, todo lo que necesita es ejecutar el comando `pppd`, algo como:

```
# pppd
```

25.3.3. Uso de pppd como Servidor

El fichero `/etc/ppp/options` debe contener algo similar a lo siguiente:

```
crtscts   # control de flujo por Hardware
netmask 255.255.255.0          # mascara de red (no es requisito)
192.114.208.20:192.114.208.165  # direcciones ip del host local y ↺
remoto
                               # la dirección ip local debe ser
                               # diferente a la que le haya ↺
asignado a su
                               # dispositivo de red ethernet (u ↺
otro)
                               # la dirección ip remota que será
                               # asignada a la maquina remota
domain ppp.foo.com             # su dominio
passive                        # espera por LCP
modem                          # línea de modem
```

El siguiengte script, llamado `/etc/ppp/pppserv` habilitará pppd, para que actue como servidor:

```
#!/bin/sh
ps ax |grep pppd |grep -v grep
pid=`ps ax |grep pppd |grep -v grep|awk '{print $1;}'`
if [ "X${pid}" != "X" -] -; then
        echo 'killing pppd, PID=' ${pid}
        kill ${pid}
fi
ps ax |grep kermit |grep -v grep
pid=`ps ax |grep kermit |grep -v grep|awk '{print $1;}'`
if [ "X${pid}" != "X" -] -; then
        echo 'killing kermit, PID=' ${pid}
        kill -9 ${pid}
```

```
fi

# reset ppp interface
ifconfig ppp0 down
ifconfig ppp0 delete

# enable autoanswer mode
kermit -y /etc/ppp/kermit.ans

# run ppp
pppd /dev/tty01 19200
```

Utilice el script `/etc/ppp/pppservdown` para detener el servidor:

```
#!/bin/sh
ps ax |grep pppd |grep -v grep
pid=`ps ax |grep pppd |grep -v grep|awk '{print $1;}'`
if [ "X${pid}" != "X" -] -; then
        echo 'killing pppd, PID=' ${pid}
        kill ${pid}
fi
ps ax |grep kermit |grep -v grep
pid=`ps ax |grep kermit |grep -v grep|awk '{print $1;}'`
if [ "X${pid}" != "X" -] -; then
        echo 'killing kermit, PID=' ${pid}
        kill -9 ${pid}
fi
ifconfig ppp0 down
ifconfig ppp0 delete

kermit -y /etc/ppp/kermit.noans
```

El siguiente script de kermit (`/etc/ppp/kermit.ans`) habilita/deshabilita el modo de autorespuesta en su módem. Y debe verse algo similar a lo siguiente:

```
set line /dev/tty01
set speed 19200
set file type binary
set file names literal
set win 8
set rec pack 1024
set send pack 1024
set block 3
set term bytesize 8
set command bytesize 8
set flow none

pau 1
out +++
inp 5 OK
out ATH0\13
inp 5 OK
echo \13
```

```
out ATS0=1\13   -; cambiar esto a quedar out ATS0=0\13 si desea ↺
deshabilitar el modo
                -; de autorespuesta
inp 5 OK
echo \13
exit
```

Un script llamado `/etc/ppp/kermit.dial` es utilizado para llamar y autentificarse en un host remoto. Es necesario que edite este fichero, de acuerdo a sus necesidades. Escriba su nombre de usuario (login) y contraseña (password) en este fichero, también será necesario cambiar su metodo de conexión, de acuerdo a lo que se ajuste a sus necesidades.

```
;
; ingrese el dispositivo que esta apuntando a su módem:
;
set line /dev/tty01
;
; escriba la velocidad del módem:
;
set speed 19200
set file type binary            -; full 8 bit file xfer
set file names literal
set win 8
set rec pack 1024
set send pack 1024
set block 3
set term bytesize 8
set command bytesize 8
set flow none
set modem hayes
set dial hangup off
set carrier auto                -; Posteriormente SET CARRIER si es ↺
necesario
set dial display on             -; despues SET DIAL si es necesario
set input echo on
set input timeout proceed
set input case ignore
def \%x 0
goto slhup

:slcmd                          -; cambio a modo de comando
echo Put the modem in command mode.
clear                           -; Limpieza del buffer de entrada
pause 1
output +++
input 1 OK\13\10                -; esperar para OK
if success goto slhup
output \13
pause 1
output at\13
input 1 OK\13\10
if fail goto slcmd              -; si el modem no responde Ok, ↺
intentar de nuevo
```

```
:slhup                          -; colgar el teléfono
clear                           -; Limpieza del buffer de entrada
pause 1
echo Hanging up the phone.
output ath0\13
input 2 OK\13\10
if fail goto slcmd              -; si no hay un OK como respuesta, ↺
poner el modem en modo de comando

:sldial                         -; marcar el numero telefonico
pause 1
echo Dialing.
output atdt9,550311\13\10               -; escriba el numero de ↺
telefono
assign \%x 0                    -; asignar cero al contador

:look
clear                           -; Limpieza del buffer de entrada
increment \%x                   -; Conteo de segundos
input 1 {CONNECT }
if success goto sllogin
reinput 1 {NO CARRIER\13\10}
if success goto sldial
reinput 1 {NO DIALTONE\13\10}
if success goto slnodial
reinput 1 {\255}
if success goto slhup
reinput 1 {\127}
if success goto slhup
if < \%x 60 goto look
else goto slhup

:sllogin                        -; login
assign \%x 0                    -; asignar cero al contador
pause 1
echo Looking for login prompt.

:slloop
increment \%x                   -; Conteo de segundos
clear                           -; Limpieza del buffer de entrada
output \13
;
; escriba su login prompt aqui:
;
input 1 {Username: }
if success goto sluid
reinput 1 {\255}
if success goto slhup
reinput 1 {\127}
if success goto slhup
if < \%x 10 goto slloop         -; intentar 10 veces para obtener ↺
un login
```

```
else goto slhup                 -; colgar y empezar de nuevo si a ↺
la decima falla

:sluid
;
; escriba su nombre de usuario:
;
output ppp-login\13
input 1 {Password: }
;
; escriba su contraseña:
;
output ppp-password\13
input 1 {Entering SLIP mode.-}
echo
quit

:slnodial
echo \7No dialtone.  Check the telephone line!\7
exit 1

; local variables:
; mode: csh
; comment-start: "; "
; comment-start-skip: "; "
; end:
```

25.4. Uso de PPP sobre Ethernet (PPPoE)

Contribuido (tomado de http://node.to/freebsd/how-tos/how-to-freebsd-pppoe.html) por Jim Mock.

En esta sección veremos como configurar PPP sobre una red Ethernet (PPPoE).

25.4.1. Configurando el kernel

Ya no es necesario realizar una configuración especial para que nuestro kernel cuente con soporte para PPPoE. Siempre y cuando el soporte de redes necesario se encuentre en él, ppp se encargará de cargarlo de una manera dinámica.

25.4.2. Editando el fichero `ppp.conf`

He aqui un ejemplo de un fichero de configuración `ppp.conf` completamente funcional:

```
default:
  set log Phase tun command # puede añadir más dispositivos si lo ↺
desea
  set ifaddr 10.0.0.1/0 10.0.0.2/0

nombre_del_proveedor_del_servicio_de_internet:
```

```
  set device PPPoE:xl1 # sustituya xl1 con su dispositivo ethernet
  set authname SuNombreDeUsuario
  set authkey SuContraseña
  set dial
  set login
  add default HISADDR
```

25.4.3. Ejecutando PPP

Estando en modo `superusuario` (root) puede ejecutar:

```
# ppp -ddial nombre_del_proveedor_de_inet
```

25.4.4. Ejecutando PPP al inicio de sesión

Añada las siguientes líneas a su archivo `/etc/rc.conf` :

```
ppp_enable="YES"
ppp_mode="ddial"
ppp_nat="YES" # siempre y cuando desee habilitar nat para su red ↺
local
ppp_profile="nombre_del_proveedor_de_inet"
```

25.4.5. Diferenciando el uso del Servicio de PPPoE

En ocasiones es necesario utilizar una pequeña marca para diferenciar el servicio que vamos a utilizar para establecer la conexión. Las marcas (“tags”)de servicio son utilizadas para distinguir entre diferentes servidores de una red, a los que nos podemos conectar utilizando PPPoE.

Su proveedor de internet debe haberle provisto de la información necesaria para crear esta marca. Si esto no fué así, puede solicitar a su proveedor que le brinde esta información.

Como último recurso, puede intentar el método sugerido por el programa Roarging Penguin PPPoE, que puede encontrarse en la colección de ports. Al utilizar este programa debe tener en mente, que este puede desconfigurar su módem por completo, por esta razón piense biena antes de utilizarlo. Simplemente instale el programa controlador del módem, provisto por su porveedor. Posteriormente, debe acceder al menú de `Sistema` del programa. El nombre de su perfil debe aparecer listado. Que normalmente es *ISP*.

El nombre del perfil (marca del servicio) será utilizada por la configuración de PPPoE en el fichero de configuración `ppp.conf` como el proveedor para la opción del comando `set device` (puede ver la página de ayuda ppp(8) para más detalles). Esto debe verse algo similar a lo siguiente:

```
set device PPPoE:xl1:ISP
```

No olvide cambiar *x11* por el dispositivo Ethernet que este utilizando.

No olvide cambiar *ISP* por el nombre del perfil que le fué descrito anteriormente (por lo general el nombre de su Proveedor de Servicio de Internet).

Para información adicional consulte:

- Cheaper Broadband with FreeBSD on DSL por Renauld Waldura.

25.4.6. Uso de PPPoE en Casa con un Modem Dual ADSL 3Com

Este módem no sigue el estandar establecido en el RFC 2516 (*Un metodo que describe el uso de PPP por medio de un dispositivo Ethernet (PPoE)*, escrito por L. Mamakos, K. Lidl, J. Evarts, D. Carrel, D. Simone y R. Wheeler). En su lugar, el código de diferentes tipos de paquetes ha sido utilizado para el manejo del entorno Ethernet. Si cree que esto es incorrecto y que se debiera ajustar a las especificaciones de PPPoE, por favor comentelo en 3Com.

Para poder hacer que FreeBSD sea capaz de comunicarse con este dispositivo, se debe establecer un control de sistema (sysctl). Esto puede hacerse de forma automática al momento del arranque, editando el fichero `/etc/sysctl.conf` :

```
net.graph.nonstandard_pppoe=1
```

o bien pude hacerse desde la línea de comandos, para un efecto inmediato, por medio del comando `sysctl -w net.graph.nonstandard_pppoe=1` .

Desafortunadamente y dado que esto implica una configuración general del sistema, por lo que no es posible comunicarnos con un dispositivo cliente — servidor que utilice PPPoE y con un módem casero 3Com ADSL, al mismo tiempo.

25.5. Uso de PPP sobre ATM (PPPoA)

Lo siguiente describe como configurar PPP utilizando ATM, alias PPPoA. PPPoA es una alternativa muy común entre proveedores de DSL en Europa.

25.5.1. Uso de PPPoA con un Alcatel Speedtouch USB

El soporte bajo FreeBSD para este dispositivo se puede encontrar como un port, por que el firmware es distribuido bajo la licencia de Alcatel.

Para instalar este software, simplemente utilice la colección de ports. Instale el port net/pppoa y siga las instrucciones provistas por el port.

25.5.2. Uso de mpd

Puede usar mpd para conectarse a una gran variedad de servicios, en particular servicios pptp. Puede encontrar mpd en la colección de ports, bajo net/mpd.

Primero debe instalar el port, y posteriormente configurar mpd para que se ajuste a sus necesidades y a la configuración del proveedor. El port instala un conjunto de ficheros de configuración de ejemplo, que estan bien documentados en `PREFIX/etc/mpd/` . Note que *PREFIX* se refiere al directorio donde sus ports son instalados, que normalmente es en `/usr/local` . Una guía completa en formato HTML, esta disponible una vez que se ha instalado el port. Esta se localiza en `PREFIX/share/mpd/` . Aqui tenemos un ejemplo simple de configuración para conectarse a un servicio ADSL con mpd. La configuración se divide en dos ficheros, primero tenemos el fichero `mpd.conf` .

```
default:
     load adsl

 adsl:
     new -i ng0 adsl adsl
     set bundle authname usuario ❶
     set bundle password contraseña ❷
     set bundle disable multilink

     set link no pap actcomp protocomp
     set link disable chap
     set link accept chap
     set link keep-alive 30 10

     set ipcp no vjcomp
     set ipcp ranges 0.0.0.0/0 0.0.0.0/0

     set iface route default
     set iface disable on-demand
     set iface enable proxy-arp
     set iface idle 0

     open
```

❶ El nombre de usuario para autentificar con su proveedor.
❷ La contraseña para autentificar con su proveedor.

El fichero `mpd.links` contiene información a cerca de la, o las conecciones, que desee establecer. Un ejemplo de `mpd.links` y que sea acompañante del ejemplo anterior, se muestra a continuación.

```
adsl:
     set link type pptp
     set pptp mode active
     set pptp enable originate incoming outcall
     set pptp self 10.0.0.140
     set pptp peer 10.0.0.138
```

La conexión es fácil de inicializarla, al ingresar los siguientes comandos como `root`.

```
# mpd -b adsl
```

El estatus de la conexión la puede ver con el comando.

```
% ifconfig ng0
 : flags=88d1<UP,POINTOPOINT,RUNNING,NOARP,SIMPLEX,MULTICAST> mtu ↺
1500
      inet 216.136.204.117 --> 204.152.186.171 netmask 0xffffffff
```

Usar mpd es la forma recomendada para conectarse con servicios ADSL con FreeBSD.

25.5.3. Uso de pptpclient

También es posible usar FreeBSD para conectarse a otros servicios PPPoA por medio de net/pptpclient.

Para conectarse por medio de net/pptpclient a un servicio DSL, instale el port o paquete y edite el fichero `/etc/ppp/ppp.conf` . Debe ser `root` para hacer estas operaciones. Un ejemplo de la sección de `ppp.conf` , se muestra a continuación. Para mayor información sobre las opciones de `ppp.conf` , consulte la página de ayuda de ppp; ppp(8).

```
adsl:
  set log phase chat lcp ipcp ccp tun command
  set timeout 0
  enable dns
  set authname usuario ❶
  set authkey contraseña ❷
  set ifaddr 0 0
  add default HISADDR
```

❶ Nombre de usuario de la cuenta DSL.
❷ La contraseña de su cuenta.

Aviso

Debido a que debe poner su contraseña en el fichero `ppp.conf` en texto plano, debe asegurarse que nadie tenga acceso de lectura a este fichero. Los siguientes comandos se aseguran de que el fichero solo pueda se leido por `root`. Ve las páginas de ayuda chmod(1) y chown(8) para mayor información.

```
# chown root:wheel /etc/ppp/ppp.conf
 # chmod 600 /etc/ppp/ppp.conf
```

Esto abrirá una sesion por medio de PPP con su ruteador DSL. Los módems Ethernet DSL cuentan con una dirección IP de LAN preconfigurada a la cual se puede conectar. En el caso del Alcatel Speedtouch, esta dirección es `10.0.0.138` . La documentación de su equipo

debe indicarle que dirección utiliza. Para abrir el "tunel" e iniciar la sesión ppp, ejecute el siguiente comando.

```
# pptp dirección proveedor
```

Sugerencia

Puede añadir un símbolo de ampersand ("&") al final de este comando, ya que pptp no retorna al shell por default.

Un dispositivo virtual `tun` será creado, para interactuar con los procesos de pptp y ppp. Una vez que regrese al shell puede examinar la conexión por medio del siguiente comando.

```
% ifconfig tun0
 tun0: flags=8051<UP,POINTOPOINT,RUNNING,MULTICAST> mtu 1500
         inet 216.136.204.21 --> 204.152.186.171 netmask 0xffffff00
         Opened by PID 918
```

Si no le es posible conectarse, verifique la configuración de su ruteador, que normalmente es accesible por medio de telnet o de su navegador web. Si aun no puede conectarse examine la salida que da el comando pptp y el contenido del fichero de registro (log) de ppp; `/var/log/ppp.log` .

25.6. Uso de SLIP

Contribución Original por Satoshi Asami.
Con comentarios por Guy Helmer y Piero Serini.

25.6.1. Configurando un cliente SLIP

Lo siguiente es una forma de configurar FreeBSD para que utilice SLIP, en un red con dirección estática. Para direcciones dinámicas (esto es, donde su dirección cambia cada vez que se conecta), probablemente sea necesario realizar algunos ajustes que complican la configuración.

En primer término, es necesario determinar a que puerto serial esta conectado nuestro módem. Mucha gente opta por} contar con un enláce simbólico, tal como `/dev//modem` , que apunta al nombre real del dispositivo, `/dev/cuaaN` . Esto permite abstenerse de usar el nombre real del dispositivo, en caso de que sea necesario cambiar de puerto nuestro módem. Lo cual puede ser de mucha ayuda, ya que puede ser un fastidio tener que editar un monton de ficheros en `/etc` y ficheros de tipo `.kermrc` en todo el sistema!.

Nota

`/dev/cuaa0` es `COM1`, `cuaa1` es `COM2`, etc.

Asegurese de contar con la siguiente opción en la configuración de su kernel:

```
pseudo-device   sl      1
```

Esta opción esta incluida en el archivo del kernel `GENERIC`, así que no debe haber problema, claro esta, a menos que lo haya borrado intencionalmente.

25.6.1.1. Cosas Que Tiene Que Hacer Solo Una Vez

1. Añada el nombre de su maquina, gateway, servidores de nombre a su fichero `/etc/hosts`. Este es un ejemplo de mi fichero:

   ```
   127.0.0.1               localhost loghost
   136.152.64.181          water.CS.Example.EDU water.CS water
   136.152.64.1            inr-3.CS.Example.EDU inr-3 slip-gateway
   128.32.136.9            ns1.Example.EDU ns1
   128.32.136.12           ns2.Example.EDU ns2
   ```

2. Asegurese de que cuenta con la opción `hosts` antes de la opción `bind`, en su fichero `/etc/host.conf` . De lo contrario pueden ocurrir cosas graciosas en su sistema.

3. Edite el fichero `/etc/rc.conf` .

   1. Especifique su nombre host al editar la línea que dice:

      ```
      hostname=“minombre.mi.dominio”
      ```

      El nombre completo de su sistema para internet, debe ser escrito en este punto.

   2. Añada el dispositivo `sl0` a la lista de dispositivos de red, al cambiar la línea que dice:

      ```
      network_interfaces="lo0"
      ```

      a quedar:

      ```
      network_interfaces=“lo0 sl0”
      ```

   3. Añada los parámetros de inicialización del dispositivo sl0, al añadir la línea:

      ```
      ifconfig_sl0="inet ${hostname} slip-gateway netmask ↺
      0xffffff00 up"
      ```

4. Especificque cual será su ruteador por omisión al editar la línea:

```
defaultrouter="NO"
```

a quedar:

```
defaultrouter="slip-gateway"
```

4. Edite su fichero `/etc/resolv.conf` (si no existe debe crearlo), a que contenga lo siguiente:

```
domain CS.Ejemplo.EDU
nameserver 128.32.136.9
nameserver 128.32.136.12
```

Como puede ver, lo anterior define el nombre de host, de su servidor de nombres. Claro esta, el nombre de dominio y las direcciones IP, dependen de su sistema específico.

5. Establezca la contraseña del superusuario `root` y de su símil `toor` (y de cualquier otro usuario que aun no cuente con la misma).

6. Reinicie su sistema y asegurese que cuenta con el nombre de host (hostname) correcto.

25.6.1.2. Haciendo una Conexión con SLIP

1. Marque el número, teclee en el signo de comando `slip`, ingrese el nombre y la contraseña. Lo que se requiere ingresar, depende de su sistema. Si utiliza kermit, puede utilizar un script similar al siguiente:

```
# kermit setup
set modem hayes
set line /dev/modem
set speed 115200
set parity none
set flow rts/cts
set terminal bytesize 8
set file type binary
# El siguiente macro se encarga de llamar e ingresar al sistema
define slip dial 643-9600, input 10 =>, if failure stop, -
output slip\x0d, input 10 Username:, if failure stop, -
output silvia\x0d, input 10 Password:, if failure stop, -
output ***\x0d, echo \x0aCONNECTED\x0a
```

Claro esta, que debe cambiar el nombre y contraseñ a quedar de acuerdo a sus necesidades. Después de hacer esto, puede simplemente teclear `slip` en el símbolo de sistema (prompt) de kermit, para realizar la conexión.

Nota

El dejar su contraseña en texto plano, en cualquier parte del sistema, generalmente es una *mala* idea. Hágalo bajo su propio riesgo.

2. Dejé a kermit en ese punto trabajando (puede suspenderlo tecleando Ctrl+z) y como `root`, teclee:

```
# slattach -h -c -s 115200 /dev/modem
```

Si puede hacer `ping` a cualquier host que se encuentre del otro lado del ruteador, usted esta conectado!. Si esto no funciona, puede intentar como argumento del comando `slattach`, la opción `-a` en lugar de utilizar la optción `-c`.

25.6.1.3. Como Terminar la Conexión

Para terminar la conexión haga lo siguiente:

```
# kill -INT `cat /var/run/slattach.modem.pid`
```

esto terminaraá `slattach`. Recuerde que para hacer esto, usted debe estar firmado como superusuario (root). Posteriormente dirijase a kermit (puede hacer esto con `fg` si lo envio a segundo plano) y salga (tecleando `q`).

La página de ayuda de `slattach` indica que debe utilizar el comando `ifconfig sl0 down`, para marcar como terminado el uso del dispositivo, pero tal parece que esto no hace una gran diferencia para mi. (`ifconfig sl0` da el mismo resultado.)

En algunas ocasiones, puede que su módem se niegue a cortar la comunicación (el mio lo hace a veces). Si ese es el caso, simplemente inicie de nuevo kermit y vuelva a salir. Normalmente en el segundo intento hay exito.

25.6.1.4. Problemas Comunes

Si esto no funciona, sientase libre de preguntarme. Lo siguiente es una recapitulación de los problemas que más comunmente se presentan:

- El no utilizar la opción `-c` o `-a` con el comando `slattach` (Esto debiera se fatal, pero algunos usuarios han reportado que esto ha solucionado sus problemas.
- Usar la opción `s10` en vez de usar la opción `sl0` (puede ser difícil ver la diferencia con algunos tipos de letras).

- Intente `ifconfig sl0` para visualizar el estatus de sus dispositivos de red. Por ejemplo, puede ser que obtenga algo similiar a lo siguiente:

```
# ifconfig sl0
sl0: flags=10<POINTOPOINT>
        inet 136.152.64.181 --> 136.152.64.1 netmask ffffff00
```

- También el comando `netstat -r` le mostrará la tabla de ruteo, en caso de que obtenga el mensaje "no route to te host", al hacer `ping`. Un ejemplo de esto se muestra a continuación:

```
# netstat -r
Routing tables
Destination      Gateway            Flags     Refs     Use  IfaceMTU
    Rtt    Netmasks:

(root node)
(root node)

Route Tree for Protocol Family inet:
(root node) =>
default          inr-3.Example.EDU  UG          8   224515  sl0 -
    -
localhost.Exampl localhost.Example. UH          5    42127  lo0 -
     0.438
inr-3.Example.ED water.CS.Example.E UH          1        0  sl0 -
    -
water.CS.Example localhost.Example. UGH        34 47641234  lo0 -
     0.438
(root node)
```

  Esto es después de que el sistema ha estado conectado por un tiempo. Los numeros pueden variar en su sistema.

25.6.2. Estableciendo un Servidor SLIP

Este documento provee sugerencias, para establecer un servidor de SLIP, bajo FreeBSD, lo que generalmente significa configurar su sistema, para que de manera automática inicie los servicios, al firmarse usuarios—clientes remotos en su sistema.

25.6.2.1. Prerequisitos

Esta sección es de naturaleza muy técnica, así que contar con antecedentes sobre esto es requerido. Este documento supone que usted cuenta con conocimientos sobre el protocolo de redes TCP/IP, y particularmente, redes y direcciones de nodos, mascaras de red, subneteo, ruteo y protocolos de ruteo, tal como RIP. El configurar servicios SLIP, en un servidor, requiere un conocimiento de estos conceptos, y si usted no esta familiarizado con estos, puede leer una copia, ya sea del libro de Craig Hunt; *TCP/IP Networking Adminis-*

tration, publicado por O'Reilly & Associates, Inc. (Numero ISBN 0-937175-82-X), o alguno de los libros de Douglas Comer, sobre protocolos TCP/IP.

Se da por un hecho, que usted ha instalado y configurado correctamente su(s) módem(s), así como la configuración de su sistema, para efecto de utilizar el mismo para realizar la conexión. Si usted no lo ha hecho, por favor lea el tutorial sobre configuración de estos servicios; si cuenta con un navegador para la World-Wide Web, puede ver los tutoriales disponibles en http://www.FreeBSD.org/.

Puede ser que también desee revisar las páginas de ayuda (*man*), sio(4) para información referente a los controladores de dispositivos de puertos en serie, y ttys(5), gettytab(5), getty(8), & init(8), para ver información relevante, sobre la configuración de su sistema, para efecto de que acepte accesos (logins) por medio de un módem, y quizás stty(1) para información sobre los parámetros de configuración de puertos en serie (tal como `clocal`, que se utiliza para la conexión directa por medio de puertos seriales).

25.6.2.2. Echemos un Vistazo

En su configuración típica, el desarrollo de FreeBSD como un servidor SLIP, funciona de la siguiente manera: un Usuario SLIP, se conecta del Servidor SLIP FreeBSD e ingresa al sistema con una identificación especial, que utiliza `/usr/sbin/sliplogin` como shell del usuario. El programa `sliplogin` busca en el fichero `/etc/sliphome/slip.hosts` la línea que haya sido creada especialmente para el usuario, conecta la línea serial a una interfaz SLIP disponible y posteriormente ejecuta el script `/etc/sliphome/slip.login` , para configurar la interfaz SLIP.

25.6.2.2.1. Un Ejemplo de Acceso al Servidor SLIP

Por ejemplo si la clave de acceso de un usuario SLIP fuese `Shelmerg`, la entrada del usuario `Shelmerg`, en el fichero `/etc/master.passwd` se vera algo similar a lo siguiente:

```
Shelmerg:password:1964:89::0:0:Guy Helmer - SLIP:/usr/users/
Shelmerg:/usr/sbin/sliplogin
```

Cuando `Shelmerg` accese al sistema, el comando `sliplogin`, buscará en el fichero `/etc/sliphome/slip.hosts` , una línea, en la cual el ID (identificación) del usuario coincida, por ejemplo, puede ser que en el fichero `/etc/sliphome/slip.hosts` exista una línea simliar a la siguiente:

```
Shelmerg        dc-slip sl-helmer       0xfffffc00      autocomp
```

El comando `sliplogin` encontrará la línea que coincida, enganchará la línea serial a cualquier interfaz SLIP disponible y posteriormente ejecutará `/etc/sliphome/slip.login` de manera similar a:

```
/etc/sliphome/slip.login 0 19200 Shelmerg dc-slip sl-helmer ↺
0xfffffc00 autocomp
```

Si todo marcha bien, `/etc/sliphome/slip.login` creará una configuración, por medio de `ifconfig`, para la interfaz SLIP, a la cual `sliplogin` se ha adjuntado (la interfaz slip `0`, que era el primer parámetro dado en la lista de `slip.login`), para establecer la dirección local IP (`dc-slip`), la interfaz de la direción IP Remota (`sl-helmer`), la submascara de red para la interfaz SLIP (`0xffffc00`) y cualquier otra opción adicional (`autocomp`). Si algo no va del todo bien, normalmente `sliplogin` guarda bastante información para depurar, por medio del `demonio` (`daemon`) syslog, que usualmente guarda dicha infomración en `/var/log/messages` (vea la página de ayuda syslogd(8) así como syslog.conf(5) y quizas el fichero `/etc/syslog.conf` , para ver que es lo que `syslogd` esta almacenando y donde es que lo almacena.

OK, basta de ejemplos — entremos de lleno en la configuración del sistema.

25.6.2.3. Configuración del Kernel

El kernel de FreeBSD, por omisión, cuenta con 2 dispositivos SLIP definidos (`sl0` y `sl1`); usted puede utilizar `netstat -i`, para verificar si estos dispositivos se encuentran en el kernel de su sistema.

Un ejemplo del resultado de `netstat -i`:

```
Name  Mtu   Network     Address              Ipkts Ierrs    Opkts Oerrs
  Coll
ed0   1500  <Link>0.0.c0.2c.5f.4a         291311     0   174209     ↺
0   133
ed0   1500  138.247.224 ivory            291311     0   174209     ↺
0   133
lo0   65535 <Link>                           79     0       79     ↺
0     0
lo0   65535 loop        localhost            79     0       79     ↺
0     0
sl0*  296   <Link>                          0     0        0     ↺
0     0
sl1*  296   <Link>                          0     0        0     ↺
0     0
```

En este ejemplo vemos que existen dos dispositivos SLIP en el kernel, que son; `sl0` y `sl1` (el asterisco que aparece después de `sl0` y `sl1` indica que los dispositivos no estan “trabajando”.)

Aun cuando el kernel cuente con los dispositivos, por omisión el kernel de FreeBSD, no viene configurado para enviar paquetes (de hecho su sistema FreeBSD no trabajara como ruteador, por default) esto en base a los requerimientos para Internet, establecidos por los RFCs (vea 1009 [Requerimentos para Pasarelas (Gateway) en Internet], 1122 [Requerimientos para hosts de Internet — Capas de comunicación] y quizás 1127 [RFC sobre Una Perspectiva de los Requerimientos de Hosts]). Si usted desea que su servidor SLIP sobre FreeBSD, opere como un ruteador, será necesario que edite el fichero `/etc/rc.conf` y cambie la opción `gateway_enable` , a quedar `YES`, esto habilitará esta función.

Será necesario que reinicie su sistema, para efecto de que estos cambios surtan efecto.

Al verificar su fichero de configuración del kernel (`/sys/i386/conf/GENERIC`), podrá notar que cerca del final, hay una línea como la siguiente:

```
pseudo-device sl 2
```

Esta línea es la que define el numero de dispositivos SLIP disponibles en el kernel; el numero al final de la línea es el numero máximo de conecciones SLIP que puede manejar el servidor simultaneamente.

Para ayuda con relación a la configuración y compilación del kernel en su sistema FreeBSD, por favor refierase Capítulo 8, *Configuración del kernel de FreeBSD* al apartado correspondiente.

25.6.2.4. Configuración de Sliplogin

Como se menciono anteriormente, existen tres ficheros en el directorio `/etc/sliphome` , que son parte de la configuración de `/usr/sbin/sliplogin` (vea la pagina de ayuda sliplogin(8) de para ver la ayuda del comando `sliplogin`): `slip.hosts` , que es el fichero que define a los usuarios SLIP, así como sus direcciones IP correspondientes; `slip.login` , que normalmente es utilizado para configurar la interfaz de SLIP; y (opcionalmente) `slip.logout` , que hace lo opuesto a `slip.login` , cuando la conexión serial ha terminado.

25.6.2.4.1. Configuració de `slip.hosts`

El fichero `/etc/sliphome/slip.hosts` contiene líneas, que al menos cuentan con cuatro partes, separadas por espacios en blanco:

- Identificador (nombre) del usuario SLIP
- Dirección Local (local para el servidor SLIP) de la liga a SLIP
- Dirección Remota de la liga a SLIP
- Mascara de red

Las direcciones local y remota, pueden ser nombres del host (la resolución de los mismos, es llevada a cabo, por medio de `/etc/hosts` o por el servidor de nombres de dominio (DNS), dependiendo de lo que haya especificado en el fichero `/etc/host.conf`), y la mascara de red puede ser un nombre, que puede ser resuelto revisando `/etc/networks` . En un sistema de ejemplo, el fichero `/etc/sliphome/slip.hosts` , puede verse así:

```
#
# login local-addr      remote-addr     mask            opt1    opt2
#                                                 ↺
(normal,compress,noicmp)
#
```

```
Shelmerg  dc-slip       sl-helmerg      0xfffffc00      autocomp
```

Al final de la línea puede ver que existen una o más opciones.

- `normal` — sin compresión de los encabezados.
- `compress` — compresión de los encabezados.
- `autocomp` — compresión automática, si el host remoto lo permite.
- `noicmp` — deshabilitar los paquetes ICMP (de tal forma que cualquier paquete enviado por "ping" seráa rechazado, en lugar de ocupar de su ancho de banda).

La elección sobre la dirección local y remota depende si usted va a utilizar una conexión TCP/IP dedicada o bien si va a utilizar una conexión por medio de "proxy ARP" en su servidor SLIP (no es correcto "proxy ARP", pero es la terminología utilizada en esta sección para describirlo). Si usted no esta seguro que metodo manejar o como asignar la dirección IP, por favor refierase a alguno de los libros sobre TCP/IP, que se mencionan en los Prerequisitos de SLIP (Sección 25.6.2.1, "Prerequisitos") y/o consulte al administrador de IP de su red.

Si usted piensa subnetear para los diferentes clientes SLIP, será necesario que la dirección de la subred (subnet), salga de la dirección IP que tenga asignada su red, y el numero de cada cliente, del numero que asigne a su subred. Posteriormente puede que sea necesario, o bien configurar una ruta estática a la subred SLIP, por medio de su servidor SLIP en su ruteador más cercano por IP.

De otra forma, si usted piensa utilizar un metodo "proxy ARP", será necesario que a sus clientes SLIP, se les asigne una dirección IP, que se encuentre dentro del rango que este utilizando para su subred Ethernet, y tambié será necesario que haga algunos ajustes en los ficheros script `/etc/sliphome/slip.login` y en `/etc/sliphome/slip.logout` , para que usen arp(8), para que maneje la tabla ARP del servidor SLIP y llamados del proxy-ARP.

25.6.2.4.2. `slip.login` Configuración

El típico fichero `/etc/sliphome/slip.login` se ve de la siguiente manera:

```
#!/bin/sh -
#
#       @(#)slip.login  5.1 (Berkeley) 7/1/90

#
# generic login file for a slip line.  sliplogin invokes this with
# the parameters:
#      1        2         3        4          5         6     7-n
#   slipunit ttyspeed loginname local-addr remote-addr mask opt-args
#
/sbin/ifconfig sl$1 inet $4 $5 netmask $6
```

This `slip.login` file merely runs `ifconfig` for the appropriate SLIP interface with the local and remote addresses and network mask of the SLIP interface.

If you have decided to use the "proxy ARP" method (instead of using a separate subnet for your SLIP clients), your `/etc/sliphome/slip.login` file will need to look something like this:

```
#!/bin/sh -
#
#       @(#)slip.login  5.1 (Berkeley) 7/1/90

#
# generic login file for a slip line.  sliplogin invokes this with
# the parameters:
#      1        2         3        4          5         6     7-n
#   slipunit ttyspeed loginname local-addr remote-addr mask opt-args
#
/sbin/ifconfig sl$1 inet $4 $5 netmask $6
# Answer ARP requests for the SLIP client with our Ethernet addr
/usr/sbin/arp -s $5 00:11:22:33:44:55 pub
```

La línea adicional, `arp -s $5 00:11:22:33:44:55 pub` del script `slip.login` , crea una entrada ARP en la tabla del servidor SLIP. Esta entrada le indica al servidor SLIP que debe responder con la dirección MAC de su dispositivo Ethernet, cuando cualquier otro nodo IP en la red, solicite información a la IP del cliente SLIP.

Al tomar en cuenta el ejemplo anterior, es importante que sustituya la dirección Ethernet MAC (`00:11:22:33:44:55`), con la dirección que corresponde a su tarjeta de red, o definitivamente su "proxy ARP" no va a funcionar!. Para efecto de conocer cual es la dirección MAC del dispositivo Ethernet (tarjeta de red), de su servidor SLIP, puede ejecutar el comando `netstat -i`, el cual tendrá como resultado algo similar a lo siguiente:

```
ed0   1500  <Link>0.2.c1.28.5f.4a         191923 0   129457     0   ↺
116
```

Esto indica que la dirección MAC de su dispositivo Ethernet, en este sistema es `00:02:c1:28:5f:4a` — los puntos que aparecen en la salida del comando `netstat -i` deben cambiarse por dos puntos, así mismo deberá de anteponerse un cero, a cada dígito hexadecimal que aparezca sólo (no en pares), de tal forma que convirtamos la dirección en lo que arp(8) requiere para trabajar; vea la página de ayuda arp(8), para ver información completa sobre su uso.

Nota

Recuerde que cuando cree los ficheros `/etc/sliphome/slip.login` y `/etc/sliphome/slip.logout` , deben contar con permisos de ejecución (`chmod 755 /etc/sliphome/slip.login /`

> `etc/sliphome/slip.logout`), de otra forma estos scripts no podrán llevar a cabo su función.

25.6.2.4.3. Configuración de `slip.logout`

El fichero `/etc/sliphome/slip.logout` no es indispensable (a menos que vaya a utilizar "proxy ARP"), pero si aun así decide crearlo, el siguiente es un ejemplo básico del script `slip.logout` :

```
#!/bin/sh -
#
#       slip.logout

#
# logout file for a slip line.  sliplogin invokes this with
# the parameters:
#      1        2         3        4          5         6     7-n
#   slipunit ttyspeed loginname local-addr remote-addr mask opt-args
#
/sbin/ifconfig sl$1 down
```

Si usted esta utilizando "proxy ARP", es recomendable que le indique a `/etc/sliphome/slip.logout` , que desea eliminar la entrada ARP, para el cliente SLIP:

```
#!/bin/sh -
#
#       @(#)slip.logout

#
# logout file for a slip line.  sliplogin invokes this with
# the parameters:
#      1        2         3        4          5         6     7-n
#   slipunit ttyspeed loginname local-addr remote-addr mask opt-args
#
/sbin/ifconfig sl$1 down
# Dejar de solicitar respuesta ARP al cliente SLIP
/usr/sbin/arp -d $5
```

El comando `arp -d $5`, elimina la entrada ARP, que `slip.login` de "proxy ARP" añadió al cliente SLIP al ingresar al sistema.

Para esta más seguros: asegurese de que el fichero `/etc/sliphome/slip.logout` cuenta con los permisos adecuados para su ejecución, una vez que lo ha creado (ej. `chmod 755 /etc/sliphome/slip.logout`).

25.6.2.5. Consideraciones sobre el Enrutamiento

Si usted no esta utilizando el metodo "proxy ARP", para efecto de rutear los paquetes entre sus clientes SLIP y el resto de la red (y quizás Internet), deberá de hacer una de

las siguientes dos acciones, o bien añadir direcciones estáticas, a su(s) ruteador(es) más cercanos, para que se reenvien los paquetes de la subred de sus clientes SLIP, por medio de su servidor SLIP, o bien tendrá que instalar y configurar `gated` en su servidor SLIP (que corre FreeBSD!), de tal forma que le indique a su(s) ruteador(es), por medio del protocolo correcto, a cerca de su subred SLIP.

25.6.2.5.1. Direcciones de Enrutamiento Estáticas

Añadir direcciones estáticas de enrutamiento puede ser un problema (o imposible si usted no cuenta con la autoridad para hacerlo...). Si usted cuenta con una red de ruteo-múltiple en su organización, algunos ruteadores, tales como los fabricados por Cisco y Proteon, puede ser que no sea suficiente con el hecho de configurar las rutas estáticas de su subred SLIP, sino que sea necesario indicar que rutas utilizar para informar a cerca de otras rutas, así que algo de experiencia así como determinación para la resolución de problemas serán necesarias para poner la ruta basada-en- ruteo-estático a trabajar.

25.6.2.5.2. Ejecutando `gated`

Una alternativa para los dolores de cabeza que pueden dar las redes con ruteo estático, es intalar `gated` en su servidor SLIP bajo FreeBSD y configurarlo, para que utilice los protocolos de ruteo apropiados (RIP/OSPF/BGP/EGP) para informar a otros ruteadores, a cerca de su subred SLIP. Una vez que lo ha compilado e instalado, deberá crear el fichero `/etc/gated.conf` , que configurará a `gated`; aqui hay un ejemplo, similar al que el autor utilizó en un servidor SLIP FreeBSD:

Nota

`gated` es un software propietario y su código fuente no estará disponible al público más (más información en el sitio gated). Esta sección solo existe para asegurarse de la compatibilidad con aquellos que usan la versió antigua.

```
#
# gated configuration file for dc.dsu.edu; for gated version ↺
3.5alpha5
# Only broadcast RIP information for xxx.xxx.yy out the ed Ethernet ↺
interface
#
#
# tracing options
#
traceoptions "/var/tmp/gated.output" replace size 100k files 2 ↺
general -;

rip yes {
```

```
  interface sl noripout noripin -;
  interface ed ripin ripout version 1 -;
  traceoptions route -;
} -;

#
# Turn on a bunch of tracing info for the interface to the kernel:
kernel {
  traceoptions remnants request routes info interface -;
} -;

#
# Propagate the route to xxx.xxx.yy out the Ethernet interface via ↺
RIP
#

export proto rip interface ed {
  proto direct {
      xxx.xxx.yy  mask 255.255.252.0 metric 1; # SLIP connections
  } -;
} -;

#
# Accept routes from RIP via ed Ethernet interfaces

import proto rip interface ed {
  all -;
} -;
```

En el ejemplo anterior, el fichero de configuración `gated.conf` transmite información sobre la subred SLIP *xxx.xxx.yy* , por medio de RIP al dispositivo Ethernet; si usted esta utilizando un dispositivo de red, diferente de `ed`, será necesario que modifique el parámetro `ed` por el correspondiente. En este ejemplo, el fichero también realiza una busqueda por el fichero `/var/tmp/gated.output` , que es un fichero que nos sirve para depurar cualquier error que se presente en la actividad de `gated`; usted puede desactivar la opción de depuración (debug), si es que `gated` esta funcionando correctamente. Será necesario que modifique *xxx.xxx.yy.* , a quedar con la dirección correcta de su subred SLIP (asegurese de modificar también la máscara de red, en la cláusula `proto direct` también).

Una vez que ha instalado y configurado `gated` en su sistema, necesitará indicarle a FreeBSD que al iniciar el sistema, ejecute el script para `gated`, en lugar de ejecutar `routed`. La forma más fácil de hacer esto, es editar las variables de `route` y `router_flags` , en el fichero `/etc/rc.conf` . Por favor vea la página de ayuda de `gated`, para ver información sobre los parámetros de la línea—de—comandos.

Capítulo 26. Cortafuegos

Texto de Joseph J. Barbish.
Convertido a SGML y actualizado por Brad Davis.

26.1. *

Pendiente de traducción

Capítulo 27. Correo Electrónico

Escrito por Bill Lloyd.
Reescrito por Jim Mock.

27.1. Sinopsis

El “Correo Electrónico”, o “email”, es una de las formas de comunicación más usadas hoy en dia. Este capítulo es una introducción básica de cómo poner en marcha un servidor de correo en FreeBSD, aunque no es una guía completa y se han omitido muchos conceptos importantes. Si se necesita información exhaustiva sobre el tema puede recurrirse a los libros listados en Capítulo 31, *Bibliografía*.

Después de leer este capítulo usted sabrá:

- Qué software está involucrado en el envío y recepción de correo electrónico.
- Dónde se encuentran en FreeBSD los ficheros básicos de configuración de sendmail.
- Cómo bloquear a los “spammers” y evitar el uso no autorizado de su servidor de correo.
- Cómo instalar y configurar agentes de transferencia de correo (MTA) en su sistema, para reemplazar sendmail.
- Cómo resolver problemas comunes en servidores de correo.
- Cómo usar SMTP con UUCP.
- Cómo usar correo con una conexión dialup.
- Cómo configurar SMTP con autentificación para más seguridad.

Antes de leer este capítulo debería usted:

- Configurar apropiadamente su conexión de red (Capítulo 29, *Networking avanzado*).
- Configurar apropiadamente la información de DNS de su servidor de correo (Capítulo 29, *Networking avanzado*).
- Saber como instalar software adicional (Capítulo 4, *Instalación de aplicaciones: «packages» y ports*).

27.2. Utilización del correo electrónico

Existen varios elementos relacionados con el intercambio de correo electrónico. A saber: El agente de usuario (Mail User Agent o MUA), El agente de transporte de correo (Mail Transport Agent o MTA), El DNS, Las carpetas de correo (Mailboxes), y por supuesto, la máquina servidora de correo (mail host).

27.2.1. El Agente de Usuario

Entre las opciones más conocidas tenemos: mutt, pine, elm, y `mail`, junto con programas con interfaz gráfica del estilo de balsa o xfmail por nombrar unos pocos. También existen lectores de correo basados en navegadores web. Los programas de correo basados en navegadores web actúan de pasarela para las transacciones de correo electrónico, entregando dichas transacciones al "servidor de correo" local, llamando a uno de los agentes de transporte de correo disponibles en la máquina local, o entregando dichas transacciones a un agente de transporte remoto utilizando el protocolo TCP.

27.2.2. Agente de Transporte de Correo

FreeBSD viene con sendmail por defecto, pero también se soportan otros dæmones, entre los cuales se encuentran:

- exim;
- postfix;
- qmail.

El agente de transporte de correo normalmente posee dos funcionalidades, por un lado se responsabiliza de la recepción y por otro se encarga de entregar el correo de salida. *No* es responsable de la recolección automática de correo mediante la utilización de protocolos como POP o IMAP, ni se utiliza para que el usuario pueda acceder a las carpetas de correo locales. Para realizar estas otras tareas, se necesitan dæmons adicionales.

Aviso

Versiones antiguas de sendmail poseen varios problemas de seguridad que pueden facilitar a un atacante el acceso local o remoto a la máquina que ejecuta sendmail. Para evitar dichos problemas de seguridad se recomienda utilizar una versión actualizada de sendmail. Tiene más opciones a la hora de elegir MTA en los Ports de FreeBSD.

27.2.3. Correo electrónico y DNS

El Sistema de Nombres de Dominio (en inglés DNS) y su correspondiente dæmon (llamado `named`) constituyen una parte fundamental del procedimiento de entrega de correo electrónico. Para entregar el correo electrónico al destinatario adecuado el dæmon servidor de correo busca el sitio remoto dentro del sistema de DNS.

El DNS es la entidad responsable de asociar nombres con direcciones IP, pero además se encarga de almacenar información específica relacionada con la entrega de correo mediante registros de tipo MX. Los registros MX (Mail eXchanger) especifican qué máquina o máquinas están encargadas de recibir correo electrónico para un determinado nombre de dominio. En caso de no existir ningún registro MX para el dominio del destinatario, se busca información almacenada en registros de tipo A para enviar el correo al destino final.

Se pueden consultar los registros MX para cualquier dominio utilizando el comando host(1), como se puede observar en el siguiente ejemplo:

```
% host -t mx FreeBSD.org
        FreeBSD.org mail is handled (pri=10) by mx1.FreeBSD.org
```

27.2.4. Recepción de correo

La recepción de correo electrónico para su dominio se realiza mediante lo que se conoce como la máquina de correo (mail host). Ésta máquina recoge todo el correo dirigido a su dominio y lo almacena en `mbox` (el método por defecto para el almacenamiento de correo) o en formato Maildir, dependiendo de la configuración de la máquina. Una vez que el correo ha sido almacenado con éxito se puede leer en local utilizando aplicaciones como mail(1) o como mutt, o de forma remota mediante un conjunto de protocolos tales como POP o IMAP. Esto significa que si usted va a leer el correo de forma local no necesita instalar ningún servidor de POP o IMAP.

27.2.4.1. Acceso a carpetas de correo remotas mediante POP e IMAP

Para poder acceder a carpetas de correo de forma remota se necesita tener acceso a un servidor de POP o IMAP. Éstos protocolos permiten a los usuarios conectarse a sus carpetas de correo desde ubicaciones remotas de una forma sencilla. Aunque ambos, POP y IMAP, permiten este acceso remoto IMAP ofrece algunas ventajas añadidas, algunas de las cuales son:

- El acceso mediante IMAP permite almacenar los correos en el servidor remoto sin necesidad de extraerlos y tener que almacenarlos en local.
- IMAP soporta actualizaciones concurrentes.
- IMAP resulta ser extremadamente útil bajo enlaces de baja velocidad puesto que permite a los usuarios recuperar la estructura de los mensajes sin necesidad de bajarse

todo el contenido. Además puede realizar tareas tales como búsquedas directas en el servidor con el fin de minimizar la utilización de la red.

Para instalar un servidor de POP o de IMAP se deben dar los siguientes pasos:

1. Seleccionar el servidor IMAP o POP que mejor cumpla a sus necesidades. Los siguientes servidores POP e IMAP son bien conocidos y son firmes candidatos para ello:
   - qpopper;
   - teapop;
   - imap-uw;
   - courier-imap;
2. Instalar el dæmon POP o IMAP de su elección desde el árbol de "ports".
3. Modifique donde sea necesario `/etc/inetd.conf` para que el servidor POP o IMAP se ejecute automáticamente.

Aviso

Tenga en cuenta que tanto POP como IMAP transmiten información, en especial el usuario y la contraseña, en texto plano. Eso significa que si se desea seguridad en la transmisión de la información a través de la red se deben considerar mecanismos adicionales como por ejemplo el encapsulado de la sesión mediante ssh(1). El encapsulado de sesiones se explica en Sección 14.11.8, "Túneles SSH".

27.2.4.2. Acceso a carpetas de correo locales

Las carpetas de correo pueden abrirse de forma local utilizando un agente de correo de usuario (MUA) en el servidor donde reside la carpeta. Se suelen usar los programas mutt or mail(1).

27.2.5. El Servidor de Correo (Mail Host)

El servidor de correo es el nombre que se usa para identificar a la máquina responsable de la entrega y recepción de correo electrónico dentro de una organización. Ésta máquina puede recibir correo de varios usuarios dentro de su dominio.

27.3. Configuración de sendmail

Contribuido por Christopher Shumway.

sendmail(8) es el agente de transporte de correo (MTA) por defecto de FreeBSD. La responsabilidad de sendmail consiste en aceptar correo de agentes de correo de usuario (MUA) y en entregar dichos correos al agente de transporte de correo apropiado, según se especifique en su archivo de configuración. Sendmail también acepta conexiones de red provinientes de otros agentes de transporte y puede depositar el correo recibido en carpetas locales o o entregarlo a otros programas.

sendmail utiliza los siguientes ficheros de configuración:

Filename	Function
`/etc/mail/access`	Base de datos de accesos de sendmail
`/etc/mail/aliases`	Carpeta de alias
`/etc/mail/local-host-names`	Listados de máquinas para las que sendmail acepta correo
`/etc/mail/mailer.conf`	Configuración del programa de correo
`/etc/mail/mailertable`	Tabla de entregas de correo
`/etc/mail/sendmail.cf`	Archivo de configuración principal de sendmail
`/etc/mail/virtusertable`	Usuarios virtuales y tablas de dominio

27.3.1. `/etc/mail/access`

La base de datos de accesos define qué máquinas o direcciones IP pueden acceder al servidor de correo y quée clase de acceso tienen permitido. Las máquinas se listan junto con las opciones `OK`, `REJECT`, `RELAY` o simplemente junto con un mensaje de error que se entrega a la rutina de gestión de excepciones de sendmail. Las máquinas que se listan junto con la opción `OK`, que es el valor por defecto, tienen permiso para enviar correo a la máquina servidora siempre y cuando la dirección de correo de destino sea la máquina servidora de correo. Las máquinas listadas junto con la opción `REJECT` tienen el acceso prohibido a conexiones de correo electrónico con el servidor. Por último las máquinas que poseen la etiqueta `RELAY` para sus nombres tienen permitido enviar correo para cualquier destino a través de la máquina servidora de correo.

Ejemplo 27.1. Configuración de la base de datos de acceso de sendmail

```
cyberspammer.com                    550 We don't accept mail from ↺
spammers
```

```
FUENTE.DE.CORREO.INDISCRIMINADO@ 550 We don't accept mail ↺
from spammers
otra.fuente.de.spam             REJECT
okay.cyberspammer.com           OK
128.32                          RELAY
```

En el ejemplo se pueden observar cinco entradas. Los generadores de correo que coinciden con la parte izquierda de la tabla se ven afectados por la parte acción especificada en la parte derecha. Los primeros dos ejemplos emiten un código de error para la rutina de excepciones de sendmail. El mensaje de error se transmite a la máquina remota cuando se recibe un correo que coincide con la parte izquierda de la tabla. La siguiente entrada rechaza correo de una determinada máquina de internet, `otra.fuente.de.spam` . La siguiente entrada acepta conexiones de correo de la máquina `okay.cyberspammer.com` , lo cual es más exacto que la línea de arriba de `cyberspammer.com` . Las coincidencias más completas tienen precedencia sobre las menos específicas. La última entrada permite actuar como "relay" o pasarela de correo electrónico para aquellas máquinas que posean una dirección IP que comience por `128.32` . Éstas máquinas podrían enviar correo destinado a otros servidores de correo utilizando el nuestro.

Cuando se actualiza este fichero se debe ejecutar `make` dentro de `/etc/mail/` para que se actualice la base de datos.

27.3.2. `/etc/mail/aliases`

La base de datos de alias contiene una lista de directorios virtuales que son traducidas a otros usuarios, ficheros, programas o incluso otros alias. A continuación se muestran unos pocos ejemplos de la sintáxis que se puede utilizar dentro del fichero `/etc/mail/aliases`:

Ejemplo 27.2. Mail Aliases

```
root: usuariolocal
ftp-bugs: joe,eric,paul
bit.bucket:  /dev/null
procmail: "|/usr/local/bin/procmail"
```

El formato del fichero es sencillo; el nombre de la carpeta de correo que aparece a la izquierda de los dos puntos se traduce al/los destinos de la derecha. El primer ejemplo simplemente traduce la carpeta `root` a la carpeta `usuariolocal`, la cual se examina de nuevo utilizando la misma base de datos de alias, y si no existe ninguna otra coincidencia el

mensaje se entrega al usuario local `usuariolocal`. En el ejemplo siguiente se muestra una lista de correo. Todo correo que se envía a la carpeta `ftp-bugs` se traduce en un envío para tres carpetas locales diferentes: `joe`, `eric` y `paul`. Es importante señalar que también se pueden especificar carpetas remotas mediante la forma `usuario@ejemplo.com`. El siguiente ejemplo muestra la escritura del correo a un fichero, en este caso en `/dev/null`. El último ejemplo muestra el envió de correo a un programa; en este caso el mensaje de correo se escribe en la entrada estándar del programa `/usr/local/bin/procmail` utilizando una tubería (o " pipe") de UNIX®.

Cuando se actualiza este fichero se debe ejecutar `make` dentro de `/etc/mail/` para actualizar la base de datos.

27.3.3. `/etc/mail/local-host-names`

Este archivo es una lista de nombres de máquinas que sendmail(8) acepta como nombres locales. Se suele utilizar para escribir aquellos dominios o máquinas de los cuales sendmail va a recibir correo. Por ejemplo, si nuestro servidor de correo va a aceptar correo proveniente del dominio `ejemplo.com` y también de la máquina `mail.ejemplo.com` nuestro `local-host-names` debería ser algo así:

```
ejemplo.com
mail.ejemplo.com
```

Cuando se actualiza este fichero sendmail(8) necesita ser reiniciado para que tenga en cuenta los cambios.

27.3.4. `/etc/mail/sendmail.cf`

Archivo de configuración principal de sendmail, controla el comportamiento global de sendmail, incluyendo cualquier tarea desde la reescritura de direcciones de correo electrónico hasta la devolución de mensajes de error a los servidores de correo remotos. Es evidente que con un abanico tan diverso el fichero de configuración acaba por ser bastante complejo y sus detalles quedan fuera de los objetivos de esta sección. Afortunadamente este fichero raras veces necesita ser modificado, al menos en lo que respecta a servidores de correo estándar.

El fichero de configuración principal de sendmail se puede construir a partir de m4(1), es decir, macros que se utilizan para definir características y comportamientos específicos de sendmail. Se ruega al lector consultar `/usr/src/contrib/sendmail/cf/README` para obtener más detalles acerca de las distintas macros que se pueden utilizar.

Cuando se realizan cambios a este fichero sendmail debe ser reiniciado para que los cambios surtan efecto.

27.3.5. /etc/mail/virtusertable

El fichero `virtusertable` asocia direcciones de correo pertenecientes a dominios y carpetas virtuales con carpetas reales. Estas carpetas pueden ser locales, remotas, alias definidos en `/etc/mail/aliases` o incluso ficheros.

Ejemplo 27.3. Ejemplo de asociación de correo de dominio virtual

```
root@ejemplo.com                root
postmaster@ejemplo.com          postmaster@noc.ejemplo.net
@ejemplo.com                    joe
```

En el ejemplo superior se observa una asociación para el dominio `ejemplo.com` . Este fichero se procesa de arriba a abajo buscando la primera coincidencia. La primera entrada asocia `root@ejemplo.com` con la carpeta de correo local denominada `root`. La siguiente entrada asocia `postmaster@ejemplo.com` con la carpeta `postmaster` situada en la máquina `noc.ejemplo.net` . Por último, si no se ha encontrado ninguna coincidencia para `ejemplo.com` se le asigna la última asociación, la cual asocia cualquier mensaje de correo proveniente de `ejemplo.com` con la carpeta de correo local denominada `joe`.

27.4. Sustitución del Agente de Transferencia de Correo

Escrito por Andrew Boothman.
Información obtenida de correos escritos por Gregory Neil Shapiro.

Como ya se ha comentado FreeBSD viene con sendmail ya instalado como agente de transferencia de correo por defecto. De esta forma sendmail se encarga de gestionar el correo entrante y saliente.

No obstante, debido a distintas razones algunos administradores de sistemas prefieren utilizar otro MTA. Estas razones varían desde simplemente querer probar otros programas de transferencia de correo, hasta la necesidad de utilizar un determinado programa que hace uso de una función específica de un agente determinado. Por suerte cualesquiera que sean estas razones FreeBSD posee un sencillo procedimiento para sustituir a sendmail.

27.4.1. Instalación de un nuevo MTA

Existen una amplia gama de MTA alternativos a sendmail. Un buen punto de partida es el Sistema de Ports de FreeBSD, donde se pueden localizar varios. Por supuesto el usuario

tiene libertad para utilizar cualquier MTA, siempre y cuando se pueda ejecutar en FreeBSD sin problemas.

Lo primero es instalar el nuevo MTA. Una vez que está instalado normalmente se tiene la oportunidad para decidir si realmente cubre las necesidades y también se tiene la oportunidad de configurar el nuevo software antes de sustituir a sendmail. El usuario debe tener en cuenta que el nuevo MTA puede sobreescribir algunos binarios del sistema como por ejemplo `/usr/bin/sendmail` . En cualquier caso el nuevo software de correo suele entrar en funcionamiento con una configuración por defecto.

Por favor, recuerde que se recomienda leer la documentación del MTA seleccionado para obtener más información.

27.4.2. Desactivación de la aplicación sendmail

El procedimiento utilizado para ejecutar sendmail cambió significativamente entre las releases 4.5-RELEASE y 4.6-RELEASE. De esta forma el procedimiento utilizado para la desactivación hoy en día es sutílmente distinto al utilizado en dichas distribuciones.

27.4.2.1. FreeBSD 4.5-STABLE antes de 2002/4/4 y anteriores (Incluyendo 4.5-RELEASE y anteriores)

Introducir:

```
sendmail_enable="NO"
```

dentro de `/etc/rc.conf` . Esta variable desactiva el servicio de recepción de correo de sendmail, pero salvo que se modifique (ver más adelante) el fichero `/etc/mail/mailer.conf` sendmail todavía será la aplicación elegida para enviar correo electrónico.

27.4.2.2. FreeBSD 4.5-STABLE desde de 2002/4/4 (Incluyendo 4.6-RELEASE y posteriores)

Para poder desactivar completamente sendmail haga lo siguiente:

```
sendmail_enable="NONE"
```

dentro del fichero `/etc/rc.conf`.

Aviso

Se desactiva el servicio de correo de salida de sendmail. Es importanque que se reemplace con un sistema de entrega de correo alternativo que sea totalmente funcional. En caso contrario funciones del sistema FreeBSD tales como periodic(8) no podrán entregar

> sus resultados por correo eletrónico tal y como normalmente hacen. Varias partes del sistema FreeBSD esperan disponer de un sistema de correo funcional compatible con sendmail. Si las aplicaciones continúan utilizando los binarios de sendmail para realizar envíos de correo después de su desactivación el correo podría ser almacenado en una cola inactiva de sendmail, en cuyo caso nunca se entregaría.

Si sólo se quiere desactivar el servicio de correo de entrada de sendmail, basta con establecer la variable:

```
sendmail_enable="NO"
```

dentro de `/etc/rc.conf` . En rc.sendmail(8) tiene más información sobre las opciones de arranque de sendmail.

27.4.3. Ejecución del nuevo MTA en el arranque

Existen dos métodos alternativos para ejecutar el nuevo MTA en el arranque, dependiendo de la versión de FreeBSD que se esté ejecutando.

27.4.3.1. FreeBSD 4.5-STABLE antes de 2002/4/11 (Incluyendo 4.5-RELEASE y anteriores)

Se debe añadir un script en `/usr/local/etc/rc.d/` cuyo nombre termine en `.sh` y que sea ejecutable por `root`. El script debe aceptar los parámetros `start` y `stop`. Cuando el sistema FreeBSD se está inicializando, los scripts de arranque ejecutarán el siguiente comando:

```
/usr/local/etc/rc.d/supermailer.sh start
```

La misma orden se puede utilizar también para ejecutar el servidor de forma manual. Cuando el sistema se está reiniciando los scripts del sistema ejecutan los ficheros ubicados en `/usr/local/etc/rc.d/` utilizando la opción `stop`, en nuestro caso:

```
/usr/local/etc/rc.d/supermailer.sh stop
```

Dicho comando tambíen se puede utilizar para detener el servidor de correo de forma manual cuando el sistema FreeBSD se ejecuta con normalidad.

27.4.3.2. FreeBSD 4.5-STABLE después de 2002/4/11 (Incluyendo 4.6-RELEASE y posteriores)

Con las últimas versiones de FreeBSD se puede utilizar el método anterior pero también se puede especificar

```
mta_start_script="nombre_de_fichero"
```

dentro de `/etc/rc.conf` , donde *`nombre_de_fichero`* es el nombre de algún script que se ejecuta en tiempo de arranque para inicializar el nuevo MTA.

27.4.4. Sustitución de sendmail como el agente de transporte de correo predeterminado.

El programa sendmail es tan imprescindible y es utilizado por tal multitud de programas en los sistemas UNIX® que algunos programas simplemente asumen que sendmail se encuentra instalado y configurado dentro del sistema. Por esta razón varios MTAs alternativos proporcionan su propia implementación de la interfaz de línea de comandos que posée sendmail; esto facilita que se puedan utilizar como sustitutos de sendmail sin mayores dificultades.

Por lo tanto si desea utilizar un agente de transporte de correo alternativo debe asegurarse de que todo software que intente ejecutar binario de sendmail estándar, `/usr/bin/sendmail`, realmente ejecute el nuevo MTA en su lugar. Por fortuna FreeBSD proporciona un sistema llamado mailwrapper(8) que realiza precisamente esta tarea.

Cuando sendmail está funcionando se debe localizar algo como lo siguiente dentro del fichero `/etc/mail/mailer.conf` :

```
sendmail  /usr/libexec/sendmail/sendmail
send-mail /usr/libexec/sendmail/sendmail
mailq  /usr/libexec/sendmail/sendmail
newaliases /usr/libexec/sendmail/sendmail
hoststat /usr/libexec/sendmail/sendmail
purgestat /usr/libexec/sendmail/sendmail
```

Esto significa que cuando cualquiera de estos comandos (por ejemplo `sendmail` mísmamente) se ejecutan el sistema ejecutará en su lugar una copia del el sistema ejecuta en su lugar una copia del " mailwrapper" denominada `sendmail` que chequea el fichero `mailer.conf` y ejecuta `/usr/libexec/sendmail/sendmail` . Este sistema permite cambiar de una forma sencilla los binarios que se ejecutan realmente cuando se invocan las funciones de `sendmail`.

Si se quiere que ejecutar `/usr/local/supermailer/bin/sendmail-compat` en lugar de sendmail se puede cambiar el fichero `/etc/mail/mailer.conf` para que contenga lo siguiente:

```
sendmail  /usr/local/supermailer/bin/sendmail-compat
send-mail /usr/local/supermailer/bin/sendmail-compat
mailq  /usr/local/supermailer/bin/mailq-compat
newaliases /usr/local/supermailer/bin/newaliases-compat
hoststat /usr/local/supermailer/bin/hoststat-compat
purgestat /usr/local/supermailer/bin/purgestat-compat
```

27.4.5. Últimos Pasos

Una vez que todo estáconfigurado a su gusto hay que matar los procesos de sendmail que ya no se necesitan y ejecutar los procesos pertenecientes al nuevo software de MTA, o utilizar la opción más sencilla: reiniciar la máquina. Reinicar la máquina nos brinda la oportunidad de comprobar que se ha configurado correctamente el arranque del sistema para que ejecute de forma automática el nuevo MTA.

27.5. Depuración de Problemas

P: ?Por qué tengo que utilizar el FQDN para las máquinas de mi organización?

R: Probablemente se deba a que la máquina de correo se encuentra en un dominio diferente; si por ejemplo la máquina de correo se encuentra en `foo.bar.edu` y se desea alcanzar una máquina llamada `mumble` en el dominio `bar.edu` se tiene que referir a ella mediante un nombre de dominio completo ("fully-quailified domain name" o FQDN), en éste caso `mumble.bar.edu` en lugar de referirse a ella simplemente como `mumble`.

Tradicionalmente, la referencia incompleta era posible utilizando "resolvers" de BSD BIND. No obstante la versión de BIND que en la actualidad se ofrece con FreeBSD ya no permite por defecto el uso de dichas abreviaturas salvo para aquellas máquinas que pertenecen al dominio al que su sistema pertenezca. Una máquina como `mumble` será buscada como `mumble.foo.bar.edu` o la búsqueda será redireccionada al servidor de dominio raíz del DNS.

Esto es distinto a lo que ocurría en versiones anteriores de BIND, donde la búsqueda se producía a través de `mumble.bar.edu` y de `mumble.edu`. Se recomienda consultar a la RFC 1535 para conocer el motivo que se considerara una práctica errónea o incluso un agujero de seguridad.

Una buena solución a este problema puede ser incluír la siguiente línea

```
search foo.bar.edu bar.edu
```

en lugar de

```
domain foo.bar.edu
```

dentro del fichero `/etc/resolv.conf`. No obstante se debe asegurar de que el orden de búsqueda no se expande más allá del "límite entre la administración local y la administración pública", tal y como se le denomina en la RFC 1535.

P: sendmail dice mail loops back to myself (el correo vuelve a mis manos)

R: Esta pregunta se responde en las FAQ de sendmail de la siguiente forma:

```
Estoy obteniendo los siguientes mensajes de error:

553 MX list for domain.net points back to relay.domain.net
554 <user@domain.net>... Local configuration error

¿Cómo puedo solucionar esto?

Usted ha especificado que el correo para el dominio (por ejemplo,
para el dominio dominio.net) sea reenviado a una máquina ↺
determinada
(en este caso relay.dominio.net), para lo que se utiliza un ↺
registro
de DNS de tipo MX,
pero la máquina que actúa de relay no se
reconoce a sí misma como perteneciente al dominio dominio.net.
Se debe añadir dominio.net al fichero /etc/mail/local-host-names,
que recibe el nombre de /etc/sendmail.cw en versiones de ↺
sendmail previas
a la 8.10.  Se puede utilizar la macro FEATURE(use_cw_file) ↺
para indicar
dónde se encuentra el fichero local-host-names; también se
puede añadir “Cw dominio.net” directamente
al fichero /etc/mail/sendmail.cf
```

Las FAQ de sendmail se pueden encontrar en `http://www.sendmail.org/faq/` y son de lectura obligada si se quiere depurar el comportamiento y la configuración de sendmail.

P: ?Cómo puedo ejecutar un servidor de correo utilizando una máquina que se conecta a internet mediante modem analógico (dial-up) ?

R: Se quiere conectar una máquina FreeBSD dentro de una LAN a Internet. La máquina FreeBSD será una pasarela de correo para dicha LAN. La conexión mediante PPP no es dedicada.

Existen al menos dos formas distintas de hacerlo. Una de ellas consiste en utilizar UUCP.

Otra forma consiste en hacerse con un servidor de internet a tiempo completo para proporcionar servicios de agente de transporte secundario para nuestro dominio. Si por ejemplo el dominio de nuestra compañía es `ejemplo.com` , nuestro proveedor de acceso a internet puede instalar lo siguiente en el DNS:

```
ejemplo.com.          MX        10      ejemplo.com.
                      MX        20      ejemplo.net.
```

Nótese que el agente de correo primario es nuestro dominio, ejemplo.com, y además se encuentra configurado un agente de transporte secundario en la máquina

ejemplo.net. En este caso sólamente se debe especificar una máquina como receptor final de correo (añadiendo `Cw ejemplo.com`) al fichero `/etc/mail/sendmail.cf` de la máquina `example.com`)

Cuando el `sendmail` que está enviando el correo trata de entregar dicho correo primero intentará conectarse con nosotros (`ejemplo.com`) utilizando el enlace de modem. Lo más probable es que la operación termine después de un tiempo de espera debido a que el enlace modem esté caído. La aplicación sendmail automáticamente entregará el correo al servidor especificado como agente de transporte de correo secundario (segundo registro MX), es decir, entregará el correo a nuestro proveedor de servicios de internet (`ejemplo.net`). El sitio MX secundario tratará de conectarse con nuestra máquina de una forma periódica con el objeto de entregar el correo a la máquina que actúa como agente servidor de correo primario (`ejemplo.com`).

Puede ser de mucha utilidad un script de "login" como el que se muestra a continuación:

```
#!/bin/sh
# Ponme en /usr/local/bin/pppmiconexion
( sleep 60 -; /usr/sbin/sendmail -q ) &
/usr/sbin/ppp -direct pppmiconexion
```

Si vamos a crear un script de "login" separado para un usuario determinado se puede utilizar `sendmail -qRejemplo.com` en lugar del script anterior. Esto obliga a que se procesen de forma inmediata todos los correos que se encuentren en la cola de `ejemplo.com` .

Vamos a dar una vuelta más de tuerca a la situación:

Mensaje robado a la lista de correo de Proveedores de Servicios de Internet en FreeBSD.

```
> Nosotros proporcionamos servicio de MX secundario a un ↺
cliente nuestro.
> El cliente se conecta a nuestro servidor varias veces al día
> de  forma automática para recoger sus correos para ↺
almacenarlos en
> su servidor MX primario (nosotros no llamamos a su organización
> justo cuando nos llega un correo suyo).
> Nuestro sendmail envía la cola de correos cada 30 minutos. En
> estos momentos nuestro cliente tiene que estar al menos 30 ↺
minutos
> conectado para asegurarnos de que todo su correo ha sido ↺
enviado al
> servidor MX primario.
>
> ?Existe algún comando que permita indicar a sendmail que
> envíe todos los correos de la cola cuando quiera el cliente?
```

```
> El cliente no tiene permisos de superusuario en la máquina que
> alberga nuestro agente de transporte, por supuesto.

En la sección de “privacy flags” del fichero sendmail.cf
existe una definición como ésta:
Opgoaway,restrictqrun

Basta con eliminar restrictqrun para permitir que usuarios sin ↺
permisos de
superusuario arranquen el procesamiento de la cola.
Sería conveniente además que reordenaran los registros MX.
Nosotros somos el primer MX para nuestros clientes.
Además de esto hay que especificar:

# Si somos el mejor MX para una determinada máquina, intenta
# utilizarnos directamente en vez de generar un error de
# configuración local.
OwTrue

en el archivo de configuración de sendmail.
Mediante la configuración anterior,
una organización remota entregará sus correos directamente a
usted, sin necesidad de intentar conectarse primero a través de
la conexión del cliente.  La etiqueta "OwTrue" se necesita para ↺
evitar
que sendmail genere un mensaje de error.
A continuación ustedes se encargan de entregar el
correo a su(s) respectivo(s) cliente(s) tal como vienen haciendo.

Esta configuración sólo funciona para
“máquinas individuales”,
de tal forma que se necesita que el cliente especifique su ↺
servidor de correo
mediante entradas de tipo A en el DNS.  En concreto se necesita ↺
una entrada de
tipo A en el DNS para el dominio del cliente (“cliente.com”).
```

P: ?Por qué me siguen saliendo mensajes de error del tipo Relaying Denied cuando se trata de enviar correo proveniente de otras máquinas?

R: En las instalaciones del sistema FreeBSD por defecto sendmail se configura para enviar correo sólamente desde la máquina en la cual se está ejecutando. Por ejemplo si un servidor POP está disponible los usuarios serán capaces de consultar su correo desde la universidad, el trabajo u otras localizaciones remotas, pero dichos usuarios podrán enviar correo desde dichas ubicaciones. Es habitual que unos instantes después del envío del correo dichos usuarios reciban un mensaje proveniente del MAILER-DAEMON con un error como 5.7 Relaying Denied.

Existen varias formas de solventar este problema. La más sencilla consiste en escribir la dirección IP de su proveedor de servicios dentro del fichero `/etc/mail/relay-domains`. Una forma rápida de hacerlo sería:

```
# echo "un.isp.ficticio.com" > /etc/mail/relay-domains
```

Después de crear o editar dicho fichero se debe reiniciar sendmail. Esto funciona perfectamente si usted es el administrador del servidor y no desea enviar correo localmente o si prefiere utilizar un cliente de correo o cualquier otro sistema en otra máquina distinta a la que alberga el servidor de correo. Es muy útil sobre todo cuando sólamente se tienen una o dos direcciones de correo eletrónico. Si hay en liza un gran número de direcciones de correo, edite el fichero anterior con su editor de texto favorito y añada uno a uno los correspondientes dominios.

```
un.isp.ficticio.com
otro.isp.ficticio.net
y.otro.isp.ficticio.org
www.ejemplo.org
```

Ahora, cualquier correo enviado a través de su sistema por cualquier máquina que se encuentre en este fichero (siempre y cuando el usuario tenga una cuenta en nuestro sistema) podrá ser enviado con éxito. Es una manera elegante de permitir a los usuarios enviar correo eletrónico desde nuestro servidor de correo sin permitir al resto del mundo que haga lo mismo (lo que se conoce como SPAM).

27.6. Conceptos Avanzados

La siguiente sección trata conceptos más específicos relacionados con la configuración del correo y la implantación del servicio de correo en una organización.

27.6.1. Configuración Básica

Por defecto debemos ser capaces de enviar correo a máquinas externas, siempre y cuando tengamos nuestro `/etc/resolv.conf` bien configurado o ejecutemos nuestro propio servidor de nombres. Si queremos que el correo para nuestra máquina se nos entregue en nuestra propia máquina, es decir, a nuestro propio sendmail, en lugar de tener que ir a recogerlo al servidor de correo de nuestra organización, podemos usar dos métodos:

- Ejecutar nuestro propio servidor de nombres y comprar nuestro propio dominio. Por ejemplo `FreeBSD.org`
- Conseguir la entrega de correo directa hacia nuestra máquina. Esto se logra entregando el correo a la dirección IP que se asocia al nombre de DNS de nuestra máquina. Por ejemplo `ejemplo.FreeBSD.org` .

Independientemente de la opción elegida para tener entrega directa en nuestra máquina debemos poseer una dirección IP estática (a diferencia de las direcciones dinámicas, que son utilizadas en configuraciones donde se utiliza el protocolo PPP). Si nos encontra-

mos detrás de un cortafuegos se debe permitir el tráfico SMTP (puerto 25) hacia nuestra máquina. Si además queremos recibir correo directamente en nuestra máquina se deben cumplir los siguientes requisitos:

- Asegurar que el registro MX de menor numeración de nuestro DNS apunta a la dirección IP de nuestra máquina.
- Asegurar que no existe ninguna entrada MX en nuestro DNS para nuestra máquina. Es decir, mientras que el registro MX del punto anterior hace referencia al dominio administrativo que gestionamos con nuestro servidor de nombres, en este apartado se quiere destacar que no debe existir ningún registro MX específico para el nombre concreto de nuestra máquina.

Cumpliendo las dos puntualizaciones anteriores podemos recibir correo electrónico mediante entrega directa en nuestra máquina.

Por ejemplo:

```
# hostname
ejemplo.FreeBSD.org
# host ejemplo.FreeBSD.org
ejemplo.FreeBSD.org has address 204.216.27.XX
```

Si se observa esta configuración la entrega directa de correo para <su_login@ejemplo.FreeBSD.org > debería funcionar sin problemas (suponiendo que sendmail se está ejecutando correctamente en `ejemplo.FreeBSD.org`).

Si en lugar de lo anterior ve algo como esto:

```
# host ejemplo.FreeBSD.org
ejemplo.FreeBSD.org has address 204.216.27.XX
ejemplo.FreeBSD.org mail is handled (pri=10) by hub.FreeBSD.org
```

Todos los correos enviados a nuestro host (`ejemplo.FreeBSD.org`) serán recogidos por `hub` bajo el mismo nombre de usuario en lugar de ser enviados directamente a nuestra máquina.

La información anterior se gestiona utilizando el servidor de DNS. El registro de DNS que transporta la información de encaminamiento de correo eletrónico es el registro *M*ail e*X*change. Si no existe ningún registro MX el correo se entregará a la dirección IP que se obtenga de resolver el nombre de dominio que se encuentre a continuación del nombre de usuario en la dirección de correo de destino (esto es, (después de la @).

En un cierto momento la entrada MX para `freefall.FreeBSD.org` tenía este aspecto:

```
freefall  MX 30 mail.crl.net
freefall  MX 40 agora.rdrop.com
freefall  MX 10 freefall.FreeBSD.org
freefall  MX 20 who.cdrom.com
```

Como se puede observar, `freefall` tenía varias entradas MX. El número de MX más bajo es la máquina que recibe correo directamente si se encuentra disponible; si dicha máquina no está accesible por algún motivo las otras máquinas (llamadas también "MXs de backup") aceptarán los mensajes temporalmente, y los transmitirán de nuevo cuando alguna máquina perteneciente a alguna entrada MX de numeración más baja se encuentre disponible y el proceso se repetirá hasta que se alcance la máquina que tenga el registro MX más bajo.

Las organizaciones donde residen los servidores (MX) de backup deberían poseer acceso a internet de una forma independiente para minimizar el riesgo de pérdida de conectividad. Nuestro ISP o cualquier otra organización independiente debería poder proporcionarnos este servicio sin problemas.

27.6.2. Correo para Nuestro Dominio

Para establecer un "mailhost" (servidor de correo) en nuestra organización debemos ser capaces de redirigir el correo destinado a cualquier máquina de nuestra organización hacia nuestro servidor de correo. Básicamente queremos "reclamar" como nuestro cualquier correo destinado a cualquier máquina de nuestro dominio (en este caso `*.FreeBSD.org`) y desviarlo a nuestro servidor de tal forma que los usuario lean su correo utilizando nuestra máquina servidora.

Para hacer las cosas lo más sencillas posible se debe crear una cuenta de usuario (con el mismo *nombre de usuario*) tanto en el servidor de correo como en la máquina del usuario o destinatario final del correo. adduser(8) puede usarse para ello.

El servidor de correo debe funcionar como el agente de transporte predeterminado para todas las máquinas de nuestra organización. Esto se realiza mediante la siguiente configuración del DNS:

```
ejemplo.FreeBSD.org A 204.216.27.XX  ; Workstation
   MX 10 hub.FreeBSD.org ; Mailhost
```

Esta configuración redirigirá el correo para cualquier estación de trabajo hacia nuestro servidor de correo sin que tengan importancia las direcciones IP asignadas mediante el registro de tipo A. Recordemos que el correo siempre se encamina utilizando primero los registros de tipo MX.

Normalmente no podremos realizar esta configuración salvo que estemos ejecutando nuestro propio servidor de DNS para nuestro dominio. Si no es el caso y no es posible ejecutar nuestro propio servidor de DNS debemos comunicarnos con nuestro proveedor de servicios o con quien pueda proporcionarnos servicio de DNS y solicitarle una modificación como la anterior.

Si además ofrecemos servicios de alojamiento virtual de correo la siguiente información puede resultar útil. Asumiremos que tenemos un cliente con su propio dominio, por ejemplo `cliente1.org` y queremos que todo el correo enviado a `cliente1.org` sea redirigido

hasta nuestro servidor de correo, `mail.nuestroservidor.com` . La entrada necesaria en el DNS debería ser la siguiente:

```
cliente1.org  MX 10 mail.nuestroservidor.com
```

No necesitamos *ningún* registro de tipo A para `cliente1.org` si sólamente queremos gestionar el correo para ese dominio.

Nota

Tenga en cuenta que un ping a `cliente1.org` no funcionará a menos que exista un registro de tipo A para dicha máquina.

La última cosa que debemos realizar en nuestro servidor de correo es comunicar a sendmail para qué dominios y/o máquinas debe aceptar correo. Existen varias formas en las que se puede realizar esta tarea. Cualquiera de las siguiente funcionará:

- Añadir las máquinas deseadas al fichero `/etc/mail/local-host-names` si se está utlizando la macro `FEATURE(use_cw_file)` . Si se está utilizando una versión de sendmail anterior a la 8.10 el fichero que se debe utilizar es `/etc/sendmail.cw` .
- Añadir la línea `Cwsu.servidor.com` al fichero `/etc/sendmail.cf` o `/etc/mail/sendmail.cf` si se está utilizando una versión de sendmail posterior a la versión 8.10.

27.7. SMTP con UUCP

La configuración de sendmail que se proporciona con la distribución de FreeBSD está diseñada para organizaciones que se conectan directamente a internet. Las organizaciones que deseén enviar y recibir sus correos utilizando UUCP deben instalar otro fichero de configuración para sendmail.

El ajuste de forma manual del archivo `/etc/mail/sendmail.cf` es un tema para expertos. La versión 8 de sendmail genera ficheros de configuración mediante el preprocesador m4(1), gracias al que las opciones de configuración se pueden escribir utilizando un nivel de abstracción mayor. Los archivos de configuración de m4(1) se pueden encontrar en `/usr/src/usr.sbin/sendmail/cf` .

Si no se instaló el sistema base con todas las fuentes el conjunto de ficheros de configuración de sendmail se puede obtener a partir de un paquete de fuentes determinado. Suponiendo que tengamos el CDROM con el código fuente de FreeBSD montado se puede ejecutar:

```
# cd /cdrom/src
# cat scontrib.?? | tar xzf - -C /usr/src/contrib/sendmail
```

Este comando extrae sólamente unos pocos cientos de kilobytes. El fichero `README` que hay en el directorio `cf` puede servirle como una introducción básica a la configuración mediante m4(1).

La mejor forma de soportar la entrega de correo mediante UUCP es utilizando la característica `mailertable`. Esta característica crea una base de datos que sendmail utiliza para tomar decisiones de encaminamiento.

En primer lugar creamon el fichero `.mc`. El directorio `/usr/src/usr.sbin/sendmail/cf/cf` alberga varios ejemplos del mismo. Suponiendo que nuestro fichero configuración se llama `foo.mc` para convertir dicho archivo en un fichero `sendmail.cf` válido basta con ejecutar lo siguiente:

```
# cd /usr/src/usr.sbin/sendmail/cf/cf
# make foo.cf
# cp foo.cf /etc/mail/sendmail.cf
```

Un fichero `.mc` suele tener este aspecto:

```
VERSIONID(`Su número de versión ') OSTYPE(bsd4.4)

FEATURE(accept_unresolvable_domains)
FEATURE(nocanonify)
FEATURE(mailertable, `hash -o /etc/mail/mailertable')

define(`UUCP_RELAY', su.relay.uucp )
define(`UUCP_MAX_SIZE', 200000)
define(`confDONT_PROBE_INTERFACES')

MAILER(local)
MAILER(smtp)
MAILER(uucp)

Cw    alias.de.su.servidor
Cw    nombredesunodouucp.UUCP
```

Las líneas que contienen `accept_unresolvable_domains` , `nocanonify`, y `confDONT_PROBE_INTERFACES` prohíben la utilización del DNS durante la entrega de correo. La cláusula `UUCP_RELAY` es necesaria para soportar entrega mediante UUCP. Lo único que hay que hacer es escribir un nombre de máquina en ese punto. Dicha máquina debe ser capaz de gestionar las direcciones del pseudo-dominio .UUCP; en la mayoría de los casos se escribe en este punto el nombre de la máquina perteneciente al proveedor de servicios que hace de relay.

Una vez que tenemos esto configurado se necesita un fichero `/etc/mail/mailertable` . Si solamente tenemos un enlace con el exterior, que usamos para todos nuestro correos, basta una configuración como la que se muestra a continuación:

```
#
```

```
# makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
.                              uucp-dom:su.relay.uucp
```

Un ejemplo más complejo puede parecerse al siguiente:

```
#
# makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
#
horus.interface-business.de   uucp-dom:horus
.interface-business.de        uucp-dom:if-bus
interface-business.de         uucp-dom:if-bus
.heep.sax.de                  smtp8:%1
horus.UUCP                    uucp-dom:horus
if-bus.UUCP                   uucp-dom:if-bus
.                             uucp-dom:
```

Las primeras tres líneas se encargan de manejar casos especiales en los que el correo dirigido directamente al dominio no se envía a la ruta por defecto sino a algún vecino UUCP para acortar el número de saltos involucrados en la entrega de dichos correos. La siguiente línea gestiona el correo para el dominio ethernet local, el cual puede ser entregado utilizando SMTP. Finalmente los vecinos UUCP se mencionan en la notación de pseudo-dominio .UUCP para permitir que un `vecino UUCP receptor` de correo pueda sobreescribir las reglas por defecto. La última línea siempre es un punto; se asocia con cualquier otra cosa que no ha sido tratada en reglas anteriores y donde se realiza entrega UUCP a un vecino UUCP que sirve como pasarela de correo universal para todo el mundo. Todos los nombres de máquinas bajo la clave `uucp-dom:` deben ser vecinos UUCP válidos, lo cual se puede verificar utilizando el comando `uuname`.

Recuerde que este fichero debe convertirse en una base de datos DBM antes de que usarse. El comando que se utiliza para realizar esta tarea se suele especificar como un comentario al principio del fichero `mailertable`. Cada vez que se modifique el fichero `mailertable` se debe ejecutar dicho comando.

Un consejo final: si dudamos sobre una determinada ruta de encaminamiento de correo se puede ejecutar sendmail con el parámetro `-bt`. Este parámetro ejecuta sendmail en *modo prueba de direcciones*; simplemente basta con escribir `3,0` seguido por la dirección de correo de la que queremos comprobar su correcto encaminamiento. La última línea nos dice el agente de correo interno que se utiliza, la máquina de destino con que el agente será invocado y la dirección (posiblemente traducida) de correo. Se puede abandonar este modo de funcionamiento escribiendo Ctrl+D.

```
% sendmail -bt
ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
Enter <ruleset> <address>
> 3,0 prueba@ejemplo.com
canonify           input: foo @ example . com
...
parse            returns: $# uucp-dom $@ su.relay.uucp  $: prueba < ↺
@ ejemplo . com . >
```

```
> ^D
```

27.8. Configuración para sólamente enviar correo

Escrito por Bill Moran.

Existen multitud de casos en los que puede bastarnos con enviar nuestro correo a través de una pasarela o relay. He aquí algunos de ellos:

- Nuestra computadora es una máquina de escritorio, pero queremos ser capaces de utilizar programas como send-pr(1). Para ello se debería utilizar el relay de nuestro ISP.
- Nuestra computadora es un servidor que no gestiona correo de forma local, si no que necesita pasar todos los correos recibidos una pasarela que se encarga de su procesamiento y entrega final.

Casi cualquier MTA es capaz de actuar como pasarela o relay. Por desgracia configurar un MTA para que sólo gestione correo saliente puede ser muy complicado. Programas del estilo de sendmail y postfix son demasiado pesados para realizar sólamente esta tarea.

Si además estamos utilizando un servicio de acceso a internet típico nuestro contrato puede prohibir explícitamente la ejecución de un servidor de correo (o los puertos pueden estar filtrados).

La forma más sencilla de utilizar un servicio de pasarela es mediante la instalación del port mail/ssmtp. Basta con ejecutar el siguiente comando como `root`:

```
# cd /usr/ports/mail/ssmtp
# make install replace clean
```

Una vez que ha sido instalado mail/ssmtp podemos configurarlo mediante un fichero de sólo cuatro líneas ubicado en `/usr/local/etc/ssmtp/ssmtp.conf` :

```
root=sudireccionrealdecorreo@ejemplo.com
mailhub=mail.ejemplo.com
rewriteDomain=ejemplo.com
hostname=_HOSTNAME_
```

Debemos asegurarnos de que se utiliza una dirección de correo real para `root`. Se debe introducir nuestra pasarela de correo en lugar de `mail.ejemplo.com` (algunos ISP llaman a la pasarela “servidor de correo saliente” o simplemente “servidor SMTP”).

Debemos asegurarnos de que se desactiva sendmail mediante `sendmail_enable="NONE"` en `/etc/rc.conf` .

mail/ssmtp acepta algunas otras opciones. Consulte el fichero de ejemplo que encontrará en `/usr/local/etc/ssmtp` ; consulte también la página de manual de ssmtp, en la que hay más ejemplos e información al respecto.

Ejecutar ssmtp de esta forma permite que cualquier software de nuestra computadora que necesite enviar correo funcione sin problemas y a la vez poder cumplir con las normas estipuladas en el contrato con nuestro ISP. Al mismo tiempo evitamos el uso de nuestro servidor de correo por parte de “ spammers”.

27.9. Utilización del correo con una conexión mediante módem analógico (dial-up)

Si se dispone de una dirección IP privada no es necesario realizar ningún ajuste a partir de la configuración por defecto. Basta con asignar como nombre de nuestra máquina el nombre que tenemos registrado en el DNS y sendmail se encargará del resto.

Por otra parte si utilizamos una conexión temporal a internet mediante PPP y se nos asigna una dirección IP de forma dinámica, lo más normal es tener nuestras carpetas de correo alojadas en el servidor de correo de nuestro proveedor de servicios. Supongamos que el dominio de nuestro ISP es `ejemplo.net` y que nuestro nombre de usuario es `usuario`; además hemos llamado a nuestra `user`, además, hemos llamado a nuestra máquina `bsd.home` , y nuestro ISP nos ha comunicado que debemos utilizar como pasarela la máquina `relay.ejemplo.net` .

Para recuperar correo de nuestra carpeta de correo se debe instalar un agente de recuperación automática de correo. fetchmail es una buena elección puesto que permite utilizar varios protocolos. Este programa está disponible como un paquete y también desde la colección de ports (mail/fetchmail). Normalmente nuestro ISP proporciona POP POP. Si utilizamos ppp a nivel de usuario se puede recuperar automáticamente el correo cuando se establece la conexión ppp utilizando el fichero `/etc/ppp/ppp.linkup` :

```
MYADDR:
!bg su user -c fetchmail
```

Si utilizamos sendmail (como se muestra más adelante) para entregar correo a cuentas remotas probablemente queramos que sendmail procese nuestras colas de correo tan pronto como nuestra conexión de internet se establezca. Para ello escriba el siguiente comando tras el comando de `fetchmail` que hemos escrito antes en el fichero `/etc/ppp/ppp.linkup` :

```
  !bg su user -c "sendmail -q"
```

Asumiendo que tenemos una cuenta para el usuario `usuario` en `bsd.home` . En el directorio “home” del usuario `usuario` en la máquina `bsd.home` debemos crear un fichero `.fetchmailrc` con el siguiente contenido:

```
poll ejemplo.net protocol pop3 fetchall pass Secr3To
```

Este fichero debe tener permisos de lectura sólo para el propio dueño ya que contiene la contraseña de acceso a nuestra cuenta de POP en nuestro ISP (`Secr3To`).

Para poder enviar correo con la cabecera `from:` correcta, debemos decir a sendmail que utilice `usuario@ejemplo.net` en vez de `usuario@bsd.home`. Siguiendo con nuestro ejemplo es necesario decirle a sendmail que envíe todo el correo a través de la pasarela `relay.ejemplo.net`.

El siguiente fichero de configuración `.mc` debe ser suficiente para cumplir con las anteriores tareas:

```
VERSIONID(`bsd.home.mc version 1.0')
OSTYPE(bsd4.4)dnl
FEATURE(nouucp)dnl
MAILER(local)dnl
MAILER(smtp)dnl
Cwlocalhost
Cwbsd.home
MASQUERADE_AS(`ejemplo.net')dnl
FEATURE(allmasquerade)dnl
FEATURE(masquerade_envelope)dnl
FEATURE(nocanonify)dnl
FEATURE(nodns)dnl
define(`SMART_HOST', `relay.ejemplo.net')
Dmbsd.home
define(`confDOMAIN_NAME',`bsd.home')dnl
define(`confDELIVERY_MODE',`deferred')dnl
```

En la sección anterior se explica cómo convertir este fichero `.mc` en un fichero de configuración para sendmail, `sendmail.cf`. No debemos olvidar reiniciar sendmail después de mofidificar el fichero `sendmail.cf`.

27.10. Autentificación utilizando SMTP

Escrito por James Gorham.

La autentificación mediante SMTP puede proporcionarnos diversas ventajas. Añade una capa adicional de seguridad a a sendmail y además proporciona a los usuarios móviles (usuarios que cambian de máquina) la posibilidad de mantener el mismo servidor de correo sin necesidad de reconfigurar sus agentes de usuario de correo cada vez que se trasladan.

1. Instalar security/cyrus-sasl desde los ports. Se puede encontrar dicho port en security/cyrus-sasl. security/cyrus-sasl posée varias opciones en tiempo de compilación pero para el método en particular que se va a explicar en esta sección basta con asegurarse de seleccionar la opción `pwcheck`.

2. Después de instalar security/cyrus-sasl, edite `/usr/local/lib/sasl/Sendmail.conf` (o créelo si no existe) y añada la siguiente línea:

   ```
   pwcheck_method: passwd
   ```

Este método activa la autentificación de sendmail contra nuestra base de datos de FreeBSD, `passwd`. Esto nos evita el problema de tener que crear un nuevo conjunto de usuarios y contraseñas para cada usuario que necesite validarse mediante SMTP y además nos permite mantener el mismo "login" y contraseña que los usuarios utilizan para acceder a sus cuentas para el acceso al correo electrónico.

3. Editar `/etc/make.conf` y añadir las siguientes líneas:

```
SENDMAIL_CFLAGS=-I/usr/local/include/sasl1 -DSASL
SENDMAIL_LDFLAGS=-L/usr/local/lib
SENDMAIL_LDADD=-lsasl
```

Estas líneas proporcionan a sendmail las opciones de configuración necesarias para enlazar con cyrus-sasl en tiempo de compilación. Debemos asegurarnos de que cyrus-sasl ha sido instalado correctamente recompilar sendmail.

4. Recompile sendmail utilizando el siguiente comando:

```
# cd /usr/src/usr.sbin/sendmail
# make cleandir
# make obj
# make
# make install
```

La compilación de sendmail no debería dar problemas siempre y cuando `/usr/src` no haya cambiado sustancialmente y siempre y cuando las bibliotecas compartidas necesarias se encuentren disponibles.

5. Una vez que sendmail se ha compilado y reinstalado con correctamente debemos editar el fichero `/etc/mail/freebsd.mc` (o el fichero que se utilice como `.mc` de referencia. Hay administradores que escogen utilizar la salida de hostname(1) como el nombre del fichero `.mc` que se utiliza para la configuración de sendmail por motivos de uniformidad). Añada las siguientes líneas a dicho fichero:

```
dnl set SASL options
TRUST_AUTH_MECH(`GSSAPI DIGEST-MD5 CRAM-MD5 LOGIN')dnl
define(`confAUTH_MECHANISMS', `GSSAPI DIGEST-MD5 CRAM-MD5 ↺
LOGIN')dnl
define(`confDEF_AUTH_INFO', `/etc/mail/auth-info')dnl
```

Estas opciones configuran los distintos métodos de que dispone sendmail para validar a los usuarios de correo. Si se desea utilizar otro método distinto a pwcheck por favor consulte la documentación.

6. Para terminar ejecutamos make(1) mientras nos encontramos dentro de `/etc/mail`. Este comando trata el fichero `.mc` y crea el fichero `.cf` correspondiente (con el mismo nombre que el anterior pero terminado en .cf). A continuación se utiliza el comando `make install restart`, el cual copia el fichero .cf recién generado al fichero

`sendmail.cf` y a continuación reinicia sendmail. Para más información sobre este proceso puede consultarse el contenido de `/etc/mail/Makefile` .

Si todo lo anteriormente comentado ha funcionado correctamente deberíamos ser capaces de introducir la información de nuestro "login" en nuestro cliente de correo y enviar un mensaje de prueba. Para investigar más a fondo estos temas se puede habilitar la opción `LogLevel` de sendmail al valor 13 y observar detenidamente el archivo `/var/log/maillog` en busca de posibles mensajes de error.

Puede ser necesario añadir las siguientes líneas al fichero `/etc/rc.conf` de modo que el servicio explicado en esta sección se encuentre disponible automáticamente desde el arranque:

```
sasl_pwcheck_enable="YES"
sasl_pwcheck_program="/usr/local/sbin/pwcheck"
```

Esto permite que la inicialización de SMTP_AUTH se produzca durante el arranque del sistema.

Para más información por favor visite la página autentificación SMTP de sendmail·

27.11. Agente de Correo de Usuario

Escrito por Marc Silver.

Un agente de correo de usuario (MUA en inglés Mail User Agent) es una aplicación que se utiliza para enviar y recibir correo. Coomo el correo electrónico está en constante evolución y cada vez se vuelve más complejo y con más opciones, los MUAs son cada vez más complejos y potentes. Esto permite a los usuarios disponer de mayor flexibilidad y funcionalidad. FreeBSD admite para muchísimos agentes de correo de usuario, todos los cuales pueden instalarse desde los Ports. Los usuarios pueden elegir entre lientes de correo con interfaz gráfica como evolution o balsa o entre clientes basados en consola como mutt, pine o `mail`, e incluso utilizar interfaces web.

27.11.1. mail

mail(1) es el agente de correo de usuario (MUA) que viene por defecto con FreeBSD. Es un MUA de consola que ofrece toda la funcionalidad básica necesaria para enviar y recibir correos, aunque resulta limitado limitado en su capacidad para manejar adjuntos y sólamente soporta carpetas de correo locales.

Aunque `mail` no soporta de forma nativa la interacción con servidores de correo mediante POP o IMAP estas carpetas de correo remotas pueden descargarse a un fichero `mbox` local utilizando una aplicación de descarga como fetchmail, que se describe en este mismo capítulo en (Sección 27.12, "Manejo de fetchmail").

Para enviar y recibir correo eletrónico basta con ejecutar el comando `mail`. Veamos un ejemplo:

```
% mail
```

El contenido de la carpeta de usuario en el directorio `/var/mail` se leen automáticamente. Si la carpeta se encuentra vacía la aplicación termina su ejecución con un mensaje que indica que no ha podido encontrar correo. Una vez que la carpeta ha sido leída la interfaz de la aplicación entra en funcionamiento y se muestra por pantalla un listado de los mensajes recuperados. Los mensajes se numeran automáticamente y pueden leerse como se observa en el siguiente ejemplo:

```
Mail version 8.1 6/6/93.  Type ? for help.
"/var/mail/marcs": 3 messages 3 new
>N  1 root@localhost        Mon Mar  8 14:05  14/510   "test"
 N  2 root@localhost        Mon Mar  8 14:05  14/509   "user account"
 N  3 root@localhost        Mon Mar  8 14:05  14/509   "sample"
```

Los mensajes se pueden leer utilizando el comando `t` de `mail` escribiendo a continuación el número del mensaje que queremos leer. En este ejemplo vamos a leer el primer correo:

```
% t 1
Message 1:
From root@localhost  Mon Mar  8 14:05:52 2004
X-Original-To: marcs@localhost
Delivered-To: marcs@localhost
To: marcs@localhost
Subject: test
Date: Mon,  8 Mar 2004 14:05:52 +0200 (SAST)
From: root@localhost (Charlie Root)

This is a test message, please reply if you receive it.
```

Como podemos observar en el ejemplo anterior el comando t muestra el contenido del correo eletrónico con todas sus cabeceras. Para mostrar el listado con todos los correos de nuevo, se debe utilizar la tecla h.

Si el correo eletrónico requiere una contestación se puede utilizar la aplicación `mail` para responder utilizando la tecla R o r. La tecla R indica a `mail` que conteste sólo al origen (remitente) del correo, mientras que la tecla r tanto al remitente a los otros usuarios receptores del mensaje original. Además ambos comandos se pueden ejecutar escribiendo a continuación el número que identifica al mensaje que se quiere responder. Tras esto la respuesta puede redactarse , y se debe indicar el final del mensaje mediante un punto (.) a continuación de un salto de línea. Veamos un ejemplo:

```
% R 1
To: root@localhost
Subject: Re: test

Thank you, I did get your email.
.
```

```
EOT
```

Para enviar nuevos correos eletrónicos se debe utilizar la tecla m seguida de la dirección de de correo del destinatario. Se pueden especificar varios destinatarios de correo separando cada dirección de correo con una coma (,). El asunto del mensaje de correo se puede escribir a continuación seguido por el cuerpo del mensaje. El final del mensaje se especifica como en el caso anterior, utilizando un . tras un saldo de línea y pulsando la tecla “enter”.

```
% mail root@localhost
Subject: I mastered mail

Now I can send and receive email using mail ... :)
.
EOT
```

Mientras nos encontremos dentro de la `mail` el comando ? puede utilizarse para mostrar la ayuda en línea aunque la principal fuente de información detallda sobre esta aplicación es la página man mail(1).

Nota

Tal y como se ha dicho ya la aplicación mail(1) no fue diseñada originalmente para gestionar adjuntos, por lo que su forma de gestionarlos resulta ser extremadamente mala. MUA más modernos como mutt gestionan los adjuntos de correo de una forma mucho más inteligente. Si se desea utilizar el comando `mail` el port converters/mpack le puede resultar bastante útil.

27.11.2. mutt

mutt es un agente de correo de usuario pequeño pero muy potente, funcional y con excelentes características; veamos algunas:

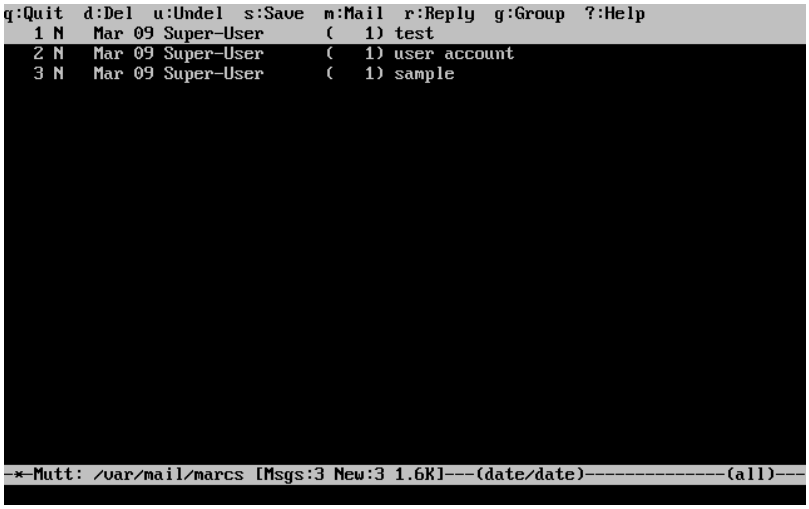
- La habilidad de agrupar mensajes en hilos.
- Soporte de PGP para cifradon y firma digital de correos eletrónicos.
- Soporte de tipos MIME.
- Soporte de gestión de correo en formato Maildir.
- Altamente configurable por el usuario.

Toda estas características hacen de mutt uno de los agentes de correo más avanzados del momento. Consulte `http://www.mutt.org` para más información sobre mutt.

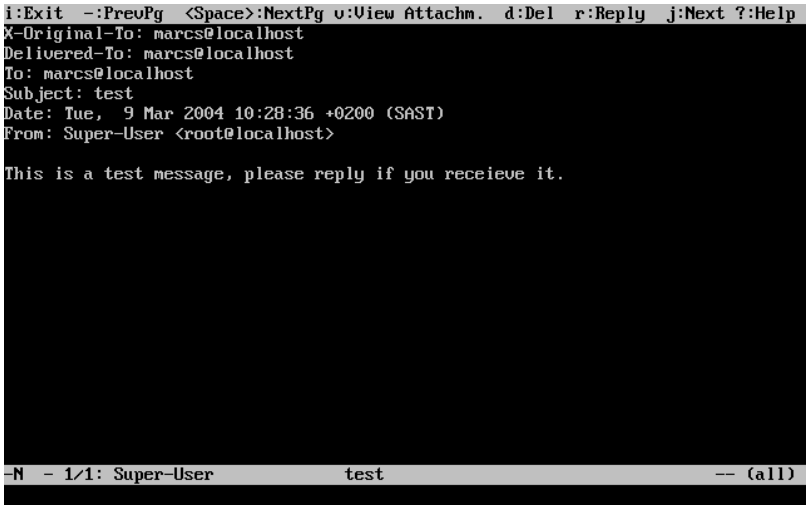
La versión estable de mutt se puede instalar usando el port mail/mutt mientras que la versión de desarrollo está en mail/mutt-devel. Una vez que se ha instalado el port, mutt puede ejecutarse mediante el siguiente comando:

```
% mutt
```

mutt lee automáticamente el contenido de la carpeta de correo del usuario dentro del directorio `/var/mail` y muestra por pantalla su contenido. Si el directorio está vacío mutt quedará a la espera de los comandos que pueda pasarle el usuario. En el ejemplo que se muestra a continuación puede verse cómo mutt facilita la lista de mensajes al usuario:



Para leer un correo basta con seleccionarlo usando las teclas de cursor, y presionando la tecla Enter. Veamos cómo muestra mutt un correo electrónico:



Al igual que ocurre con mail(1) mutt permite que los usuarios contesten al remitente de cualquier mensaje así como a los demás receptores. Para responder sólo al remitente se puede utilizar la tecla r. Para responder a un grupo, es decir a todos los receptores y al remitente del correo eletrónico pulse g.

Nota

mutt tiene vi(1) como editor por defecto para crear y responder a los mensajes de correo eletrónico. Si prefiere emplear otro editor modifique el valor de la variable `editor` en `.muttrc`.

Para escribir un mensaje de correo presione la tecla m. Después de escribir el asunto mutt ejecuta vi(1) y el cuerpo del mensaje puede escribirse. Una vez escrito el correo salga de `vi` y mutt se ejecutará de nuevo mostrando por pantalla un resumen del correo que está a punto de ser enviado. Para enviar ese correo hay que pulsar y. Este es un ejemplo de uno de esos resúmenes:

```
y:Send  q:Abort  t:To  c:CC  s:Subj  a:Attach file  d:Descrip  ?:Help
    From: Marc Silver <marcs@localhost>
      To: Super-User <root@localhost>
      Cc:
     Bcc:
 Subject: Re: test
Reply-To:
     Fcc:
Security: Clear

-- Attachments
- I     1 /tmp/mutt-bsd-c0hobscQ             [text/plain, 7bit, us-ascii, 1.1K]

-- Mutt: Compose  [Approx. msg size: 1.1K   Atts: 1]--------------------------
```

mutt también contiene una amplia ayuda a la que se accede desde la mayoría de los menús pulsando la tecla ?. La primera lína de la pantalla también muestra las teclas de método abreviado cuando es posible utilizarlas.

27.11.3. pine

pine es una aplicación de correo enfocada a los usuarios principiantes o inexpertos pero también incluye algunas características avanzadas.

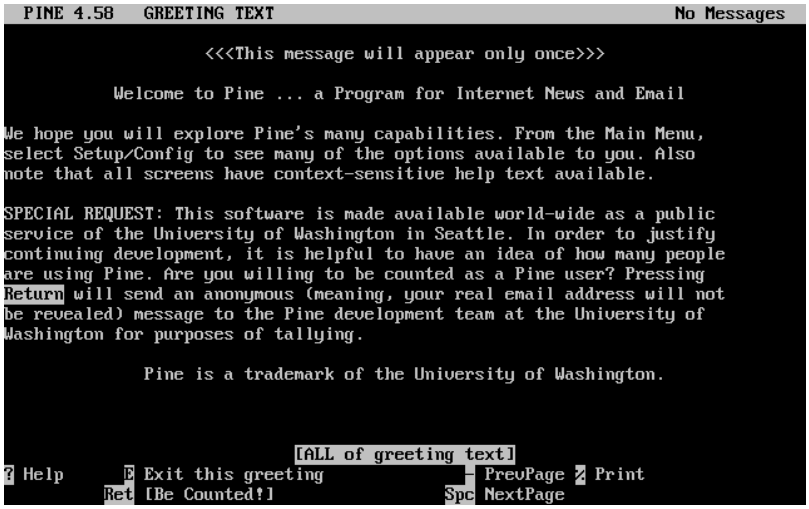
Aviso

Se han descubierto en pine varias vulnerabilidades que pueden explotarse de forma remota. Esas vulnerabilidades permiten que atacantes remotos puedan ejecutar código como si fueran usuarios locales del sistema mediante el envío de correos con un formato determinado. Todos los problemas *conocidos* se han resuelto pero el código de pine está escrito de una forma insegura y el "Security Officer" de FreeBSD opina que es probable que existan todavía vulnerabilidades sin descubrir. Si decide instalar pine debe asumir los riesgos que ello puede implicar.

La versión actual de pine se puede instalar utilizando el port mail/pine4. Una vez que se ha instalado pine se puede ejecutar mediante el siguiente comando:

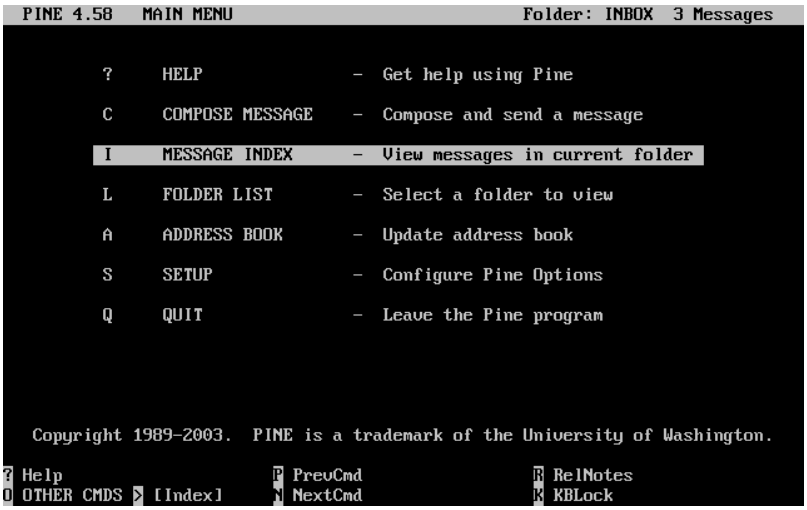
```
% pine
```

La primera vez que se ejecuta pine se muestra un mensaje de bienvenida con una pequeña introducción a la herramienta junto con una petición del equipo de desarrollo de pine en la que se solicita que se envíe un correo de forma anónima un correo de forma anónima para que puedan hacerse una idea s de cuántos usuarios están utilizando la herramienta. Para enviar dicho correo hay que presionar la tecla Enter, o bien puede pulsar la tecla E para salir de la ventana de bienvenida sin enviar dicho correo. A continuación se muestra un ejemplo de la página de bienvenida:



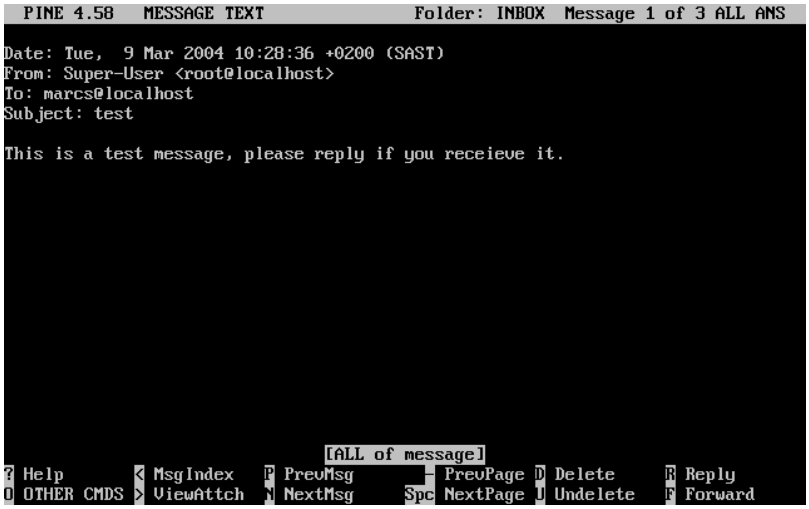
Los usuarios disponen de un menú principal, que puede navegarse utilizando las flechas. Este menú proporciona atajos para la composición de nuevos correos, para navegar a través de las carpetas de correo, e incluso para la administración de la libreta de direcciones. Justo debajo del menú principal, se muestran las teclas de método abreviado (atajos) que pueden utilizarse en cada momento.

El directorio por defecto que pine intenta abrir es `inbox`. Para ver el índice de todos los mensajes recibidos pulse la tecla I o seleccione la opción de menú denominada MESSAGE INDEX como se muestra a continuación:

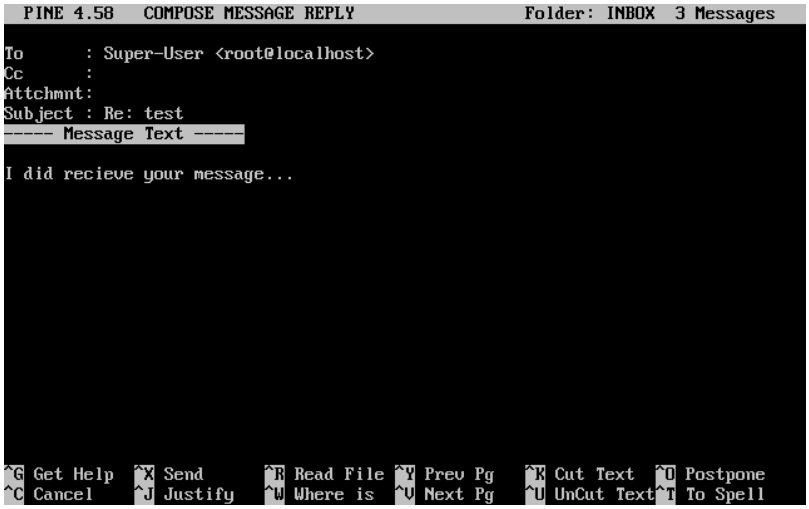


El índice muestra los mensajes en el directorio actual y puede navegarse en él utilizando las teclas del cursor. El mensaje seleccionado se puede leer presionando la tecla Enter.

En la captura de pantalla que se muestra a continuación se muestra un mensaje de ejemplo. Las teclas de atajo se muestran como referencia en la parte baja de la pantalla. Un ejemplo de dichas teclas de método abreviado es la tecla r que permite responder al mensaje que se muestra en dicho momento.



La contestación a un mensaje de correo eletrónico en pine se realiza mediante el editor pico, que se instala por defecto junto con pine. pico permite navegar de forma sencilla por los distintos mensajes de correo y es ligeramente más sencilla de manejar que vi(1), sobre todo para los usuarios noveles. Una vez que se ha escrito la réplica al correo se envía pulsando la tecla Ctrl+X. pine pedirá confirmación antes de enviarlo.



pine puede configurarse utilizando la entrada SETUP del menú principal. Se ruega consultar `http://www.washington.edu/pine/` para obtener más información.

27.12. Manejo de fetchmail

Escrito por Marc Silver.

fetchmail es un cliente de IMAP y POP que permite a los usuarios descargar automáticamente el correo de cuentas remotas en servidores IMAP y POP y almacenarlos en carpetas de correo locales; una vez en local, se puede acceder a los correos de una forma más sencilla y utilizando multitud de programas cliente. fetchmail se puede instalar a partir del port mail/fetchmail. Veamos algunas de sus características más útiles:

- Suporte de POP3, APOP, KPOP, IMAP, ETRN y ODMR.
- Puede reenviar correo utilizando SMTP lo que permite que las reglas de filtrado, reenvío y "aliasing" funcionen correctamente.
- Se puede ejecutar en modo dæmon comprobar periódicamente el correo entrante.
- Puede recuperar correo de múltiples carpetas y reenviarlos, en función de la configuración establecida, a varios usuarios locales.

Queda fuera del objetivo de este documento explicar todas las características de fetchmail pero algunas de ellas se exponen a ontinuación. fetchmail usa un fichero de configuración denominado `.fetchmailrc`. Este fichero incluye información sobre el servidor de correo remoto y los datos necesarios para poder hacer login en él. Debido a la naturaleza sensible de la información que se almacena en dicho fichero se recomienda modificar los permisos para que sea de sólo sea legible por su propietario. Lo conseguirá mediante el siguiente comando:

```
% chmod 600 .fetchmailrc
```

El siguiente ejemplo muestra un fichero de configuración `.fetchmailrc`. Este ejemplo sirve para automatizar la descarga del correo de un determinado usuario mediante POP. El fichero de configuración hace que fetchmail se conecte a `ejemplo.com` utilizando como nombre de usuario `joesoap` y como contraseña XXX. En el ejemplo se asume que el usuario `joesoap` también es un usuario válido en el sistema local.

```
poll ejemplo.com protocol pop3 username "joesoap" password "XXX"
```

El siguiente ejemplo permite a fetchmail conectarse a múltiples servidores POP e IMAP y redirige los correos a diferentes usuarios locales en función de la configuración establecida:

```
poll ejemplo.com proto pop3:
user "joesoap", with password "XXX", is "jsoap" here;
user "andrea", with password "XXXX";
poll ejemplo.net proto imap:
user "john", with password "XXXXX", is "myth" here;
```

fetchmail se puede ejecutar en modo dæmon mediante el parámetro `-d` seguido seguido por un intervalo de tiempo (expresado en segundos) que indica cada cuánto tiempo debe fetchmail interrogar a los distintos servidores listados en `.fetchmailrc`. El siguiente ejemplo hace que fetchmail interroge cada 60 segundos:

```
% fetchmail -d 60
```

Se puede encontrar más información sobre fetchmail en `http://www.catb.org/~esr/fetchmail/`.

27.13. Uso de procmail

Escrito por Marc Silver.

procmail es una aplicación increíblemente potente que se utiliza para filtrar el correo de entrada. Permite a los usuarios definir "reglas" que se asocian con correos entrantes y que realizan funciones concretas, como reencaminar el correo a carpetas o direciones alternativas. procmail se puede instalar utilizando el port mail/procmail. Una vez instalado, se puede integrar directamente en la mayoría de los MTAs; por favor, consulte la documentación del MTA que utilice para saber más sobre la integración entre ambos. Por otro lado procmail se puede integrar con el MTA que prefiera de una forma sencilla añadiendo la siguiente línea al fichero `.forward` dentro del directorio home del usuario que desée usar procmail:

```
"|exec /usr/local/bin/procmail || exit 75"
```

La siguiente sección muestra algunas reglas básicas de procmail, junto con una breve descripción de las acciones que realizan. Estas reglas, y muchas otras se deben insertar dentro del fichero `.procmailrc` ubicado en el directorio home del usuario.

En la página man de "procmailex" se explica la mayoría de estas reglas.

Reenvío de todo el correo proveniente de `usuario@ejemplo.com` hacia la dirección externa `correodefiar@ejemplo.com` :

```
:0
* ^From.*usuario@ejemplo.com
! correodefiar@ejemplo.com
```

Reenvío de todos los correos que ocupen menos de 1000 bytes a la dirección `corredefiar@ejemplo2.com` :

```
:0
* < 1000
! correodefiar@ejemplo2.com
```

Envío de todos los correos dirigidos a `opcional@ejemplo.com` hacia una carpeta de correo llamada `opcional`:

```
:0
* ^TOopcional@ejemplo.com
opcional
```

Envío de todos los correos con un asunto que contenga la palabra “Spam” al dispositivo `/dev/null`:

```
:0
^Subject:.*Spam
/dev/null
```

Una útil receta para examinar mensajes de correo provenientes de listas de distribución de `FreeBSD.org` y poner cada mensaje en el directorio apropiado en función del origen del mensaje:

```
:0
* ^Sender:.owner-freebsd-\/[^@]+@FreeBSD.ORG
{
 LISTNAME=${MATCH}
 :0
 * LISTNAME??^\/[^@]+
 FreeBSD-${MATCH}
}
```

Capítulo 28. Servidores de red

Reorganizado por Murray Stokely.

28.1. *

Pendiente de traducción. Este capítulo incluye también parte del contenido del Capítulo 29, *Networking avanzado*, "networking" avanzado.

Capítulo 29. Networking avanzado

Networking avanzado

29.1. Resumen

Este capítulo cubre algunos de los servicios de red que se usan con más frecuencia en sistemas UNIX®. Para ser más concretos este capítulo explica cómo definir, ejecutar, probar y mantener todos los servicios de red que utiliza FreeBSD. Se muestran además ejemplos de ficheros de configuración que podrá utilizar para sus propios quehaceres.

Después de leer este capítulo habremos aprendido:

- Los conceptos básicos de pasarelas y "routers".
- Cómo poner en funcionamiento dispositivos IEEE 802.11 y Bluetooth®.
- Cómo configurar FreeBSD para que actúe como un "bridge".
- Cómo poner en funcionamiento un sistema de ficheros en red con NFS.
- Cómo realizar un arranque del sistema por red en máquinas sin disco duro.
- Cómo ejecutar un servidor de información en red para compartir cuentas de usuario mediante NIS.
- Cómo especificar parámetros de configuración automática de red utilizando DHCP.
- Cómo ejecutar un servidor de nombres de dominio.
- Cómo sincronizar la hora y la fecha y ejecutar un servidor horario utilizando el protocolo NTP.
- Cómo ejecutar un servicio de traducción de direcciones de red.
- Cómo gestionar el dæmon inetd.
- Cómo conectar dos computadoras a través de PLIP.
- Cómo habilitar IPv6 en una máquina FreeBSD.
- Cómo configurar ATM sobre FreeBSD 5.X.

Antes de leer este capítulo debería usted:

- Intentar comprender los conceptos básicos de los scripts de `/etc/rc` .

- Familiarizarse con la terminología básica de redes.

29.2. Pasarelas y “routers”

Escrito por Coranth Gryphon.

Para que una máquina sea capaz de encontrar otra máquina remota a través de la red debe existir mecanismo que describa cómo llegar del origen al destino. Este mecanismo se demonina *routing* o *encaminamiento*. Una “ruta” es un par definido de direcciones: una dirección de “destino” y una dirección de “pasarela”. Éste par indica que para llegar a dicho *destino* debe efectuarse una comunicación previa con dicha *pasarela*. Exiten tres tipos distintos de destinos: máquinas individuales, subredes y “por defecto”. La “ruta por defecto” se utiliza sólamente cuando no se puede aplicar ninguna de las otras rutas existentes. El tema de las rutas por defecto se tratará más adelante con más detalle. También existen tres tipos de pasarelas distintas: máquinas individuales, interfaces (también llamados “enlaces”) y direcciones hardware de ethernet (direcciones MAC).

29.2.1. Ejemplo

Para ilustrar diferentes aspectos del sistema de encaminamiento veamos el siguiente ejemplo obtenido mediante `netstat`.

```
% netstat -r
Routing tables

Destination      Gateway            Flags     Refs     Use     ↺
Netif Expire

default          outside-gw         UGSc       37      418      ppp0
localhost        localhost          UH          0      181       lo0
test0            0:e0:b5:36:cf:4f   UHLW        5    63288       ed0
    77
10.20.30.255     link#1             UHLW        1     2421
ejemplo.com      link#1             UC          0        0
host1            0:e0:a8:37:8:1e    UHLW        3     4601       lo0
host2            0:e0:a8:37:8:1e    UHLW        0        5       ↺
lo0 =>
host2.ejemplo.com link#1            UC          0        0
224              link#1             UC          0        0
```

Las primeras dos líneas especifican la ruta por defecto (la cual se comenta en la siguiente sección) y la ruta de `máquina local`.

La interfaz (columna `Netif`) que especifica esta tabla de rutas para el destino `localhost` se denomina `lo0`, y también se conoce como el dispositivo de “ loopback” a de bucle de retorno. Esto viene a decir que el tráfico no debe entregarse a la red puesto que dicho tráfico va destinado a la misma máquina que lo originó.

Lo siguiente que podemos observar son las direcciones que comienzan por `0:e0:`. Son direcciones hardware de Ethernet, llamadas también se direcciones MAC. FreeBSD identifica automáticamente cualquier máquina (`test0` en el ejemplo anterior) que se encuentre en la red local y crea una ruta del estilo que estamos comentando, para entregar el tráfico directamente a través del correspondiente interfaz Ethernet, en este caso `ed0`. Existe también un contador (`Expire`) asociado con este tipo de rutas que se usa para borrarlas cuando dicho contador expira. Las rutas para las máquinas de nuestra propia red de de área local se crean dinámicamente utilizando el protocolo ARP (Address Resolution Protocol o Protocolo de Resolución de Direcciones), que se encarga de averiguar la dirección MAC que se corresponde con la dirección IP de la máquina destino.

FreeBSD tambíen utiliza rutas de subred para direccionar la subred local (`10.20.30.255` es la dirección de broadcast para la subred `10.20.30`, y `ejemplo.com` es el nombre de dominio asociado con dicha subred.) La notación `link#1` se refiere a la primera tarjeta Ethernet de la máquina. En este tipo de redes no se especifica ningún interfaz en el campo de `Netif`.

Las rutas de subredes aparecen cuando se asigna una dirección IP a una interfaz, utilizando una máscara de red. También se pueden aprender dinámicamente utilizando demonios de encaminamiento, como routed. Por último estas rutas pueden crearse manualmente de forma explícita; es lo que se conoce con el nombre de rutas estáticas.

La línea de `host1` se refiere a nuestra máquina, que el sistema identifica por la correspondiente dirección Ethernet de la tarjeta de red. FreeBSD sabe que debe utilizar la interfaz de loopback (`lo0`) en vez de enviar los paquetes a a través de red.

Las dos líneas que comienzan por `host2` son ejemplos del uso de alias de ifconfig(8) alias (consultar la sección sobre Ethernet para averiguar por qué nos podría interesar hacer esto.) El símbolo `=>` después de la interfaz `lo0` especifica que no sólo estamos utilizando la interfaz de loopback, si no que además especifica que se trata de un alias. Estas rutas sólo aparecen en las máquinas que implementan el alias, el resto de las máquinas de la subred local solamente poseerán una línea `link#1` para dichas rutas.

La última línea (destino de subred `224`) trata sobre encaminamiento multicast, que cubriremos en otra sección.

Finalmente, se pueden observar varios atributos relacionados con las rutas en la columna de `Flags`. A continuación se muestra una pequeña tabla con el significado de algunos de esos de los atributos o “flags”.

U	Up: La ruta está activa.
H	Host: El destino de la ruta es una única máquina.
G	Gateway: Envía cualquier cosa para éste destino a través de la pasarela especificada,

	la cual decidirá cómo encaminar el paquete hasta que eventualmente se alcance el destino.
S	Static: Esta ruta se configuró manualmente, y no se ha generado de forma automática por el sistema.
C	Clone: Genera una nueva ruta para la máquina a la que nos queremos conectar basándose en la ruta actual. Este tipo de ruta se utiliza normalmente en redes locales.
W	WasCloned: Indica una ruta que se auto-configuró basándose en una ruta de red de área local con etiqueta Clone.
L	Link: Esta ruta posée referencias a hardware de Ethernet.

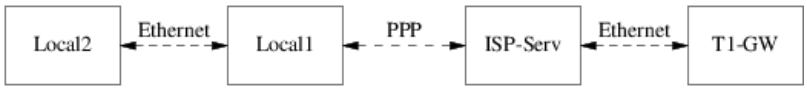
29.2.2. Rutas por defecto

Cuando el sistema local necesita realizar una conexión con una máquina remota se examina la tabla de rutas para determinar si se conoce algún camino para llegar al destino. Si la máquina remota pertenece a una subred que sabemos cómo alcanzar (rutas clonadas) entonces el sistema comprueba si se puede conectar utilizando dicho camino.

Si todos los caminos conocidos fallan al sistema le queda una única opción: la "ruta por defecto". Esta ruta está constituída por un tipo especial de pasarela (normalmente el único "router" presente en la red área local) y siempre posée el "flag" `c` en el campo de "flags". En una LAN, la pasarela es la máquina que posée conectividad con el resto de las redes (sea a través de un enlace PPP, DSL, cable modem, T1 u otra interfaz de red.)

Si se configura la ruta por defecto en una máquina que está actuando como pasarela hacia el mundo exterior la ruta por defecto será el "router" que se encuentre en posesión del proveedor de servicios de internet (ISP).

Vamos a examinar un ejemplo que utiliza rutas por defecto. A continuación se muestra una configuración bastante común:



Las máquinas `Local1` y `Local2` se encuentran en nuestro sitio u organización. `Local1` se conecta con un ISP a través de una conexión de modem PPP. El servidor PPP del ISP se conecta a través de una red de área local a otra pasarela utilizando una interfaz externa.

Las rutas por defecto para cada una de las máquinas son las siguientes:

Host	Default Gateway	Interface
Local2	Local1	Ethernet
Local1	T1-GW	PPP

Una pregunta bastante frecuente es "?Por qué (o cómo) hacer que la máquina `T1-GW` sea el 'router' por defecto para `Local1` en vez de que sea el servidor del ISP al cual se está conectando?".

Recordemos que, como la interfaz PPP está utilizando una dirección de la red local del ISP en nuestro lado de la las rutas para cualquier otra máquina en la red local del proveedor se generarán de forma automática. De este ya sabemos el modo de alcanzar la máquina `T1-GW`, de tal forma que no se necesita un paso intermedio para enviar tráfico al servidor del ISP.

Es frecuente utilizar la dirección `X.X.X.1` como la dirección de la pasarela en la red local. Siguiendo con el ejemplo anterior, si nuestro espacio de direccionamiento local fuera la clase C `10.20.30` y nuestro ISP estuviera utilizando `10.9.9` las rutas por defecto serían:

Host	Default Route
Local2 (10.20.30.2)	Local1 (10.20.30.1)
Local1 (10.20.30.1, 10.9.9.30)	T1-GW (10.9.9.1)

Se puede especificar fácilmente la entrada de la ruta por defecto utilizando el fichero `/etc/rc.conf` . En nuestro ejemplo en la máquina `Local2`, se añadió la siguiente línea en dicho fichero:

```
defaultrouter="10.20.30.1"
```

También se puede hacer directamente desde la línea de órdenes mediante route(8):

```
# route add default 10.20.30.1
```

Para obtener más información sobre la manipulación de tablas de rutas se ruega consultar la página de manual route(8).

29.2.3. Máquinas con doble pertenencia (Dual Homed Hosts)

Existe otro tipo de configuración que debemos describir y que se produce cuando una máquina se sitúa en dos redes distintas al mismo tiempo. Técnicamente hablando cualquier máquina que actúa como pasarela (en el caso anterior utilizando un enlace de PPP) pertenece al tipo de máquinas con doble pertenencia, pero normalmente el término sólo se aplica para describir máquinas que se encuentran directamente conectadas con dos redes de área local.

En un caso la máquina posée dos tarjetas de red Ethernet, cada una de ellas con una dirección de red independiente. En otro caso la máquina puede tener sólo una tarjeta de red,

pero utilizar “ aliasing” (ifconfig(8)). El primer caso se utiliza cuando se necesita usar dos redes Ethernet al mismo tiempo mientras que el segundo caso se utiliza cuando se dispone de un único segmento de red físico pero se han definido dos redes lógicas distintas

En cualquier caso la tabla de rutas se construye de tal forma que cada subred sepa que la máquina es la pasarela definida definida (“inbound route”) para la otra subred. Ésta configuración en la que la máquina actúa como “router” entre las dos subredes se usa a menudo cuando queremos implementar filtrado de paquetes o cortafuegos seguridad en un sentido o en ambos.

Si queremos que dicha máquina encamine paquetes entre las dos interfaces es necesario decirle a FreeBSD que active dicha funcionalidad. En la siguiente sección se explica cómo hacerlo.

29.2.4. Construcción de un “route”

Un “router” de red, también llamado pasarela o “route”, es simplemente un sistema que reenvía paquetes desde un interfaz hacia otro interfaz. Los estándares Internet y el sentido común aplicado a la ingeniería de redes impiden que FreeBSD incluya por defecto ésta característica. Se puede activar cambiando a `YES` el valor de la siguiente variable en el fichero rc.conf(5):

```
gateway_enable=YES          # Set to YES if this host will be a ↺
gateway
```

Esta opción modificará la variable de sysctl(8) `net.inet.ip.forwarding` al valor `1`. Si en algún momento se necesita detener el “router” de forma temporal basta con asignar a dicha variable el valor `0`. Consulte sysctl(8) para más detalles.

Nuestro recién activado “router” necesita rutas para saber a dónde debe enviar el tráfico recibido. Si nuestra red es ña se pueden definir rutas estáticas. FreeBSD incluye por defecto el dæmon de encaminamiento BSD, routed(8), que admite RIP (versión 1 y versión 2) e IRDP. El paquete net/zebra le permitirá usar otros protocolos de encaminamiento dinámico como BGP v4, OSPF v2 y muchos otros. En caso de necesitar características avanzadas de gestión puede usted recurrir a productos comerciales como GateD®.

Incluso cuando FreeBSD se configura del modo descrito no se cumple completamente con los estándares de Internet respecto a los “routers”. Bastará no obstante para poder usarse.

29.2.5. Configuración de rutas estáticas

Escrito porAl Hoang.

29.2.5.1. Configuración manual

Vamos a suponer que tenemos la siguiente topología de red:

```
    INTERNET
      | (10.0.0.1/24) Router por defecto para Internet
      |
      |Interfaz xl0
      |10.0.0.10/24
   +------+
   |      | RouterA
   |      | (pasarela FreeBSD)
   +------+
      | Interfaz xl1
      | 192.168.1.1/24
      |
  +--------------------------------+
   Red Interna 1       | 192.168.1.2/24
                       |
                  +------+
                  |      | RouterB
                  |      |
                  +------+
                      | 192.168.2.1/24
                      |
                    Red Interna 2
```

En este escenario `RouterA` es nuestra máquina FreeBSD que actúa como pasarela para acceder al resto de internet. Tiene una ruta por defecto que apunta a `10.0.0.1` que le permite conectarse con el mundo exterior. Vamos a suponer también que `RouterB` se encuentra configurado de forma adecuada que sabe cómo llegar a cualquier sitio que necesite. Esto es sencillo viendo nuestra topología de red, basta con añadir una ruta por defecto en la máquina `RouterB` utilizando `192.168.1.1` como "router".

Si observamos la tabla de rutas de `RouterA` veremos algo como lo siguiente:

```
% netstat -nr
Routing tables

Internet:
Destination        Gateway            Flags    Refs      Use  Netif
  Expire
default            10.0.0.1           UGS         0    49378    xl0
127.0.0.1          127.0.0.1          UH          0        6    lo0
10.0.0/24          link#1             UC          0        0    xl0
192.168.1/24       link#2             UC          0        0    xl1
```

Con la tabla de rutas actual `RouterA` no es capaz de alcanzar la red interna 2. Esto es así porque no posee ninguna ruta para la red `192.168.2.0/24` . Una forma de mitigar esto es añadir de forma manual la ruta que falta. La siguiente orden añade la red interna 2 a la tabla de rutas de la máquina `RouterA` utilizando `192.168.1.2` como siguiente salto:

```
# route add -net 192.168.2.0/24 192.168.1.2
```

Ahora `RouterA` puede alcanzar cualquier máquina en la red `192.168.2.0/24` .

29.2.5.2. Cómo hacer la configuración persistente

El ejemplo anterior es perfecto en tanto que resuelve el problema de encaminamiento entre redes pero existe un problema. La información de encaminamiento desaparecerá si se reinicia la máquina. La forma de evitarlo es añadir las rutas estáticas a `/etc/rc.conf` :

```
# Añade la red interna 2 como una ruta estática
static_routes="redinterna2"
route_internalnet2="-net 192.168.2.0/24 192.168.1.2"
```

La variable de configuración `static_routes` es una lista de cadenas separadas por espacios. Cada cadena identifica un nombre para la ruta que se desea definir. En el ejemplo anterior sólamente se dispone de una cadena dentro de la variable `static_routes` . Esta cadena es *`redinterna2`*. A continuación se añade otra variable de configuración denominada `route_redinterna2` donde se escriben todos los parámetros de configuración que normalmente utilizaríamos normalmente utilizaríamos con route(8). En el ejemplo que estamos comentando se utilizaría la siguiente orden:

```
# route add -net 192.168.2.0/24 192.168.1.2
```

De tal forma que la variable debería contener `"-net 192.168.2.0/24 192.168.1.2"` .

Como ya se ha comentado anteriormente podemos especificar más de una cadena en la variable `static_routes` .Esto nos permite crear varias rutas estáticas. Las siguientes línas muestran un ejemplo donde se añaden rutas estáticas para las redes `192.168.0.0/24` y `192.168.1.0/24` en un “router”imaginario:

```
static_routes="red1 red2"
route_red1="-net 192.168.0.0/24 192.168.0.1"
route_red2="-net 192.168.1.0/24 192.168.1.1"
```

29.2.6. Propagación de rutas

Ya hemos comentado cómo se definen las rutas para el mundo exterior pero no hemos comentado nada sobre cómo haremos que el mundo exterior nos encuentre a nosotros.

También hemos aprendido que las tablas de rutas se pueden construír de tal forma que un grupo de tráfico (perteneciente a un espacio de direcciones determinado) se reenvíe a una máquina específica de la red, que se encargará de reenviar los paquetes hacia adentro.

Cuando se obtiene un espacio de direcciones para la organización el proveedor de servicios modifica sus tablas de rutas para que todo el tráfico para nuestra subred se encamine a través del enlace PPP hasta alcanzarnos. Pero ?cómo conocen las organizaciones dispersas a través del país que deben enviar los paquetes dirigidos a nosotros hacia nuestro ISP?

Existe un sistema (muy similar al sistema de nombres de dominio, DNS) que se encarga de controlar todos los espacios de direcciones que se encuentran actualmente repartidos y que además define sus puntos de conexión con el “backbone” de internet. El “backbone”

está formado por las principales líneas de de comunicacion que se encargan de transportar el tráfico de internet a través del país y del mundo entero. Cada máquina del "backbone" dispone de una copia de un conjunto maestro de tablas de rutas gracias a las cuales pueden dirigir el tráfico para una red particular hacia una determinada red de transporte de dicho "backbone". Una vez en la red de transporte adecuada el tráfico se encamina a través de un número indeterminado de redes de proveedores de servicio hasta que se alcanza la red de destino final.

Una de las tareas que debe realizar el proveedor de servicio servicio consiste en anunciarse a las organizaciones del consiste en anunciarse a las organizaciones del "backbone" como el punto de conexión principal (y por tanto como el camino de entrada) para alcanzar las redes de sus clientes. Este proceso se denomina propagación de rutas.

29.2.7. Solución de problemas

En algunas ocasiones surgen problemas con la propagación de las rutas y algunas organizaciones son incapaces de conectarse con nuestra subred. Quizá la orden más útil para averiguar dónde se está interrumpiendo el sistema de encaminamiento sea traceroute(8). Se puede usar también cuando somos nosotros los que no podemos alcanzar alguna red externa (por ejemplo cuando ping(8) falla).

traceroute(8) se ejecuta pasandole como parámetro el nombre de la máquina remota a la que nos queremos conectar. Esta orden muestra por pantalla lás máquinas que actúan de pasarela a lo largo del camino. El proceso termina bien porque se alcanza el destino o bien porque algún "router" intermedio no puede conectarse con el siguiente salto, o lo desconoce.

Si quiere saber más sobre esto consulte la página man de traceroute(8).

29.2.8. Rutas multicast

FreeBSD soporta tanto aplicaciones multicast como encaminamiento multicast de forma nativa. Las aplicaciones multicast no necesitan ninguna configuración especial en FreeBSD; estas aplicaciones se ejecutan tal cual. El encaminamiento multicast necesita para ser usado que se compile dicho soporte en el núcleo de FreeBSD:

```
options MROUTING
```

Se debe configurar además el dæmon de encaminamiento multicast, mrouted(8), para establecer túneles y ejecutar DVMRP utilizando `/etc/mrouted.conf` . Se pueden encontrar más detalles sobre cómo realizar una configuración de multicast en mrouted(8).

29.3. Redes sin cables ("wireless")

Escrito por Eric Anderson.

29.3.1. Introducción

Puede resultar muy útil el ser capaz de utilizar una computadora sin la molestia de tener un cable de red colgando de la máquina en todo momento. FreeBSD puede utilizarse como un cliente de “wireless” e incluso como un “punto de acceso”.

29.3.2. Modos de operación Wireless

Existen dos formas diferentes de configurar dispositivos wireless 802.11: BSS e IBSS.

29.3.2.1. Modo BSS

El modo BSS es el que se utiliza normalmente. Este modo también se denomina modo infraestructura. En esta configuración se conectan un determinado número de puntos de acceso a una red cableada. Cada red Cada red “wireless” posée su propio nombre. Este nombre es el SSID de la red.

Los clientes “wireless” se conectan a estos puntos de acceso. El estándar IEEE 802.11 define el protocolo que se utiliza para realizar esta conexión. Un cliente “wireless” puede asociarse con una determinada red “wireless” especificando el SSID. Un cliente “wireless” también puede asociarse a cualquier red que se encuentre disponible; basta con no especificar ningún SSID.

29.3.2.2. Modo IBSS

El modo IBSS, también conocido como modo ad-hoc, se ha diseñado para facilitar las conexiones punto a punto. En realidad existen dos tipos distintos de modos ad-hoc. Uno es el modo IBSS, también conocido como modo ad-hoc o modo ad-hoc del IEEE. Este modo se encuentra especificado en el estándar IEEE 802.11. El segundo tipo se denomina modo ad-hoc de demostración o modo ad-hoc de Lucent (y algunas veces, también se le llama simplemente modo ad-hoc, lo cual es bastante confuso). Este es el modo de funcionamiento antíguo, anterior al estándar 802.11, del modo ad-hoc debería utilizarse sólo en instalaciones propietarias. No profundizaremos más sobre estos modos de funcionamiento.

29.3.3. Modo infraestructura

29.3.3.1. Puntos de acceso

Los puntos de acceso son dispositivos de red “wireless” que funcionan de forma equivalente a los “hubs” o concentradores, permitiendo que varios clientes “ wireless” se comuniquen entre sí. A menudo se utilizan varios puntos de acceso para cubrir un área determinada como una casa, una oficina u otro tipo de localización delimitada.

Los puntos de acceso poseen típicamente varias conexiones de red: la tarjeta “wireless” y una o más tarjetas Ethernet que se utilizan para comunicarse con el resto de la red.

Los puntos de acceso se pueden comprar como tales pero también se puede configurar un sistema FreeBSD para crear nuestro propio punto de acceso “wireless” utilizando un determinado tipo de tarjetas “wireless” que poseen tales capacidades de configuración. Existe una gran cantidad de fabricantes de hardware que distribuyen puntos de acceso y tarjetas de red “wireless”, aunque las capacidades de unos y otras varín.

29.3.3.2. Construcción de un punto de acceso basado en FreeBSD

29.3.3.2.1. Requisitos

Para crear nuestro propio punto de acceso con FreeBSD debemos utilizar un determinado tipo de tarjeta “wireless”. Por el momento, sólo las tarjetas con el chip Prism nos permiten hacer un punto de acceso. También vamos a necesitar una tarjeta para red cableada que sea soportada por el sistema (esto no es muy complicado dada la ingente cantidad de dispositivos de este tipo que funcionan en FreeBSD). Para este ejemplo vamos a suponer que queremos puentear (bridge(4)) todo el tráfico entre la red cableada y la red inalámbrica.

El uso como punto de acceso “wireless” (también denominado *hostap*) funciona mejor con determinadas versiones del “ firmware”. Las tarjetas con chip Prism2 deben disponer de la versión 1.3.4 (o superior) del “ firmware”. Los chips Prism2.5 y Prism3 deben disponer de la versión 1.4.9 o superior del “firmware”. Las versiones más antíguas de estos “ firmwares” pueden no funcionar correctamente. A día de hoy la única forma de actualizar el “ firmware” de las tarjetas es usando las herramientas que proporciona el fabricante para Windows®.

29.3.3.2.2. Puesta en marcha del sistema

Primero debemos asegurarnos de que el sistema reconoce la tarjeta “wireless”:

```
# ifconfig -a
wi0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet6 fe80::202:2dff:fe2d:c938%wi0 prefixlen 64 scopeid 0x7
        inet 0.0.0.0 netmask 0xff000000 broadcast 255.255.255.255
        ether 00:09:2d:2d:c9:50
        media: IEEE 802.11 Wireless Ethernet autoselect (DS/2Mbps)
        status: no carrier
        ssid ""
        stationname "nodo Wireless FreeBSD"
        channel 10 authmode OPEN powersavemode OFF powersavesleep 100
        wepmode OFF weptxkey 1
```

No se preocupe si no entiende algo de la configuración anterior, lo importante es asegurarse de que el sistema muestra algo parecido, lo cual nosindicará que la tarjeta “wireless” ha sido correctamente reconocida por FreeBSD. Si el interfaz inalámbrico no es reconocido correctamente y se está utilizando una tarjeta PC Card consulte pccardc(8) y pccardd(8), en las que tiene mucha información al respecto.

A continuación, para que podamos disponer de un “bridge” deberá cargar el módulo del kernel bridge(4) por el sencillo procedimiento de ejecutar la siguiente orden:

```
# kldload bridge
```

No debería aparecer mensaje de error alguno al ejecutar dicha orden. Si apareciera alguno quizás deba compilar el kernel del sistema con bridge(4) incluído. La sección Bridging de éste manual incluye información abundante para llevar a buen puerto esa tarea.

Una vez que tenemos el soporte de "bridging" cargado debemos indicar a FreeBSD qué interfaces se desean puentear. Para ello emplearemos sysctl(8):

```
# sysctl net.link.ether.bridge=1
# sysctl net.link.ether.bridge_cfg="wi0,xl0"
# sysctl net.inet.ip.forwarding=1
```

En FreeBSD 5.2-RELEASE y posteriores se deben emplear las siguientes opciones en lugar de las anteriormente expuestas:

```
# sysctl net.link.ether.bridge.enable=1
# sysctl net.link.ether.bridge.config="wi0,xl0"
# sysctl net.inet.ip.forwarding=1
```

Ahora es el momento de configurar la tarjeta de red inalámbrica. La siguiente orden convierte la tarjeta en un punto de acceso:

```
# ifconfig wi0 ssid mi_red channel 11 media DS/11Mbps mediaopt hostap ↺
up stationname "PA FreeBSD"
```

La línea de ifconfig(8) levanta el interfaz `wi0`, configura el SSID con el valor de *`mi_red`* y también el nombre de la estación como *`FreeBSD`*. La opción `media DS/11Mbps` configura la tarjeta a 11Mbps. Ésto es necesario para que cualquier valor que se necesite asignar a `mediaopt` surta efecto. La opción `mediaopt hostap` sitúa el interfaz en modo punto de acceso. La opción `channel 11` configura la tarjeta para que use el canal de radio número 11. En wicontrol(8) encontraráa rangos de canales válidos para varios dominios regulatorios. Por favor, tenga en cuenta que no todos los canales son legales en todos los países.

Despues de esto deberíamos disponer de un punto de acceso completamente funcional y en ejecución. Le animamos a consultar wicontrol(8), ifconfig(8) y wi(4) para máss información.

También le recomemdamos leer la sección sobre cifrado que econtrará más adelante.

29.3.3.2.3. Información de estado

Una vez que el punto de acceso estáconfigurado resulta interesante poder obtener información acerca de los clientes que estén asociados. La persona encargada de la administración del punto de acceso puede ejecutar cuando estime oportuno lo siguiente:

```
# wicontrol -l
```

```
1 station:
00:09:b7:7b:9d:16  asid=04c0, flags=3<ASSOC,AUTH>, caps=1<ESS>, ↺
rates=f<1M,2M,5.5M,11M>, sig=38/15
```

Lo que aquí se muestra indica que hay una única estación asociada y nos suministra sus parámetros. Los valores de señal que se muestran se deben tomar sólo como indicaciones aproximadas de la fuerza de dicha señal. Su traducción a dBm u otras unidades varía según la versión del " firmware" de la tarjeta que se use.

29.3.3.3. Clientes

Un cliente "wireless" es un sistema que se comunica con un punto de acceso o directamente con otro cliente "wireless".

Generalmente los clientes "wireless" sólo poseen un dispositivo de red: la tarjeta de red inalámbrica.

Existen varias formas de configurar un cliente " wireless" basadas en los distintos modos inalámbricos, normalmente reducidos a BSS (o modo infraestructura, que requiere de un punto de acceso) y el modo IBSS (modo ad-hoc, o modo punto a punto). En nuestro ejemplo usaremos el más famoso de ambos, el BSS, para comunicarnos con un punto de acceso.

29.3.3.3.1. Requisitos

Sólamente existe un requisito real para configurar un sistema FreeBSD como cliente inalámbrico: usar una tarjeta de red inalámbrica soportada por el sistema.

29.3.3.3.2. Ejecución de un cliente inalámbrico FreeBSD

Para utilizar una red inalámbrica se necesitan conocer algunos conceptos básicos de redes de redes wireless. En nuestro ejemplo queremos conectarnos a la red inalámbrica *mi_red* y queremos hacerlo con el soporte de cifrado desactivado.

Nota

En este ejemplo no se utiliza cifrado, lo cual resulta ser bastante peligroso. En la próxima sección aprenderemos cómo activar el sistema de cifrado común el los dispositivos inalámbricos, por qué resulta importante hacerlo y por qué algunas tecnologías de cifrado no son suficientes para protegernos completamente.

Asegúrese de que FreeBSD reconoce su tarjeta de red inalámbrica:

```
# ifconfig -a
wi0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
```

```
        inet6 fe80::202:2dff:fe2d:c938%wi0 prefixlen 64 scopeid 0x7
        inet 0.0.0.0 netmask 0xff000000 broadcast 255.255.255.255
        ether 00:09:2d:2d:c9:50
        media: IEEE 802.11 Wireless Ethernet autoselect (DS/2Mbps)
        status: no carrier
        ssid ""
        stationname "FreeBSD Wireless node"
        channel 10 authmode OPEN powersavemode OFF powersavesleep 100
        wepmode OFF weptxkey 1
```

A continuación debemos especificar los parámetros correctos para nuestra red:

```
# ifconfig wi0 inet 192.168.0.20 netmask 255.255.255.0 ssid mi_red
```

Sustituya `192.168.0.20` y `255.255.255.0` con una dirección IP y máscara de red que se adecúen con el espacio de direccionamiento de la red cableada. Recordemos que nuestro punto de acceso está puenteando la red inalámbrica y la red de cable, de modo que para el resto de dispositivos de la red el cliente inalábrico se muestra como un elemento más de la red cableada.

Llegados a este punto deberíamos poder hacer ping a las máquinas de la red cableada como si estuviéramos compartiendo el mismo enlace físico cableado.

Si se presentan problemas con la conexión inalámbrica se puede comprobar si la tarjeta “ wireless” se encuentra correctamente asociada (conectada) con el punto de acceso:

```
# ifconfig wi0
```

ebería devolver algún tipo de información entre la que deberíamos observar la siguiente línea:

```
status: associated
```

Si no aparece la palabra `associated` puede ser que nos encontremos fuera de la cobertura proporcionada por el punto de acceso o puede ser que necesitemos activar el cifrado, aunque éstos no son los únicos problemas con los que nos podemos encontrar.

29.3.3.4. Cifrado

El cifrado, también llamado codificación, de una red inalámbrica es un proceso importante porque, a diferencia de lo que ocurre con las redes cableadas convencionales, las redes inalámbricas no se pueden restringir a un espacio físico determinado. Los datos que viajan a través de ondas de radio se difunden a través de las paredes y alcanzan a los vecinos más cercanos. Aquí es donde entra en en juego el sistema de cifrado. El cifrado se emplea para evitar que cualquiera pueda examinar los datos enviados a través del aire.

Los dos métodos más comunes para realizar el cifrado de datos entre el cliente y el punto de acceso son WEP e ipsec(4).

29.3.3.4.1. WEP

WEP son las siglas de Wired Equivalency Protocol. WEP es un un intento de crear redes inalámbricas al menos tan seguras omo las redes cableadas o al menos de seguridad equivalente a dichas redes. Por desgracia el sistema WEP es débil y resulta bastante sencillo de romper. Esto significa que cuando se transmite información de carácter crítico no se debe confiar únicamente en este sistema de cifrado.

No obstante es mejor que no utilizar nada; puede activar WEP en el sistema que hace de punto de acceso mediante:

```
# ifconfig wi0 inet up ssid
              mi_red wepmode on wepkey
              0x1234567890 media DS/11Mbps
              mediaopt hostap
```

y en un cliente inalámbrico mediante la siguiente orden:

```
# ifconfig wi0 inet 192.168.0.20 netmask 255.255.255.0 ssid mi_red ↺
wepmode on wepkey 0x1234567890
```

Por favor, tenga un poco de sentido común y reemplace la clave *0x1234567890* por otra clave menos obvia.

29.3.3.4.2. IPsec

ipsec(4) es una herramienta más robusta y potente para cifrar datos que se mueven a través de una red. Es el mecanismo más conveniente para asegurar los datos de una red inalámbrica. Tiene más información sobre el protocolo ipsec(4) y cómo utilizarlo en la sección IPsec de este manual.

29.3.3.5. Herramientas

No hay muchas herramientas disponibles si se quiere depurar y monitorizar redes inalámbricas pero en el siguiente apartado mostraremos cómo utilizar algunas de ellas.

29.3.3.5.1. El paquete bsd-airtools

El paquete bsd-airtools es un conjunto muy completo de herramientas “wireless” que se pueden utilizar para multitud de tareas, entre las cuales podemos citar citar el descifrado de claves WEP, detección de puntos de de acceso, monitorización de la señal de radio, etc.

El paquete bsd-airtools se puede instalar como “port” desde net/bsd-airtools. La información relacionada con los “ports” puede encontrarse en la sección Capítulo 4, *Instalación de aplicaciones: «packages» y ports* de este manual.

El programa `dstumbler` es una herramienta que permite descubrir puntos de acceso y entre otras cosas muestra de forma gráfica la relación señal / ruido del enlace. Si se expe-

rimentan problemas para acceder a un determinado punto de acceso `dstumbler` puede ser muy útil.

Para probar la seguridad de la red inalámbrica se puede usar "dweputils", concretamente las órdenes `dwepcrack`, `dwepdump` y `dwepkeygen`. Estas órdenes permiten determinar hasta qué punto la seguridad que ofrece WEP es suficiente para nuestras necesidades.

29.3.3.5.2. Las utilidades `wicontrol`, `ancontrol` y `raycontrol`

Mediante estas herramientas se puede controlar el comportamiento de la tarjeta de red inalámbrica. En los ejemplos anteriores se ha utilizado wicontrol(8) debido a que la tarjeta de red del ejemplo utiliza el interfaz `wi0`. Si se posée una tarjeta "wireless" de Cisco dicha tarjeta se mostrará en el sistema mediante el interfaz `an0` y por lo tanto la orden equivalente que se debe usar será ancontrol(8).

29.3.3.5.3. `ifconfig`

Con ifconfig(8) se puede utilizar unas cuantas de las opciones que se pueden usar con wicontrol(8), pero no obstante no posée todas las funcionalidades que proporciona wicontrol(8). Se recomienda leer ifconfig(8) para conocer los detalles de los parámetros y opciones que admite.

29.3.3.6. Tarjetas de Red inalámbricas soportadas

29.3.3.6.1. Puntos de acceso

Las únicas tarjetas que soportan el modo de funcionamiento funcionamiento BSS (pueden funcionar como puntos de acceso) son los dispositivos basados en el chip Prism 2, 2.5 ó 3. Consulte wi(4) para ver una lista completa de ellos.

29.3.3.6.2. Clientes

FreeBSD soporta casi todas las tarjetas inalámbricas 802.11b 802.11b que se encuentran actualmente en el mercado. La mayoría de las tarjetas basadas en los chips Prism, Spectrum24, Spectrum24, Hermes, Aironet y Raylink tambíen funcionan en modo IBSS (modos ad-hoc, punto a punto y BSS).

29.4. Bluetooth

Escrito por Pav Lucistnik.

29.4.1. Introducción

Bluetooth es una tecnología inalámbrica que opera en banda de 2.4 GHz (donde no se necesita licencia). Se trata de una tecnología pensada para la creación de redes de ámbito personal (de cobertura reducida, normalmente de unos 10 metros). Las redes se suelen

construir en modo “ad-hoc” utilizando dispositivos heterogéneos como teléfonos móviles, dispositivos manuales (“handhelds”) y computadoras portátiles. A diferencia de otras tecnologías inalámbricas como Wi-Fi, Bluetooth ofrece perfiles de servicio más detallados; por ejemplo un perfil para actuar como un servidor de ficheros basado en FTP, para la difusión de ficheros (“file pushing”), para el transporte de voz, para la emulación de línea serie y muchos más.

La pila de Bluetooth en FreeBSD se implementa utilizando el entorno de Netgraph (véase netgraph(4)). La mayoría de los dispositivos USB Bluetooth se pueden utilizar mediante el controlador ng_ubt(4). Los dispositivos Bluetooth basados en el chip Broadcom BCM2033 están soportados mediante los controladores ubtbcmfw(4) y ng_bt3c(4). Los dispositivos Bluetooth basados en la interfaz serie o de Rayos Infrarrojos (UART) se controlan mediante sio(4), ng_h4(4) y hcseriald(8). Este capítulo describe el uso de dispositivos Bluetooth USB. El soporte para Bluetooth se encuentra en las versiones de FreeBSD 5.0 y posteriores.

29.4.2. Instalación del dispositivo

Por defecto los controladores de los dispositivos Bluetooth se encuentran disponibles como módulos del kernel. Antes de enchufar el dispositivo Bluetooth se debe cargar el módulo correspondiente dentro del núcleo.

```
# kldload ng_ubt
```

Si el dispositivo Bluetooth se encuentra conectado cuando el sistema arranca se debe cargar el módulo modificando a tal efecto el fichero `/boot/loader.conf` .

```
ng_ubt_load="YES"
```

Al conectar el dispositivo Bluetooth aparecerá en la consola (o en syslog) la siguiente información:

```
ubt0: vendor 0x0a12 product 0x0001, rev 1.10/5.25, addr 2
ubt0: Interface 0 endpoints: interrupt=0x81, bulk-in=0x82, bulk-
out=0x2
ubt0: Interface 1 (alt.config 5) endpoints: isoc-in=0x83, isoc-
out=0x3,
      wMaxPacketSize=49, nframes=6, buffer size=294
```

Se debe copiar `/usr/share/examples/netgraph/bluetooth/rc.bluetooth` a algún lugar más conveniente, por ejemplo `/etc/rc.bluetooth` . Este script se usa para ejecutar y detener la pila Bluetooth del sistema. Se suele recomendar quitar la pila antes de desenchufar el dispositivo pero si no se hace no debería producirse ningún desastre. Cuando se arranca la pila aparece un mensaje similar a este:

```
# /etc/rc.bluetooth start ubt0
BD_ADDR: 00:02:72:00:d4:1a
Features: 0xff 0xff 0xf 00 00 00 00 00
<3-Slot> <5-Slot> <Encryption> <Slot offset>
```

```
<Timing accuracy> <Switch> <Hold mode> <Sniff mode>
<Park mode> <RSSI> <Channel quality> <SCO link>
<HV2 packets> <HV3 packets> <u-law log> <A-law log> <CVSD>
<Paging scheme> <Power control> <Transparent SCO data>
Max. ACL packet size: 192 bytes
Number of ACL packets: 8
Max. SCO packet size: 64 bytes
Number of SCO packets: 8
```

29.4.3. Interfaz de la controladora de la máquina (HCI)

La interfaz de la Controladora de la Máquina (Host Controller Interface) proporciona una interfaz de órdenes para la controladora de banda base y para el gestor de enlace, y permite acceder al estado del hardware y a los registros de control. Esta interfaz proporciona una capa de acceso homogénea para todos los dispositivos Bluetooth de banda base. La capa HCI de la máquina intercambia órdenes y datos con el firmware del HCI presente en el dispositivo Bluetooth. El driver de la capa de transporte de la controladora de la máquina (es decir, el driver del bus físico) proporciona ambas capas de HCI la posibilidad de intercambiar información entre ellas.

Se crea un nodo Netgraph de tipo *HCI* para cada dispositivo Bluetooth. El nodo Netgraph HCI se conecta normalmente con el nodo que representa el controlador del dispositivo Bluetooth de la máquina (sentido de bajada) y con el nodo Netgraph L2CAP en el sentido de subida. Todas las operaciones HCI se realizan sobre el nodo Netgraph HCI y no sobre el el nodo que representa al dispositivo. El nombre por defecto para el nodo HCI es “devicehci”. Para obtener más detalles, por favor consulte la página del manual de ng_hci(4).

Una de las tareas más importantes que se deben realizar es el descubrimiento automático de otros dispositivos Bluetooth que se encuentren dentro del radio de cobertura. Esta operación se denomina en inglés *inquiry* (consulta). Esta operación o otras operaciones HCI relacionadas se realizan mediante la utilidad hccontrol(8). El siguiente ejemplo muestra cómo descubrir dispositivos en pocos segundos. Tenga siempre presente que un dispositivo remoto sólo contesta a la consulta si se encuentra configurado en modo descubrimiento (*discoverable mode*).

```
% hccontrol -n ubt0hci inquiry
Inquiry result, num_responses=1
Inquiry result #0
       BD_ADDR: 00:80:37:29:19:a4
       Page Scan Rep. Mode: 0x1
       Page Scan Period Mode: 00
       Page Scan Mode: 00
       Class: 52:02:04
       Clock offset: 0x78ef
Inquiry complete. Status: No error [00]
```

`BD_ADDR` es la dirección identificativa única del dispositivo Bluetooth, similar a las direcciones MAC de las tarjetas Ethernet. Esta dirección se necesita para transmitir otro tipo

de información a otros dispositivos. Se puede asignar un nombre más significativo para los humanos en la variable BD_ADDR. El fichero `/etc/bluetooth/hosts` contiene información relativa a los dispositivos Bluetooth conocidos. El siguiente ejemplo muestra cómo obtener un nombre significativo para los humanos que fué asignado a un dispositivo remoto:

```
% hccontrol -n ubt0hci remote_name_request 00:80:37:29:19:a4
BD_ADDR: 00:80:37:29:19:a4
Name: Pav's T39
```

Si se realiza una consulta (inquiry) sobre el dispositivo Bluetooth remoto, dicho dispositivo identificará nuestro computador como "nombre.de.su.sistema (ubt0)". El nombre asignado al dispositivo local se puede modificar en cualquier momento.

El sistema Bluetooth proporciona una conexión punto a punto (con sólo dos unidades Bluetooth involucradas) o también una conexión punto multipunto. En el último caso, la conexión se comparte entre varios dispositivos Bluetooth. El siguiente ejemplo muestra como obtener una lista de las conexiones de banda base activas en el dispositivo local:

```
% hccontrol -n ubt0hci read_connection_list
Remote BD_ADDR    Handle Type Mode Role Encrypt Pending Queue State
00:80:37:29:19:a4     41  ACL    0 MAST    NONE       0     0 OPEN
```

Resulta útil disponer de un *manejador de la conexión* cuando se necesita terminar la conexión de banda base. Es importante recalcar que normalmente no es necesario realizar esta terminación de forma manual. La pila Bluetooth puede concluír automáticamente las conexiones de banda base que se encuentren inactivas.

```
# hccontrol -n ubt0hci disconnect 41
Connection handle: 41
Reason: Connection terminated by local host [0x16]
```

Se ruega consultar la salida de la orden `hccontrol help` para obtener un listado completo de las órdenes HCI disponibles. La mayoría de estas órdenes no requiren privilegios de superusuario.

29.4.4. Protocolo de adaptación y de control de enlace a nivel lógico (L2CAP)

El protocolo L2CAP (Logical Link Control and Adaptation Protocol) proporciona servicios de datos tanto orientados a conexión como no orientados a conexión a los protocolos de las capas superiores, junto con facilidades de multiplexación y de segmentacion y reensamblaje. L2CAP permite que los protocolos de capas superiores puedan transmitir y recibir paquetes de datos L2CAP de hasta 64 kilobytes de longitud.

L2CAP se basa en el concepto de *canales*. Un canal es una conexión lógica que se sitúa sobre la conexión de banda base. Cada canal se asocia a un único protocolo. Cada paquete

L2CAP que se recibe a un canal se redirige al protocolo superior correspondiente. Varios canales pueden operar sobre la misma conexión de banda base, pero un canal no puede tener asociados más de un protocolo de alto nivel.

Para cada dispositivo Bluetooth se cre un único nodo Netgraph de tipo *l2cap*. El nodo L2CAP se conecta normalmente conectado al nodo Netgraph HCI (hacia abajo) y con nodos Bluetooth tipo "sockets" hacia arriba. El nombre por defecto para el nodo Netgraph L2CAP es "devicel2cap". Para obtener más detalles se ruega consultar la página del manual ng_l2cap(4).

l2ping(8) le será muy útil para hacer ping a otros dispositivos. Algunas implementaciones de Bluetooth no devuelven todos los datos que se envían, de tal forma que el valor *0 bytes* que se observa a continuación es normal:

```
# l2ping -a 00:80:37:29:19:a4
0 bytes from 0:80:37:29:19:a4 seq_no=0 time=48.633 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=1 time=37.551 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=2 time=28.324 ms result=0
0 bytes from 0:80:37:29:19:a4 seq_no=3 time=46.150 ms result=0
```

La herramienta l2control(8) se utiliza para realizar varias operaciones sobre los nodos L2CAP. Este ejemplo muestra cómo obtener la lista de conexiones lógicas (canales) y la lista de conexiones de banda base (física) que mantiene el dispositivo local:

```
% l2control -a 00:02:72:00:d4:1a read_channel_list
L2CAP channels:
Remote BD_ADDR     SCID/ DCID   PSM  IMTU/ OMTU State
00:07:e0:00:0b:ca    66/   64     3   132/  672 OPEN
% l2control -a 00:02:72:00:d4:1a read_connection_list
L2CAP connections:
Remote BD_ADDR    Handle Flags Pending State
00:07:e0:00:0b:ca     41 O           0 OPEN
```

Otra herramienta de diagnóstico interesante es btsockstat(1). Realiza un trabajo similar a la orden netstat(1), pero en este caso para las estructuras de datos relacionadas con el sistema Bluetooth. A continuación se muestra la información relativa a la misma conexión lógica del ejemplo anterior.

```
% btsockstat
Active L2CAP sockets
PCB      Recv-Q Send-Q Local address/PSM       Foreign address   CID
  State
c2afe900      0      0 00:02:72:00:d4:1a/3     00:07:e0:00:0b:ca 66
  OPEN
Active RFCOMM sessions
L2PCB    PCB      Flag MTU   Out-Q DLCs State
c2afe900 c2b53380 1    127   0     Yes  OPEN
Active RFCOMM sockets
PCB      Recv-Q Send-Q Local address     Foreign address   Chan ↺
DLCI State
```

```
c2e8bc80      0    250 00:02:72:00:d4:1a 00:07:e0:00:0b:ca 3    6
  OPEN
```

29.4.5. Protocolo RFCOMM

El protocolo RFCOMM proporciona emulación de puertos serie a través del protocolo L2CAP. Este protocolo se basa en el estándar de la ETSI denominado TS 07.10. RFCOMM es un protoclo de transporte sencillo, con soporte para hasta 9 puertos serie RS-232 (EIA-TIA-232-E). El protocolo RFCOMM permite hasta 60 conexiones simultaneas (canales RFCOMM) entre dos dispositivos Bluetooth.

Para los propósitos de RFCOMM, un camino de comunicación involucra siempre a dos aplicaciones que se ejecutan en dos dispositivos distintos (los extremos de la comunicación). Entre ellos existe un segmento que los comunica. RFCOMM pretende cubrir aquellas aplicaciones que utilizan los puertos serie de las máquinas donde se ejecutan. El segmento de comunicación es un enlace Bluetooth desde un dispositivo al otro (conexión directa).

RFCOMM trata únicamente con la conexión de dispositivos directamente, y también con conexiones entre el dispositivo y el modem para realizar conexiones de red. RFCOMM puede soportar otras configuraciones, tales como módulos que se comunican via Bluetooth por un lado y que proporcionan una interfaz de red cableada por el otro.

En FreeBSD el protocolo RFCOMM se implementa utilizando la capa de “sockets” de Bluetooth.

29.4.6. Enparejamiento de dispositivos

Por defecto, la comunicación Bluetooth no se valida, por lo que cualquier dispositivo puede en principio hablar con cualquier otro. Un dispositivo Bluetooth (por ejemplo un teléfono celular) puede solicitar autenticación para realizar un determinado servicio (por ejemplo para el servicio de marcación por modem). La autenticación de Bluetooth normalmente se realiza utilizando *códigos PIN*. Un código PIN es una cadena ASCII de hasta 16 caracteres de longitud. Los usuarios deben introducir el mismo código PIN en ambos dispositivos. Una vez que el usuario ha introducido el PIN adecuado ambos dispositivos generan una *clave de enlace*. Una vez generada, la clave se puede almacenar en el propio dispositivo o en un dispositivo de almacenamiento externo. La siguiente vez que se comuniquen ambos dispositivos se reutilizará la misma clave. El procedimiento descrito hasta este punto se denomina *emparejamiento (pairing)*. Es importante recordar que si la clave de enlace se pierde en alguno de los dispositivos involucrados se debe volver a ejecutar el procedimiento de emparejamiento.

El dæmon hcsecd(8) se encarga de gestionar todas las peticiones de autenticación Bluetooth. El archivo de configuración predeterminado se denomina `/etc/bluetooth/hcsecd.conf` . A continuación se muestra una sección de ejemplo de un teléfono celular con el código PIN arbitrariamente fijado al valor “1234”:

```
device {
        bdaddr  00:80:37:29:19:a4;
        name    "Pav's T39";
        key     nokey;
        pin     "1234";
      }
```

No existe ninguna limitación en los códigos PIN a excepción de su longitud. Algunos dispositivos (por ejemplo los dispositivos de mano Bluetooth) pueden obligar a escribir un número predeterminado de caracteres para el código PIN. La opción `-d` fuerza al dæmon hcsecd(8) a permanecer ejecutádose en primer plano, de tal forma que se puede observar fácilmente lo que ocurre. Si se configura el dispositivo Bluetooth remoto para aceptar el procedimiento de emparejamiento y se inicia la conexión con dicho dispositivo, el dispositivo remoto debería decir que el procedimiento de emparejamiento se ha aceptado y debería solicitar el código PIN. Si se introduce el mismo código PIN que se escribió en su momento en el fichero `hcsecd.conf` el procedimiento de emparejamiento y de generación de la clave de enlace debería terminar satisfactoriamente. Por otra parte el procedimiento de emparejamiento se puede iniciar en el dispositivo remoto. A continuación se muestra un ejemplo de la salida del dæmon `hcsecd`.

```
hcsecd[16484]: Got Link_Key_Request event from 'ubt0hci', remote ↺
bdaddr 0:80:37:29:19:a4
hcsecd[16484]: Found matching entry, remote bdaddr ↺
0:80:37:29:19:a4, name 'Pav's T39', link key doesn't exist
hcsecd[16484]: Sending Link_Key_Negative_Reply to 'ubt0hci' for ↺
remote bdaddr 0:80:37:29:19:a4
hcsecd[16484]: Got PIN_Code_Request event from 'ubt0hci', remote ↺
bdaddr 0:80:37:29:19:a4
hcsecd[16484]: Found matching entry, remote bdaddr ↺
0:80:37:29:19:a4, name 'Pav's T39', PIN code exists
hcsecd[16484]: Sending PIN_Code_Reply to 'ubt0hci' for remote ↺
bdaddr 0:80:37:29:19:a4
```

29.4.7. Protocolo de descubrimiento de servicios (SDP)

El Protocolo de Descubrimiento de Servicios (Service Discovery Protocol o SDP) permite a las aplicaciones cliente descubrir la existencia de diversos servicios proporcionados por uno o varios servidores de aplicaciones, junto con los atributos y propiedades de los servicios que se ofrecen. Estos atributos de servicio incluyen el tipo o clase de servicio ofrecido y el mecanismo o la información necesaria para utilizar dichos servicios.

SDP se basa en una determinada comunicación entre un servidor SDP y un cliente SDP. El servidor mantiene una lista de registros de servicios, los cuales describen las características de los servicios ofrecidos. Cada registro contiene información sobre un determinado servicio. Un cliente puede recuperar la información de un registro de servicio almacenado en un servidor SDP lanzando una petición SDP. Si el cliente o la aplicación asociada con el cliente decide utilizar un determinado servicio, debe establecer una conexión independiente con el servicio en cuestión. SDP proporciona un mecanismo para el descu-

brimiento de servicios y sus atributos asociados, pero no proporciona ningún mecanismo ni protocolo para utilizar dichos servicios.

Normalmente, un cliente SDP realiza una búsqueda de servicios acotada por determinadas características. No obstante hay momentos en los que resulta deseable descubrir todos los servicios ofrecidos por un servidor SDP sin que pueda existir ningún conocimiento previo sobre los registros que pueda contener. Este proceso de búsqueda de cualquier servicio ofrecido se denomina *navegación* o *browsing*.

El servidor Bluetooth SDP denominado sdpd(8) y el cliente de línea de órdenes sdpcontrol(8) se incluyen en la instalación estándar de FreeBSD. El siguiente ejemplo muestra cómo realizar una consulta de navegación una consulta de navegación SDP.

```
% sdpcontrol -a 00:01:03:fc:6e:ec browse
Record Handle: 00000000
Service Class ID List:
        Service Discovery Server (0x1000)
Protocol Descriptor List:
        L2CAP (0x0100)
                Protocol specific parameter #1: u/int/uuid16 1
                Protocol specific parameter #2: u/int/uuid16 1

Record Handle: 0x00000001
Service Class ID List:
        Browse Group Descriptor (0x1001)

Record Handle: 0x00000002
Service Class ID List:
        LAN Access Using PPP (0x1102)
Protocol Descriptor List:
        L2CAP (0x0100)
        RFCOMM (0x0003)
                Protocol specific parameter #1: u/int8/bool 1
Bluetooth Profile Descriptor List:
        LAN Access Using PPP (0x1102) ver. 1.0
```

... y así sucesivamente. Resulta importante resaltar una vez más que cada servicio posee una lista de atributos (por ejemplo en el canal RFCOMM). Dependiendo de los servicios que se quieran utilizar puede resultar necesario anotar algunos de los atributos. Algunas implementaciones de Bluetooth no soportan navegación de servicios y pueden devolver una lista vacía. En este caso se puede intentar buscar algún servicio determinado. El ejemplo siguiente muestra cómo buscar el servicio OBEX Object Push (OPUSH):

```
% sdpcontrol -a 00:01:03:fc:6e:ec search OPUSH
```

En FreeBSD los servicios a clientes Bluetooth se suministran mediante el servidor sdpd(8).

```
# sdpd
```

La aplicación local servidora que quiere proporcionar servicio Bluetooth a los clientes remotos puede registrar su servicio con el dæmon SDP local. Un ejemplo de dicha aplicación

Un ejemplo de dicha aplicación lo constituye el dæmon rfcomm_pppd(8). Una vez ejecutado el dæmon registra un servicio LAN de Bluetooth en el dæmon SDP local.

Se puede obtener la lista de servicios registrados con el servidor SDP local lanzando una consulta de navegación SDP utilizando el canal de control local.

```
# sdpcontrol -l browse
```

29.4.8. Acceso telefónico a redes (DUN) y acceso a redes mediante perfiles PPP (LAN)

El perfil de Acceso Telefónico a Redes (Dial-Up Networking o DUN) se utiliza mayoritariamente con modems y teléfonos celulares. Los escenarios cubiertos por este perfil se describen a continuación:

- Utilización de un teléfono celular o un modem por una computadora para simular un modem sin cables que se conecte a un servidor de acceso telefónico a redes o para otros servicios de acceso telefónico relacionados;
- Utilización de un teléfono celular o un modem por un computador para recibir llamadas de datos.

El Acceso a Redes con perfiles PPP (LAN) se puede utilizar en las siguientes situaciones:

- Acceso LAN para un único dispositivo Bluetooth;
- Acceso LAN para múltiples dispositivos Bluetooth;
- Conexión de PC a PC (utilizando emulación de PPP sobre una línea serie).

En FreeBSD ambos perfiles se implementan bajo las órdenes ppp(8) y rfcomm_pppd(8), un encapsulador que convierte la conexión RFCOMM de Bluetooth en algo que puede ser utilizado por PPP. Antes de que se puedan utilizar los perfiles se debe definir una nueva etiqueta PPP en el fichero de configuración `/etc/ppp/ppp.conf` . Consulte rfcomm_pppd(8) para ver algunos ejemplos.

En el siguiente ejemplo se va a utilizar rfcomm_pppd(8) para abrir una conexión RFCOMM con un dispositivo remoto con BD_ADDR 00:80:37:29:19:a4 sobre un canal RFCOMM basado en DUN (Dial-Up Networking). El número de canal RFCOMM se obtiene a partir del dispositivo remoto a través de SDP. Es posible especificar el canal RFCOMM a mano, en cuyo caso rfcomm_pppd(8) no realizará ninguna consulta SDP. Se puede utilizar la orden sdpcontrol(8) para descubrir el canal RFCOMM utilizado en el dispositivo remoto.

```
# rfcomm_pppd -a 00:80:37:29:19:a4 -c -C dun -l rfcomm-dialup
```

Para proporcionar el servicio de Acceso a Redes a través de PPP (LAN) se debe ejecutar el servidor sdpd(8). Se debe crear una nueva entrada en `/etc/ppp/ppp.conf` . Le rogamos

que consulte rfcomm_pppd(8) y observe los ejemplos que se facilitan. Por último se debe ejecutar el servidor PPP RFCOMM sobre un número de canal RFCOMM adecuado. El servidor PPP RFCOMM registrará automáticamente el servicio LAN de Bluetooth con el servidor SDP local. El ejemplo que se muestra a continuación describe cómo ejecutar el servidor PPP RFCOMM.

```
# rfcomm_pppd -s -C 7 -l rfcomm-server
```

29.4.9. Perfil OBEX Object Push (OPUSH)

OBEX es un protocolo muy utilizado para transferencias de ficheros sencillas entre dispositivos móviles. Su uso más importante se produce en comuncaciones por infrarrojos, donde se utiliza para transferencia de ficheros genéricos entre portátiles o dispositivos Palm y para enviar tarjetas de visita o entradas de la agenda entre teléfonos celulares y otros dispositivos con aplicaciones PIM.

El cliente y el servidor de OBEX se implementan como un paquete denominado obexapp disponible como “ port” en comms/obexapp.

El cliente OBEX se utiliza para introducir y para recuperar recuperar objetos del servidor OBEX. Un objeto puede por ejemplo ser una tarjeta de visita o una cita. El cliente OBEX puede obtener un número de canal RFCOMM del dispositivo remoto utilizando SDP. Esto se hace especificando el nombre del servicio en lugar del número de canal RFCOMM. Los nombres de servicios soportados son: IrMC, FTRN y OPUSH. Es posible especificar el canal RFCOMM como un número. A continuación se muestra un ejemplo de una sesión OBEX donde el objeto que posee la información del dispositivo se recupera del teléfono celular y un nuevo objeto (la tarjeta de visita) se introduce en el directorio de dicho teléfono.

```
% obexapp -a 00:80:37:29:19:a4 -C IrMC
obex> get
get: remote file> telecom/devinfo.txt
get: local file> devinfo-t39.txt
Success, response: OK, Success (0x20)
obex> put
put: local file> new.vcf
put: remote file> new.vcf
Success, response: OK, Success (0x20)
obex> di
Success, response: OK, Success (0x20)
```

Para proporcionar servicio de OBEX el servidor sdpd(8) debe estar en funcionamiento. Además se debe crear un directorio raíz donde todos los objetos van a ser almacenados. La ruta por defecto para el directorio raíz es `/var/spool/obex` . Por último se debe ejecutar el servidor OBEX en un número de canal RFCOMM válido. El servidor OBEX registra automáticamente el servicio de Object Push con el dæmon SDP local. El ejemplo que se muestra a local. El ejemplo que se muestra a continuación continuación describe cómo ejecutar el servidor OBEX.

```
# obexapp -s -C 10
```

29.4.10. Perfil de puerto serie (SP)

El perfil de puerto serie (Serial Port o SP) permite que dispositivos Bluetooth realicen emulación de RS232 (o similar). El escenario cubierto por este perfil trata con con aplicaciones comerciales que utilizan Bluetooth como un sustituto sustituto del cable, utilizando una capa de abstracción que representa un puerto serie virtual.

La aplicación rfcomm_sppd(1) implementa el perfil Puerto Serie. Usa una pseudo tty como abstracción de puerto serie virtual. El ejemplo de más abajo muestra cómo conectarse a un servicio de dispositivo remoto de Puerto Serie. Observe que no necesita especificarse el canal RFCOMM: rfcomm_sppd(1) puede obtenerlo del dispotivo remoto via SDP. Si necesita especificarlo por alguna razón hágalo en la propia línea de órdenes.

```
# rfcomm_sppd -a 00:07:E0:00:0B:CA -t /dev/ttyp6
rfcomm_sppd[94692]: Starting on /dev/ttyp6...
```

Una vez conectado el pseudo tty se puede utilizar como un puerto serie.

```
# cu -l ttyp6
```

29.4.11. Solución de problemas

29.4.11.1. Un dispositivo remoto no puede conectarse

Algunos dispositivos Bluetooh antiguos no soportan el cambio de cambio de roles. Por defecto, roles. Cuando FreeBSD acepta una nueva conexión por defecto intenta realizar un cambio de rol y convertirse en maestro. Dispositivos que no son capaces de realizar este cambio no pueden conectarse. Es interesante resaltar que el cambio de roles se realiza cuando se está estableciendo una nueva conexión de tal forma que no es posible preguntar al dispositivo si soporta intercambio de roles. Existe una opción HCI para desactivar el intercambio de roles en la parte local.

```
# hccontrol -n ubt0hci write_node_role_switch 0
```

29.4.11.2. Algo va mal ?puedo ver exactamente qué está ocurriendo?

Sí, se puede. Utilice el paquete hcidump-1.5, que se puede descargar de aquí. La herramienta hcidump es similar a la herramienta tcpdump(1). Se puede utilizar para mostrar el contenido de los paquetes Bluetooth sobre el terminal y para volcar los paquetes Bluetooth a un fichero.

29.5. Puenteado

Escrito por Steve Peterson.

29.5.1. Introducción

Algunas veces resulta útil dividir una red física (como por ejemplo un segmento Ethernet) en dos segmentos de red separados, sin tener que crear subredes IP y sin utilizar una pasarela para comunicar ambos segmentos. El dispositivo que realiza esta función se denomina “bridge”. Un sistema FreeBSD con dos interfaces de red puede actuar como un “bridge” o puente entre ambas.

El “bridge” funciona de tal forma que aprende las direcciones de la capa MAC (direcciones Ethernet) de los nodos que se encuentran conectados a cada interfaz de red de tal forma que sólo se reenvía tráfico entre los segmentos de red cuando las direcciones fuente y destino se encuentran separadas en segmentos distintos.

En varios aspectos se puede comparar un “bridge” con un “switch” de pocos puertos.

29.5.2. Situaciones donde el puenteado resulta adecuado

Existen al menos dos situaciones típicas donde se puede utilizar la funcionalidad proporcionada por los “ bridges”.

29.5.2.1. Tráfico de gran volumen en un segmentos de red

La primera situación surge cuando nos encontramos con un segmento de red congestionado pero por las razones que sean no queremos subdividir la red e interconectar las nuevas subredes mediante un “route”.

Vamos a considerar un ejemplo de un periódico donde los departamentos editoriales y de producción utilizan la misma subred. Los usuarios de la editorial utilizan el servidor `A` como servidor de ficheros y los de producción utilizan el servidor `B`. Se Se utiliza una red Ethernet para conectar ambos departamentos y se ha detectado que la alta utilización del enlace está ralentizando el funcionamiento de la red.

Si los usuarios de la editorial pudieran agregarse en un segmento de red mientras que los usuarios de producción se localizaran en otro se podrían conectar ambos segmentos mediante un “bridge”. Sólo se utilizará el “bridge” para encaminar tráfico de red destinado a interfaces que se encuentren en el *otro* lado del “bridge”, reduciendo de esta forma la congestión en cada nuevo segmento.

29.5.2.2. Cortafuegos de filtrado/conformación de tráfico

La segunda situación típica se produce cuando se necesita un cortafuegos pero no la Traducción de Direcciones de Red (NAT).

A continuación se muestra un ejemplo. Una pequeña compañía se comunica con su ISP utilizando DSL o ISDN. Dicha compañía posee 13 13 direcciones IP globalmente accesibles

delegadas por su ISP y tiene 10 ordenadores en funcionamiento. En esta situación un un cortafuegos basado en un “router” resulta difícil debido a la distribución del espacio de direccionamiento disponible (subnetting).

Un cortafuegos implementado sobre un “bridge” se puede utilizar en el camino de bajado desde el ISP hasta las oficinas de la compañía sin necesidad de tener en cuenta ningún aspecto relacionado con la distribución de las direcciones IP.

29.5.3. Configuración de un “bridge”

29.5.3.1. Selección de la interfaz de red

Un “bridge” necesita al menos dos tarjetas de red situadas en dos segmentos de red para su funcionamiento. Por desgracia no todas las interfaces de red pueden usarse para el puenteo. Consulte bridge(4), ahín encontrará más información sobre qué tarjetas puede usar.

Por favor, instale y pruebe las dos tarjetas de red antes de continuar.

29.5.3.2. Cambios en la configuración del núcleo

Para activar el soporte de “bridging” en el núcleo añada

```
options BRIDGE
```

al fichero de configuración del núcleo y recompile el kernel.

29.5.3.3. Soporte de cortafuegos

Si se desea utilizar el “bridge” como un cortafuegos, se debe añadir además la opción `IPFIREWALL`. Lea el capílo de firewalls para obtener información general sobre cómo configurar el bridge para que actúe además como cortafuegos.

Si además queremos que los paquetes que no sean IP (por ejemplo paquetes ARP) puedan atravesar el “bridge” deberemos añadir la opción `IPFIREWALL_DEFAULT_TO_ACCEPT` . Tenga en cuenta opción modifica el comportamiento del cortafuegos de tal forma que por defecto aceptará cualquier paquete. Hay que tener cuidado para asegurarse de que el comportamiento esperado del cortafuegos, que reside en el conjunto de reglas que se hayan definido, no se vea afectado por este cambio.

29.5.3.4. Soporte de conformado de tráfico

Si se quiere utilizar el “bridge” como un conformador de tráfico, es decir, como un elemento capaz de adaptar los distintos flujos según determinados patrones, se debe añadir la opción `DUMMYNET` a la configuración del núcleo. Se ruega consultar dummynet(4) para obtener más información al respecto.

29.5.4. Cómo activar el “bridge”

Añadir la línea:

```
net.link.ether.bridge=1
```

en `/etc/sysctl.conf` para habilitar el soporte de “bridging” en tiempo de ejecución y la línea:

```
net.link.ether.bridge_cfg=if1,if2
```

Para activar el “bridging” en las interfaces especificadas (sustituya *`if1`* y *`if2`* con los nombres de sus interfaces de red). Si deseamos filtrar los paquetes puenteados utilizando ipfw(8), debemos añadir también:

```
net.link.ether.bridge_ipfw=1
```

En FreeBSD 5.2-RELEASE y posteriores, se debe utilizar las siguientes líneas en lugar de las anteriores:

```
net.link.ether.bridge.enable=1
net.link.ether.bridge.config=if1,if2
net.link.ether.bridge.ipfw=1
```

29.5.5. Información adicional

Si queremos ser capaces de conectarnos al “bridge” mediante telnet(1) se puede asignar una dirección IP a una de las tarjetas de red del “bridge”. Por amplio consenso se considera una mala idea asignar más de una dirección IP al “bridge”.

Si poseemos varios “bridges” en nuestra red sólamente puede existir un único camino entre cualesquiera dos máquinas de nuestra red. Técnicamente hablando esto significa que no existe soporte para gestión de enlace mediante mecanismos basados en árboles de recubrimiento mínimos (“spanning tree”).

Un “bridge” puede añadir latencia a los tiempos de respuesta de la orden ping(8), especialmente cuando el tráfico tiene que viajar de un segmento de red al otro.

29.6. NFS

Reorganizado y ampliado por Tom Rhodes.
Escrito por Bill Swingle.

FreeBSD soporta diversos sistemas de ficheros, uno de los cuales es el Sistema de Ficheros en Red, tambíen conocido por su acrónimo en inglés NFS. NFS permite compartir directorios y ficheros a través de la red. Los usuarios del sistema NFS pueden acceder a ficheros que se encuentran físicamente en máquinas remotas de una forma transparente, como si se tratara de ficheros locales.

He aquí algunos los beneficios más destacados que NFS proporciona:

- Las estaciones de trabajo locales utilizan menos espacio de disco debido a que los datos se encuentran centralizados en un único lugar pero pueden ser accedidos y modificados por varios usuarios, de tal forma que no es necesario replicar la información.
- Los usuarios no necesitan disponer de un directorio " home" en cada una de las máquinas de la organización. Los directorios "home" pueden crearse en el servidor de NFS para posteriormente poder acceder a ellos desde cualquier máquina a través de la infraestrutura de red.
- También se pueden compartir a través de la red dispositivos de almacenamiento como disqueteras, CDROM y unidades ZIP. Esto puede reducir la inversión en dichos dispositivos y mejorar el aprovechamiento del hardware existente en la organización.

29.6.1. Cómo funciona NFS

El sistema NFS está dividido al menos en dos partes principales: un servidor y uno o más clientes. Los clientes acceden de forma remota a los datos que se encuentran almacenados en el servidor. Para que el sistema funcione correctamente se deben configurar y ejecutar unos cuantos procesos.

Nota

En FreeBSD 5.X se ha reemplazado portmap por rpcbind. de esta forma para los ejemplos que vamos a comentar a continuación se recuerda que en FreeBSD 5.X se debe reemplazar cualquier instancia de portmap por rpcbind.

El servidor de NFS debe ejecutar los siguientes dæmones:

Dæmon	Descripción
nfsd	El dæmonNFS, que atiende peticiones de clientes NFS.
mountd	El dæmon de montaje de NFS, que transporta las peticiones que nfsd(8) realiza.
portmap	El dæmon portmapper permite que los clientes NFS puedan descubrir qué puerto está utilizando el servidor de NFS.

El cliente también puede ejecutar un dæmon conocido , como nfsiod. El dæmon nfsiod atiende las peticiones provinientes del servidor NFS. Este dæmon es opcional y sirve para

mejorar el rendimiento pero no es necesario para el funcionamiento correcto del sistema. Se recomienda consultar nfsiod(8) para obtener más información.

29.6.2. Configuración de NFS

La configuración de NFS es un proceso relativamente sencillo. Para que los procesos anteriormente descritos se ejecuten en tiempo de arranque del sistema, basta con realizar paqueñas modificaciones en `/etc/rc.conf` .

En `/etc/rc.conf` del servidor de NFS se deben configurar las siguientes opciones:

```
portmap_enable="YES"
nfs_server_enable="YES"
mountd_flags="-r"
```

mountd se ejecuta automáticamente cuando se activa el servidor NFS.

En el cliente debemos asegurarnos de que se encuentra activada la activada la siguiente opción dentro de `/etc/rc.conf` :

```
nfs_client_enable="YES"
```

El archivo `/etc/exports` especifica los directorios o sistemas de ficheros que NFS exporta al exterior. Cada línea dentro de `/etc/exports/` especifia un sistema de ficheros y qué máquinas tienen derechos de acceso sobre dicho sistema. Además de los derechos de acceso se pueden definir otras opciones de acceso, tales como solo lectura o lectura y escritura. Existen multitud de opciones que pueden definirse sobre un directorio exportable pero en este manual sólo se van a comentar unas pocas. Consulte exports(5) para obtener una descripción más detallada.

Aquí se muestran algunos ejemplos de entradas para `/etc/exports` :

El siguiente ejemplo proporciona una idea de cómo exportar sistemas de ficheros, aunque los parámetros pueden diferir dependiendo de su entorno y su configuración de red. En dicho ejemplo, se exporta el directorio `/cdromm` a tres máquinas que se encuentran en el mismo dominio que el servidor (de ahí que no se especifique ningún nombre de dominio para cada máquina) o que pueden estar dadas de alta en `/etc/hosts` . En cualquier caso la opción `-ro` configura el sistema de ficheros de red como “sólo lectura” (“read-only”). Con esta opción los sistemas remotos no serán capaces de realizar cambios sobre el sistema de ficheros exportados.

```
/cdrom -ro host1 host2 host3
```

La siguiente línea exporta el directorio `/home` a tres máquinas utilizando direcciones IP. Esto resulta útil cuando disponemos de una red privada pero no disponemos de ningún servidor de DNS configurado. También se podría configurar `/etc/hosts` para que resolviera nombres de máquinas internos; consulte hosts(5) para obtener más información al respecto. La opción `-alldirs` permite que los subdirectorios del directorio `/home` también

se puedan utilizar como puntos de montaje. En otras palabras, esto permite que los clientes puedan trabajar sobre los subdirectorios en los que estén realmente interesados.

```
/home  -alldirs  10.0.0.2 10.0.0.3 10.0.0.4
```

La siguiente línea exporta el directorio `/a` de tal forma que puedan acceder a dicho directorio dos máquinas situadas en distintos dominios. La opción `-maproot=root` permite que el usuario `root` de la máquina cliente modifique los datos del sistema de ficheros en red como si fuera el usuario `root` del servidor. Si no se especifica la opción `-maproot=root` el usuario `root` del cliente puede no poseer los permisos necesarios para realizar modificaciones en el sistema de ficheros.

```
/a  -maproot=root  host.example.com box.example.org
```

Para que un cliente pueda acceder al sistema de ficheros exportado debe poseer permisos para ello. Debemos asegurarnos de que el cliente se encuentra listado en `/etc/exports`.

Dentro de `/etc/exports` cada línea representa información de exportación de un sistema de ficheros para un determinado conjunto de máquinas. Una máquina sólo puede aparecer una vez dentro de un sistema de ficheros exportable y el archivo sólo puede tener una única entrada por defecto. Por ejemplo, si suponemos que `/usr` es un único sistema de ficheros la siguiente configuración de `/etc/exports` sería incorrecta:

```
/usr/src   client
/usr/ports client
```

Existe un sistema de ficheros, concretamente `/usr`, que posee dos líneas con reglas de exportación para la misma máquina, `client`. El formato correcto para esta situación sería el siguiente:

```
/usr/src /usr/ports  client
```

Las propiedades de un sistemas de ficheros que se exporta al exterior deben aparecer agrupadas bajo la misma línea. Líneas que no poseen ningún cliente se tratan como si tuvieran una única máquina. Esto limita la forma en que pueden configurarse la exportaciones de sistemas de ficheros pero para la mayoría de la gente no suele ser un problema.

El ejemplo que se muestra a continuación es una muestra de una lista de exportación correcta, donde `/usr` y `/exports` son sistemas de ficheros locales:

```
# Exportar src y ports a cliente01 y cliente02, pero
# solo el cliente01 tiene acceso root
/usr/src /usr/ports -maproot=root    cliente01
/usr/src /usr/ports               cliente02
# Las maquinas cliente tienen acceso root y pueden montar todo lo ↺
que aparezca
# en /exports.  Cualquier sistema puede montar /exports/obj en modo
# solo lectura
/exports -alldirs -maproot=root      cliente01 cliente02
/exports/obj -ro
```

Se debe reiniciar el dæmon mountd siempre que se modifique el contenido del archivo `/etc/exports` para que los cambios surtan efecto. Esto se realiza enviando la señal HUP al proceso `mountd`:

```
# kill -HUP `cat /var/run/mountd.pid`
```

También se puede reiniciar FreeBSD para que se cargue la nueva configuración pero este mecanismo no resulta necesario si se ejecutan las órdenes como `root`, que ponen el servidor de NFS de nuevo en funcionamiento.

En el servidor de NFS:

```
# portmap
# nfsd -u -t -n 4
# mountd -r
```

En el cliente de NFS:

```
# nfsiod -n 4
```

En este punto todo debería estar preparado para poder anclar el sistema de ficheros remoto en la máquina cliente. En los siguientes ejemplos el nombre del servidor es `server` y el punto de montaje temporal utilizado por el cliente es `client`. Si se desea montar el sistema de ficheros de forma temporal o simplemente comprobar que la configuración funciona sin problemas se puede ejecutar una orden como la que se muestra a continuación con permisos de `root` en la máquina cliente:

```
# mount server:/home /mnt
```

Esta orden ancla el directorio `/home` del servidor en el directorio `/mnt` del cliente. Si todo funciona correctamente debería poder entrar en el directorio `/mnt` del cliente y ver todos los ficheros que se encuentran en el directorio `/home` del servidor.

Si queremos anclar automáticamente un sistema de ficheros remoto cuando la máquina está arrancando se puede añadir una línea como la siguiente dentro de `/etc/fstab`:

```
servidor:/home /mnt nfs rw 0 0
```

fstab(5) comenta todas las opciones disponibles.

29.6.3. Usos prácticos

El protocolo NFS tiene múltiples usos prácticos. Los más típicos se enumeran a continuación:

- Compartición de la unidad de CDROM entre varias máquinas. Esto resulta ser más barato y una forma más conveniente para instalar software en varias máquinas.
- En grandes redes puede ser más adecuado configurar un servidor central de NFS en el cual se almacenen todos los “homes” de los distintos usuarios. Estos directorios se

pueden exportar a través de la red de tal forma que los usuarios pueden trabajar con el mismo directorio independientemente de la máquina que utilicen.

- Varias máquinas pueden poseer el directorio `/usr/ports/distfiles` compartido. De este modo cuando necesitemos instalar un port en varias máquinas, se puede acceder rápidamente a las fuentes sin necesidad de bajarlas una vez para cada máquina.

29.6.4. Anclajes automáticos usando amd

Escrito por Wylie Stilwell.
Reescrito por Chern Lee.

El dæmon amd(8) ("the automatic mounter daemon", o dæmon de montaje automático) automáticamente ancla un sistema de ficheros remoto cuando se tiene que acceder a un fichero perteneciente a dicho sistema. Los sistemas de ficheros que permanecen inactivos durante un determinado periodo de tiempo son automáticamente desmontados por el mismo dæmon. Este dæmon proporciona una alternativa sencilla a la utilización de los montajes permanentes que normalmente se especifican a través del fichero `/etc/fstab`.

amd trabaja actuando como un servidor servidor de NFS para los directorios `/host` y `/net`. Cuando se accede a algún fichero ubicado bajo estos directorios amd busca el punto de montaje remoto y automáticamente lo monta. El directorio `/net` se utiliza para anclar sistemas de ficheros remotos especificados mediante direcciones IP, mientras que el directorio `/host` almacena aquellos sistemas de ficheros remotos que han sido especificados mediante un nombre de máquina.

amd detecta cualquier intento de acceder a un fichero dentro del directorio `/host/foobar/usr` y se encarga de montar el sistema de ficheros remoto (`/usr`) en la máquina, en caso de que no estuviera ya anclado.

Ejemplo 29.1. Anclaje de una exportación utilizando amd

`showmount` muestra los puntos de montaje que posee una máquina remota. Por ejemplo para conocer los montajes de un máquina llamada `foobar`, se puede utilizar:

```
% showmount -e foobar
Exports list on foobar:
/usr                               10.10.10.0
/a                                 10.10.10.0
% cd /host/foobar/usr
```

Como se observa en el ejemplo, `showmount` muestra el directorio `/usr` como una exportación. Cuando se cambia el directorio actual al directorio `/host/foobar/usr` el dæmon amd intenta resolver el nombre `foobar` y automáticamente ancla el sistema de ficheros remoto.

El dæmon amd se puede ejecutar a partir de los scripts de inicio, utilizando la siguiente línea del archivo de configuración `/etc/rc.conf` :

```
amd_enable="YES"
```

Además, amd soporta opciones adicionales que pueden definirse mediante la variable `amd_flags` . Por defecto, la variable `amd_flags` posee las siguientes opciones:

```
amd_flags="-a /.amd_mnt -l syslog /host /etc/amd.map /net /etc/amd.↺
map"
```

El archivo `/etc/amd.map` define las opciones por defecto con las cuales se anclan los sistemas de ficheros remotos. El archivo `/etc/amd.conf` define algunas características avanzadas para el dæmon amd.

Se ruega consultar las páginas del manual de amd(8) y de amd.conf(5) para obtener más información.

29.6.5. Problemas de integración con otras plataformas

Escrito por John Lind.

Determinados adaptadores Ethernet para sistemas basados en el bus ISA poseen restricciones que pueden producir serios problemas de red, en particular con el protocolo NFS. Estos problemas no son específicos de FreeBSD, pero los sistemas FreeBSD se ven afectados por ellos.

El problema surge casi siempre cuando el sistema (FreeBSD) está empotrado dentro de una red compuesta por estaciones de trabajo de alto rendimiento, como por ejemplo estaciones de Silicon Graphics y de Sun Microsystems. El montaje del sistema de ficheros remoto suele funcionar perfectamente y algunas operaciones sobre el el sistema de ficheros pueden tener éxito pero de repente el el servidor que no responde a las peticiones del cliente, aunque peticiones y respuestas de otros clientes funcionan con normalidad y se continúan procesando. Esto sucede en los sistemas clientes, tanto en sistemas FreeBSD como en otras estaciones de trabajo. En muchos sistemas, lo único que se puede hacer es resetear la máquina de forma abrupta, ya que el bloqueo producido por el protocolo NFS no se puede solucionar.

Aunque la solución “correcta” consiste en obtener un adaptador Ethernet con mayor rendimiento y capacidad, todavía se puede aplicar un parche sencillo que puede llegar a permitir un funcionamiento sin problemas. Si el sistema FreeBSD actúa como servidor de *NFS* se puede incluír la opción `w=1024` cuando el ejecute una petición de montaje sobre dicho

servidor. Si FreeBSD dicho servidor. Si FreeBSD actúa como cliente de *NFS*, se puede ejecutar mount(8) con el parámetro `-r=1024`. Estas opciones se pueden especificar en el `/etc/fstab` del cliente para que entren en funcionamiento cuando se realicen montajes automáticos y también se puede utilizar el parámetro `-o` de mount(8) cuando se realicen montajes manuales.

Resulta apropiado resaltar que existe un problema totalmente distinto que algunas veces se confunde con el que acabamos de describir, que aparece cuando el servidor y los clientes se encuentran en redes diferentes. Si nos encontramos en esta situación debemos *asegurarnos* de que nuestros " routers" están encaminando correctamente los paquetes UDP que genera el protocolo NFS pues en caso contrario el sistema no funcionará, independientemente de los ajustes que se realicen en el cliente o en el servidor.

En los siguientes ejemplos `fastws` es el nombre de una estación de trabajo de altas prestaciones y `freebox` es el nombre de un sistema FreeBSD con un adaptador Ethernet de bajas prestaciones. Se pretende además exportar el directorio `/sfcompartido` (ver exports(5)) y el directorio `/projecto`. Tenga en cuenta que en cualquier caso puede resultar útil definir opciones adicionales a las que que se muestran en el siguiente ejemplo, como pueden ser `hard`, `soft` o `bg`. Esto dependerá de la aplicación que utilice el sistema de ficheros remoto.

Ejemplos de configuración para el sistema FreeBSD (`freebox`) que actúa como cliente. Configuración del archivo `/etc/fstab` de `freebox`:

```
fastws:/sfcompartido /projecto nfs rw,-r=1024 0 0
```

Orden de ejecución manual para `freebox`:

```
# mount -t nfs -o -r=1024 fastws:/sfcompartido /projecto
```

Ejemplos de configuración para el sistema FreeBSD que actúa como servidor. Configuración de `/etc/fstab` de `fastws`:

```
freebox:/sfcompartido /projecto nfs rw,-w=1024 0 0
```

Orden de ejecución manual para `fastws`:

```
# mount -t nfs -o -w=1024 freebox:/sfcompartido /projecto
```

Casi cualquier adaptador Ethernet de 16 bits permite operar sin operar sin las restricciones anteriores sobre el tamaño de lectura o escritura especificado por defecto.

Por si alguien estuviera interesado a continuación se muestra el error que aparece en estos casos, lo cual explica por qué decimos que el error resulta irrecuperable. NFS trabaja típicamente con un tamaño de "bloque" de 8 K (aunque se pueden producir fragmentos de menor tamaño). Debido a que el máximo tamaño de los paquetes Ethernet se encuentra alrededor de los 1500 bytes el "bloque" de NFS se trocea en varios paquetes Ethernet aun-

que desde el punto de vista del protocolo NFS se trata como si fuese un único paquete. Los trozos deben reensamblarse en el destino y se debe enviar una *confirmación* para el bloque recibido. Las estaciones de trabajo de altas prestaciones pueden soltar paquetes NFS de forma contínua uno después de otro, lo más juntos posible. Por otro lado en las tarjetas de red más pequeñas y de menor capacidad puede ocurrir que un paquete recien llegado a la tarjeta sobreescriba información perteneciente a un paquete anterior antes de que llegue a ser transmitido completamente, de tal forma que al recibirse el bloque NFS no puede ser ni reconstruido ni ni reconocido. Como resultado de este proceso la máquina tratará de enviar el mismo paquete transcurridos unos instantes de espera, pero se tratarán de enviar de nuevo los 8 K que constituyen un bloque NFS, y de esta forma se repetirá el el proceso, así hasta el infinito.

Si se mantiene el tamaño del bloque por debajo del tamaño de paquete máximo de Ethernet, podemos asegurar que cualquier paquete Ethernet transporta un bloque NFS, el cual puede asentirse individualmente, evitando así la explosión de paquetes y el eventual bloqueo del sistema.

Desbordamientos circulares del “buffer” (“ overruns”) pueden producirse si nos encontramos con una estación de trabajo de altas prestaciones que envía contínuamente mucho tráfico a un sistema convencional, pero con tarjetas Ethernet de buena calidad, estos desbordamientos resultan altamente improbables para el caso de los tamaños de bloque por defecto generados por el sistema NFS. Cuando se produce un desbordamiento, las unidades afectadas se retransmiten, y existe una gran probabilidad de que se reciban, se reensamblen y se confirmen.

29.7. Ejecución sin disco duro

Actualizado por Jean-François Dockès.
Reorganizado y ampliado por Alex Dupre.

Una máquina FreeBSD se puede arrancar a través de la red y operar sin que necesite poseer ningún disco, utilizando sistemas de ficheros de un servidor de NFS. No se necesita realizar ninguna modificación al sistema, salvo configurar determinados ficheros. Este tipo de sistemas se pueden configurar fácilmente puesto que FreeBSD dispone de todos los elementos necesarios:

- Existen al menos dos formas de cargar el núcleo del sistema operativo a través de la red:
  - PXE: El sistema de Intel® conocido como Preboot Execution Environment. Se trata de una especie de arranque inteligente a partir de una memoria de sólo lectura (ROM) que se encuentra en algunas placas bases y tarjetas de red. Se puede obtener más información en pxeboot(8).
  - El port etherboot (net/etherboot) genera código de sólo lectura (código ROM) que se puede utilizar para arrancar máquinas a través de la red. Dicho código se puede

instalar en una memoria de arranque tipo PROM en algunas tarjetas de red o se puede cargar en una disquetera (o disco duro), y también en un sistema de ficheros MS-DOS® que esté en ejecución. Varias tarjetas de red soportan este mecanismo.

- Existe un script de ejemplo (`/usr/share/examples/diskless/clone_root`) que facilita la creación y el mantenimiento del sistema de ficheros raíz de la estación de trabajo en el servidor. La configuración de este "script" se debe retocar ligeramente pero sirve como punto de partida para comenzar rápidamente.

- Existen ficheros estándar de arranque bajo `/etc` que dan soporte al arranque de máquinas sin disco.

- El "swapping", en caso de ser necesario, se puede realizar usando NFS y tambíen usando un disco duro local.

Existen varias formas de ejecutar una estación de trabajo sin discos. En el proceso se involucran distintos elementos y la mayoría se pueden adaptar a las necesidades del usuario. A continuación se describen variaciones sobre la configuración de un sistema sin discos, haciendo incapié en la simplicidad y compatibilidad con los "scripts" de arranque de FreeBSD. El sistema que vamos a describir tiene las siguientes características.

- Las estaciones de trabajo sin disco utilizan un sistema de ficheros `raíz` de sólo lectura y un sistema de ficheros compartido, también de sólo lectura, bajo `/usr`.

  El sistema de ficheros `raíz` es una copia del sistema raíz estandar de FreeBSD (normalmente del sistema raíz del servidor), donde se sobreescriben algunos archivos de configuración necesarios para la ejecución sin discos y para la configuración local específica de la máquina objetivo.

  Las partes del sistema de ficheros `raíz` que tiene que tener permisos de lectura y escritura se superponen con los sistemas de ficheros mfs(8) (FreeBSD 4.X) o md(4). Cualquier cambio que se produzca en dichas partes se perderá cuando se reinicie el sistema.

- El núcleo se transmite y se carga utilizando etherboot o bien PXE, dependiendo del hardware y los mecanismos que se soporten.

Atención

Como se ha comentado con anterioridad estos sistemas son inseguros. Se debe confinar dentro de una red protegida y el resto de las máquinas por defecto no deben confiar en estos métodos.

Toda la información que se presenta en esta sección se ha probado utilizando FreeBSD 4.9-RELEASE y 5.2.1-RELEASE. El texto se encuentra estructurado principalmente para utili-

zación en sistemas 4.X. Se insertan notas para indicar cambios producidos en las versiones 5.X.

29.7.1. Conocimientos previos

Configurar estaciones de trabajo sin discos es una operación relativamente sencilla pero en la que pueden cometerse errores. Estos errores resultan algunas veces difíciles de diagnosticar debido a razones que vamos a exponer a continuación. Por ejemplo:

- Diferentes opciones de tiempo de compilación pueden determinar comportamientos distintos en tiempo de ejecución.
- Los mensajes de error a menudo resultan crípticos o incluso no existen.

Se se quieren resolver los posibles problemas que puedan surgir resulta muy útil conocer el funcionamiento conceptual del mecanismo.

Para que el arranque se produzca exitosamente se deben realizar varias operaciones:

- La máquina necesita obtener algunos parámetros iniciales, tales como su dirección IP, el fichero ejecutable, el nombre del servidor y la ruta raíz. Esto se realiza utilizando los protocolos DHCP o BOOTP. DHCP es una extensión compatible del protocolo BOOTP y utiliza los mismos números de puertos y los mismos formatos de paquete básicos.

  Es posible configurar un sistema de tal forma que utilice sólamente BOOTP. En el sistema base de FreeBSD se incluye el programa servidor bootpd(8).

  No obstante DHCP posee varias ventajas sobre BOOTP (archivos de configuración más limpios, posibilidad de ejecutar PXE, junto con otras características que no se relacionan directamente con el tema que estamos tratando tratando) por lo que principalmente se va a describir la configuración de DHCP, proporcionando ejemplos equivalentes en bootpd(8) siempre que sea posible. La configuración de ejemplo se basa en el paquete software de ISC DHCP (en el servidor de prueba se instaló la versión 3.0.1.r12).

- La máquina sin disco necesita transferir uno o varios programas a la memoria local. Para ello se usa TFTP o bien NFS. La elección entre ambos se produce mediante la configuración de la compilación que se produce en varios lugares. Una fuente de error típica aparece cuando se especifican ficheros con el protocolo incorrecto: TFTP normalmente transfiere todos los ficheros desde un único directorio del servidor, de modo que espera nombres de ficheros relativos a dicho directorio. Por otro lado NFS necesita recibir rutas de fichero absolutas.

- El kernel y los programas de arranque intermedios deben ser inicializados y ejecutados. Existen diferencias importantes en este área:

  - PXE carga pxeboot(8), una versión modificada de la tercera fase del cargador de arranque de FreeBSD. loader(8) obtiene la mayoría de los parámetros necesarios para

arrancar el sistema y los deposita en variables de entorno del kernel antes de tranferir el control. En este caso es posible utilizar un un núcleo `GENERIC` .

- etherboot carga directamente el directamente el núcleo con menos trabajo previo que el método anterior. Para ello se debe compilar un núcleo con ciertas opciones.

PXE y etherboot funcionan muy bien en los sistemas 4.X. Dado que los núcleos de los sistemas 5.X permiten que el loader(8) realice más tareas, se prefiere usar PXE.

Si su BIOS y su tarjeta de red soportan PXE lo normal es utilizarlo. No obstante se puede arrancar un sistema 5.X utilizando etherboot.

- Para acabar la tarea la máquina necesita acceder al sistema de ficheros. En todos los casos se utiliza NFS.

No olvide consultar diskless(8).

29.7.2. Instrucciones de configuración

29.7.2.1. Configuración utilizando ISC DHCP

El servidor ISC DHCP puede responder tanto a peticiones de BOOTP como a peticiones de DHCP.

ISC DHCP no forma parte de la versión 4.9 de FreeBSD por lo que se debe instalar el port net/isc-dhcp3-server o el paquete correspondiente. Por favor, consulte Capítulo 4, *Instalación de aplicaciones: «packages» y ports* para obtener más información sobre los ports y los paquetes.

Una vez que ISC DHCP se encuentra instalado necesita un fichero de configuración para poder ejecutarse `/usr/local/etc/dhcpd.conf`). A continuación se muestra un ejemplo comentado, donde la máquina `margaux` utiliza etherboot y la máquina `corbieres` utiliza PXE:

```
default-lease-time 600;
max-lease-time 7200;
authoritative;

option domain-name "example.com";
option domain-name-servers 192.168.4.1;
option routers 192.168.4.1;

subnet 192.168.4.0 netmask 255.255.255.0 {
  use-host-decl-names on; ❶
  option subnet-mask 255.255.255.0;
  option broadcast-address 192.168.4.255;
```

```
  host margaux {
    hardware ethernet 01:23:45:67:89:ab;
    fixed-address margaux.example.com;
    next-server 192.168.4.4; ❷
    filename "/data/misc/kernel.diskless"; ❸
    option root-path "192.168.4.4:/data/misc/diskless"; ❹
  }
  host corbieres {
    hardware ethernet 00:02:b3:27:62:df;
    fixed-address corbieres.example.com;
    next-server 192.168.4.4;
    filename "pxeboot";
    option root-path "192.168.4.4:/data/misc/diskless";
  }
}
```

❶ Esta opción indica a dhcpd que envíe el valor que se encuentra en las declaraciones de `host` como el nombre de máquina para la máquina sin disco. Otra forma de hacer esto sería añadiendo una opción `option host-name margaux` dentro de las declaraciones de máquina.

❷ La directiva `next-server` selecciona el servidor de TFTP o de NFS que se debe utilizar para cargar el núcleo o el fichero cargador del núcleo (por defecto se utiliza la misma máquina que actúa como servidor de DHCP).

❸ La directiva `filename` define el archivo que etherboot o PXE cargará en el siguiente paso de ejecución. Debe especificarse de acuerdo con el método de transferencia seleccionado. Etherboot se puede compilar para que use NFS o TFTP. El sistema FreeBSD se configura por defecto para NFS. PXE utiliza TFTP por lo que se utiliza una ruta relativa para especificar el nombre del fichero (esto puede depender de la configuración del servidor de TFTP pero suele ser lo normal). Además PXE no carga el núcleo, lo hace `pxeboot`. Existen otras posibilidades interesantes, como cargar `pxeboot` desde el directorio `/boot` de una unidad de CD-ROM de FreeBSD (ya que pxeboot(8) puede cargar un núcleo `GENERIC` surge la posibilidad de utilizar PXE para arrancar desde una unidad de CD-ROM remota).

❹ La opción `root-path` define la ruta para el sistema de ficheros raíz utilizando la notación típica de NFS. Cuando se utiliza PXE, es posible dejar la dirección IP siempre y cuando no se active la opción del núcleo de BOOTP. El servidor NFS será en este caso el mismo que el servidor de TFTP.

29.7.2.2. Configuración utilizando BOOTP

A continuación se muestra la configuración equivalente utilizando bootpd (reducida a un único cliente). Esta configuración se debe situar en `/etc/bootptab`.

Por favor, recuerde que etherboot se debe compilar con la opción específica de `NO_DHCP_SUPPORT` para que pueda utilizar BOOTP y que PXE *requiere* DHCP. La única ventaja obvia de bootpd es que se encuentra disponible en el sistema base.

```
.def100:\
  :hn:ht=1:sa=192.168.4.4:vm=rfc1048:\
  :sm=255.255.255.0:\
  :ds=192.168.4.1:\
  :gw=192.168.4.1:\
  :hd="/tftpboot":\
  :bf="/kernel.diskless":\
  :rp="192.168.4.4:/data/misc/diskless":

margaux:ha=0123456789ab:tc=.def100
```

29.7.2.3. Preparación de un programa de arranque con Etherboot

La página web de Etherboot contiene una amplia documentación enfocada principalmente a los sistemas Linux pero en cualquier caso contiene información que puede resultar útil. En los siguientes párrafos se describe brevemente como se puede utilizar etherboot en un sistema FreeBSD.

Lo primero es instalar el port o paquete net/etherboot. El port de etherboot está en `/usr/ports/net/etherboot` . Si el árbol de ports está instalado en el sistema basta con ejecutar `make` en dicho directorio. Por favor, lea Capítulo 4, *Instalación de aplicaciones: «packages» y ports* para saber más sobre los ports y los paquetes.

Se puede modificar la configuración de etherboot (por ejemplo, para que use TFTP en lugar de NFS) editando el fichero `Config` que se encuentra en el directorio fuente de etherboot.

Para nuestros propósitos se utilizará un disquete de arranque. Para utilizar otros métodos (PROM o un programa MS-DOS®) por favor consulte la documentación de etherboot.

Para crear un disco de arranque se debe insertar un disco en la unidad de disquetes de la máquina donde se ha instalado etherboot, cambiar al directorio `src` dentro del árbol de directorios de etherboot y teclear:

```
# gmake bin32/tipo_de_dispositivo.fd0
```

`tipo_de_dispositivo` depende del tipo de tarjeta Ethernet que se encuentre instalada en la estación de trabajo sin disco. Consulte el fichero `NIC` en el mismo directorio para determinar cúal es el *`tipo_de_dispositivo`* que debe usted usar.

29.7.2.4. Arranque con PXE

Por defecto el cargador pxeboot(8) carga, valga la redundancia, el kernel vía NFS. El El cargador se puede compilar para que utilice TFTP en lugar de NFS especificando la op-

ción `LOADER_TFTP_SUPPORT` dentro de `/etc/make.conf`. Observe los comentarios de `/etc/defaults/make.conf` (o de `/usr/share/examples/etc/make.conf` para sistemas 5.X) para saber más detalles.

Existen otras dos opciones de `make.conf` no documentadas que pueden ser útiles para arrancar una máquina sin disco a través del puerto serie: `BOOT_PXELDR_PROBE_KEYBOARD` y `BOOT_PXELDR_ALWAYS_SERIAL` (esta última sólo existe en FreeBSD 5.X).

Para utilizar PXE cuando arranca la máquina normalmente el usuario tiene que seleccionar la opción `Boot from network` dentro del menú de opciones de la BIOS o pulsar un tecla de función cuando la máquina se está inicializando.

29.7.2.5. Configuración de servidores de TFTP y de NFS

Si PXE o etherboot se encuentran configurados para utilizar TFTP se necesita activar tftpd en el servidor de ficheros:

1. Crear un directorio desde el cual el dæmon tftpd servirá los ficheros, por ejemplo `/tftpboot`.

2. Añadir la siguiente línea a `/etc/inetd.conf`:

```
tftp dgram udp wait root /usr/libexec/tftpd tftpd -l -s /tftpboot
```

> **Nota**
>
> Parece que al menos algunas versiones de PXE utilizan la versión TCP de TFTP. En este caso se puede añadir una segunda línea, donde se reemplace `dgram udp` por `stream tcp`.

3. Indicar a inetd que vuelva a leer su fichero de configuración:

```
# kill -HUP `cat
                /var/run/inetd.pid`
```

Se puede situar el directorio `tftpboot` en cualquier parte del servidor. Debe asegurarse de que la localización se encuentra correctamente configurada tanto en `inetd.conf` como en `dhcpd.conf`.

En todos los casos también resulta necesario activar el sistema de NFS y exportar los sistemas de ficheros adecuados, todo ello en el servidor de NFS.

1. Añadir lo siguiente a `/etc/rc.conf`:

```
nfs_server_enable="YES"
```

2. Exportar el sistema de ficheros donde el directorio raíz sin disco se encuentra localizado añadiendo lo siguiente a `/etc/exports` (ajuste el punto de montaje de la unidad y sustituya *`margaux corbieres`* por el nombre de las estaciones de trabajo sin disco, según corresponda):

```
/data/misc  -alldirs -ro margaux corbieres
```

3. Indicar a mountd que vuelva a leer su archivo de configuración. Si en un primer paso se ha configurado la activación automática del sistema de NFS en `/etc/rc.conf` lo mejor es reiniciar para que los cambios surtan efecto.

```
# kill -HUP `cat
                /var/run/mountd.pid`
```

29.7.2.6. Construcción de un kernel sin disco

Si se utiliza etherboot, se necesita crear un archivo de configuración para el kernel de la máquina sin disco que posea las siguientes opciones (además de las opciones del núcleo habituales):

```
options     BOOTP          # Use BOOTP to obtain IP address/hostname
options     BOOTP_NFSROOT  # NFS mount root filesystem using BOOTP ↺
info
```

Puede resultar interesante utilizar además `BOOTP_NFSV3`, `BOOT_COMPAT` y `BOOTP_WIRED_TO` (consultar `LINT` en 4.X o `NOTES` en sistemas 5.X).

Los nombres de estas opciones son nombres históricos y ligeramente confusos ya que permiten un uso indistinto tanto de DHCP como de BOOTP dentro del núcleo (también resulta posible forzar la utilización única de o bien BOOTP o bien de DHCP).

Contruir el núcleo (vea Capítulo 8, *Configuración del kernel de FreeBSD*) y copiarlo al lugar especificado en el archivo `dhcpd.conf`.

Nota

Cuando se utiliza PXE, la construcción del núcleo con las opciones anteriores no resulta ser algo estrictamente necesario (aunque se recomienda). Activar dichas opciones provoca un mayor tráfico de peticiones de DHCP durante el arranque del núcleo, lo que puede dar lugar a pequeñas inconsistencias entre los nuevos valores y los

los valores recuperados por pxeboot(8) en casos muy específicos. La ventaja de utilizarlas consiste en que como un efecto colateral se configurará el nombre de la máquina. De otro modo tendríamos que configurar dicho nombre mediante otro método por ejemplo mediante la configuración específica de la máquina cliente a través del archivo `rc.conf`.

Nota

Para que el núcleo se pueda cargar sin problemas con etherboot en sistemas 5.X dicho núcleo tiene que tener compilado el soporte para *device hints*. Para ello normalmente se especifica la siguiente opción dentro del fichero de configuración del núcleo (consulte los comentarios del fichero `NOTES`):

```
hints  "GENERIC.hints"
```

29.7.2.7. Preparación del sistema de ficheros raíz

Se debe crear un sistema de ficheros raíz en las estaciones de trabajo sin disco, concretamente en la localización especificada por `root-path` dentro de `dhcpd.conf`. Las siguientes secciones describen dos formas de hacer esto.

29.7.2.7.1. Utilización del "script" `clone_root`

Este es el modo más rápido de crear un sistema de ficheros raíz, pero actulamente sólo se encuentra soportado en FreeBSD 4.X. El "script" de shell se encuentra en `/usr/share/examples/diskless/clone_root` y debe ser configurado al menos para ajustar el lugar donde se construirá el sistema de ficheros (concretamente la variable `DEST`).

Consulte los comentarios que se encuentran al comienzo del "script" para conocer cuales son las instrucciones que debe seguir. Allí se explica cómo se construye el sistema de ficheros base y como determinados ficheros se pueden sobreescribir de manera selectiva por versiones específicas para funcionar sin discos, para toda una subred o para una máquina individual. También allí se muestran ejemplos de los ficheros `/etc/fstab` y `/etc/rc.conf` para máquinas sin disco.

Los archivos `README` que se encuentran dentro de `/usr/share/examples/diskless` contienen mucha información de base, que junto con el resto de ejemplos dentro del directorio `diskless` sirven para documentar un método de configuración distinto del que se utiliza en `clone_root` y en los " scripts" del sistema de `/etc`, que resultan ser un tanto confusos. No obstante se pueden utilizar a modo de referencia, excepto si se prefiere uti-

lizar el método que se describe en ellos, en cuyo caso se necesitará modificar y adaptar los “scripts” de forma adecuada.

29.7.2.7.2. Utilización del procedimiento estándar de `make world`

Este método se puede utilizar tanto en FreeBSD 4.X o 5.X y se instalará un sistema completamente nuevo (no sólo el sistema de ficheros raíz) dentro de `DESTDIR`. Basta con ejecutar el siguiente “ script”:

```
#!/bin/sh
export DESTDIR=/data/misc/diskless
mkdir -p ${DESTDIR}
cd /usr/src; make world && make kernel
cd /usr/src/etc; make distribution
```

Una vez ejecutado puede ser necesario ajustar los ficheros `/etc/rc.conf` y `/etc/fstab` que se encuentran en `DESTDIR` de acuerdo con nuestras necesidades.

29.7.2.8. Configuración de la partición de intercambio

En caso de ser necesario se puede acceder a un fichero de intercambio (swap) a través del sistema NFS. Uno de los métodos típicamente utilizados para realizar esta tarea ha sido retirado de la distribución 5.X.

29.7.2.8.1. NFS swap en sistemas FreeBSD 4.X

La ubicación del fichero de intercambio y su tamaño se puede especificar con las opciones FreeBSD-specific 128 y 129 de BOOTP/DHCP. A continuación se muestran varios ejemplos de ficheros de de configuración para ISC DHCP 3.0 o bootpd:

1. Añadir las siguientes líneas al fichero `dhcpd.conf` :

   ```
   # Global section
   option swap-path code 128 = string;
   option swap-size code 129 = integer 32;

   host margaux {
     ... # Standard lines, see above
     option swap-path "192.168.4.4:/netswapvolume/netswap"  ;
     option swap-size 64000;
   }
   ```

   `swap-path` es la ruta al directorio donde se instalarán los archivos de intercambio. Cada Cada fichero se denomina `swap.direccion-ip-del-cliente` .

   Versiones más antiguas de dhcpd usaban una sintáxis del estilo de `option option-128 "...`, lo cual ya no está soportado.

   `/etc/bootptab` normalmente utiliza la siguiente sintaxis:

```
T128="192.168.4.4:/netswapvolume/netswap":T129=0000fa00
```

Nota

El tamaño del fichero dedicado a intercambio se debe expresar en `/etc/bootptab` en formato hexadecimal.

2. En el servidor de ficheros NFS donde va a residir el fichero de "swap" se debe(n) crear dicho(s) fichero(s)

```
# mkdir /volumenintercambiored/intercambiored
# cd /volumenintercambiored/intercambiored
# dd if=/dev/zero bs=1024 count=64000 of=swap.192.168.4.6
# chmod 0600 swap.192.168.4.6
```

   `192.168.4.6` es la dirección IP del cliente sin disco.

3. En el servidor NFS añadir a `/etc/exports` la siguiente línea:

```
/volumenintercambiored    -maproot=0:10 -alldirs margaux corbieres
```

   A continuación indicar a mountd que vuelva a leer el fichero `/etc/exports` como se ha indicado anteriormente.

29.7.2.8.2. NFS swap en FreeBSD 5.X

El núcleo no soporta la activación del intercambio a través de NFS en tiempo de arranque. De esta forma la "swap" se debe activar mediante los "scripts" montando un sistema de ficheros de lectura-escritura y creando y activando el fichero de intercambio. Para crear un fichero de intercambio de un determinado tamaño se puede ejecutar lo siguiente:

```
# dd if=/dev/zero of=/ruta/al/fichero/de/intercambio bs=1k count=1 ↺
oseek=100000
```

Para activar el intercambio se tiene que añadir la siguiente línea al fichero de configuración `rc.conf`:

```
swapfile=/ruta/al/fichero/de/intercambio
```

29.7.2.9. Varios

29.7.2.9.1. Ejecución con un `/usr` de sólo lectura

Si la estación de trabajo sin disco se configura para utilizar el sistema X-Window se tiene que ajustar el fichero de configuración de xdm debido a que dicho fichero sitúa por defecto el fichero de "logs" de errores en el directorio `/usr`.

29.7.2.9.2. Uso de un servidor no-FreeBSD

Cuando el servidor del sistema de ficheros raíz no ejecuta FreeBSD se tiene que crear un sistema de ficheros raíz sobre una máquina FreeBSD para después copiarlo al servidor original mediante las órdenes `tar` o `cpio`.

En esta situación algunas veces surgen varios problemas relacionados con los dispositivos especiales que se encuentran en el directorio `/dev` debido a los diferentes tamaños de los enteros mayor/menor. Una solución para este problema consiste en exportar un directorio del servidor no-FreeBSD, montar este directorio en la máquina FreeBSD anterior y ejecutar `MAKEDEV` en dicha máquina para crear las entradas de dispositivo correctas (FreeBSD 5.0 y posteriores utilizan devfs(5) para ubicar nodos de dispositivos de forma transparente para el usuario de tal modo que la ejecución de `MAKEDEV` en estos sistemas no sirve para nada).

29.8. RDSI

la página de RDSI de Dan Kegel constituye un recurso de información bastante bueno sobre la tecnología RDSI (ISDN en inglés) y sobre el hardware relacionado.

A continuación se comenta un esquema rápido sobre RDSI:

- Si usted vive en Europa le puede resultar útil leer la sección sobre tarjetas RDSI.
- Si se prevee utilizar RDSI principalmente para conectarse a Internet a través de un Proveedor de Servicios utilizando un mecanismo de marcación automática no dedicado (dial-up), se puede echar un vistazo a los Adaptadores de Terminal. Dichos adaptadores proporciona la mayor flexibilidad y garantiza los mínimos problemas en caso de cambio de cambio de proveedor.
- Si estamos conectados a dos LAN o conectando a Internet con una conexión RDSI dedicada puede ser interesante considerar la opción de usar un "router/bridge" único.

El coste es un factor importante a la hora de determinar qué solución se debe escoger. Las siguientes opciones se encuentran ordenadas desde las más baratas hasta las más caras.

29.8.1. Tarjetas RDSI

Escrito por Hellmuth Michaelis.

La implementación de RDSI que posee FreeBSD soporta sólamente el estandar DSS1/Q.931 (también conocido como Euro-RDSI) utilizando tarjetas pasivas. A partir de FreeBSD 4.4

se soportan también algunas tarjetas activas usando firmware que además soporta otros protocolos de señalización; esto también sucede con la primera tarjeta RDSI de acceso primario (PRI) soportada.

El software isdn4bsd permite conectar con otras pasarelas RDSI utilizando IP sobre HDLC o bien PPP PPP síncrono: ambos mediante el uso del PPP del núcleo con `isppp`, una versión modificada del controlador sppp(4) o mediante la utilización del PPP de entorno de usuario, ppp(8). Si se utiliza ppp(8) de entorno de usuario se pueden agrupar dos o mas canales B de RDSI (channel bonding). Existe también software que permite a una máquina responder a llamadas de teléfono y algunas cosas más como un modem de 300 baudios.

Cada vez se soportan más tarjetas RDSI bajo FreeBSD y los informes existentes muestra que FreeBSD se utiliza con dichas tarjetas de forma satisfactoria en toda Europa y también en otras partes del mundo.

Las tarjetas RDSI pasivas soportadas en FreeBSD son principalmente las que poseen el chip Infineon (antiguamente Siemens) ISAC/HSCX/IPAC. También las tarjetas RDSI con los chips de Cologne (en bus ISA exclusivamente), tarjetas PCI con el chip Winbond W6692, algunas tarjetas con combinaciones de los chips Tiger 300/320/ISAC y también algunas tarjetas basadas en chips propietarios como las AVM Fritz! PCI V.1.0 y AVM Fritz! PnP.

Actualmente las tarjetas RDSI activas soportadas son las AVM B1 (ISA y PCI) BRI, y las AVM T1 PCI PRI.

Se puede consultar `/usr/share/examples/isdn/` para obtener documentación sobre isdn4bsd y también en la página principal de isdn4bsd, donde hay enlaces de ayuda, erratas y mucha más información útil, como por ejemplo el manual de isdn4bsd.

Si se quiere añadir soporte para un protoclo RDSI distinto para una tarjeta RDSI que no se encuentra soportada o para mejorar isdn4bsd en algún aspecto por favor póngase en contacto con Hellmuth Michaelis.

Para realizar consultas referentes a la instalación, configuración y depuración de problemas relacionados con isdn4bsd le recomendamos recurrir a la lista de correo freebsd-isdn.

29.8.2. Adaptadores de terminal RDSI

Los Adaptadores de Terminal (TA), son para RDSI lo que los modems son para las líneas de teléfono convencionales.

La mayor parte de los TAs utilizan el conjunto de instrucciones AT de los modem Hayes y se pueden utilizar en lugar del modem.

Un TA opera básicamente de igual forma que un modem, diferenciándose en que las velocidades de conexión y “throughput” son mucho más grandes. La configuración de PPP se realiza exactamente igual que para una configuración de modem convencional.

La ventaja principal de utilizar un TA para conectarse a un proveedor de servicios de internet consiste en que se puede usar PPP dinámico. Ya que el espacio de direcciones de IP se está direcciones de IP se está convirtiendo cada vez convirtiendo en un recurso cada dí más limitado y escaso los proveedores ya no desean proporcionar direcciones IP estáticas a sus clientes. No obstante la mayoría de los "routers standalone" no son capaces de adquirir direcciones IP dinámicas.

Los TAs confían completamente en el dæmon de PPP que se está ejecutando para proporcionar fiabilidad y estabilidad en la conexión. De esta forma si se tiene configurado PPP se puede migrar fácilmente de la utilización de modems analógicos al uso de RDSI. No obstante si existía algún problema con PPP antes de efectuar la migración dichos problemas persistirán en RDSI.

Si se desea máxima estabilidad se puede utilizar la opción PPP, no el PPP a nivel de usuario.

Se sabe que los siguientes TAs funcionan con FreeBSD:

- Motorola BitSurfer y Bitsurfer Pro
- Adtran

La mayoría de los demás TAs probablemente también funcionen puesto que los fabricantes siempre tratan de que sus productos puedan aceptar la mayoría de las órdenes AT.

El problema que existe con los TAs es que, como sucede con los modems, se necesita tener una buena tarjeta serie instalada en el sistema.

Se recomienda consultar el tutorial FreeBSD Serial Hardware para obtener una comprensión detallada del funcionamiento de los dispositivos serie en FreeBSD y para comprender las diferencias entre puertos serie síncronos y asíncronos.

Un TA que se ejecuta a través de un puerto serie (asíncrono) está limitado a 115.2 Kbs, aunque la conexión RDSI sea de 128 Kbs. Para utilizar completamente el ancho de banda que RDSI proporciona, se debe conectar el TA a una tarjeta serie síncrona.

No se engañe creyendo que comprando un TA interno hará desaparecer los problemas síncronos/asíncronos. Los TA internos simplemente disponen de un chip de puerto serie instalado de fábrica. Lo único que se consigue con estos dispositivos es no tener que enchufarlos a la red elétrica ahorrando así un enchufe y no tener que comprar un cable serie, pero los problemas dichos anteriormente permanecen.

Una tarjeta asíncrona con un TA resulta ser al menos tan rápida como un "router standalone" y si FreeBSD controla dicha tarjeta se puede adaptar más fácilmente.

La elección de una tarjeta síncrona/TA versus un "router standalone" se trata en la mayoría de los casos de una cuestión cuasi-religiosa. Han existido diversas discusiones sobre

este tema en varias listas de correo. Nosotros recomendamos que busque información en los históricos para para poder sopesar los pros y los contras que se han esgrimido en tales discusiones.

29.8.3. “bridges/routers” RDSI “Stand-alone”

Los “bridges” o “routers” RDSI no son específicos de FreeBSD o de cualquier otro sistema operativo. Para una descripción completa de la tecnología de “bridge” y de pasarela de red por favor consulte cualquier libro sobre redes.

En el contexto de esta sección los términos “router”, pasarela y “bridge” se utilizarán indistintamente.

Según va bajando el coste de los “ routers/bridges” RDSI su utilización entre el público en general va en aumento. Un “router” RDSI es una pequeña caja que se conecta directamente a la red Ethernet local y que gestiona sus propias conexiones con el “bridge/router” remoto. Posee un software preconfigurado para comunicarse vía PPP y tambíen utilizando otros protocolos de uso común.

Un router sopota una mayor tasa de paquetes (throughput) que un “standalone TA”, ya que utiliza una conexión RDSI síncrona de forma completa.

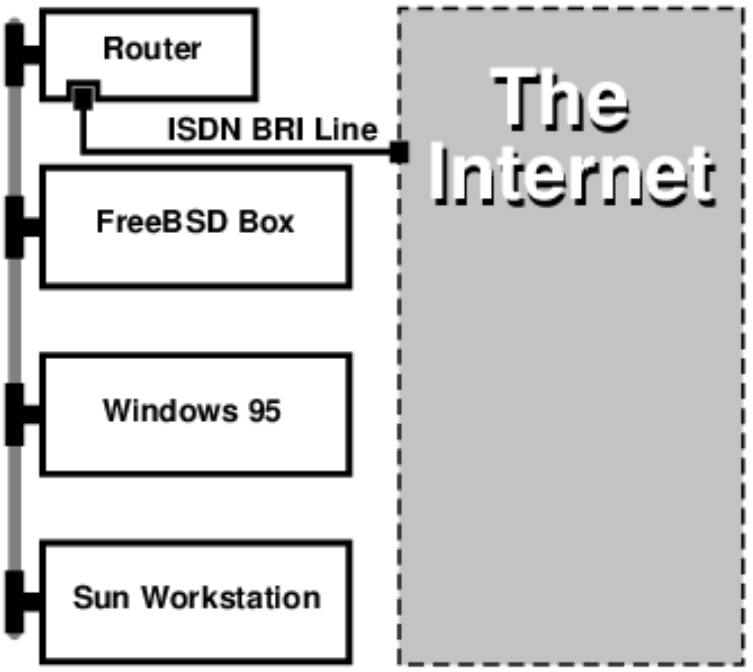
El problema principal que surge con los “routers” y los “bridges” RDSI es que la interoperatibilidad entre fabricantes muchas veces causa problemas. Si se está planificando conectarse a un proveedor de servicios resulta conveniente discutir previamente con ellos las necesidades y requisitos.

Si se tiene en mente conectar dos segmentos LAN tales como su LAN de casa y la LAN de su oficina RDSI proporciona la solución más simple y menos costosa de gestionar. Esto es así porque al comprar usted mismo el equipamiento necesario para ambos extremos de la conexión tiene usted el control sobre el enlace y puede asegurar su correcto funcionamiento.

Por ejemplo, si queremos conecar una computadora casera o una sucursal de la red de oficinas con la oficinal central, se puede utilizar una configuración como la que se muestra a continuación.

Ejemplo 29.2. Sucursal o red doméstica

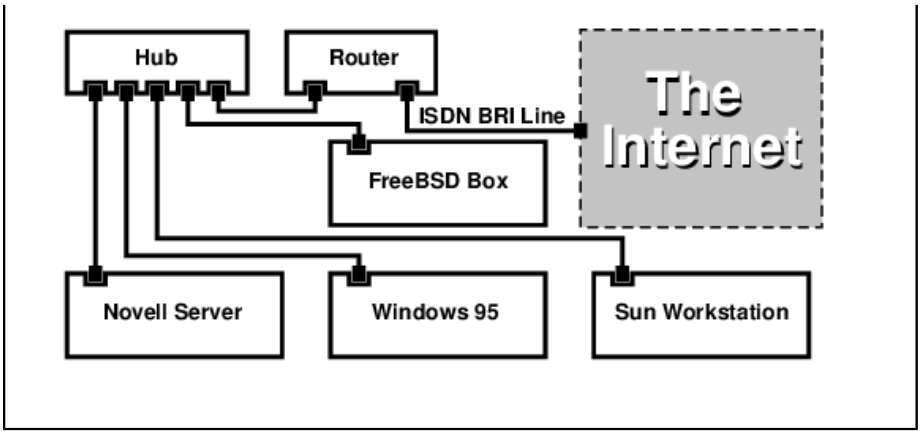
La red utiliza una topología basada en bus con Ethernet tipo 10 base 2 (“thinnet”). Se conecta, en caso de ser necesario, el “router” a la red cableada mediante un “transceiver” AUI/10BT.



Si nuestra sucursal o red hogar está compuesta únicamente por una computadora se puede utilizar un cable cruzado de par trenzado para conectar con el "router standalone" de forma directa.

Ejemplo 29.3. Oficina central u otra LAN

La red utiliza una topología en estrella basada en Ethernet de 10 base T ("Par Trenzado").



Una gran ventaja que poseen la mayoría de los "routers/bridges" es que pueden gestionar al mismo tiempo dos conexiones PPP *independientes* destinadas a dos organizaciones distintas. Esta funcionalidad no se proporciona en la mayoría de los TAs, excepto para determinados modelos (normalmente más caros) que se fabrican con dos puertos serie. No confunda esto con la agrupación de canales, MPP, etc.

Esta característica puede resultar muy útil si, por ejemplo, se dispone de una conexión RDSI dedicada con la oficina y queremos introducirnos en ella pero no queremos utilizar otra línea RDSI en el trabajo. Un "router" situado en las instalaciones de la oficina puede gestionar una conexión de canal B dedicada (64 Kpbs) hacia internet y utilizar el otro canal B como una conexión de datos independiente. El segundo canal B se puede utilizar para marcación remota ("dial-in" y " dial-out") o para agrupación dinámica de canales (MPP, etc) en conjunción con el primer canal B con el objetivo de obtener un mayor ancho de banda.

Un "bridge" Ethernet permite transmitir más tráfico aparte del tráfico IP. Se puede transmitir IPX/SPX o cualquier otro protocolo que se esté utilizando.

29.9. NIS/YP

Escrito por Bill Swingle.
Ampliado por Eric Ogren y Udo Erdelhoff.

29.9.1. ?Qué es esto?

NIS, siglas de Network Information Services (Servicios de Información de Red), fué un servicio desarrollado por Sun Microsystems para centralizar la administración de sistemas UNIX® (originalmente SunOS™). Actualmente se considera como un estándar de la

industria; los principales sistemas tipo UNIX® (Solaris™, HP-UX, AIX®, Linux, NetBSD, OpenBSD, FreeBSD, etc) implementan NIS.

NIS también se conocía como el servicio de páginas amarillas pero debido a problemas legales debidos a la propiedad de marcas comerciales, Sun tuvo que cambiar el nombre. El antíguo término ("Yellow Pages" o yp) todavía se ve y se utiliza con frecuencia.

Se trata de un sistema cliente servidor basado en llamadas RPC que permite a un grupo de máquinas que se encuentran definidas dentro de un dominio administrativo NIS compartir un conjunto de ficheros de configuración. Esto permite al administrador de sistemas por un lado configurar clientes NIS de forma minimalista y por otro lado centralizar la gestión de los ficheros de configuración en una única ubicación (una sola máquina).

Se trata de algo similar al sistema de dominio de Windows NT® aunque la implementación interna no se puede comparar, la funcionalidad y el servicio obtenido son similares.

29.9.2. Términos/procesos que debe usted conocer

Existen varios conceptos y varios procesos de usuario que el usuario no versado en estos temas suele encontrarse la primera vez que se intenta implantar un servicio de NIS en FreeBSD, tanto si se intenta configurar un servidor como si se intenta configurar un cliente:

Term	Description
NIS domainname	Un servidor maestro de NIS y todos sus clientes (incluyendo a sus servidores esclavos) poseen el mismo nombre dominio NIS. Al igual que ocurre con el nombre de dominio de Windows NT®, el nombre de dominio de NIS no tiene nada que ver con el nombre de dominio de DNS.
portmap	Debe ejecutarse para que se activen las llamadas a procedimientos remotos (Remote Procedure Call o RPC) que son utilizadas por NIS. Si portmap no se está ejecutando no se podrá ejecutar ni clientes ni servidores de NIS.
ypbind	"Asocia" un cliente con un servidor NIS. Primeramente se lee el nombre de dominio NIS del sistema y utilizando RPC se conecta con el servidor. ypbind es la parte central de la comunicación cliente servidor del sistema NIS; si ypbind muere en una máquina cliente, dicha máquina no podrá acceder al servidor NIS.

Term	Description
ypserv	Debe ejecutarse sólamente en los servidores NIS; se trata del proceso servidor de NIS. Si ypserv(8) muere, el servidor no será capaz de responder a peticiones NIS (no obstante, si se definen servidores NIS esclavos la situación puede recuperarse). Existen algunas implementaciones de NIS (no es el caso de FreeBSD) que no intentan conectarse con otro servidor si el servidor con otro servidor si el servidor que se estaba que se estaba utilizando muere. A menudo lo único que se puede hacer en estos casos es reiniciar el servidor (el proceso o la propia máquina) o el proceso ypbind del cliente.
rpc.yppasswdd	Otro proceso que sólo debe ejecutarse en el servidor maestro de NIS; se trata de un dæmon que permite a los clientes de NIS modificar las contraseñas de los usuarios. Si no se ejecuta este dæmon los usuarios tendrán que entrar en el servidor maestro de NIS para cambiar sus contraseñas allí.

29.9.3. ?Cómo funciona?

Existen tres tipos de máquinas dentro del entorno NIS: los servidores maestros, los servidores esclavos y los clientes de NIS. Los servidores actúan como repositorios centrales para almacenamiento de información de configuración. Los servidores maestros mantienen una copia maestra de dicha información, mientras que los servidores esclavos mantienen copias de la información maestra por motivos de redundancia. Los servidores se encargan de transmitir la información necesaria a los clientes a petición de estos últimos.

De esta forma se pueden compatir mucha información contenida en varios archivos. Los ficheros `master.passwd`, `group` y `hosts` normalmente se comparten a través de NIS. Siempre que un proceso en un cliente necesita información que, en caso de no utilizar NIS, se podría recuperar de ficheros locales, en este caso se envía una solicitud al servidor NIS con el que nos encontramos asociados.

29.9.3.1. Clases de máquinas

- *Servidor de NIS maestro*. Este servidor, semejante a un controlador de dominio primario de Windows NT® mantiene todos los archivos que utilizan los clientes. Los ficheros `passwd`, `group` y algunos otros se encuentran ubicados en el servidor maestro.

Nota

Resulta posible configurar una máquina para que actúe como servidor NIS maestro para más de un dominio NIS. No obstante esta configuración no se va a tratar en esta introducción, en la cual asumimos un entorno NIS de tamaño relativamente pequeño.

- *Servidores de NIS esclavos.* Semejantes a los controladores de backup de Windows NT®, los servidores NIS esclavos se utilizan para proporcionar redundancia en entornos de trabajo donde la disponibilidad del servicio resulta muy importante. Además se utilizan para distribuir la carga que normalmente soporta un servidor maestro: los clientes de NIS siempre se asocian con el servidor de NIS que posee mejor tiempo de respuesta, y esto y esto también incluye a los servidores de NIS esclavos.

- *Clientes NIS.* Los clientes NIS, de forma semejante a las estaciones de trabajo de Windows NT®, se validan contra un servidor NIS (en el caso de Windows NT® se validan contra un controlador de dominio) para acceder al sistema.

29.9.4. Uso de NIS/YP

Esta sección trata sobre cómo configurar y poner en funcionamiento un entorno de NIS sencillo.

Nota

Esta sección supone que se está utilizando utilizando FreeBSD 3.3 o posteriores. Las instrucciones dadas aquí *probablemente* funcionen también en cualquier versión de FreeBSD superior a la 3.0 pero no podemos garantizar que esto sea así.

29.9.4.1. Planificación

Vamos a suponer que somos el administrador de un pequeño laboratorio de una universidad. En este laboratorio, compuesto por 15 máquinas FreeBSD, actualmente no existe ningún punto de administración centralizada; cada máquina posee sus sus propios `/etc/passwd` y `/etc/master.passwd`. Estos ficheros se encuentran sincronizados el uno con el otro mediante intervención manual; por tanto, cuando queramos añadir un usuario a

nuestro laboratorio tendremos que ejecutar `adduser` en todas las máquinas. Claramente esta situación tiene que cambiar, de tal forma que hemos decidido crear un dominio NIS en el laboratorio usando dos máquinas como servidores NIS.

La configuración de nuestro laboratorio debería ser algo parecido a lo siguiente:

Nombre de máquina	Dirección IP	Papel
`ellington`	`10.0.0.2`	servidor NIS maestro
`coltrane`	`10.0.0.3`	Servidor NIS esclavo
`basie`	`10.0.0.4`	Estación de trabajo del profesorado
`bird`	`10.0.0.5`	máquina cliente
`cli[1-11]`	`10.0.0.[6-17]`	Resto de máquinas clientes

Si se está configurando un esquema de NIS por primera vez es una buena idea detenerse a pensar cómo queremos implantar el sistema. Existen varias decisiones que se deben tomar independientemente del tamaño de nuestra red.

29.9.4.1.1. Elección del nombre de dominio NIS

Este nombre puede no ser el "nombre de dominio" al que estamos acostumbrados. Resulta más preciso llamarlo "nombre de dominio NIS". Cuando un cliente genera peticiones de NIS que llegan a todas las máquinas (broadcast) solicitando información se incluye el nombre de dominio NIS que tiene configurado. De esta forma, varios servidores de dominios distintos situados en la misma red pueden discriminar las peticiones recibidas. Se puede pensar en el nombre de dominio NIS como un identificador de grupos de máquinas que se encuentran relacionados administrativamente de alguna forma.

Algunas organizaciones eligen utilizar su nombre de dominio de Internet como nombre de dominio NIS. Esto no se recomienda ya que puede causar confusión cuando se intentan depurar problemas de red. El nombre de dominio NIS debería ser un nombre único dentro de nuestra red y resulta más útil aún si el nombre elegido puede describir de alguna forma al conjunto de máquinas que representa. Por ejemplo el departamento de arte de la empresa Acme puede utilizar como nombre de dominio "acme-art". En nuestro ejemplo hemos utilizado el nombre `test-domain`.

No obstante algunos sistemas operativos (de forma notable SunOS™) utilizan como nombres de dominio nombres de Internet. Si se poseen máquinas con esta restricción no queda más remedio que *utilizar* los nombres de dominio de Internet como nombres de dominio NIS.

29.9.4.1.2. Requisitos físicos de los servidores

Existen varias cosas que debemos tener en cuenta cuando se selecciona una máquina para actuar como servidor NIS. Una de las características desafortunadas del servicio de pági-

nas amarillas es el alto nivel de dependencia que llegan a tener los clientes respecto del servidor de NIS. Si el cliente no puede contactar con el servidor NIS normalmente la máquina se queda en un estado totalmente inutilizable. La carencia de información de usuarios y grupos provoca que las máquinas se bloqueen. Con esto en mente debemos debemos asegurarnos de escoger un servidor de NIS que no se reinicie de forma habitual o uno que no se utilice para para desarrollar. Si se dispone de una red con poca carga puede resultar aceptable colocar el servidor de NIS en una máquina donde se ejecuten otros servicios pero en todo momento se debe tener presente que si por cualquier motivo el servidor de NIS quedara inutilizable afectaría a *todas* las máquinas de forma negativa.

29.9.4.2. Servidores NIS

Las copias canónicas de toda la información que mantiene el sistema de páginas amarillas se almacenan en una única máquina denominada servidor maestro de NIS. Las bases de datos utilizadas para almacenar la información se denominan mapeos NIS. En FreeBSD estas asociaciones o mapeos se almacenan en el directorio `/var/yp/[nombrededominio]` donde `[nombrededominio]` es el nombre del dominio de NIS que el servidor gestiona. Un único servidor NIS puede gestionar varios dominios al mismo tiempo de forma que resulta posible tener varios directorios como el anterior, uno por cada dominio soportado. Cada dominio posee su conjunto de mapeos independiente y propio.

Los servidores maestro y esclavos manejan todas las peticiones de a través del dæmon `ypserv`. `ypserv` se responsabiliza de recibir peticiones de los clientes NIS. Estas peticiones se traducen a una ruta dentro del servidor. Esta ruta localiza un fichero de base de datos determinado del servidor de NIS, y finalmente `ypserv` se encarga de transmitir la información de dicha base de datos de vuelta al cliente que la solicitó.

29.9.4.2.1. Configuración de un servidor de NIS maestro

La configuración de un servidor de NIS maestro puede resultar relativamente sencilla dependiendo de las necesidades que se tengan. FreeBSD viene preconfigurado por defecto con un servicio NIS. Todo lo que necesitamos es añadir la siguiente línea en `/etc/rc.conf` y FreeBSD se encarga del resto.

1. ```
   nisdomainname="test-domain"
   ```

   Esta línea establece el nombre de dominio NIS como `test-domain`, cuando se realiza la configuración de la red (por ejemplo, después de un reinicio).

2. ```
   nis_server_enable="YES"
   ```

   Esta variable indica a FreeBSD que ejecute los procesos necesarios para actuar como un servidor de NIS la próxima vez que se configure el subsistema de red.

3. ```
   nis_yppasswdd_enable="YES"
   ```
   ```

Esto permite activar el dæmon `rpc.yppasswdd` el cual, como se ha mencionado anteriormente, permite a los usuarios realizar cambios de contraseña desde las máquinas clientes de NIS.

Nota

Dependiendo de la configuración de NIS podemos necesitar añadir algunas entradas más. Consulte la sección sobre servidores NIS que también actúan como clientes, más adelante en el texto, para saber más sobre esto.

Una vez hecho esto todo lo que tenemos que hacer es ejecutar `/etc/netstart` como superusuario. Esta orden realiza los pasos de configuración necesarios utilizando los valores de las variables definidas en `/etc/rc.conf` .

29.9.4.2.2. Inicialización de los mapeos de NIS

Las *asociaciones o mapeos de NIS* no son más que ficheros de base de datos. Estos ficheros se generan a partir de los ficheros de configuración contenidos en el directorio `etc/` excepto para el caso del fichero `etc/master.passwd` . Esto es así por una buena razón ya que no suele ser buena idea propagar las contraseñas de `root` y de otras cuentas de administración a todos los servidores NIS del dominio. servidores NIS del dominio. Así, antes de inicializar los mapeos se debe ejecutar:

```
# cp /etc/master.passwd /var/yp/master.passwd
# cd /var/yp
# vi master.passwd
```

Se deben borrar todas las entradas que hagan referencia a cuentas del sistema (`bin`, `tty`, `kmem`, `games`, etc), junto con cualquier cuenta que no deseemos que se transmita a los clientes NIS (por ejemplo la cuenta de `root` y cualquier otra cuenta con UID 0 (el del superusuario)).

Nota

Asegúrese de que `/var/yp/master.passwd` no se puede leer ni por grupos ni por el resto de usuarios (modo 600). Utilice `chmod` en caso de necesidad.

Una vez hecho esto es hora de inicializar las asociaciones de NIS. FreeBSD incluye un "script" denominado `ypinit` para realizar esta tarea (consulte su página del manual para obtener más información). Recuerde que este "script" se encuentra disponible en la mayoría de los sistemas UNIX®, pero no en todos. En sistemas Digital UNIX/Compaq Tru64 UNIX se denomina `ypsetup`. Debido a que se pretende generar asociaciones para un servidor NIS maestro vamos a ejecutar `ypinit` con la opción `-m`. A modo de ejemplo, suponiendo que todos los pasos comentados anteriormente se han realizado con éxito, ejecute:

```
ellington# ypinit -m test-domain
Server Type: MASTER Domain: test-domain
Creating an YP server will require that you answer a few questions.
Questions will all be asked at the beginning of the procedure.
Do you want this procedure to quit on non-fatal errors? [y/n: n] n
Ok, please remember to go back and redo manually whatever fails.
If you don't, something might not work.
At this point, we have to construct a list of this domains YP ↺
servers.
rod.darktech.org is already known as master server.
Please continue to add any slave servers, one per line. When you are
done with the list, type a <control D>.
master server   :  ellington
next host to add:  coltrane
next host to add:  ^D
The current list of NIS servers looks like this:
ellington
coltrane
Is this correct?  [y/n: y] y

[..salida de la generacion de mapeos..-]

NIS Map update completed.
ellington has been setup as an YP master server without any errors.
```

`ypinit` debería haber creado el fichero `/var/yp/Makefile` a partir de `/var/yp/Makefile.dist` . Una vez creado este archivo presupone que se está usando un entorno NIS con un único servidor utilizando sólamente máquinas FreeBSD. Debido a que `test-domain` posee también un servidor NIS esclavo se debe editar el fichero `var/yp/Makefile` :

```
ellington# vi
             /var/yp/Makefile
```

Se debe comentar la línea que dice:

```
NOPUSH = "True"
```

(si es que no se encuentra ya comentada).

29.9.4.2.3. Configuración de un servidor NIS esclavo

La configuración de un servidor NIS esclavo resulta ser incluso más sencilla que la del maestro. Basta con entrar en el servidor esclavo y editar `/etc/rc.conf` de foma seme-

jante a como se realizó en el apartado anterior. La única diferencia consiste en que ahora debemos utilizar la opción `-s` cuando ejecutemos ejecutemos `ypinit`. A continuación del parámetro `-s` se debe especificar el nombre del servidor maestro de modo que la orden tendría que ser algo parecido a esto:

```
coltrane# ypinit -s ellington test-domain

Server Type: SLAVE Domain: test-domain Master: ellington

Creating an YP server will require that you answer a few questions.
Questions will all be asked at the beginning of the procedure.

Do you want this procedure to quit on non-fatal errors? [y/n: n]  n

Ok, please remember to go back and redo manually whatever fails.
If you don't, something might not work.
There will be no further questions. The remainder of the procedure
should take a few minutes, to copy the databases from ellington.
Transferring netgroup...
ypxfr: Exiting: Map successfully transferred
Transferring netgroup.byuser...
ypxfr: Exiting: Map successfully transferred
Transferring netgroup.byhost...
ypxfr: Exiting: Map successfully transferred
Transferring master.passwd.byuid...
ypxfr: Exiting: Map successfully transferred
Transferring passwd.byuid...
ypxfr: Exiting: Map successfully transferred
Transferring passwd.byname...
ypxfr: Exiting: Map successfully transferred
Transferring group.bygid...
ypxfr: Exiting: Map successfully transferred
Transferring group.byname...
ypxfr: Exiting: Map successfully transferred
Transferring services.byname...
ypxfr: Exiting: Map successfully transferred
Transferring rpc.bynumber...
ypxfr: Exiting: Map successfully transferred
Transferring rpc.byname...
ypxfr: Exiting: Map successfully transferred
Transferring protocols.byname...
ypxfr: Exiting: Map successfully transferred
Transferring master.passwd.byname...
ypxfr: Exiting: Map successfully transferred
Transferring networks.byname...
ypxfr: Exiting: Map successfully transferred
Transferring networks.byaddr...
ypxfr: Exiting: Map successfully transferred
Transferring netid.byname...
ypxfr: Exiting: Map successfully transferred
Transferring hosts.byaddr...
ypxfr: Exiting: Map successfully transferred
```

```
Transferring protocols.bynumber...
ypxfr: Exiting: Map successfully transferred
Transferring ypservers...
ypxfr: Exiting: Map successfully transferred
Transferring hosts.byname...
ypxfr: Exiting: Map successfully transferred

coltrane has been setup as an YP slave server without any errors.
Don't forget to update map ypservers on ellington.
```

En estos momentos debemos tener un nuevo directorio llamado `/var/yp/test-domain` . Las copias de los mapeos del servidor maestro se almacenan en dicho directorio. Es nuestra responsabilidad como administradores asegurar que dichas copias permanecen actualizadas en todo momento. La siguiente entrada en el archivo `/etc/crontab` del servidor esclavo se encarga de realizar esta tarea:

```
20      *       *       *       *       root    /usr/libexec/ypxfr ↺
passwd.byname
21      *       *       *       *       root    /usr/libexec/ypxfr ↺
passwd.byuid
```

Estas dos líneas obligan al servidor esclavo a sincronizar los mapeos con el servidor maestro. Estas entradas no son obligatorias ya que el servidor maestro siempre intenta comunicar cualquier cambio que se produzca en sus asociaciones a todos los servidores esclavos ya que la información de, esclavos, ya que la información de, por ejemplo, contraseñas es de vital importancia para el funcionamiento de los sistemas que dependen del servidor. No obstante es una buena idea obligar a que se realicen estos cambios mediante las entradas anteriores. Esto resulta ser incluso más importante en redes de sobrecargadas donde las actualizaciones de asociaciones pueden algunas veces no llegar a realizarse de forma completa.

A continuación se ejecuta `/etc/netstart` en el servidor esclavo de igual de igual forma que se hizo con el servidor maestro; esto relanza de nuevo el servidor de NIS.

29.9.4.3. Clientes NIS

Un cliente de NIS establece lo que se conoce con el nombre de asociación (bind en inglés) con un servidor NIS NIS determinado utilizando el dæmon `ypbind`. `ypbind` comprueba el dominio por defecto del sistema (especificado mediante `domainname`) y comienza a enviar peticiones RPC a todos los elementos de la red local (broadcast). Estas peticiones especifican el nombre del dominio con el que se quiere establecer la asociación. Si esta petición alcanza una petición alcanza un servidor NIS configurado para servir dicho dominio el servidor responde a la petición e `ypbind` almacenará la dirección de dicho servidor. Si existen varios servidores disponibles (un maestro y varios esclavos, por ejemplo), `ypbind` utilizará la dirección del primero en responder. A partir de este punto el cliente dirigirá el resto de sus peticiones NIS directamente a la dirección IP almacenada. Ocasionalmente `ypbind` envía un “ping” sobre el servidor para comprobar que en efecto se encuentra

funcionando. Si no se recibe contestación al ping dentro de un espacio de tiempo determinado `ypbind` marca el dominio como "sin asociar" y comienza de nuevo a inundar la red con la esperanza de localizar algún otro servidor NIS.

29.9.4.3.1. Configuración de un cliente NIS

La configuración de un cliente FreeBSD de NIS resulta ser una operación extremadamente sencilla.

1. Editar el fichero `/etc/rc.conf` y añadir las siguientes líneas para establecer el nombre de dominio NIS y para que se ejecute `ypbind` al arranque de la red:

```
nisdomainname="test-domain"
nis_client_enable="YES"
```

2. Para importar todas las entradas de contraseñas del servidor de NIS hay que eliminar todas las cuentas de usuario de `/etc/master.passwd` y utilizar `vipw` para añadir la siguiente línea al final de dicho fichero:

```
+:::::::::
```

Nota

Esta línea permite que cualquiera abra una cuenta en local, siempre que dicha cuenta se encuentre definida en las asociaciones de cuentas y contraseñas del servidor NIS. Existen varias formas de configurar los clientes NIS modificando esta línea. Consulte la sección sobre netgroups que se encuentra más adelante en este mismo texto. Si quiere saber más puede consultar el libro de O'Reilly `Managing NFS and NIS`.

Nota

Se debe mantener al menos una cuenta local (por ejemplo, una cuenta que no se importe a través de NIS) en el fichero `/etc/master.passwd` y además dicha cuenta debería ser miembro del grupo `wheel`. Si algo va mal con el procedimiento descrito esta cuenta se puede utilizar para entrar en la máquina cliente de forma remota para posteriormente convertirse en superusuario e intentar solucionar el problema.

3. Para importar todas las entradas de grupo posibles del servidor NIS se debe añadir la siguiente línea al fichero `/etc/group` :

```
+:*::
```

Después de completar estos pasos deberímos ser capaces de ejecutar `ypcat passwd` y ver la asociación de contraseñas que se encuentra en el servidor de NIS

29.9.5. Seguridad en NIS

En general cualquier usuario remoto puede realizar peticiones de RPC a ypserv(8) y recuperar los contenidos de las asociaciones de NIS siempre y cuando el usuario remoto conozca el nombre de dominio de NIS. Para evitar este tipo de transacciones no autorizadas, ypserv(8) soporta una característica denominada “securenets” la cual se puede utilizar para limitar el acceso a un determinado conjunto de máquinas. En el arranque ypserv(8) intenta cargar la información de “securenets” a partir de un fichero denominado `/var/yp/securenets`.

Nota

Esta ruta de fichero varía dependiendo del camino especificado con la opción `-p`. Dicho fichero contiene entradas compuestas de, por un lado, un rango de red y por otro una máscara de red, separados por espacios en blanco. Las líneas que comienzan por “`#`” se consideran comentarios. A continuación se muestra un ejemplo de un fichero “securenet”:

```
# admitir conexiones desde localhost -- obligatorio
127.0.0.1     255.255.255.255
# admitir conexiones desde cualquier host
# on the 192.168.128.0 network
192.168.128.0 255.255.255.0
# admitir conexiones desde cualquier host
# between 10.0.0.0 to 10.0.15.255
# esto incluye las maquinas en el 'testlab'
10.0.0.0      255.255.240.0
```

Si ypserv(8) recibe una petición de una dirección que coincide con alguna de las reglas especificadas en el fichero se procesa la petición. Si no existe ninguna coincidencia la petición se rechaza y se graba un mensaje de aviso. Si el archivo `/var/yp/securnets` no existe `ypserv` acepta conexiones de cualquier máquina.

El programa `ypserv` también posee soporte para el paquete de Wietse Venema denominado tcpwrapper. Este paquete permite utilizar los ficheros de configuración de tcpwrapper para realizar control de acceso en lugar de utilizar `var/yp/securenets` .

Nota

Aunque ambos mecanismos de control de acceso proporcionan un grado de seguridad mayor que no utilizar nada resultan vulnerables a ataques de "falsificación de IPs". El cortafuegos de la red donde se implanta el servicio de NIS debería encargarse de bloquear el tráfico específico de dicho servicio.

Los servidores que utilizan `/var/yp/securenets` pueden no prestar servicio a clientes NIS legítimos cuando se trabaja con implementaciones arcaicas de TCP/IP. Algunas de estas implementaciones ponen a cero todos los bits de máquina cuando se realizan broadcast y/o pueden fallar al especificar la máscara de red en el mismo caso, por citar algunos ejemplos. Mientras que algunos de estos problemas se pueden solucionar variando la configuración del cliente en otros casos podemos vernos obligados a prescindir del cliente en cuestión o a prescindir del fichero `var/yp/securenets`.

Se desaconseja la utilización de `var/yp/securenets` en un sistema con una implementación arcaica de TCP/IP y puede suponer una pérdida de funcionalidad para grandes zonas de la red.

La utilización del paquete tcpwrapper incrementa la latencia del servidor NIS. El retardo adicional introducido puede ser lo suficientemente grande como para causar la expiración de ciertos temporizadores de los programas clientes, especialmente en redes muy cargadas o con servidores de NIS lentos. Si se experimentan estos síntomas en varias máquinas clientes, puede ser conveniente convertir dichas máquinas en servidores NIS esclavos y obligarlas a asociarse con ellas mismas.

29.9.6. Prohibir el acceso a determinados usuarios

En nuestro laboratorio de ejemplo existe una máquina denominada `basie` que actúa sólo como una estación de trabajo para el profesorado. No queremos sacar a esta máquina del dominio NIS pero nos damos cuenta de que el fichero `passwd` que se encuentra en el servidor de NIS posee cuentas tanto para profesores como para alumnos. ?Qué podemos hacer?.

Existe una forma de prohibir el acceso a determinados usuarios sobre una determinada máquina incluso aunque se encuentren dados de alta en la base de datos de NIS. Para realizar esto todo lo que debemos hacer es añadir `-nombredeusuario` al final del fiche-

ro `/etc/master.passwd` en la máquina cliente donde *`nombredeusuario`* es el nombre de usuario del usuario al que queremos prohibirle el acceso. Esto se debe realizar a poder ser mediante `vipw` ya que `vipw` realiza comprobaciones de seguridad sobre los cambios realizados y además se encarga de reconstruir la base de datos de contraseñas cuando se termina la edición. Por ejemplo, si quisiéramos prohibir el acceso al usuario `bill` a la máquina `basie` haríamos:

```
basie# vipw
[añadimos -bill al final y salimos]
vipw: rebuilding the database...
vipw: done

basie# cat /etc/master.passwd

root:[password]:0:0::0:0:The super-user:/root:/bin/csh
toor:[password]:0:0::0:0:The other super-user:/root:/bin/sh
daemon:*:1:1::0:0:Owner of many system processes:/root:/sbin/nologin
operator:*:2:5::0:0:System &:/:/sbin/nologin
bin:*:3:7::0:0:Binaries Commands and Source,,,:/:/sbin/nologin
tty:*:4:65533::0:0:Tty Sandbox:/:/sbin/nologin
kmem:*:5:65533::0:0:KMem Sandbox:/:/sbin/nologin
games:*:7:13::0:0:Games pseudo-user:/usr/games:/sbin/nologin
news:*:8:8::0:0:News Subsystem:/:/sbin/nologin
man:*:9:9::0:0:Mister Man Pages:/usr/share/man:/sbin/nologin
bind:*:53:53::0:0:Bind Sandbox:/:/sbin/nologin
uucp:*:66:66::0:0:UUCP pseudo-user:/var/spool/uucppublic:/usr/
libexec/uucp/uucico
xten:*:67:67::0:0:X-10 daemon:/usr/local/xten:/sbin/nologin
pop:*:68:6::0:0:Post Office Owner:/nonexistent:/sbin/nologin
nobody:*:65534:65534::0:0:Unprivileged user:/nonexistent:/sbin/
nologin
+:::::::::
-bill

basie#
```

29.9.7. Uso de Netgroups

Escrito por Udo Erdelhoff.

El método descrito en la sección anterior funciona razonablemente bien si las reglas especiales se definen para un conjunto pequeño de usuarios y/o máquinas. En dominios admnistrativos grandes puede que se nos *olvide* prohibir el acceso a algún usuario en determinadas máquinas perdiendo de esta forma el principal beneficio de utilizar el servicio de páginas amarillas: *administración centralizada*.

La solución creada por los desarrolladores de NIS se denomina *netgroups*. Su propósito se asemeja al concepto de grupos utilizado por los sistemas UNIX®. La diferencia principal es la carencia de un identificador numérico y la habilidad para definir un “netgroup” que incluye tanto a cuentas de usuario como a otros “netgroups”.

Los netgroups se desarrollaron para gestionar redes grandes y y complejas con cientos de usuarios y máquinas. Por un lado esto es una Cosa Buena si nos encontramos en tal situación pero por otro lado esta complejidad añadida hace que sea casi imposible de explicar a través de ejemplos sencillos. El ejemplo que va a utilizar en lo que queda de sección ilustrará este hecho.

Vamos a suponer que la exitosa introducción del servicio de páginas amarillas en nuestro laboratorio ha llamado la atención de nuestros jefes. Nuestra siguiente tarea consiste en extender el dominio de NIS para que cubra otras máquinas del campus. Las tablas que se muestran a continuación contienen los nombres de los nuevos usuarios y máquinas junto con una breve descripción de ellas.

Nombre del Usuario/usuarios	Descripción
`alpha`, `beta`	Empleados normales del departamento de IT
`charlie`, `delta`	Los nuevos aprendices del departamento de IT
`echo`, `foxtrott`, `golf`, ...	Empleados normales
`able`, `baker`, ...	Los actuales internos

Nombre de la Máquina(s)	Descripción
`guerra`, `muerte`, `hambre`, `peste`	Los servidores más importantes. Sólo los empleados de IT pueden entrar en estas máquinas
`orgullo`, `avaricia`, `envidia`, `ira`, `lujuria`, `pereza`	Servidores de menor importancia. Todos los miembros del departamento de IT pueden entrar en ellos.
`uno`, `dos`, `tres`, `cuatro`, ...	Estaciones de trabajo ordinarias. Sólo los empleados *actuales* pueden utilizar estas máquinas.
`trashcan`	Una máquina muy vieja sin ningún dato dato crítico. Incluso los internos pueden utilizar esta máquina.

Si se trata de implementar estas restricciones a nivel de usuario particular tendríamos que añadir una línea `-usuario` a cada fichero `passwd` del sistema para cada usuario que tuviera prohibido el acceso a dicho sistema. Si nos olvidamos de una sola entrada podrímos tener serios problemas. Puede ser factible realizar esta configuración cuando se instala el servicio pero no obstante el riesgo que corremos al mantener este sistema de restricciones en el día día es muy alto. Después de todo Murphy era un optimista.

La gestión de esta situación mediante netgroups ofrece varias ventajas. Cada usuario no tiene que tratarse de una forma individualizada; basta con añadir un usuario a uno o más

netgroups y posteriormente permitir o prohibir el acceso para todos los usuarios del netgroup. Si se añade una nueva máquina al servicio de NIS simplemente tendremos que definir restricciones de acceso para los netgroups definidos en la red. Si se añade un nuevo usuario bastará con añadir dicho usuario a un netgroup existente. Estos cambios son independientes unos de otros: se acabó la necesidad de aplicar la frase "por cada combinación de usuario y máquina haga esto y esto". Si hemos planificado nuestro servicio de NIS cuidadosamente, sólo tendremos que modificar un fichero de configuración en un determinado servidor para permitir o denegar estos accesos.

El primer paso consiste en la inicialización de la asociación o mapeo del netgroup. La orden de FreeBSD ypinit(8) no crea este mapeo por defecto pero una vez creado será tenido en cuenta por la implementación de NIS. Para crear una asociación vacía simplemente escriba:

```
ellington# vi /var/yp/netgroup
```

y comienze a añadir contenido. En nuestro ejemplo necesitamos al menos cuatro netgroups: empleados de IT, aprendices de IT, empleados normales e internos.

```
IT_EMP  (,alpha,test-domain)    (,beta,test-domain)
IT_APP  (,charlie,test-domain)  (,delta,test-domain)
USERS   (,echo,test-domain)     (,foxtrott,test-domain) \
        (,golf,test-domain)
INTERNS (,able,test-domain)     (,baker,test-domain)
```

`IT_EMP`, `IT_APP` etc. son los nombres de los netgroups. Cada conjunto delimitado por paréntesis define una o más cuentas como pertenecientes al netgroup. Existen tres campos definidos dentro de dichos de dichos grupos:

1. El nombre de las máquinas donde los siguientes items son aplicables. Si no se especifica un nombre de máquina la entrada se aplica a todas las máquinas existentes. Si se especifica una máquina determinada entraremos en un mundo lleno de horrores y confusiones así que mejor no hacerlo.

2. El nombre de la cuenta que pertenece al netgroup que estamos definiendo.

3. El dominio NIS donde reside la cuenta. Se pueden importar cuentas de otros dominios NIS (en caso de que usted pertenezca al extraño grupo de personas que gestionan más de un dominio NIS.

Cada uno de estos campos puede contener comodines. Consulte netgroup(5) para obtener más detalles.

Nota

No se deben utilizar nombres de los netgroups superiores a ocho caracteres, especialmente si las máquinas de nuestro dominio utilizan sistemas operativos variados. Los nombres son sensibles a las mayúsculas/minúsculas: se recomienda utilizar nombres en mayúsculas para distinguirlos de los usuarios o máquinas.

Algunos clientes de NIS (distintos de los clientes FreeBSD) no pueden gestionar netgroups con un gran número de entradas. Por ejemplo algunas versiones antiguas de SunOS™ comienzan a presentar problemas si un netgroup contiene más de *quince entradas*. Se puede solventar este problema creando varios sub-netgroups de como mucho quince usuarios y finalmente creando el verdadero netgroup compuesto por los sub-netgroups:

```
BIGGRP1  (,joe1,domain)  (,joe2,domain)  ↺
(,joe3,domain) [...-]
BIGGRP2  (,joe16,domain)  (,joe17,domain) [...-]
BIGGRP3  (,joe31,domain)  (,joe32,domain)
BIGGROUP  BIGGRP1 BIGGRP2 BIGGRP3
```

Se puede repetir este proceso si se tienen que definir más de 225 usuarios dentro de un único netgroup.

La activación y distribución de nuestro nuevo mapeo NIS resulta sencillo:

```
ellington# cd /var/yp
ellington# make
```

Esto genera las tres asociaciones NIS `netgroup`, `netgroup.byhost` y `netgroup.byuser`. Utilice ypcat(1) para comprobar si el nuevo mapeo NIS se encuentra disponible:

```
ellington% ypcat -k netgroup
ellington% ypcat -k netgroup.byhost
ellington% ypcat -k netgroup.byuser
```

La salida de la primera orden debería parecerse a los contenidos del fichero `/var/yp/netgroup`. La segunda orden no mostrará ninguna salida salvo que se hayan especificado netgroups específicos para máquinas. La tercera orden se puede utilizar para obtener la lista de los netgroups a los que petenece un determinado usuario.

La configuración del cliente es bastante simple. Para configurar el servidor `guerra` se debe ejecutar vipw(8) y sustituír la línea

```
+::::::::::
```

por

```
+@IT_EMP:::::::::
```

Ahora sólo se importan los datos para los usuarios que se encuentren definidos en el netgroup `IT_EMP`; dichos datos se importan en la base de datos de contraseñas de `guerra` y sólo se permite el acceso a estos usuarios.

Por desgraciaesta información también se aplica a la función ~ del shell y a todas las rutinas que traducen nombres de usuarios con los correspondientes identificadores numéricos de usuario (uid). En otras palabras, la orden `cd ~` no va a funcionar, `ls -l` muestra el identificador numérico en lugar del nombre de usuario y `find . -user joe -print` produce un error de No such user ("Usuario desconocido"). Para solucionar esto debemos importar todas las entradas de usuario en la máquina cliente NIS pero sin permitirles el acceso.

Esto se puede realizar añadiendo otra línea al fichero `/etc/master.passwd` . Esta línea debería contener lo siguiente:

`+:::::::::/sbin/nologin` , lo que significa que se "importen todas las entradas pero que se reemplace el shell por `/sbin/nologin` ". Se puede sustituir cualquier campo de la entrada de contraseñas especificando un valor concreto para dicho campo en el fichero local local `/etc/master.passwd` .

Aviso

Asegúrese de que la línea `+:::::::::/sbin/nologin` se sitúa después de `+@IT_EMP:::::::::` . Si esto no se cumple todas las cuentas de usuario importadas del servidor NIS poseerán `/sbin/nologin` como shell de acceso.

Después de este cambio si se introduce un nuevo empleado en el departamento de IT basta con cambiar una asociación de NIS. Se podría aplicar una aproximación similar para los servidores menos importantes sustituyendo la cadena `+:::::::::` en las versiones locales del fichero `/etc/master.passwd` por algo parecido a esto:

```
+@IT_EMP:::::::::
+@IT_APP:::::::::
+:::::::::/sbin/nologin
```

Las líneas correspondientes para las estaciones de trabajo normales podrían ser:

```
+@IT_EMP:::::::::
+@USERS:::::::::
+:::::::::/sbin/nologin
```

Y parece que todos nuestros problemas de gestión han desaparecido; hasta que unas semanas más tarde se produce un cambio en la política de gestión: el depatamento de IT

comienza a alquilar interinos. Los interinos de IT pueden utilizar las estaciones de trabajo normales y los servidores menos importantes y los aprendices de IT pueden acceder a los servidores principales. No nos queda más remedio que añadir un nuevo netgroup denominado `IT_INTERN` y añadir los nuevos interinos de IT a este nuevo grupo y comenzar a cambiar la la configuración de cada máquina, una por una... Como dice el antiguo proverbio: "Errores en la planificación centralizada conllevan grandes quebraderos de de cabeza globales".

La habilidad que posee NIS para crear netgroups a partir de otros netgroups se puede utilizar para evitar la situación anterior. Una posibilidad consiste en la creación de netgroups basados en roles. Por ejemplo, se podría crear un netgroup denominado `BIGSRV` para definir las restricciones de acceso para los servidores importantes, otro grupo denominado `USERBOX` para las estaciones de trabajo... Cada uno de estos netgroups podría contener los netgroups que pueden acceder a dichas máquinas. Las nuevas entradas para nuestro mapeo NIS de netgroups sería algo así:

```
BIGSRV    IT_EMP  IT_APP
SMALLSRV  IT_EMP  IT_APP  ITINTERN
USERBOX   IT_EMP  ITINTERN USERS
```

Este método de definir restricciones de acceso funciona razonablemente bien si podemos definir grupos de máquinas que posean restricciones semejantes. Por desgracia lo normal es que este caso resulta ser una excepción más que una regla. En la mayor parte de las ocasiones necesitaremos definir restricciones de acceso en función de máquinas individuales.

Las definiciones de netgroups específicos para determinadas máquinas constituyen el segundo método que se puede utilizar para gestionar el cambio de política del ejemplo que estamos explicando. En nuestro caso el fichero `/etc/master.passwd` de cada máquina contiene dos líneas que comienzan por "+". La primera de ellas añade un netgroup con las cuentas que pueden acceder a esa máquina, mientras que la segunda añade el resto de cuentas con shell el resto de cuentas con shell `/sbin/nologin` . Es una buena idea utilizar la versión "todo en mayúsculas" del nombre de máquina como el nombre del netgroup. En otras palabras, las líneas deberían ser como la siguiente:

```
+@BOXNAME:::::::::
+:::::::::/sbin/nologin
```

Una vez que hemos completado esta tarea para todas las máquinas nunca más resultará necesario modificar las versiones locales de `/etc/master.passwd` . Los futuros cambios se pueden gestionar simplemente modificando el mapeo o asociación de NIS. A continuación se muestra un mapeo de netgroups para el escenario que se está explicando junto con algunas buenas ideas:

```
# Definimos antes que nada los grupos de usuarios
IT_EMP    (,alpha,test-domain)    (,beta,test-domain)
IT_APP    (,charlie,test-domain)  (,delta,test-domain)
DEPT1     (,echo,test-domain)     (,foxtrott,test-domain)
```

```
DEPT2     (,golf,test-domain)     (,hotel,test-domain)
DEPT3     (,india,test-domain)    (,juliet,test-domain)
ITINTERN  (,kilo,test-domain)     (,lima,test-domain)
D_INTERNS (,able,test-domain)     (,baker,test-domain)
#
# Ahora definimos unos pocos grupos basados en roles
USERS     DEPT1   DEPT2     DEPT3
BIGSRV    IT_EMP  IT_APP
SMALLSRV  IT_EMP  IT_APP    ITINTERN
USERBOX   IT_EMP  ITINTERN  USERS
#
# Y un grupo para tareas especiales
# Permitimos a echo y golf acceso a nuestra maquina-anti-virus
SECURITY  IT_EMP  (,echo,test-domain)  (,golf,test-domain)
#
# netgroups basados en maquinas
# Nuestros servidores principales
GUERRA   BIGSRV
HAMBRE   BIGSRV
# El usuario india necesita acceso a este servidor
PESTE  BIGSRV  (,india,test-domain)
#
# Este es realmente importante y necesita mas restricciones de
# acceso
MUERTE     IT_EMP
#
# La maquina-anti-virus que mencionabamos mas arriba
ONE       SECURITY
#
# Restringimos una maquina a un solo usuario
TWO       (,hotel,test-domain)
# [...otros grupos]
```

Si estamos utilizando algun tipo de base de datos para gestionar cuentas de usuario debemos ser capaces de crear la primera parte del mapeo utilizando las herramientas proporcionadas por dicho sistema de base de datos. De este modo los nuevos usuarios tendrán automáticamente derechos de acceso sobre las máquinas.

Una última, por precaución: puede no ser siempre aconsejable utilizar netgroups basados en máquinas. Si se están desplegando, por ejemplo, un par de docenas o incluso cientos de máquinas idénticas en laboratorios de estudiantes, es mejor utilizar netgroups basados en roles en lugar de netgroups basados en máquinas individuales para mantener el tamaño de la asociación NIS dentro de unos límites razonables.

29.9.8. Conceptos importantes a tener muy en cuenta

Todavía quedan un par de cosas que tendremos que hacer de forma distinta a lo comentado hasta ahora en caso de encontrarnos dentro de un entorno de NIS.

- Cada vez que deseemos añadir un usuario a nuestro laboratorio debemos añadirlo al servidor NIS maestro *únicamente* y es tarea fundamental del administrador *acordarse de*

reconstruir los mapeos NIS. Si nos olvidamos de esto el nuevo usuario será incapaz de acceder a ninguna máquina excepto al servidor NIS. Por ejemplo, si necesitáramos añadir el nuevo usuario `jsmith` al laboratorio tendríamos que ejecutar lo siguiente:

```
# pw useradd jsmith
# cd /var/yp
# make test-domain
```

Se puede ejecutar también `adduser jsmith` en lugar de `pw useradd jsmith`.

- *No introduzca las cuentas de administración dentro de los mapeos de NIS*. No es buena idea propagar cuentas y contraseñas de administración a máquinas donde residen usuarios que normalmente no deberían poder acceder a dichas cuentas.

- *Mantenga el servidor maestro y esclavo de NIS seguros y minimize el tiempo de interrupción del servicio*. Si alguien fuera capaz de comprometer la integridad de estas máquinas o de apagarlas los usuarios del dominio no podrían acceder a sus cuentas en ningún sistema.

  Esto es la debilidad principal de cualquier sistema de administración centralizada. Si no se protegen los servidores NIS tendremos frente a nosotros a una horda de usuarios bastante enfadados.

29.9.9. Compatibilidad con NIS v1

El servicio ypserv de FreeBSD puede servir también a clientes NIS v1. La implementación de NIS de FreeBSD sólo utiliza el protocolo NIS v2 aunque otras implementaciones incluyen soporte para el protocolo v1 por razones de compatibilidad con sistemas antiguos. Los dæmones ypbind suministrados con estos sistemas tratan de establecer una asociación con un servidor NIS versión 1 aunque puede que nunca la lleguen a utilizar (y pueden continuar realizando búsquedas mediante broadcast incluso cuando reciben una respuesta de un servidor versión 2). Tenga muy presente que mientras se soportan las llamadas normales de clientes v1, la versión de ypserv actualmente suministrada no gestiona peticiones de transferencia de mapeos a través de la versión 1; en consecuencia no se puede utilizar como maestro o esclavo junto con servidores de NIS antiguos que sólo soporten el protocolo v1. Por suerte quedan muy pocos servidores de este estilo en funcionamiento hoy en día.

29.9.10. Servidores NIS que actúan también como clientes NIS

Se debe tener cuidado cuando se ejecuta ypserv en un entorno multi-servidor donde las máquinas servidoras actúan también como máquinas clientes de NIS. Normalmente es una buena idea obligar a los servidores a que se asocien con ellos mismos mejor que permitirles emitir peticiones de asociación en broadcast, lo que posiblemente terminará con los servidores asociados entre sí. Se pueden producir extraños fallos si un servidor del que dependen otros deja de funcionar. Puede darse que los contadores de todos los clientes expiren e intenten asociarse a otro servidor, pero el retardo puede ser considerable y los

fallos todavía podrín persistir debido a que los servidores se asocian contínuamente los unos a los otros.

Se puede obligar a una máquina a asociarse con un servidor en particular ejecutando `ypbind` con la opción `-S`. Si no se quiere ejecutar esto manualmente cada vez que se reinice el servidor NIS se puede puede añadir la siguiente línea al fichero `/etc/rc.conf` :

```
nis_client_enable="YES" # ejecuta tambien el soft cliente
nis_client_flags="-S NIS domain ,server"
```

Consulte ypbind(8) para obtener más información.

29.9.11. Formatos de contraseñas

Uno de los problemas más comunes que se encuentra la gente a la hora de implantar un servicio de NIS es la compatibilidad del formato de las contraseñas. Si nuestro servidor de NIS utiliza contraseñas cifradas mediante DES sólo podrá aceptar clientes que utilicen DES. Por ejemplo, si poseemos clientes de NIS Solaris™ en nuestra red casi seguro que necesitaremos utilizar contraseñas cifradas mediante DES.

Para comprobar qué formato utilizan los servidores y los clientes, se puede mirar en `/etc/login.conf` . Si la máquina se configura para utilizar cifrado de contraseñas mediante DES, entonces la clase por defecto debe poseer una entrada como la siguiente:

```
default:\
 :passwd_format=des:\
 :copyright=/etc/COPYRIGHT:\
 [Se han omitido otras entradas]
```

Otros posibles valores para característica de `passwd_format` pueden ser `blf` y `md5` (para utilizar los algoritmos Blowfish y MD5 respectivamente).

Se debe reconstruir la base de datos de acceso siempre que se modifique el fichero `/etc/login.conf` mediante la ejecución de la siguiente orden como `root`:

```
# cap_mkdb /etc/login.conf
```

Nota

El formato de las contraseñas que ya se encuentran definidas en `/etc/master.passwd` no se actualizará hasta que el usuario cambie su contraseña, después de que se haya reconstruido la base de datos de tipos de acceso.

A continuación para asegurarse de que las contraseñas se cifran con el formato seleccionado también debemos comprobar que la variable `crypt_default` dentro del fichero `/etc/auth.conf` da preferencia al formato de contraseña elegido. Por ejemplo cuando se utilizan contraseñas cifradas con DES la entrada debe ser:

```
crypt_default = des blf md5
```

Si se realizan los pasos anteriores en cada una de las máquinas clientes y servidoras de nuestro entorno NIS podemos asegurar que todas utilizan el mismo formato de contraseña dentro de nuestra red. Si se presentan problemas de validación con determinados usuarios en una determinada máquina cliente se puede empezar a investigar sobre el asunto. Recuerde: si se quiere desplegar un servicio de páginas amarillas sobre un entorno de red heterogéneo probablemente se deba utilizar DES en todos los sistemas puesto que DES es el mínimo común denominador.

29.10. DHCP

Escrito por Greg Sutter.

29.10.1. ?Qué es DHCP?

DHCP, el Protocolo de Configuración Dinamica de Máquinas (“Dynamic Host Configuration Protocol”), especifica un método para configurar dinámicamente los parámetros de red necesarios para que un sistema pueda comunicarse efectivamente. FreeBSD utiliza la implementación de DHCP proporcionada por el Internet Software Consortium (ISC) de tal forma que toda la información relativa a la configuración de DHCP se basa en la distribución proporcionada por el ISC.

29.10.2. Contenido de esta seccións

Esta sección describe tanto los componentes de la parte servidora como los componentes de la parte cliente del sistema DHCP del ISC. El programa cliente, denominado forma parte por defecto de los sistemas FreeBSD y el programa servidor se puede instalar a partir del “port” net/isc-dhcp3-server. Las principales fuentes de información son las páginas de manual dhclient(8), dhcp-options(5) y dhclient.conf(5) junto con las referencias que se muestran a continuación en esta misma sección.

29.10.3. Cómo funciona

Cuando el cliente de DHCP, `dhclient`, se ejecuta en una máquina cliente, valga la redundancia, comienza a enviar peticiones “broadcast” solicitando información de configuración. Por defecto estas peticiones se realizan contra el puerto UDP 68. El servidor responde a través del puerto UDP 67 proporcionando al cliente una dirección IP junto con otros

parámetros relevantes para el correcto funcionamiento del sistema en la red, tales como la máscara de red, el " router" por defecto y los servidores de DNS. Toda esta información se "presta" y es válida sólo durante un determinado período de tiempo (configurado por el administrador del servidor de DHCP). De esta forma direcciones IP asignadas a clientes que ya no se encuentran conectados a la red pueden ser reutilizadas al pasar determinado periodo de tiempo.

Los clientes de DHCP pueden obtener una gran cantidad de información del servidor. Se puede encontrar una lista completa en dhcp-options(5).

29.10.4. Integración dentro de los sistemas FreeBSD

FreeBSD se integra totalmente con el cliente DHCP del ISC, `dhclient`. Este soporte se proporciona tanto en el proceso de instalación como en la instalación por defecto del sistema base obviando la necesidad de poseer un conocimiento detallado de aspectos relacionados con la configuración de redes siempre y cuando se disponga de servicio de DHCP en la red dada. `dhclient` se incluye en todas las distribuciones de FreeBSD desde la versión 3.2.

sysinstall soporta DHCP. Cuando se configura la interfaz de red la primera pregunta es: " ? Quiere intentar configurar el interfaz mediante DHCP?". Si se responde afirmativamente se ejecutará `dhclient` y si tiene éxito se procede con los siguientes pasos de configuración rellenandose automáticamente las variables de arranque necesarias para completar la configuración de la red.

Existen dos cosas que se deben realizar de tal forma que nuestro sistema utilice la configuración de red mediante DHCP al arrancar:

- Asegurarse de que el dispositivo `bpf` se encuentra compilado en el kernel. Para ello basta añadir `device bpf` (`pseudo-device bpf` en los sistemas FreeBSD 4.X) al fichero de configuración del kernel y recompilarlo e instalarlo. Para más información sobre la construcción de núcleos consulte Capítulo 8, *Configuración del kernel de FreeBSD*.

  El dispositivo `bpf` se encuentra activado por defecto dentro del fichero de configuración del núcleo (`GENERIC` que encontrará en su sistema FreeBSD de forma que si no se está utilizando un fichero de configuración del núcleo específico (hecho a medida y/o por usted) no es necesario crear uno nuevo y se puede utilizar directamente `GENERIC`.

Nota

Para aquellas personas especialmente preocupadas por la seguridad debemos advertir de que el dispositivo `bpf` es el dispositivo que las aplicaciones de captura de paquetes utilizan para acceder a los mismos (aunque dichas aplicaciones deben ser ejecutadas como `root`). DHCP *requiere* la presencia de `bpf` pero si la seguri-

dad del sistema es más importante que la configuración automática de la red no se recomienda instalar `bpf` en el núcleo.

- Editar el fichero `/etc/rc.conf` para para incluir lo siguiente:

```
ifconfig_fxp0="DHCP"
```

Nota

Asegúrese de sustituir `fxp0` con el nombre de interfaz que desea que se configure dinámicamente, como se describe en Sección 11.1, “Configuración de Tarjetas de Red”.

Si se utiliza una ubicación distinta para `dhclient` o si se desea añadir opciones adicionales a `dhclient` se puede incluir, adaptándolo a las condiciones particulares de cada usuario, lo siguiente:

```
dhcp_program="/sbin/dhclient"
dhcp_flags=""
```

El servidor de DHCP (dhcpd) forma parte del “port” net/isc-dhcp3-server. Este “ port” también contiene la documentación de ISC DHCP.

29.10.5. Ficheros

- `/etc/dhclient.conf`

  `dhclient` necesita un fichero de configuración denominado `/etc/dhclient.conf` . Normalmente este fichero sólo contiene comentarios de forma que las opciones que se definen por defecto son razonablemente inocuas. Este fichero de configuración se describe en la página de manual de dhclient.conf(5).

- `/sbin/dhclient`

  `dhclient` se encuentra enlazado de forma estática y reside en `/sbin`. La página de manual de dhclient(8) proporciona más información sobre la orden `dhclient`.

- `/sbin/dhclient-script`

  `dhclient-script` es el “ script” de configuración del cliente de DHCP específico de FreeBSD. Tiene todos los detalles en dhclient-script(8) pero no necesita hacer ninguna modificación en él para que todo funcione correctamente.

- `/var/db/dhclient.leases`

  El cliente de DHCP mantiene una base de datos de préstamos en este fichero que se escribe de forma semejante a un "log". En dhclient.leases(5) puede consultar una explicación ligeramente más detallada.

29.10.6. Lecturas recomendadas

El protocolo DHCP se describe completamente en RFC 2131. También tiene más información en dhcp.org.

29.10.7. Instalación y configuración de un servidor de DHCP

29.10.7.1. Qué temas se tratan en esta sección

Esta sección proporciona información sobre cómo configurar un sistema FreeBSD de forma que actúe como un servidor de DHCP utilizando la implementación proporcionada por el Internet Software Consortium (ISC).

La parte servidora del paquete proporcionado por el ISC no se instala por defecto en los sistemas FreeBSD pero se puede intalar como "port" desde net/isc-dhcp3-server. Consulte Capítulo 4, *Instalación de aplicaciones: «packages» y ports* si necesita saber más sobre la Colección de "ports".

29.10.7.2. Instalación del servidor DHCP

Para configurar un sistema FreeBSD como servidor de DHCP debe asegurarse de que el dispositivo bpf(4) está compilado dentro del núcleo. Para ello basta añadir `device bpf` (`pseudo-device bpf` en FreeBSD 4.X) al fichero de configuración del núcleo y reconstruir el mismo. Si necesita saber más sobre el proceso de compilar e instalar el núcleo consulte Capítulo 8, *Configuración del kernel de FreeBSD*.

El dispositivo `bpf` ya se encuentra activado en el fichero de configuración `GENERIC` del núcleo que se facilita con FreeBSD de tal forma que no resulta imprescindible crear un núcleo a medida para ejecutar DHCP.

Nota

Para aquellas personas especialmente preocupadas por la seguridad debemos advertir de que el dispositivo `bpf` es el dispositivo que las aplicaciones de captura de paquetes utilizan para acceder a los mismos (aunque dichas aplicaciones deben ser ejecutadas como `root`). DHCP *requiere* la presencia de `bpf` pero si la seguridad del sistema

es más importante que la configuración automática de la red no se recomienda instalar `bpf` en el núcleo.

El siguiente paso consiste en editar el fichero de ejemplo `dhcpd.conf` que se crea al instalar el "port" net/isc-dhcp3-server. Por defecto el fichero se llama `/usr/local/etc/dhcpd.conf.sample`, así que se debe copiar este fichero a `/usr/local/etc/dhcpd.conf` y a continuación realizar todos los cambios sobre este último.

29.10.7.3. Configuración del servidor de DHCP

El fichero `dhcpd.conf` se compone de un conjunto de declaraciones que hacen referencia a máquinas y a subredes. Esto se entenderá mejor mediante el siguiente ejemplo:

```
option domain-name "ejemplo.com";❶
option domain-name-servers 192.168.4.100;❷
option subnet-mask 255.255.255.0;❸

default-lease-time 3600;❹
max-lease-time 86400;❺
ddns-update-style none;❻

subnet 192.168.4.0 netmask 255.255.255.0 {
  range 192.168.4.129 192.168.4.254;❼
  option routers 192.168.4.1;❽
}

host mailhost {
  hardware ethernet 02:03:04:05:06:07;❾
  fixed-address mailhost.ejemplo.com;❿
}
```

❶ Esta opción especifica el dominio que se proporciona a los clientes y que dichos clientes utilizan como dominio de búsqueda por defecto. Consulte resolv.conf(5) si necesita más información sobre qué significa el dominio de búsqueda.

❷ Esta opción especifica la lista de servidores de DNS (seperados por comas) que deben utilizar los clientes.

❸ La máscara de red que se proporciona a los clientes.

❹ Un cliente puede solicitar un determinado tiempo de vida para el préstamo. En caso contrario el servidor asigna un tiempo de vida por defecto mediante este valor (expresado en segundos).

❺ Este es el máximo tiempo que el servidor puede utilizar para realizar préstamos a los clientes. Si un cliente solicita un tiempo mayor como máximo se responderá con el valor aquí configurado, ignorándose la petición de tiempo del cliente.

❻ Esta opción especifica si el servidor de DHCP debe intentar actualizar el servidor de DNS cuando se acepta o se libera un préstamo. En la implementación proporcionada por el ISC esta opción es *obligatoria*.

❼ Esto indica qué direcciones IP se pueden utilizar para ser prestadas a los clientes que las soliciten. Las direcciones IP pertenecientes a este rango, incluyendo los extremos, se pueden entregar a los clientes.
❽ Declara cúal es la pasarela por defecto que se proporcionará a los clientes.
❾ Especifica la dirección MAC de una máquina, de tal forma que el servidor de DHCP pueda identificar a la máquina cuando realice una petición.
❿ Especifica que la máquina cliente debería obtener siempre la misma dirección IP. Recuerde que se puede utilizar un nombre de máquina para esto ya que el servidor de DHCP resolverá el nombre por sí mismo antes de devolver la información al cliente.

Una vez que se ha acabado de configurar el fichero `dhcpd.conf` se puede proceder con la ejecución del servidor mediante la siguiente orden:

```
# /usr/local/etc/rc.d/isc-dhcpd.sh start
```

Si posteriormente se necesitan realizar cambios en la configuración anterior tenga en cuenta que el envío de la señál `SIGHUP` a la aplicación dhcpd *no* provoca que se lea de nuevo la configuración como suele ocurrir en la mayoría de los dæmones. Tendrá que enviar la señal `SIGTERM` para parar el proceso y posteriormente relanzar el dæmon utilizando la orden anterior.

29.10.7.4. Ficheros

- `/usr/local/sbin/dhcpd`

  dhcpd se encuentra enlazado de forma estática y reside en el directorio `/usr/local/sbin`. La página de manual dhcpd(8) que se instala con el "port" le proporcionará más información sobre dhcpd.

- `/usr/local/etc/dhcpd.conf`

  dhcpd necesita un fichero de configuración, `/usr/local/etc/dhcpd.conf` . Este fichero contiene toda la información relevante que se quiere proporcionar a los clientes que la soliciten, junto con información relacionada con el funcionamiento del servidor. Este fichero de configuración se describe en la página del manual dhcpd.conf(5) que instala el " port".

- `/var/db/dhcpd.leases`

  El servidor de DHCP mantiene una base de datos de préstamos o alquileres dentro de este fichero, que presenta un formato de fichero de "log". La página del manual dhcpd.leases(5) que se instala con el "port" proporciona una descripción ligeramente más larga.

- `/usr/local/sbin/dhcrelay`

  dhcrelay se utiliza en entornos de red avanzados donde un servidor de DHCP reenvía una petición de un cliente hacia otro servidor de DHCP que se encuentra localizado en

otra subred. Si se necesita esta funcionalidad se debe instalar el "port" net/isc-dhcp3-server. La página del manual dhcrelay(8) proporcionada por el "port" contiene más detalles sobre esto.

29.11. DNS

Enviado por Chern Lee.

29.11.1. Resumen

FreeBSD utiliza por defecto una versión de BIND (Berkeley Internet Name Domain) que proporciona la implementación más común del protocolo de DNS. DNS es el protocolo a través del cual los nombres de máquinas se asocian con direcciones IP y viceversa. Por ejemplo una consulta preguntando por `www.FreeBSD.org` recibe una respuesta con la dirección IP del servidor web del Proyecto FreeBSD, mientras que una pregunta sobre `ftp.FreeBSD.org` recibe como respuesta la dirección IP correspondiente al servidor de FTP. El proceso inverso sucede de una forma similar. Una pregunta relativa a una determinada dirección IP se resuelve al nombre de la máquina que la posee. No se necesita ejecutar un servidor de DNS para poder realizar consultas y búsquedas de DNS.

El DNS se coordina de forma distribuida a través de Internet utilizando un sistema en cierta forma complejo de servidores de nombres raíz autorizados y mediante otros servidores de nombres de menor escala que se encargan de replicar la información de dominios individuales con el objetivo de mejorar los tiempos de respuesta de búsquedas reiteradas de la misma información.

Este documento hace referencia a la versión estable BIND 8.X. BIND 9.X se puede instalar a través del "port" net/bind9.

El protocolo de DNS se encuentra definido en la RFC1034 y la RFC1035.

El Internet Software Consortium (www.isc.org) se encarga de de mantener el software de BIND.

29.11.2. Terminología

Para comprender este documento se deben definir los siguientes términos:

Término	Definición
DNS directo (Forward DNS)	Asociación de nombres de máquinas con direcciones IP
Origen	Se refiere al dominio cubierto por un determinado fichero de zona

Término	Definición
named, BIND, servidor de nombres (name server)	Nombres típicos que hacen referencia al paquete servidor de nombres de BIND de FreeBSD
Resolver	Un proceso del sistema que utilizan las aplicaciones para hacer preguntas al servidor de nombres.
DNS inverso (Reverse DNS)	Lo contrario de lo que realiza el DNS directo; asocia direcciones IP con nombres de máquinas
Zona Raíz	El comienzo de la jerarquía de zonas de Internet. Todas las zonas surgen a partir de una zona raíz de forma similar a como todos los directorios de un sistema de ficheros se encuentran a partir de un directorio raíz inicial.
Zona	Un dominio individual, subdominio o porción del DNS que se encuentra administrado por la misma autoridad.

Ejemplos de zonas:

- `.` es la zona raíz
- `org.` es una zona localizada bajo la zona raíz
- `ejemplo.org` es una zona localizada bajo la zona `org.`
- `foo.ejemplo.org.` es un subdominio o una zona ubicada bajo la zona `ejemplo.org.`
- `1.2.3.in-addr.arpa` es una zona que referencia a a todas las direcciones IP que se encuentran dentro del espacio de direcciones de 3.2.1.*.

Como se puede observar la parte más específica de una máquina aparece más a la izquierda. Por ejemplo `ejemplo.org` es más específico que `org.` y del mismo modo `org.` es más específico que la zona raíz. El formato de cada parte del nombre de la máquina es muy similar al formato de un sistema de ficheros: el directorio `/dev` se encuentra dentro del directorio raíz, y así sucesivamente.

29.11.3. Razones para ejecutar un servidor de nombres

Los servidores de nombres normalmente son de dos tipos: autoritarios y de cache.

Se necesita un servidor de nombres autoritario cuando:

- uno quiere proporcionar información de DNS al resto del mundo respondiendo con información autoritaria a las consultas recibidas.
- un dominio, por ejemplo `ejemplo.org` , está registrado y se necesita añadir nombres de máquinas bajo dicho dominio.
- un bloque de direcciones IP necesita entradas de DNS inversas (de IP a nombre de máquina).
- un servidor de nombres de "backup", llamado esclavo, debe responder a consultas cuando el servidor primario se encuentre caído o inaccesible.

Se necesita un servidor caché cuando:

- un servidor de DNS local puede responder más rápidamente de lo que se haría si se tuviera que preguntar al servidor de nombres externo.
- se desea reducir el tráfico global de red (se ha llegado a comprobar que el tráfico de DNS supone un 5% o más del total del tráfico que circula por Internet).

Cuando se pregunta por `www.FreeBSD.org` el " resolver" normalmente pregunta al servidor de nombres del ISP de nivel superior y se encarga de recibir la respuesta. Si se utiliza un servidor de DNS caché local la pregunta sólo se dirige una única vez hacia el exterior. Dicha pregunta la realiza el servidor caché. Posteriores consultas sobre el mismo nombres son respondidas directamente por este servidor.

29.11.4. Cómo funciona

En FreeBSD el dæmon de BIND se denomina named por razones obvias.

Fichero	Descripción
named	El dæmon de BIND
`ndc`	El programa de control del dæmon
`/etc/namedb`	El directorio donde se almacena la información de las zonas de BIND
`/etc/namedb/named.conf`	El archivo de configuración del dæmon

Los ficheros de zonas se encuentran normalmente bajo el directorio `/etc/namedb` y contienen la información que proporciona el servidor de nombres al resto de máquinas de Internet.

29.11.5. Ejecución de BIND

Debido a que BIND se instala por defecto la configuración resulta ser bastante sencilla.

Para asegurarnos de que el dæmon se ejecuta al inicio del sistema se deben añadir las siguientes modificaciones en `/etc/rc.conf` :

```
named_enable="YES"
```

Para arrancar el dæmon de forma manual (una vez configurado)

```
# ndc start
```

29.11.6. Ficheros de configuración

29.11.6.1. Uso de `make-localhost`

Asegúrese de hacer los siguiente

```
# cd /etc/namedb
# sh make-localhost
```

para que se cree el archivo de zona inversa `/etc/namedb/localhost.rev` de forma apropiada.

29.11.6.2. `/etc/namedb/named.conf`

```
// $FreeBSD$
//
// Consulte la pÃigina man de named(8) para mÃis detalles.  tiene
// alguna vez la necesidad de configurar un servidor primario
// asegÃºree de que entiende a la perfecciÃ³n los detalles peliagudos
// del funcionamiento del DNS.  Si hay errores, incluso triviales,
// puede sufrir pÃ©rdidas de conectividad ogenerar cantidades ↺
ingentes
// de trÃifico inÃºtil hacia o desde Interne

options {
        directory "/etc/namedb";

// AdemÃis de con la lÃiusula "forwarders" puedeobligar a su servidor
// de nombres a que nunca lance bÃºsquedas por su cuenta sino que
// se las pida a sus "forwarders". Esto se hace del siguiente modo:
//
//      forward only;

// Si su proveedor de acceso tiene a su alcance un servidor DNS
// escriba aquÃ su direcciÃ³n IP y descomente la lÃneaPodrÃi usar
// su cachÃ© y por lo tanto reducir el trÃifico DNS de Internet.
//

/*
        forwarders {
                127.0.0.1;
        };
```

```
*/
```

Tal y como se dice en los comentarios del ejemplo para beneficiarnos de la caché se puede activar `forwarders`. En circunstancias normales un servidor de nombres busca de forma recursiva a través de Internet tratando de localizar un servidor de nombres que sea capaz de responder una determinada pregunta. Si se activa esta opción por defecto se pasa a preguntar primero al servidor de nombres especificado (servidor o servidores) pudiendo aprovecharse de la información de caché que dicho servidor tuviera disponible. Si el servidor de nivel superior al nuestro se encuentra congestionado puede merecer la pena la activación de esta característica de "redirección" ya que se puede disminuir la carga de tráfico que dicho servidor tiene que soportar.

Aviso

La dirección IP `127.0.0.1` *no* funciona aí. Se debe cambiar esta dirección IP por un servidor de nombres válido.

```
        /*
  * Si hay un cortafuegos entre usted y los servidores de
  * nombres que quiere consultar tendrÃi que descoentar la
  * siguiente directiva, "query-source".  Las versiones
  * anteriores de BIND siempre hacÃan sus consultas a travÃ©s
  * del puerto 53 pero BIND 8.1 utiliza por defecto un puerto no
  * privilegiado.
         */
        // query-source address * port 53;

        /*
  * Si lo va a ejecutar en un "cajÃ³n de arena" (o "sandbox")
  * tendrÃi que declarar una uicaciÃ³n diferente para el
  * fichero de volcado de named.
         */
        // dump-file "s/named_dump.db";
};

// Nota: lo siguiente serÃi incluÃdo en futuras versiones.
/*
host { any; } {
        topology {
                127.0.0.0/8;
        };
};
*/

// La configuraciÃ³n de secundarios se explica de modo secillo a
// partir de aquÃ-.
//
```

```
// Si activa un servidor de nombres local no olvide incluÃr
// 127.0.0.1 en su /etc/resolv.conf para que sea ese servidor el
// primero al que se consulte.
// AsegÃºrese tambiÃ©n de activarlo en /etc/rc.con

zone "." {
        type hint;
        file "named.root";
};

zone "0.0.127.IN-ADDR.ARPA" {
        type master;
        file "localhost.rev";
};

zone
"0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.IP6.↺
INT" {
        type master;
        file "localhost.rev";
};

// Nota: No use las direcciones IP que se muestran aquÃ-, son falsas
// y sÃ³lo se usancomo demostraciÃ³n y para una mejor comprensiÃ³n.
//
// Ejemplo de entradas en la configuraciÃ³n de secundarios.  Puede ↺
ser
// conveniente convertirse en secundario al menos del dominio en cuya
// zona estÃi su dominio.  Cnsulte con su administrador de red para
// que le facilite la direcciÃ³n IP del servidor primario.
//
// No olvide incluÃr la zona del bucle inverso (IN-ADDR.ARPA).  (Son
// los primeros bytes de la direcciÃ³n IP correspondiente, en orden
// inverso, con ".IN-ADDR.ARPA" al final.)
//
// Antes de configurar una zona primara asegÃºresede haber ↺
comprendido
// completamente cÃ³mo funcionan DNS y BIND.  Hay errores que no son
// visibles fÃicilmente.  La configuraciÃ³n de un secundario es, por
// el contrario, muchÃsimo mÃis sencilla.
//
// Nota: No se limite a copiar los ejemplos de mÃis arriba. :-)
// Utilice nombres y direcciones reales.
//
// ADVERTENCIA: FreeBSD ejecuta bind en un sandbok (observe los
// parÃimeros de named (named_flags) en rc.conf).  El directorio que
// contiene las zonas secundarias debe tener permisos de escritura
// para bind.  Le sugerimos la siguiente secuencia de órdenes:
//
//      mkdir /etc/namedb/s
//      chown bind:bind /etc/namedb/s
//      chmod 750 /etc/namedb/s
```

Si quiere más información sobre cómo ejecutar BIND en un "sandbox" consulte Ejecución de named en un sandbox.

```
/*
zone "ejemplo.com" {
        type slave;
        file "s/ejemplo.com.bak";
        masters {
                192.168.1.1;
        };
};

zone "0.168.192.in-addr.arpa" {
        type slave;
        file "s/0.168.192.in-addr.arpa.bak";
        masters {
                192.168.1.1;
        };
};
*/
```

Dentro del fichero `named.conf` se muestran ejemplos de entradas de esclavo tanto para las zonas directas como para las inversas.

Para cada nueva zona administrada se debe crear una entrada de zona dentro del fichero `named.conf`

Por ejemplo la entrada de zona más simple para el dominio `ejemplo.org` puede ser algo como esto:

```
zone "ejemplo.org" {
 type master;
 file "ejemplo.org";
};
```

Esta zona es una zona maestra (observe la línea de `type`, y mantiene la información de la zona en `/etc/namedb/ejemplo.org` tal como se indica en la línea de `file`.

```
zone "ejemplo.org" {
 type slave;
 file "ejemplo.org";
};
```

En el caso del esclavo la información de la zona se transmite desde el servidor de nombres maestro y se almacena en el fichero especificado. Cuando el servidor maestro " muere" o no puede ser alcanzado el servidor de nombres esclavo puede responder a las peticiones debido a que posee la información de la zona.

29.11.6.3. Ficheros de zona

A continuación se muestra un fichero de una zona maestra para el dominio `ejemplo.org` , que se encuentra ubicado en `/etc/namedb/ejemplo.org` :

```
$TTL 3600

example.org. IN SOA ns1.ejemplo.org. admin.ejemplo.org. (
                        5               -; Serial
                        10800           -; Refresh
                        3600            -; Retry
                        604800          -; Expire
                        86400 )         -; Minimum TTL

; DNS Servers
@       IN NS           ns1.ejemplo.org.
@       IN NS           ns2.ejemplo.org.

; Machine Names
localhost       IN A    127.0.0.1
ns1             IN A    3.2.1.2
ns2             IN A    3.2.1.3
mail            IN A    3.2.1.10
@               IN A    3.2.1.30

; Aliases
www             IN CNAME        @

; MX Record
@               IN MX   10      mail.ejemplo.org.
```

Tenga muy en cuenta que todo nombre de máquina que termina en "." es tratado como si fuera un nombre de máquina completo mientras que cualquier otro nombre sin el "." final se trata como una referencia relativa al dominio de origen de la zona. Por ejemplo `www` se traduce a `www + origen`. En nuestro fichero de zona ficticio nuestro origen es `ejemplo.org` de forma que `www` se convierte en `www.ejemplo.org`

El formato de un fichero de zona es el siguiente:

```
nombrederegistro IN tipodeentrada valor
```

Los registros de DNS que más se utilizan son:

SOA
: Comienzo de Zona con Autoridad (Start Of zone Authority)

NS
: Un servidor de nombres con autoridad para una una determinada zona

A
: Una dirección IP de una máquina

CNAME
: El nombre canónico de una máquina para definir un alias

MX
: mail exchanger

PTR
: Un puntero a un nombre de dominio (utilizados para definir el DNS inverso)

```
ejemplo.org. IN SOA ns1.ejemplo.org. admin.ejemplo.org. (
                        5               -; Serial
                        10800           -; Refresh after 3 hours
                        3600            -; Retry after 1 hour
                        604800          -; Expire after 1 week
                        86400 )         -; Minimum TTL of 1 day
```

`ejemplo.org.`
: el nomre de dominio, también el origen para el fichero de zona

`ns1.ejemplo.org.`
: el servidor de nombres primario/autoritario para esta zona

`admin.ejemplo.org.`
: la persona responsable de esta zona; observe que la dirección de correo electrónico aparece con la @ sustituida por un punto. (<admin@ejemplo.org > se escribe `admin.ejemplo.org`)

`5`
: el número de serie del fichero. Este número se debe incrementar cada vez que se modifique el fichero de zona. Muchos administradores prefieren un formato expresado del siguiente modo `aaaammddss` . 2001041002 significa (según dicho formato) que el fichero se modificó por última vez el 04/10/2001 y se indica con los dos últimos dígitos (02) que es la segunda vez en el día que se ha modificado el fichero. El número de serie es importante ya que para avisar a los servidores de nombres esclavos de que se ha actualizado la zona.

```
@       IN NS           ns1.ejemplo.org.
```

Esta es una entrada de tipo `NS`. Cada servidor de nombres que contesta de forma autoritaria a las consultas de un determinado dominio debe tener una de estas entradas. El caracter `@` se sustituye por `ejemplo.org.` , es decir, se sustituye por el origen.

```
localhost       IN A    127.0.0.1
ns1             IN A    3.2.1.2
ns2             IN A    3.2.1.3
mail            IN A    3.2.1.10
@               IN A    3.2.1.30
```

El registro de tipo A hace referencia a nombres de máquinas . Como puede verse más arriba `ns1.ejemplo.org` se resuelve a `3.2.1.2`. Vemos que se utiliza otra vez el origen `@`, que significa que `ejemplo.org` se resuelve a `3.2.1.30`.

```
www             IN CNAME        @
```

Los registros de nombres canónicos se utilizan normalmente como alias de máquinas. En el ejemplo `www` es un alias de `ejemplo.org` (`3.2.1.30`). `CNAME`s se puede utilizar para proporcionar alias de nombres de máquinas, o también para proporcionar un mecanismo de vuelta cíclica ("round robin") de un nombre de máquina mapeado a un determinado conjunto de máquinas intercambiables.

```
@               IN MX   10      mail.ejemplo.org.
```

El registro `MX` indica qué servidores de correo se encargan de recibir correos para esta zona. `mail.example.org` es el nombre del servidor de correo y 10 significa la prioridad de dicho servidor.

Se pueden especificar varios servidores de correo con prioridades de, por ejemplo,3, 2 y 1. Un servidor de correo que intenta entregar correo para el dominio `ejemplo.org` primero intentará contactar con el servidor especificado en el registro MX de mayor prioridad, después con el siguiente y así sucesivamente hasta que lo logre entregar.

Para los ficheros de zona de in-addr.arpa (DNS inverso) se utiliza el mismo formato excepto que se especifican registros `PTR` en lugar de registros `A` o `CNAME`.

```
$TTL 3600

1.2.3.in-addr.arpa. IN SOA ns1.ejemplo.org. admin.ejemplo.org. (
                        5               -; Serial
                        10800           -; Refresh
                        3600            -; Retry
                        604800          -; Expire
                        3600 )          -; Minimum

@       IN NS   ns1.ejemplo.org.
@       IN NS   ns2.ejemplo.org.

2       IN PTR  ns1.ejemplo.org.
3       IN PTR  ns2.ejemplo.org.
10      IN PTR  mail.ejemplo.org.
30      IN PTR  ejemplo.org.
```

Este fichero proporciona las asociaciones de direcciones IP con nombres de máquinas adecuadas para nuestro dominio ficticio.

29.11.7. Servidor de nombres de cache

Un servidor de nombres de tipo caché es un servidor de nombres que no es autoritario para ninguna zona. Simplemente realiza consultas por sí mismo y recuerda las respuestas para futuros usos. Para configura uno de estos servidores se configura el servidor de la forma habitual pero se omite la inclusión de zonas.

29.11.8. Ejecución de named en una " Sandbox"

Para obtener una mayor seguridad se puede ejecutar named(8) como un usuario sin privilegios y configurarlo mediante chroot(8) dentro del directorio especificado como el directorio del "sandbox". Esto hace que cualquier cosa que se encuentre fuera de dicho directorio resulte inaccesible para el dæmon named. En caso de que se comprometiera la seguridad de named esta restricción puede ayudar a limitar el daño sufrido. FreeBSD dispone por defecto de un usuario y un grupo destinado a este uso: `bind`.

Nota

Hay quien recomienda que en lugar de configurar named con `chroot` es mejor configurarlo dentro de jail(8). En esta sección no se va a explicar esa alternativa.

Debido a que named no va a poder acceder a nada que se encuentre fuera del directorio " sandbox" (y esto incluye cosas tales como bibliotecas compartidas, "sockets" de "log", etc) se debe efectuar una serie de cambios para que named pueda funcionar con normalidad. En la siguiente lista se supone que la ruta del "sandbox" es `/etc/namedb` y que no se ha modificado anteriormente dicho directorio. Por favor, ejecute los pasos que se muestran a continuación:

- Cree todos los directorios que named espera tener a su disposición:

```
# cd /etc/namedb
# mkdir -p bin dev etc var/tmp var/run master slave
# chown bind:bind slave var/* ❶
```

❶ named sólamente necesita escribir en estos directorios así que eso es todo lo que debemos crear.

- Reorganizar y crear los archivos de configuración de zona básicos:

```
# cp /etc/localtime etc ❶
# mv named.conf etc && ln -sf etc/named.conf
# mv named.root master

# sh make-localhost && mv localhost.rev localhost-v6.rev master
# cat > master/named.localhost
```

```
$ORIGIN localhost.
$TTL 6h
@ IN SOA localhost. postmaster.localhost. (
   1 ; serial
   3600 ; refresh
   1800 ; retry
   604800 ; expiration
   3600 ) ; minimum
 IN NS localhost.
 IN A  127.0.0.1
^D
```

❶ Esto permite que named pueda escribir al archivo de log la hora correcta a través del syslogd(8)

- Si está usando una versión de FreeBSD anterior a 4.9-RELEASE se debe construir una copia estáticamente enlazada de named-xfer y copiarla dentro del directorio del “sandbox”:

```
# cd /usr/src/lib/libisc
# make cleandir && make cleandir && make depend && make all
# cd /usr/src/lib/libbind
# make cleandir && make cleandir && make depend && make all
# cd /usr/src/libexec/named-xfer
# make cleandir && make cleandir && make depend && make ↵
NOSHARED=yes all
# cp named-xfer /etc/namedb/bin && chmod 555 /etc/namedb/bin/named-
xfer❶
```

Despueés de instalar la versión estática de `named-xfer` se deben realizar algunas tareas de limpieza para evitar dejar copias de bibliotecas o de programas en nuestros ficheros de fuentes:

```
# cd /usr/src/lib/libisc
# make cleandir
# cd /usr/src/lib/libbind
# make cleandir
# cd /usr/src/libexec/named-xfer
# make cleandir
```

❶ En algunas ocasiones este paso puede fallar. Si es su caso ejecute lo siguiente:

```
# cd /usr/src && make cleandir && make cleandir
```

y borre su directorio `/usr/obj` :

```
# rm -fr /usr/obj && mkdir /usr/obj
```

Esto limpia cualquier “impureza” del árbol de fuentes y si se repiten los pasos anteriores todo debería funcionar.

Si se está usando FreeBSD version 4.9-RELEASE o posterior el ejecutable de `named-xfer` del directorio `/usr/libexec` ya se encuentra enlazado estáticamente y se puede utilizar cp(1) para copiarlo directamente en nuestro "sandbox".

- Cree el fichero `dev/null` de tal forma que named pueda verlo y pueda escribir sobre él:

```
# cd /etc/namedb/dev && mknod null c 2 2
# chmod 666 null
```

- Enlace simbólicamente `/var/run/ndc` con `/etc/namedb/var/run/ndc` :

```
# ln -sf /etc/namedb/var/run/ndc /var/run/ndc
```

Nota

Esto simplemente evita el tener que especificar la opción `-c` de ndc(8) cada vez que se ejecute. Dado que los contenidos de /var/run se borran al inicio del sistema, si se piensa que esto puede resultar útil, se puede añadir esta orden al " crontab" del usuario root utilizando la opción `@reboot`. Consulte crontab(5) para saber más información sobre esto.

- Configure syslogd(8) para que cree un "socket" de `log` adicional de tal forma que named pueda escribir sobre él. Añada `-l /etc/namedb/dev/log` a la variable `syslogd_flags` dentro del fichero `/etc/rc.conf` .

- Reorganice la ejecución de las aplicaciones named y `chroot` para que se ejecuten dentro del "sandbox" añadiendo lo siguiente al fichero`/etc/rc.conf` :

```
named_enable="YES"
named_flags="-u bind -g bind -t /etc/namedb /etc/named.conf"
```

Nota

Recuerde que el fichero de configuración */etc/named.conf* tiene una ruta completa *que comienza en el directorio del "sandbox"*; por ejemplo, en la línea superior el fichero que aparece es en realidad `/etc/namedb/etc/named.conf` .

El siguiente paso consiste en editar el fichero `/etc/namedb/etc/named.conf` de tal forma que named pueda conocer qué zonas cargar y donde encontrarlas en disco. A continuación se muestra un ejemplo comentado (cualquier cosa que no se comenta en el ejemplo es porque resulta igual que la configuración del servidor de DNS del caso normal):

```
options {
        directory "/";❶
        named-xfer "/bin/named-xfer";❷
        version "";  // No muestra la versiÃn de BIND
        query-source address * port 53;
};
// ndc control socket
controls {
        unix "/var/run/ndc" perm 0600 owner 0 group 0;
};
// A partir de aquÃvan las zonas:
zone "localhost" IN {
        type master;
        file "master/named.localhost";❸
        allow-transfer { localhost; };
        notify no;
};
zone "0.0.127.in-addr.arpa" IN {
        type master;
        file "master/localhost.rev";
        allow-transfer { localhost; };
        notify no;
};
zone "0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.↺
ip6.int" {
 type master;
 file "master/localhost-v6.rev";
 allow-transfer { localhost; };
 notify no;
};
zone "." IN {
        type hint;
        file "master/named.root";
};
zone "private.example.net" in {
        type master;
        file "master/private.example.net.db";
 allow-transfer { 192.168.10.0/24; };
};
zone "10.168.192.in-addr.arpa" in {
        type slave;
        masters { 192.168.10.2; };
        file "slave/192.168.10.db";❹
};
```

❶ La línea que contiene `directory` se especifica como `/`, ya que todos los ficheros que named necesita se encuentran dentro de este directorio (recuerde que esto es equivalente al fichero `/etc/namedb` de un usuario "normal".

❷ Especifica la ruta completa para el binario `named-xfer` binary (desde el marco de referencia de named). Esto resulta necesario ya que por defecto named se compila de tal forma que trata de localizar `named-xfer` dentro de `/usr/libexec`.

❸ Especifica el nombre del fichero (relativo a la línea (relativo a la línea) `directory` anterior) donde named puede encontrar el fichero de zona para esta zona.

❹ Especifica el nombre del fichero (relativo a la líena `directory` anterior) donde named debería escribir una copia del archivo de zona para esta zona después de recuperarla exitosamente desde el servidor maestro. Este es el motivo por el que en las etapas de configuración anteriores necesitábamos cambiar la propiedad del directorio `slave` al grupo `bind`.

Después de completar los pasos anteriores reinicie el servidor o reinicie syslogd(8) y ejecute named(8) asegurándose de que se utilicen las nuevas opciones especificadas en `syslogd_flags` y `named_flags`. En estos momentos deberíamos estar ejecutando una copia de named dentro de un "sandbox".

29.11.9. Seguridad

Aunque BIND es la implementación de DNS más utilizada existe siempre el asunto relacionado con la seguridad. De vez en cuando se encuentran agujeros de seguridad y vulnerabilidades.

Es una buena idea suscribirse a CERT y a freebsd-security-notifications para estar al día de los problemas de seguridad relacionados con named.

Sugerencia

Si surge un problema nunca está de más actualizar los fuentes y recompilar los ejecutables desde dichas fuentes.

29.11.10. Lecturas recomendadas

Las páginas del manual de BIND/named: ndc(8) named(8) named.conf(5)

- Página oficial de ISC Bind
- Preguntas más frecuentes sobre BIND
- Libro de O'Reilly "DNS and BIND", cuarta edición
- RFC1034 - Nombre de Dominio - Conceptos y Características

- RFC1035 - Nombres de Domninio - Implementación y Especificación

29.12. NTP

Escrito por Tom Hukins.

29.12.1. Resumen

Según pasa el tiempo el reloj de un computador está expuesto a ligeros desplazamientos. NTP (Protocolo de Hora en Red, en inglés "Network Time Protocol") es un protocolo que permite asegurar la exactitud de nuestro reloj.

Existen varios servicios de internet que confían y se pueden beneficiar de relojes de computadores precisos. Por ejemplo un servidor web puede recibir peticiones de un determinado fichero si ha sido modificado posteriormente a una determinada fecha u hora. Servicios como cron(8) ejecutan órdenes en determinados instantes. Si el reloj no se encuentra ajustado estas órdenes pueden ejecutarse fuera de la hora prevista.

FreeBSD viene con el servidor NTP ntpd(8) que se puede utilizar para preguntar a otros servidores NTP, de tal forma que podemos ajustar nuestro reloj según la hora de otros servidores e incluso proporcionar servicio de hora nosotros mismos.

29.12.2. Elección de los servidores de hora adecuados

Para sincronizar nuestro reloj necesitamos comunicarnos con uno o más servidores NTP. El administrador de nuestra red o nuestro proveedor de servicios de Internet muy posiblemente hayan configurado algún servidor NTP para estos propósitos. Consulte la documentación de que disponga. Existe una lista de servidores NTP públicamente accesibles que se pueden utilizar para buscar un servidor NTP que se encuentre geográficamente próximo. Asegúrese de que conoce la política de uso de estos servidores públicos ya que en algunos casos es necesario pedir permiso al administrador antes de de poder utilizarlos, principalmente por motivos estadísticos.

Le recomendamos seleccionar servidores NTP que no se encuentren conectados entre sí por si alguno de los servidores que use sea inaccesible o su reloj se averíe. ntpd(8) utiliza las respuestas que recibe de otros servidores de una forma inteligente. servidores de una forma inteligente (Tiene a hacer más caso a los más fiables.

29.12.3. Configuración de la máquina

29.12.3.1. Configuración básica

Si sólamente deseamos sincronizar nuestro reloj cuando se arranca la máquina se puede utilizar ntpdate(8). Esto puede ser adecuado en algunas máquinas de escritorio que se

reinician frecuentemente y donde la sincronización no suele ser un objetivo prioritario pero normalmente la mayoría de las máquinas deberían ejecutar ntpd(8).

La utilización de ntpdate(8) en tiempo de arranque es también una buena idea incluso para las máquinas que ejecutan ntpd(8). El programa ntpd(8) modifica el reloj de forma gradual, mientras que ntpdate(8) ajusta directamente el reloj sin importar que tamaño tenga la diferencia de tiempo existente entre la máquina y el servidor de tiempo de referencia.

Para activar ntpdate(8) en tiempo de arranque, añada `ntpdate_enable="YES"` al fichero `/etc/rc.conf` . También es necesario especificar todos los servidores que deseamos utilizar para realizar el proceso de sincronización y cualquier parámetro que deseemos pasar a ntpdate(8) utilizando la variable `ntpdate_flags` .

29.12.3.2. Configuración general

NTP se configura mediante el archivo `/etc/ntp.conf` utilizando el formato descrito en ntp.conf(5). A continuación se muestra un sencillo ejemplo:

```
server ntplocal.ejemplo.com prefer
server timeserver.ejemplo.org
server ntp2a.ejemplo.net

driftfile /var/db/ntp.drift
```

La opción `server` especifica qué servidores se van a utilizar, especificando un servidor por línea. Si se especifica un servidor con el argumento `prefer`, como en `ntplocal.ejemplo.com` dicho servidor se prefiere sobre los demás. No obstante la respuesta de su servidor preferido se descartará si difiere sustancialmente de la respuesta recibida por parte del resto de los servidores especificados; en caso contrario sólo se tendrá en cuenta la respuesta del servidor preferido sin importar la información suministrada por el resto. El argumento `prefer` se utiliza normalmente en servidores NTP altamente precisos, como aquellos que poseen hardware de tiempo específico.

La opción `driftfile` especifica qué fichero se utiliza para almacenar el desplazamiento de la fracuencia de reloj de la máquina. El programa ntpd(8) utiliza este valor para automáticamente compensar el desvío que experimenta de forma natural el reloj de la máquina, permitiendo mantener una precisión acotada incluso cuando se pierde la comunicación con el resto de referencias externas.

La opción `driftfile` especifica qué fichero se utiliza para almacenar la información sobre espuestas anteriores de servidores NTP. Este fichero contiene información útil para la implementación de NTP. No debería ser modificada por ningún otro proceso.

29.12.3.3. Control de acceso al servidor NTP

Por defecto nuestro servidor de NTP puede ser accedido por cualquier máquina de Internet. La opción `restrict` se puede utilizar para controlar controlar qué máquinas pueden acceder al servicio.

Si queremos denegar el acceso a todas las máquinas existentes basta con añadir la siguiente línea a `/etc/ntp.conf` :

```
restrict default ignore
```

Si sólo queremos permitir el acceso al servicio de hora a las máquinas de nuestra red y al menos tiempo nos queremos asegurar de que dichos clientes no pueden a su vez configurar la hora del servidor o utilizarse ellos mismos como nuevos servidores de hora basta con añadir lo siguiente en lugar de lo anterior:

```
restrict 192.168.1.0 mask 255.255.255.0 notrust nomodify notrap
```

donde `192.168.1.0` es la dirección IP de nuestra red y `255.255.255.0` es la máscara de red.

`/etc/ntp.conf` puede contener varias opciones de tipo `restrict`. Para más detalles consulte la sección `Soporte de Control de Acceso` de ntp.conf(5).

29.12.4. Ejecución del servidor de NTP

Para asegurarnos de que el servidor de NTP se ejecuta en tiempo de arranque se debe añadir la línea `xntpd_enable="YES"` al fichero `/etc/rc.conf` . Si deseamos pasar opciones adicionales a ntpd(8) se puede modificar la variable `xntpd_flags` del fichero `/etc/rc.conf` .

Para ejecutar el servidor sin reiniciar la máquina ejecute `ntpd` junto con todos aquellos parámetros que haya especificado en la variable de arranque `xntpd_flags` del fichero `/etc/rc.conf` . Por ejemplo:

```
# ntpd -p /var/run/ntpd.pid
```

Nota

Bajo FreeBSD 5.X se han renombrado algunas opciones del fichero `/etc/rc.conf` . Se debe reemplazar cualquier aparición `xntpd` por por `ntpd`.

29.12.5. Utilización de ntpd junto con una conexión temporal a Internet

El programa ntpd(8) no necesita una conexión permanente a Internet para poder funcionar correctamente. No obstante si la conexión a Internet se encuentra configurada con marcación bajo demanda es una buena idea impedir que el tráfico de NTP lance una marcación automática o que mantenga la conexión viva. Si se utiliza el PPP de entorno de

usuario se pueden utilizar las directivas `filter` dentro del fichero `/etc/ppp/ppp.conf` para evitar esto. Por ejemplo:

```
set filter dial 0 deny udp src eq 123
# Evita que el trÃ¡fico NTP inice una llamada saliente
set filter dial 1 permit 0 0
set filter alive 0 deny udp src eq 123
Evita que el trÃ¡ficoNTP entrante mantenga abierta la conexiÃ³n
set filter alive 1 deny udp dst eq 123
Evita que el trÃ¡fico NTP saliente mantenga abierta la conexiÃ³n
set filter alive 2 permit 0/0 0/0
```

Para ás detalles consulte la sección `PACKET FILTERING` de ppp(8) y los ejemplos que se encuentran en `/usr/share/examples/ppp/` .

Nota

Algunos proveedores de acceso a Internet bloquean paquetes que utilizan números de puertos bajos impidiendo que los paquetes de vuelta alcancen nuestra máquina.

29.12.6. Información adicional

Hay documentación sobre el servidor NTP en formato HTML en `/usr/share/doc/ntp/` .

29.13. Traducción de direcciones de red

Escrito por Chern Lee.

29.13.1. Overview

El dæmon de FreeBSD que se encarga de traducir direcciones de red, más conocido como natd(8), es un dæmon que acepta paquetes IP, modifica la dirección IP fuente de dichos paquetes y los reinyecta en el flujo de paquetes IP de salida. natd(8) ejecuta este proceso modificando la dirección de origen y el puerto de tal forma que cuando se reciben paquetes de contestación natd(8) es capaz de determinar el destino real y reenviar el paquete a dicho destino.

El uso más común de NAT es para Compartir la Conexión a Internet.

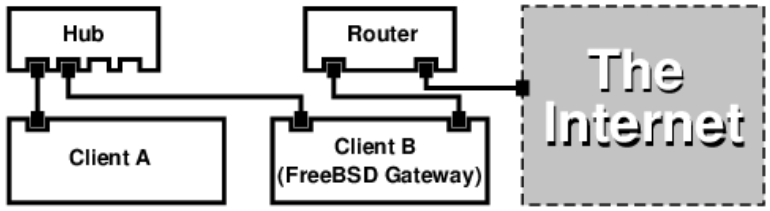
29.13.2. Configuración

Debido al pequeño espacio de direccionamiento que se encuentra actualmente disponible en IPv4 y debido también al gran aumento que se está produciendo en cuanto a número

de usuarios de líneas de conexión a Internet de alta velocidad como cable o DSL la gente necesita utilizar cada vez más la salida de Compartición de Conexión a Internet. La característica de poder conectar varios computadores a través de una única conexión y una única dirección IP hacen de natd(8) una elección razonable.

Cada vez con más frecuencia un usuario típico dispone de una máquina conectada mediante cable o DSL pero desearía utilizar dicha máquina como un " router" de acceso para el resto de los ordenadores de su red de área local.

Para poder hacerlo la máquina (FreeBSD por supuesto) debe configurarse para actuar como pasarela. Debe tener al menos dos tarjetas de red, una para conectarse a la red de área local y la otra para conectarse con el "router" de acceso a Internet. Todas las máquinas de la LAN se conectan entre sí mediante un "hub" o un " switch".



Una configuración como esta se utiliza frecuentemente para compartir el acceso a Internet. Una de las máquinas de la LAN está realmente conectada a Internet. El resto de las máquinas acceden a Internet utilizando como "pasarela" la máquina inicial.

29.13.3. Configuración

Se deben añadir las siguientes opciones al fichero de configuración del núcleo:

```
options IPFIREWALL
options IPDIVERT
```

Además, según se prefiera, se pueden añadir también las siguientes:

```
options IPFIREWALL_DEFAULT_TO_ACCEPT
options IPFIREWALL_VERBOSE
```

Lo que viene a continuación se tiene que definir en `/etc/rc.conf` :

```
gateway_enable="YES"
firewall_enable="YES"
firewall_type="OPEN"
natd_enable="YES"
natd_interface="fxp0"
natd_flags=""
```

gateway_enable="YES"	Configura la máquina para que actúe como "router" o pasarela de red. Se pue-

	de conseguir lo mismo ejecutando `sysctl net.inet.ip.forwarding=1` .
firewall_enable="YES"	Activa las reglas de cortafuegos que se encuentran definidas por defecto en `/etc/rc.firewall` y que entran en funcionamiento en el arranque del sistema.
firewall_type="OPEN"	Especifica un conjunto de reglas de cortafuegos que permite el acceso a todos los paquetes que se reciban. Consulte `/etc/rc.firewall` para obtener información sobre el resto de tipos de reglas que se pueden configurar.
natd_interface="fxp0"	Indica qué interfaz se utiliza para reenviar paquetes (la interfaz que se conecta a Internet).
natd_flags=""	Define cualesquiera otras opciones que se deseen proporcionar a natd(8) en tiempo de arranque.

Si se definen las opciones anteriores, en el arranque del sistema el fichero `/etc/rc.conf` configurará las variables de tal forma que se ejecutaría `natd -interface fxp0` . Evidentemente esta orden también se puede ejecutar de forma manual.

Nota

También es posible utilizar un fichero de configuración para natd(8) en caso de que deseemos especificar muchos parámetros de arranque. Tendremos que declarar la ubicación del fichero de configuración mediante la inclusión de lo siguiente en `/etc/rc.conf` :

```
natd_flags="-f /etc/natd.conf"
```

El fichero `/etc/natd.conf` debe contener una lista de opciones de configuración una opción por línea. Por ejemplo, en el caso que se comenta en la siguiente sección se utilizaría un fichero de configuración con la siguiente información:

```
redirect_port tcp 192.168.0.2:6667 6667
redirect_port tcp 192.168.0.3:80 80
```

> Para obtener más información sobre el fichero de configuración se puede consultar la opción `-f` que se describe en la página del manual de natd(8).

Cada máquina (y cada interfaz) que se encuentra conectada a la LAN debe poseer una dirección IP perteneciente al espacio de direcciones IP privado tal y como se define en RFC 1918 y debe poseer como pasarela por defecto la dirección IP de la interfaz interna (la interfaz que se conecta a la LAN) de la máquina que ejecuta natd.

Por ejemplo los clientes `A` y `B` se encuentran en la LAN utilizando las direcciones IP `Â192.168.0.2` y `192.168.0.3` , respectivamente. La máquina que ejecuta natd posee la dirección IP `192.168.0.1` en la interfaz que se conecta a la LAN. El "router" por defecto tanto de `A` omo de `B` se establece al valor `192.168.0.1` . La interfaz externa de la máquina que ejecuta natd, la interfaz que se conecta con Internet, no necesita de ninguna especial en relación con el tema que estamos tratando en esta sección.

29.13.4. Redirección de puertos

El incoveniente que se presenta con la utilización de natd(8) es que los clientes de la LAN no son accesibles desde Internet. Dichos clientes pueden establecer conexiones con el exterior pero no pueden recibir intentos de conexión desde pueden recibir intentos de conexion desde Internet. Esto supone un gran problema cuando se quieren ejecutar servicios de acceso global en una o varias máquinas de la red LAN. Una forma sencilla de solucionar parcialmente este problemma consiste en redirigir determinados puertos en el servidor natd hacia determinadas máquinas de la LAN.

Supongamos por ejemplo que en `A` se ejecuta un servidor de IRC y que en `B` se ejecuta un servidor web. Para que funcione lo que hemos comentado anteriormente se tienen que redirigir las conexiones recibidas en los puertos 6667 (IRC) y 80 (web) a dichas máquinas, respectivamente.

Se debe pasar la opción `-redirect_port` a natd(8) con los valores apropiados. La sintaxis es como sigue:

```
-redirect_port proto IPdestino:PUERTOdestino[-PUERTOdestino]
               [aliasIP:]aliasPUERTO[-aliasPUERTO]
 [IPremota[:PUERTOremoto[-PUERTOremoto]]]
```

Continuando con el ejemplo anterior los valores serían:

```
  -redirect_port tcp 192.168.0.2:6667 6667
  -redirect_port tcp 192.168.0.3:80 80
```

Esto redirigirá los puertos *tcp* adecuados a las máquinas situadas en la LAN.

La opción `-redirect_port` se puede utilizar para indicar rangos de puertos en vez de puertos individuales. Por ejemplo, *`tcp 192.168.0.2:2000-3000 2000-3000`* redirige todas las conexiones recibidas desde los puertos 2000 al 3000 a los puertos puertos 2000 a 3000 de la máquina `A`.

Estas opciones se pueden utilizar cuando se ejecute directamente natd(8) se pueden situar en la variable `natd_flags=""` en `/etc/rc.conf` y también se pueden pasar mediante un archivo de configuración.

Para obtener más información sobre opciones de configuración por favor consulte natd(8)

29.13.5. Redirección de direcciones

La redirección de direcciones es una característica útil si se dispone de varias direcciones IP pero todas ellas se ubican en una única máquina. Gracias a esto natd(8) puede asignar a cada cliente de la red LAN su propia dirección IP externa. natd(8) reescribe los paquetes que salen de la red LAN con la dirección IP externa adecuada y redirige todo el tráfico recibido de vuelta al cliente en función de la dirección IP de destino: esto se conoce como NAT estático. Por ejemplo las direcciones IP `128.1.1.1`, `128.1.1.2` y `128.1.1.3` pertenecen al "router" natd. `128.1.1.1` se puede utilizar como la dirección IP externa del natd, mientras que `128.1.1.2` y `128.1.1.3` se redirigen a los clientes `A` y `B`, respectivamente.

La sintaxis de la opción `-redirect_address` es la siguiente:

```
-redirect_address IPlocal IPpública
```

IPlocal	La dirección IP interna del cliente de la LAN.
IPpública	La dirección IP externa que se corresponde con un determinado cliente de la LAN.

En nuestro ejemplo esta opción se especificaría de la siguiente forma:

```
-redirect_address 192.168.0.2 128.1.1.2
-redirect_address 192.168.0.3 128.1.1.3
```

De forma semejante a la opción `-redirect_port` estos argumentos se pueden especificar directamente sobre la variable `natd_flags=""` del fichero `/etc/rc.conf` o también se pueden pasar vía archivo de configuración de natd. Si se utiliza redirección de direcciones ya no es necesario utilizar redirección de puertos ya que todos los paquetes que se reciben en una determinada dirección IP son redirigidos a la máquina especificada.

Las direcciones IP externas de la máquina que ejecuta natd se deben activar y deben formar parte de un alias configurado sobre la interfaz externa que se conecta a Internet. Consulte rc.conf(5) para aprender a hacerlo.

29.14. El “Superservidor” inetd

Escrito por Chern Lee.

29.14.1. Resumen

inetd(8) se conoce como el “Super Servidor de Internet” debido a que gestiona las conexiones de varios dæmones. Los dæmones son programas que proporcionan servicios de red. inetd actúa como un servidor de servidor de gestión de otros dæmones. Cuando inetd(8) recibe una conexión se determina qué dæmon debería responder a dicha conexión, se lanza un proceso que ejecuta dicho dæmon y se le entrega el “ socket”. La ejecución de una única instancia de inetd reduce la carga del sistema en comparación con lo que significaría ejecutar cada uno de los dæmones que gestiona de forma individual.

inetd se utiliza principalmente para lanzar procesos que albergan a otros dæmones pero inetd también se utiliza para gestionar determinados protocolos triviales como chargen, auth y daytime.

Esta sección trata la configuración básica de inetd a través de sus opciones de línea de órdenes y utilizando su fichero de configuración, denominado `/etc/inetd.conf` .

29.14.2. Configuraciones

inetd se inicializa a través del fichero `/etc/rc.conf` en tiempo de arranque. La opción `inetd_enable` posee el valor `NO` por defecto, pero a menudo la aplicación sysinstall la activa cuando se utiliza la configuración de perfil de seguridad medio. Estableciendo

```
inetd_enable="YES"
```

o

```
inetd_enable="NO"
```

dentro de `/etc/rc.conf` se puede activar o desactivar la la ejecución de inetd en el arranque del sistema.

Se pueden además aplicar distintas opciones de línea de órdenes mediante la opción `inetd_flags` .

29.14.3. Opciones de línea de órdenes

sinopsis de inetd:

```
inetd [-d] [-l] [-w] [-W] [-c máximo] [-C tasa] [-a dirección | nombre_de_host] [-p nombre_de_fichero] [-R tasa] [fichero de configuración ]
```

-d
: Activa la depuración.

-l
: Activa el “logging” de las conexiones efectuadas con écute.

-w
: Activa el recubrimiento de TCP para servicios externos (activado por defecto).

-W
: Activa el recubrimiento de TCP para los servicios internos, ejecutados directamente por el dæmon inetd (activado por defecto).

-c máximo
: Especifica el máximo número de invocaciones simultáneas de cada servicio; el valor por defecto es ilimitado. Se puede sobreescribir para cada servicio utilizando la opción `max-child`.

-C tasa
: Especifica el máximo número de veces que se puede llamar a un servicio desde un dirección IP determinada por minuto; el valor por defecto es ilimitado. Se puede redefinir para cada servicio utilizando la opción `max-connections-per-ip-per-minute`.

-R tasa
: Especifica el máximo número de veces que se puede invocar un servicio en un minuto; el valor por defecto es 256. Un valor de `0` permite un número ilimitado de llamadas.

-a
: Especifica una dirección IP a la cual se asocia y sobre la cual se queda esperando recibir conexiones. Puede declararse también un nombre de máquina, en cuyo caso se utilizará la dirección (o direcciones si hay más de una) IPv4 o IPv6 que estén tras dicho nombre. Normalmente se usa un nombre de máquina cuando inetd se ejecuta dentro de un jail(8), en cuyo caso el nombre de máquina se corresponde con el entorno jail(8).

  Cuando se desea asociarse tanto a direcciones IPv4 como a direcciones IPv6 y se utiliza un nombre de máquina se necesita una entrada para cada protocolo (IPv4 o IPv6) para cada servicio que se active a través de `/etc/inetd.conf`. Por ejemplo un servicio basado en TCP necesitaría dos entradas, una utilizando `tcp4` para el protocolo IPv4 y otra con `tcp6` para las conexiones a través del procolo de red IPv6.

-p
: Especifica un fichero alternativo en el cual se guarda el ID del proceso.

Estas opciones se pueden declarar dentro de las variables `inetd_flags` del fichero `/etc/rc.conf`. Por defecto `inetd_flags` tiene el valor `-wW`, lo que activa el recubrimiento de TCP para los servicios internos y externos de inetd. Los usuarios inexpertos no suelen introducir estos parámetros y por ello ni siquiera necesitan especificarse dentro de `/etc/rc.conf`.

Nota

Un servicio externo es un dæmon que se ejecuta fuera de inetd y que se lanza cuando se recibe un intento de conexión. Un servicio interno es un servicio que inetd puede servir directamente sin necesidad de lanzar nuevos procesos.

29.14.4. `inetd.conf`

La configuración de inetd se realiza a través del ficherode configuración `/etc/inetd.conf`.

Cuando se realiza una modificación en el fichero `/etc/inetd.conf` se debe obligar a inetd a releer dicho fichero de configuración, lo cual se realiza enviando una señal "HANGUP" al proceso inetd como se muestra a continuación:

Ejemplo 29.4. Envío de una señal HANGUP a inetd

```
# kill -HUP `cat /var/run/inetd.pid`
```

Cada línea del fichero de configuración especifica un dæmon individual. Los comentarios se preceden por el caracter "`#`". El formato del fichero de configuración `/etc/inetd.conf` es el siguiente:

```
service-name
socket-type
protocol
{wait|nowait}[/max-child[/max-connections-per-ip-per-minute]]
user[:group][/login-class]
server-program
server-program-arguments
```

A continuación se muestra una entrada de ejemplo para el dæmon ftpd para IPv4:

```
ftp     stream  tcp     nowait  root    /usr/libexec/ftpd       ftpd
 -l
```

service-name
: Este es el nombre del servicio que proporciona un determinado dæmon. Se debe corresponder con el nombre del nombre de servicio que se declara en el fichero `/etc/`

`services`. Este fichero determina sobre qué puerto debe ponerse a escuchar inetd. Si se crea un nuevo servicio se debe especificar primero en `/etc/services` .

socket-type
: Puede ser `stream` , `dgram` , `raw` o `seqpacket`. `stream` se debe utilizar obligatoriamente para dæmones orientados a conexión (dæmones TCP) mientras que `dgram` se utiliza en dæmones basados en el protocolo de transporte UDP.

protocol
: Uno de los siguientes:

Protocolo	Explicación
tcp, tcp4	TCP IPv4
udp, udp4	UDP IPv4
tcp6	TCP IPv6
udp6	UDP IPv6
tcp46	TCP IPv4 e IPv6 al mismo tiempo
udp46	UDP IPv4 e IPv6 al mismo tiempo

{wait|nowait}[/max-child[/max-connections-per-ip-per-minute]]
: `wait|nowait` indica si el dæmon puede gestionar su propio “socket” o no. Los “ sockets” de tipo `dgram` deben utilizar obigatoriamente la opción `wait` mientras que los dæmones basados en “sockets” de tipo “stream”, los cuales se implementan normalmente mediante hilos, debería utilizar la opción `nowait`. La opción `wait` normalmente entrega varios “ sockets” a un único dæmon, mientras que la opción `nowait` lanza un dæmon “hijo” por cada nuevo “ socket”.

  El número máximo de dæmones “ hijo” que puede lanzar inetd se puede especificar mediante la opción `max-child` . Si se necesita por ejemplo un límite de diez instancias para un dæmon en particular se puede especificar el valor `10` justo después de la opción `nowait`.

  Además de `max-child` se puede activar otra opción para limitar en número máximo de conexiones que se aceptan desde un determinado lugar mediante la opción `max-connections-per-ip-per-minute` . Esta opción hace justo lo que su nombre indica. Un valor de, por ejemplo, diez en esta opción limitaría cualquier máquina remota a un máximo de diez intentos de conexión por minuto. Esto resulta útil para prevenir un consumo incontrolado de recursos y ataques de Denegación de Servicio (“Denial of Service” o DoS) sobre nuestra máquina.

  Cuando se especifica este campo las opciones `wait` o `nowait` son obligatorias `max-child` y `max-connections-per-ip-per-minute` son opcionales.

  Un dæmon de tipo “stream” sin la opción `max-child` y sin la opción `max-connections-per-ip-per-minute` simplemente especificaría la opción `nowait`.

El mismo dæmon con el límite máximo de diez dæmones "hijos" sería: `nowait/10`.

La misma configuración con un límite de veinte conexiones por dirección IP por minuto y un máximo total de diez dæmones "hijos" sería: `nowait/10/20`.

Todas estas opciones son utilizadas por el dæmon fingerd que se muestra a continuación a modo de ejemplo:

```
finger stream  tcp     nowait/3/10 nobody /usr/libexec/fingerd ↺
fingerd -s
```

user
: Este es el nombre de usuario con el que debería ejecutarse un determinado dæmon. Normalmente los dæmones se suelen ejectar con permisos de `root`. Por motivos de seguridad, resulta bastante común encontrarse con algunos servidores que se ejecutan bajo el usuario `daemon` o incluso por el usuario menos privilegiado de todos que es el usuario `nobody`.

server-program
: La ruta completa de la localización del dæmon que se quiere ejecutar cuando se recibe un intento de conexión. Si el dæmon es un servicio proporcionado por el propio inetd se debe utilizar la opcion `internal` en su lugar.

server-program-arguments
: Esto funciona en conjunción con `server-program`, ya que especifica los argumentos, comenzando por `argv[0]` , que se pasan al dæmon cuando se le invoca. Si la línea de órdenes es `mydaemon -d`, `midæmon -d` debería ser el valor de la opción `server-program-arguments`. Si el dæmon es un servicio interno se debe utilizar la utilizar la opción `internal` en lugar de la que estamos comentando.

29.14.5. Seguridad

Dependiendo del perfil de seguridad establecido cuando se instaló el sistema FreeBSD varios dæmones de inetd pueden estar desactivados o activados. Si no se necesita un dæmon determinado, *no lo active*. Especifique un "`#`" al comienzo de la línea del dæmon que quiere desactivar y envíe una señal hangup a inetd. No se aconseja ejecutar algunos dæmones determinados (un caso típico es fingerd) porque pueden proporcionar información valiosa para un atacante.

Algunos dæmones no presentan ninguna característica de seguridad y poseen grandes o incluso no poseen tiempos de expiración para los intentos de conexión. Esto permite que un atacante sature los recursos de nuestra máquina realizando intentos de conexión a una tasa relativamente baja contra uno de estos ingenuos dæmones. Pueder ser una buena idea protegerse de esto utilizando las opciones `max-connections-per-ip-per-minute` y `max-child` para este tipo de dæmones.

El recubrimiento de TCP está activado por defecto tal y como ya se ha comentado anteriormente. Consulte la página del manual de hosts_access(5) para obtener más informa-

ción sobre cómo aplicar restricciones relacionadas con TCP a los dæmones invocados por inetd.

29.14.6. Varios

daytime, time, echo, discard, chargen y auth son servicios que inetd proporciona de forma interna y propia.

El servicio auth proporciona servicios de identificación a través de la red (ident, identd) y se puede configurar hasta en cierto grado.

Consulte la página del manual de inetd(8) si quiere conocer todos los detalles.

29.15. Línea IP paralela (PLIP)

PLIP nos permite configurar TCP/IP a través del puerto paralelo. Es útil para conectar máquinas que no poseen tarjetas de red, o para instalar FreeBSD en ciertos viejos modelos de portátiles. En esta sección se explica lo siguiente:

- Construcción de un cable paralelo (laplink).
- Conexión de dos computadores utilizando PLIP.

29.15.1. Construcción de un cable paralelo

Se puede comprar un cable paralelo en cualquier tienda de componentes informáticos. No obstante si no es posible comprarlo o simplemente queremos saber cómo hacerlo nosotros mismos, en la siguiente tabla mostramos como hacer un cable de impresora paralelo.

Tabla 29.1. Cableado de una conexión de cable paralelo para redes

Nombre-A	Extremo-A	Extremo-B	Descr.	Post/Bit
DATA0 -ERROR	2 15	15 2	Data	0/0x01 1/0x08
DATA1 +SLCT	3 13	13 3	Data	0/0x02 1/0x10
DATA2 +PE	4 12	12 4	Data	0/0x04 1/0x20
DATA3 -ACK	5 10	10 5	Strobe	0/0x08 1/0x40
DATA4 BUSY	6 11	11 6	Data	0/0x10 1/0x80
GND	18-25	18-25	GND	-

29.15.2. Configuración de PLIP

En primer lugar debemos tener en nuesras manos un cable “ laplink”. A continuación se debe comprobar que ambos sistemas poseen núcleos con soporte para el controlador lpt(4):

```
# grep lp /var/run/dmesg.boot
lpt0: <Printer> on ppbus0
lpt0: Interrupt-driven port
```

El puerto paralelo debe ser un puerto controlado por alguna “ irq”. En FreeBSD 4.X se debería tener un línea como la siguiente en el fichero de configuración del kernel:

```
device ppc0 at isa? irq 7
```

En FreeBSD 5.X el fichero `/boot/device.hints` debe contener las siguientes líneas:

```
hint.ppc.0.at="isa"
hint.ppc.0.irq="7"
```

A continuación se debe comprobar que el fichero de configuración del núcleo posee una línea con `device plip` o también puede comprobar si se ha cargado el módulo del núcleo `plip.ko`. Tanto en un caso como en el otro, cuando se ejecute ifconfig(8) debería aparecer el interfaz de red paralelo. En FreeBSD 4.X se muestra algo parecido a lo siguiente:

```
# ifconfig lp0
lp0: flags=8810<POINTOPOINT,SIMPLEX,MULTICAST> mtu 1500
```

y en FreeBSD 5.X:

```
# ifconfig plip0
plip0: flags=8810<POINTOPOINT,SIMPLEX,MULTICAST> mtu 1500
```

Nota

El nombre del dispositivo utilizado para la interfaz paralela es distinto en FreeBSD 4.X (`lpX`) y en FreeBSD 5.X (`plipX`).

Enchufe el cable “laplink” en los interfaces de ambos computadores.

Configure los parámetros de la interfaz de red en ambas máquinas como `root`. Por ejemplo, si queremos conectar la máquina `host1` ejecutando FreeBSD 4.X con la máquina `host2` que ejecuta FreeBSD 5.X:

```
                host1 <-----> host2
Dirección IP 10.0.0.1      10.0.0.2
```

Configure la interfaz de `host1` así:

```
# ifconfig lp0 10.0.0.1 10.0.0.2
```

Configure la interfaz de `host2` por medio de:

```
# ifconfig plip0 10.0.0.2 10.0.0.1
```

Tras esto debería disponerse de una conexión totalmente funcional. Por favor, consulte lp(4) y lpt(4) si quiere saber más.

Además se debe añadir ambas máquinas al fichero `/etc/hosts` :

```
127.0.0.1               localhost.mi.dominio localhost
10.0.0.1                host1.mi.dominio host1
10.0.0.2                host2.mi.dominio
```

Para comprobar que efectivamente la conexión funciona se puede probar a hacer un ping desde cada máquina. Por ejemplo en la máquina `host1`:

```
# ifconfig lp0
lp0: flags=8851<UP,POINTOPOINT,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet 10.0.0.1 --> 10.0.0.2 netmask 0xff000000
# netstat -r
Routing tables

Internet:
Destination        Gateway           Flags     Refs     Use      ↺
Netif Expire
host2              host1               UH          0       0       lp0
# ping -c 4 host2
PING host2 (10.0.0.2): 56 data bytes
64 bytes from 10.0.0.2: icmp_seq=0 ttl=255 time=2.774 ms
64 bytes from 10.0.0.2: icmp_seq=1 ttl=255 time=2.530 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=255 time=2.556 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=255 time=2.714 ms

--- host2 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 2.530/2.643/2.774/0.103 ms
```

29.16. IPv6

Texto original de Aaron Kaplan.
Reestructurado y ampliado por Tom Rhodes.

IPv6 (también conocido como IPng o “IP de nueva generación”) es la nueva versión del conocido protocolo de red IP, tambíen llamado IPv4. Como sucede con el resto de los sistemas *BSD FreeBSD proporciona una implementación de referencia que desarrolla el pro-

yecto japonés KAME. FreeBSD dispone de todo lo necesario para experimentar con el nuevo protocolo de red. Esta sección se centra en conseguir configurar y ejecutar correctamente el protocolo IPv6.

Al comienzo de los años 90 la gente comenzó a preocuparse por el rápido consumo del espacio de direcciones de IPv4. Dada la expansión actual de Internet existen dos preocupaciones principales:

- Agotamiento de las direcciones disponibles. Actualmente no se trata del principal problema debido al uso generalizado del del espacio de direccionamiento privado (`10.0.0.0/8` , `192.168.0.0/24` , etc.) junto con NAT.

- El número de entradas de las tablas de rutas comenzaba a ser imposible de manejar. Esto todavia es un problema prioritario.

IPv6 trata de resolver estos problemas y algunos más de la siguiente forma:

- IPv6 posee un espacio de direccionamiento de 128 bits. En otras palabras, en teoría existen 340,282,366,920,938,463,463,374,607,431,768,211,456 direcciones disponibles. Esto significa que existen aproximadamente 6.67 * 10^27 direcciones IPv6 por metro cuadrado disponibles para todo el planeta Tierra.

- Los "routers" sólo almacenan direcciones de red agregadas así que se reduce el número de entradas para cada tabla de rutas a un promedio de 8192.

Existen además muchas otras caracterísiticas interesantes que IPv6 proporciona, como:

- Autoconfiguración de direcciones (RFC2462)

- Direcciones anycast ("una-de-varias")

- Soporte de direcciones multicast predefinido

- IPsec (Seguridad en IP)

- Estructura de la cabecera simplificada

- IP móvil

- Mecanismos de traducción de IPv6 a IPv4 (y viceversa)

Si quiere saber más sobre IPv6 le recomendamos que consulte:

- Resumen de IPv6 en playground.sun.com

- KAME.net

- 6bone.net

29.16.1. Conceptos básicos sobre las direcciones IPv6

Existen varios tipos distintos de direcciones IPv6: Unicast, Anycast y Multicast.

Las direcciones unicast son direcciones bien conocidas. Un paquete que se envía a una dirección unicast deberín llega a la interfaz identificada por dicha dirección.

Las direcciones anycast son sintácticamente indistinguibles de las direcciones unicast pero sirven para identificar a un *conjunto* de interfaces. Un paquete destinado a una dirección anycast llega a la interfaz "más cercana" (en términos de métrica de "routers"). Las direcciones anycast sólo se pueden utilizar en "routers".

Las direcciones multicast identifican un grupo de interfaces. Un paquete destinado a una dirección multicast llega a todos los los interfaces que se encuentran agrupados bajo dicha dirección.

Nota

Las direcciones IPv4 de tipo broadcast (normalmente `xxx.xxx.xxx.255`) se expresan en IPv6 mediante direcciones multicast.

Tabla 29.2. Direcciones IPv6 reservadas

Dirección IPv6	Longitud del Prefijo (Bits)	Descripción	Notas
`::`	128 bits	sin especificar	como `0.0.0.0` en Pv4
`::1`	128 bits	dirección de bucle local (loopback)	como las `127.0.0.1` en IPv4
`::00:xx:xx:xx:xx`	96 bits	direcciónes IPv6 compatibles con IPv4	Los 32 bits más bajos contienen una dirección IPv4. También se denominan direcciones "empotradas."
`::ff:xx:xx:xx:xx`	96 bits	direcciones IPv6 mapeadas a IPv4	Los 32 bits más bajos contienen una dirección IPv4. Se usan para representar direcciones IPv4 mediante direcciones IPv6.

Dirección IPv6	Longitud del Prefijo (Bits)	Descripción	Notas
`fe80:: - feb::`	10 bits	direcciones link-local	equivalentes a la dirección de loopback de IPv4
`fec0:: - fef::`	10 bits	direcciones site-local	Equivalentes al direccionamiento privado de IPv4
`ff::`	8 bits	multicast	
`001` (base 2)	3 bits	direcciones unicast globales	Todas las direcciones IPv6 globales se asignan a partir de este espacio. Los primeros tres bits siempre son "001".

29.16.2. Lectura de las direcciones IPv6

La forma canónica que se utiliza para representar direcciones IPv6 es: `x:x:x:x:x:x:x:x` , donde cada "x" se considera un valor hexadecimal de 16 Bit. Por ejemplo `FEBC:A574:382B:23C1:AA49:4592:4EFE:9982`

A menudo una dirección posee alguna subcadena de varios ceros consecutivos de forma que se puede abreviar dicha cadena (sólo una vez, para evitar ambigúedades) mediante "::". También se pueden omitir los ceros a la ceros a la izquierda dentro de un valor "x". Por ejemplo `fe80::1` se corresponde con la forma canónica `fe80:0000:0000:0000:0000:0000:0000:0001` .

Una tercera forma de escribir direciones IPv6 es utilizando la ya tradicional notación decimal de IPv4 pero sólamente para los 32 bits más bajos de la dirección IPv6. Por ejemplo `2002::10.0.0.1` se correspondería con la representation hexadecimal canónica `2002:0000:0000:0000:0000:0000:0a00:0001` la cual es equivalente también a escribir `2002::a00:1` .

A estas alturas el lector debería ser capaz de comprender lo siguiente:

```
# ifconfig
```

```
rl0: flags=8943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX,MULTICAST> mtu ↺
1500
        inet 10.0.0.10 netmask 0xffffff00 broadcast 10.0.0.255
        inet6 fe80::200:21ff:fe03:8e1%rl0 prefixlen 64 scopeid 0x1
        ether 00:00:21:03:08:e1
        media: Ethernet autoselect (100baseTX )
        status: active
```

`fe80::200:21ff:fe03:8e1%rl0` es una dirección link-local autoconfigurada. Se construye a partir de la dirección MAC de la tarjeta de red.

Si quiere saber más sobre la estructura de las direcciones IPv6 puede consultar RFC3513.

29.16.3. Establecimiento de conectividad

Actualmente existen cuatro formas distintas de conectarse con otras máquinas y redes IPv6:

- Unirse a la red experimental denominada 6bone
- Obtener una red IPv6 a través de nuestro proveedor de acceso a Internet. Consulte a su proveedor de servicios para para más información.
- Encapsulación de IPv6 sobre IPv4 (RFC3068)
- Utilización del "port" net/freenet6 si se dispone de una de una conexión de marcación por modem.

Vamos a explicar cómo conectarse al 6bone ya que parece ser la forma más utilizada en la actualidad.

En primer lugar se recomienda consultar el sitio web de 6bone para saber cuál es la conexión del 6bone (físicamente) más próxima. Se debe escribir a la persona responsable de ese nodo y con un poco de suerte dicha persona responderá con con un conjunto de instrucciones y pasos a seguir para establecer la la conexión con ellos y a través de ellos con el resto de los nodos IPv6 que forman parte del 6bone. Normalmente esta conexión se establece usando túneles GRE (gif).

Veamos un ejemplo típico de configuración de un de un túnel gif(4):

```
# ifconfig gif0 create
# ifconfig gif0
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
# ifconfig gif0 tunnel MI_DIRECCIÓn_IPV4  SU_DIRECCIÓn_IPV4
# ifconfig gif0 inet6 alias DIRECCIÓn_DE-
SALIDA_IPv6_DEL_TÚNEL_ASIGNADO
```

Sustituya las palabras en mayúsculas por la información recibida del nodo 6bone al que nos queremos conectar.

La orden anterior establece el túnel. Compruebe que el túnel funciona correctamente mediante ping(8). Haga un ping6(8) a `ff02::1%gif0` . Deberíamos recibir recibir "dos" respuestas.

Nota

Para que el lector no se quede pensando en el significado significado de la dirección `ff02:1%gif0` le podemos decir que se trata de de una dirección IPv6 multicast de tipo link-local. `%gif0` no forma parte del protocolo IPv6 como tal sino que se trata de un detalle de implementación relacionado con las direcciones link-local y se añade para especificar la interfaz de salida que se debe utilizar para enviar los paquetes de ping6(8). Como estamos haciendo ping a una dirección multicast a la que se unen todos los interfaces pertenecientes al mismo enlace debería responder al ping tanto nuestro propio interfaz como el interfaz remoto.

A continuación se configura la ruta por defecto hacia nuestro enlace 6bone; observe que es muy semejante a lo que hay que hacer en IPv4:

```
# route add -inet6 default -interface gif0
# ping6 -n MI_UPLINK
```

```
# traceroute6 www.jp.FreeBSD.org
(3ffe:505:2008:1:2a0:24ff:fe57:e561) from 3ffe:8060:100::40:2, 30 ↺
hops max, 12 byte packets
     1  atnet-meta6  14.147 ms  15.499 ms  24.319 ms
     2  6bone-gw2-ATNET-NT.ipv6.tilab.com  103.408 ms  95.072 ms *
     3  3ffe:1831:0:ffff::4  138.645 ms  134.437 ms  144.257 ms
     4  3ffe:1810:0:6:290:27ff:fe79:7677  282.975 ms  278.666 ms  ↺
292.811 ms
     5  3ffe:1800:0:ff00::4  400.131 ms  396.324 ms  394.769 ms
     6  3ffe:1800:0:3:290:27ff:fe14:cdee  394.712 ms  397.19 ms  ↺
394.102 ms
```

Esta captura de pantalla variará dependiendo de la localización de la máquina. Tras seguir estos pasos deberíamos poder alcanzar el sitio IPv6 de www.kame.net y ver la tortuga bailarina, que es una imagen animada que sólo se muestra cuando se accede al servidor web utilizando el protocolo IPv6 (para ellos se encesita utilizar un navegador web que soporte IPv6, IPv6, por ejemplo www/mozilla o Konqueror, que forma parte de x11/kdebase3, o también con www/epiphany.

29.16.4. DNS en el mundo IPv6

Existen dos tipos de registros de DNS para IPv6. No obstante el IETF ha declarado los registros A6 y CNAME como registros para uso experimental. Los registros de tipo AAAA son los únicos estandar a día de hoy.

La utilización de registros de tipo AAAA es muy sencilla. Se asocia el nombre de la máquina con la dirección IPv6 de la siguiente forma:

```
NOMBREDEMIMÁQUINA AAAA   MIDIRECCIÓNIPv6
```

De igual forma que en IPv4 se utilizan los registros de tipo A. En caso de no poder administrar su propia zona de DNS se puede pedir esta configuración a su proveedor de servicios. Las versiones actuales de bind (versiones 8.3 y 9) y el "port" dns/djbdns (con el parche de IPv6 correspondiente) soportan los registros de tipo AAAA.

29.17. ATM en FreeBSD 5.X

Escrito por Harti Brandt.

29.17.1. Configuración de IP clásico sobre ATM (PVCs)

IP clásico sobre ATM (CLIP) es el método más sencillo de utilizar ATM con IP. Se puede utilizar con conexiones conmutadas (SVC) y con conexiones permanentes (PVCs). En esta sección se describe cómo configurar una red basada en PVCs.

29.17.1.1. Configuraciones en red mallada completa

El primer método para configurar CLIP con PVCs consiste en conectar unas máquinas con otras mediante circuitos PVC dedicados. Aunque la configuración parece sencilla llega a resultar imposible de manejar cuando se posee un número grande de máquinas. El ejemplo que se muestra a continuación supone que nuestra red posee cuatro máquinas y que cada una se conecta a la red ATM mediante una tarjeta de red ATM. El primer paso consiste en planificar las direcciones IP y las conexiones ATM que se van a configurar en las máquinas.

Máquina	Dirección IP
hostA	192.168.173.1
hostB	192.168.173.2
hostC	192.168.173.3
hostD	192.168.173.4

Para construir una red completamente mallada necesitamos una conexión ATM entre cada par de máquinas:

Máquinas	Pareja VPI.VCI
hostA - hostB	0.100
hostA - hostC	0.101

Máquinas	Pareja VPI.VCI
hostA - hostD	0.102
hostB - hostC	0.103
hostB - hostD	0.104
hostC - hostD	0.105

Los valores VPI y VCI en cada extremo de la conexión pueden ser diferentes pero por simplicidad suponemos que son iguales. A continuación necesitamos configurar las interfaces ATM en cada máquina:

```
hostA# ifconfig hatm0 192.168.173.1 up
hostB# ifconfig hatm0 192.168.173.2 up
hostC# ifconfig hatm0 192.168.173.3 up
hostD# ifconfig hatm0 192.168.173.4 up
```

Suponiendo que la interfaz ATM es `hatm0` en todas las máquinas. Ahora necesitamos configurar los PVCs en las máquinas (suponemos que ya se han configurado de forma correcta en el "switch" ATM, para lo cual puede ser necesario consultar el manual del "switch").

```
hostA# atmconfig natm add 192.168.173.2 hatm0 0 100 llc/snap ubr
hostA# atmconfig natm add 192.168.173.3 hatm0 0 101 llc/snap ubr
hostA# atmconfig natm add 192.168.173.4 hatm0 0 102 llc/snap ubr

hostB# atmconfig natm add 192.168.173.1 hatm0 0 100 llc/snap ubr
hostB# atmconfig natm add 192.168.173.3 hatm0 0 103 llc/snap ubr
hostB# atmconfig natm add 192.168.173.4 hatm0 0 104 llc/snap ubr

hostC# atmconfig natm add 192.168.173.1 hatm0 0 101 llc/snap ubr
hostC# atmconfig natm add 192.168.173.2 hatm0 0 103 llc/snap ubr
hostC# atmconfig natm add 192.168.173.4 hatm0 0 105 llc/snap ubr

hostD# atmconfig natm add 192.168.173.1 hatm0 0 102 llc/snap ubr
hostD# atmconfig natm add 192.168.173.2 hatm0 0 104 llc/snap ubr
hostD# atmconfig natm add 192.168.173.3 hatm0 0 105 llc/snap ubr
```

Por supuesto que se pueden utilizar otras especificaciones de tráfico siempre y cuando las tarjetas de red las soporten. En este caso la especificación del tipo de tráfico se completa con los parámetros del tráfico. Puede acceder a la ayuda de atmconfig(8) así:

```
# atmconfig help natm add
```

y por supuesto en la página de manual de atmconfig(8).

Se puede crear la misma configuración utilizando el fichero `/etc/rc.conf` . Para la máquina `hostA` sería algo así:

```
network_interfaces="lo0 hatm0"
ifconfig_hatm0="inet 192.168.173.1 up"
```

```
natm_static_routes="hostB hostC hostD"
route_hostB="192.168.173.2 hatm0 0 100 llc/snap ubr"
route_hostC="192.168.173.3 hatm0 0 101 llc/snap ubr"
route_hostD="192.168.173.4 hatm0 0 102 llc/snap ubr"
```

El estado de todas las rutas CLIP se puede obtener en todo momento con:

```
hostA# atmconfig natm show
```

Parte V. Apéndices

Tabla de contenidos

30. Cómo obtener FreeBSD 777
30.1. Servidores FTP 777
30.2. Uso de CVSup 777
31. Bibliografía 779
31.1. Libros y revistas específicas sobre FreeBSD 779
31.2. Guías de usuario 779
31.3. Guías de administrador 780
31.4. Guías de programadores 780
31.5. El sistema operativo por dentro 781
31.6. Referencia de seguridad 782
31.7. Referencia de hardware 782
31.8. Historia de UNIX 782
31.9. Diarios y revistas 783
A. Recursos en Internet 785
A.1. Listas de correo 785
A.2. Grupos de noticias de Usenet 797
A.3. Servidores WWW 798
A.4. Direcciones de correo electrónico 801
A.5. Cuentas shell 801
32. PGP keys 803
32.1. Responsables 803
32.2. Miembros del Core Team 804
Índice 813

Capítulo 30. Cómo obtener FreeBSD

Cómo obtener FreeBSD

30.1. Servidores FTP

30.2. Uso de CVSup

30.2.1. Instalación

30.2.2. Servidores

30.2.2.1. El fichero `refuse`

Capítulo 31. Bibliografía

Aunque las páginas del manual proveen la referencia definitiva para partes individuales del sistema operativo FreeBSD, son notorias por no ilustrar como poner todas las piezas juntas para hacer que todo el sistema operativo funcione fácilmente. Debido a esto, no hay sustituto para un buen libro de administración de sistemas UNIX y un buen manual de usuario.

31.1. Libros y revistas específicas sobre FreeBSD

Libros y revistas internacionales:

- Usando FreeBSD (en Chino).
- FreeBSD for PC 98'ers (en japonés), publicado por SHUWA System Co, LTD. ISBN 4-87966-468-5 C3055 P2900E.
- FreeBSD (en japonés), publicado por CUTT. ISBN 4-906391-22-2 C3055 P2400E.
- Introducción completa a FreeBSD (en japonés), publicado por Shoeisha Co., Ltd. ISBN 4-88135-473-6 P3600E.
- Kit personal del principiante UNIX FreeBSD (en japonés), publicado por ASCII. ISBN 4-7561-1733-3 P3000E.
- Manual FreeBSD (traducción del japonés), publicado por ASCII. ISBN 4-7561-1580-2 P3800E.
- FreeBSD mit Methode (en alemán), publicado por Computer und Literatur Verlag/Vertrieb Hanser, 1998. ISBN 3-932311-31-0.
- Manual de instalación y utilización de FreeBSD (en japonés), publicado por Mainichi Communications Inc..

Libros y revistas en inglés:

- The Complete FreeBSD, publicado por Walnut Creek CDROM.

31.2. Guías de usuario

- Computer Systems Research Group, UC Berkeley. *4.4BSD User's Reference Manual*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-075-9
- Computer Systems Research Group, UC Berkeley. *4.4BSD User's Supplementary Documents*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-076-7

- *UNIX in a Nutshell*. O'Reilly & Associates, Inc., 1990. ISBN 093717520X
- Mui, Linda. *What You Need To Know When You Can't Find Your UNIX System Administrator*. O'Reilly & Associates, Inc., 1995. ISBN 1-56592-104-6
- La Ohio State University ha escrito un Curso de introducción a UNIX disponible en línea en formato HTML y postscript.
- Jpman Project, Japan FreeBSD Users Group. FreeBSD User's Reference Manual (traducción japonesa). Mainichi Communications Inc., 1998. ISBN4-8399-0088-4 P3800E.

31.3. Guías de administrador

- Albitz, Paul and Liu, Cricket. *DNS and BIND*, 2nd Ed. O'Reilly & Associates, Inc., 1997. ISBN 1-56592-236-0
- Computer Systems Research Group, UC Berkeley. *4.4BSD System Manager's Manual*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-080-5
- Costales, Brian, et al. *Sendmail*, 2nd Ed. O'Reilly & Associates, Inc., 1997. ISBN 1-56592-222-0
- Frisch, Æleen. *Essential System Administration*, 2nd Ed. O'Reilly & Associates, Inc., 1995. ISBN 1-56592-127-5
- Hunt, Craig. *TCP/IP Network Administration*. O'Reilly & Associates, Inc., 1992. ISBN 0-937175-82-X
- Nemeth, Evi. *UNIX System Administration Handbook*. 2nd Ed. Prentice Hall, 1995. ISBN 0131510517
- Stern, Hal *Managing NFS and NIS* O'Reilly & Associates, Inc., 1991. ISBN 0-937175-75-7
- Jpman Project, Japan FreeBSD Users Group. FreeBSD System Administrator's Manual (traducción japonesa). Mainichi Communications Inc., 1998. ISBN4-8399-0109-0 P3300E.

31.4. Guías de programadores

- Asente, Paul. *X Window System Toolkit*. Digital Press. ISBN 1-55558-051-3
- Computer Systems Research Group, UC Berkeley. *4.4BSD Programmer's Reference Manual*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-078-3
- Computer Systems Research Group, UC Berkeley. *4.4BSD Programmer's Supplementary Documents*. O'Reilly & Associates, Inc., 1994. ISBN 1-56592-079-1

- Harbison, Samuel P. and Steele, Guy L. Jr. *C: A Reference Manual*. 4rd ed. Prentice Hall, 1995. ISBN 0-13-326224-3
- Kernighan, Brian and Dennis M. Ritchie. *The C Programming Language.*. PTR Prentice Hall, 1988. ISBN 0-13-110362-9
- Lehey, Greg. *Porting UNIX Software*. O'Reilly & Associates, Inc., 1995. ISBN 1-56592-126-7
- Plauger, P. J. *The Standard C Library*. Prentice Hall, 1992. ISBN 0-13-131509-9
- Stevens, W. Richard. *Advanced Programming in the UNIX Environment*. Reading, Mass. : Addison-Wesley, 1992 ISBN 0-201-56317-7
- Stevens, W. Richard. *UNIX Network Programming*. 2nd Ed, PTR Prentice Hall, 1998. ISBN 0-13-490012-X
- Wells, Bill. "Writing Serial Drivers for UNIX". *Dr. Dobb's Journal*. 19(15), December 1994. pp68-71, 97-99.

31.5. El sistema operativo por dentro

- Andleigh, Prabhat K. *UNIX System Architecture*. Prentice-Hall, Inc., 1990. ISBN 0-13-949843-5
- Jolitz, William. "Porting UNIX to the 386". *Dr. Dobb's Journal*. January 1991-July 1992.
- Leffler, Samuel J., Marshall Kirk McKusick, Michael J Karels and John Quarterman *The Design and Implementation of the 4.3BSD UNIX Operating System*. Reading, Mass. : Addison-Wesley, 1989. ISBN 0-201-06196-1
- Leffler, Samuel J., Marshall Kirk McKusick, *The Design and Implementation of the 4.3BSD UNIX Operating System: Answer Book*. Reading, Mass. : Addison-Wesley, 1991. ISBN 0-201-54629-9
- McKusick, Marshall Kirk, Keith Bostic, Michael J Karels, and John Quarterman. *The Design and Implementation of the 4.4BSD Operating System*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-54979-4
- Stevens, W. Richard. *TCP/IP Illustrated, Volume 1: The Protocols*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-63346-9
- Schimmel, Curt. *Unix Systems for Modern Architectures*. Reading, Mass. : Addison-Wesley, 1994. ISBN 0-201-63338-8
- Stevens, W. Richard. *TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP and the UNIX Domain Protocols*. Reading, Mass. : Addison-Wesley, 1996. ISBN 0-201-63495-3

- Vahalia, Uresh. *UNIX Internals -- The New Frontiers*. Prentice Hall, 1996. ISBN 0-13-101908-2
- Wright, Gary R. and W. Richard Stevens. *TCP/IP Illustrated, Volume 2: The Implementation*. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-63354-X

31.6. Referencia de seguridad

- Cheswick, William R. and Steven M. Bellovin. *Firewalls and Internet Security: Repelling the Wily Hacker*. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-63357-4
- Garfinkel, Simson and Gene Spafford. *Practical UNIX Security*. 2nd Ed. O'Reilly & Associates, Inc., 1996. ISBN 1-56592-148-8
- Garfinkel, Simson. *PGP Pretty Good Privacy* O'Reilly & Associates, Inc., 1995. ISBN 1-56592-098-8

31.7. Referencia de hardware

- Anderson, Don and Tom Shanley. *Pentium Processor System Architecture*. 2nd Ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40992-5
- Ferraro, Richard F. *Programmer's Guide to the EGA, VGA, and Super VGA Cards*. 3rd ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-62490-7
- La corporación Intel publica documentación sobre sus CPUs, chipsets y estándares en su web para desarrolladores, normalmente en archivos con formato PDF.
- Shanley, Tom. *80486 System Architecture*. 3rd ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40994-1
- Shanley, Tom. *ISA System Architecture*. 3rd ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40996-8
- Shanley, Tom. *PCI System Architecture*. 3rd ed. Reading, Mass. : Addison-Wesley, 1995. ISBN 0-201-40993-3
- Van Gilluwe, Frank. *The Undocumented PC*. Reading, Mass: Addison-Wesley Pub. Co., 1994. ISBN 0-201-62277-7

31.8. Historia de UNIX

- Lion, John *Lion's Commentary on UNIX, 6th Ed. With Source Code*. ITP Media Group, 1996. ISBN 1573980137

- Raymond, Eric S. *The New Hacker's Dictionary, 3rd edition*. MIT Press, 1996. ISBN 0-262-68092-0. Also known as the Jargon File
- Salus, Peter H. *A quarter century of UNIX*. Addison-Wesley Publishing Company, Inc., 1994. ISBN 0-201-54777-5
- Simon Garfinkel, Daniel Weise, Steven Strassmann. *The UNIX-HATERS Handbook*. IDG Books Worldwide, Inc., 1994. ISBN 1-56884-203-1
- Don Libes, Sandy Ressler *Life with UNIX* — special edition. Prentice-Hall, Inc., 1989. ISBN 0-13-536657-7
- *The BSD family tree*. 1997. ftp://ftp.FreeBSD.org/pub/FreeBSD/FreeBSD-current/src/share/misc/bsd-family-tree o local on a FreeBSD-current machine.
- *The BSD Release Announcements collection*. 1997. http://www.de.FreeBSD.org/de/ftp/releases/
- *Networked Computer Science Technical Reports Library*. http://www.ncstrl.org/
- *Antiguas releases BSD procedentes del Computer Systems Research Group (CSRG)*. http://www.mckusick.com/csrg/: El paquete de 4 CDs cubre todas las versiones de BSD desde la 1BSD hasta la 4.4BSD y 4.4BSD-Lite2 (pero no la 2.11BSD, desafortunadamente). El último disco contiene el código fuente final y los archivos SCCS.

31.9. Diarios y revistas

- *The C/C++ Users Journal*. R&D Publications Inc. ISSN 1075-2838
- *Sys Admin — The Journal for UNIX System Administrators* Miller Freeman, Inc., ISSN 1061-2688

Apéndice A. Recursos en Internet

La velocidad del desarrollo de FreeBSD hace imposible el uso de medios impresos como forma de seguir los últimos desarrollos. Los recursos electrónicos son la mejor, y con frecuencia la única, manera de estar informados de los últimos avances. Dado que FreeBSD es sacado adelante mediante el trabajo de voluntarios la propia comunidad de usuarios suele ejercer las funciones de lo que sería un "departamento de soporte técnico", siendo el correo electrónico y USENET la manera más efectiva de contactar con esa comunidad.

Las formas de contacto con la comunidad de usuarios de FreeBSD están detalladas a continuación. Si usted sabe de algún otro medio que no figure aquí envíelo por favor a lista de correo del proyecto de documentación de FreeBSDpara que pueda ser incluída.

A.1. Listas de correo

Aunque la práctica totalidad de los desarrolladores de FreeBSD léen USENET no podemos garantizar de modo rotundo que recibiremos sus dudas rápidamente (o siquiera que las recibamos) si usted las envía a uno de los grupos de `comp.unix.bsd.freebsd.*` . Enviando sus dudas a la lista de correo apropiada cumplirá dos objetivos, llegar a los desarrolladores y a una audiencia específica, lo que le asegurará la mejor (o al menos la más rápida) respuesta.

Las normas de las diversas listas están al principio de éste documento. *Por favor, léa las normas antes de suscribirse o enviar correo a ninguna lista.* Muchos suscriptores de nuestras listas reciben varios cientos de mensajes relacionados con FreeBSD cada día y estableciendo las normas de uso de las listas intentamos mantener alto el interés de los mensajes que en ella circulan. Bajarlo haría fallar a las listas de correo como un medio de comunicación efectivo para el proyecto.

Todas las listas de correo son archivadas y se pueden hacer búsquedas en ellas desde el servidor WWW de FreeBSD usando éste enlace. El archivo ofrece la posibilidad de usar palabras clave, lo que lo convierte en una excelente manera de buscar respuestas a preguntas frecuentes y debería ser consultado antes de enviar ninguna duda.

A.1.1. Índice de listas

*Listas generales:*Las siguientes son listas generales de suscripción libre (y muy recomendable):

Lista	Propósito
cvs-all	Cambios realizados en el árbol de código de FreeBSD
freebsd-advocacy	Proselitismo de FreeBSD
freebsd-announce	Sucesos importantes e hitos del proyecto
freebsd-arch	Debates de arquitectura y diseños
freebsd-bugs	Informes de errores
freebsd-chat	Temas no técnicos relacionados con la comunidad FreeBSD.
freebsd-config	Desarrollo de herramientas de instalación y configuración de FreeBSD
freebsd-current	Debates acerca del uso de FreeBSD-current
freebsd-isp	Consultas de Proveedores de Servicios de Internet que usan FreeBSD
freebsd-jobs	Oportunidades de trabajo y consultoría bajo FreeBSD
freebsd-newbies	Actividades y discusiones de nuevos usuarios de FreeBSD
freebsd-policy	Decisiones estratégicas del Core Team de FreeBSD. Bajo volumen y sólo lectura
freebsd-questions	Preguntas de usuarios y soporte técnico
freebsd-stable	Debates acerca del uso de FreeBSD-stable
freebsd-test	Un sitio al que mandar sus mensajes de prueba en lugar de a una de las demás listas

Listas Técnicas: Las siguientes listas son para debates técnicos. Debería leer cuidadosamente las normas de cada lista antes de suscribirse o enviar correos, dado que hay normas estrictas en cuanto a su uso y contenidos.

Lista	Propósito
freebsd-afs	Porte de AFS a FreeBSD
freebsd-alpha	Porte FreeBSD a Alpha
freebsd-arm	Porte de FreeBSD para procesadores ARM
freebsd-atm	Uso de redes ATM con FreeBSD
freebsd-audit	Proyecto de auditoría del código fuente

Lista	Propósito
freebsd-binup	Diseño y desarrollo del sistema de actualización binaria
freebsd-cluster	Uso de FreeBSD en entornos cluster
freebsd-database	Debates sobre uso de bases de datos y su desarrollo bajo FreeBSD
freebsd-doc	Creación de documentación sobre FreeBSD
freebsd-emulation	Emulación de otros sistemas como Linux/DOS/Windows
freebsd-firewire	Debates técnicos sobre Firewire (iLink, IEEE 1394)
freebsd-fs	Sistemas de ficheros
freebsd-hackers	Debates técnicos generales
freebsd-hardware	Debates generales sobre harware y su uso en FreeBSD
freebsd-i18n	Internacionalización de FreeBSD
freebsd-ia64	Porte de FreeBSD a los próximos sistemas IA64 de Intel
freebsd-ipfw	Debates técnicos sobre el rediseño del código del cortafuegos IP
freebsd-isdn	Desarrolladores de RDSI
freebsd-java	Desarrolladores de Java y personas portando los JDK a FreeBSD
freebsd-libh	La segunda generación del sistema de instalación y paquetes
freebsd-mobile	Debates sobre equipos portátiles
freebsd-mozilla	Porte de mozilla a FreeBSD
freebsd-multimedia	Aplicaciones multimedia
freebsd-new-bus	Debates técnicos sobre la arquitectura de bus
freebsd-net	Debates sobre el código fuente de Redes y TCP/IP
freebsd-platforms	Específica sobre plataformas de arquitectura no Intel
freebsd-ports	Debates sobre la colección de ports

Lista	Propósito
freebsd-ppc	Porte de FreeBSD a PowerPC
freebsd-qa	Debates sobre Control de Calidad, generalmente al salir una nueva release
freebsd-realtime	Desarrollo de extensiones en tiempo real en FreeBSD
freebsd-scsi	El subsistema SCSI
freebsd-security	Temas de seguridad
freebsd-security-notifications	Avisos de seguridad
freebsd-small	Uso de FreeBSD en aplicaciones embebidas
freebsd-smp	Debates sobre diseño de Multiproceso [A]Simétrico
freebsd-sparc	Porte de FreeBSD a sistemas Sparc
freebsd-standards	Cumplimiento de las normas C99 y POSIX en FreeBSD
freebsd-tokenring	Soporte de Token Ring en FreeBSD

*Listas limitadas:*Las siguientes listas son para una audiencia más especializada (e interesada)y probablemente no son de interés para el público en general. Es una buena idea tener una presencia estable en las listas técnicas antes de suscribirse a alguna de las limitadas, de modo que se pueda entender la etiqueta de la comunicación que en ellas se usa.

Lista	Propósito
freebsd-core	FreeBSD Core Team
freebsd-hubs	Mantenimiento de mirrors (mantenimiento de infraestructuras)
freebsd-install	Desarrollo de la Instalación
freebsd-user-groups	Coordinación de grupos de usuarios
freebsd-www	Mantenimiento de www.FreeBSD.org

*Listas Compendio:*La mayoría de las listas citadas son accesibles como compendio. Los nuevos mensajes enviados a la lista son guardados y enviados como un único correo cuando el archivo llega a un tamaño cercano a los 100 Kb. Las listas accesibles como compendio son:

Lista
freebsd-afs-digest

Lista
freebsd-alpha-digest
freebsd-chat-digest
freebsd-current-digest
freebsd-cvs-all-digest
freebsd-database-digest
freebsd-hackers-digest
freebsd-ia64-digest
freebsd-isdn-digest
freebsd-java-digest
freebsd-questions-digest
freebsd-security-digest
freebsd-sparc-digest
freebsd-stable-digest
freebsd-test-digest

*Listas CVS:*Las siguientes listas son para gente interesada en llevar un seguimiento de los mensajes en el registro para conocer los cambios hechos en las diferentes áreas del árbol de código fuente. Son listas *de sólo lectura* y no se debe enviar correo a ellas.

Lista	Área de código	Descripción de área de código (código fuente)
cvs-all	/usr/src	Todos los cambios al árbol de código (superconjunto)

A.1.2. Cómo suscribirse

Todas las listas de correo están en `FreeBSD.org` , de manera que para enviar correo a la lista "nombredelista" simplemente hay que escribir a <*`nombredelista`* @FreeBSD.org>. Desde ahí será redistribuído a los miembros de la lista de correo a lo largo y ancho del mundo.

Para suscribirse a una lista envíe un correo a <`majordomo@FreeBSD.org` > incluyendo

```
subscribe <listname> [<optional address>]
```

en el cuerpo del mensaje. Por ejemplo, para suscribirse a `freebsd-announce` usted haría esto:

```
% mail majordomo@FreeBSD.org
```

```
subscribe freebsd-announce
^D
```

Si quisiera suscribirse bajo otro nombre o enviar una petición de suscripción para una lista de correo local (un sistema muy eficiente si dispone de varias personas interesadas que tengan cuentas de correo en un mismo servidor ¡esto nos facilita mucho el trabajo!) ésto es lo que debe hacer:

```
% mail majordomo@FreeBSD.org
subscribe freebsd-announce local-announce@ejemplo.com
^D
```

Por último, también es posible desuscribirse de una lista, obtener una lista de los suscriptores de una lista u obtener una lista de las listas de correo disponibles enviando otro tipo de mensajes de control a majordomo. Para obtener una lista completa de las órdenes disponibles haga esto:

```
% mail majordomo@FreeBSD.org
help
^D
```

De nuevo quisiéramos pedirle que procure mantener los debates de las listas técnicas dentro de temas técnicos. Si lo único que usted quiere es recibir avisos importantes le sugerimos que se suscriba a freebsd-announce, que está pensada para tener un tráfico muy bajo.

A.1.3. Normas de las listas

Todas las listas de correo de FreeBSD tienen ciertas normas elementales que han de ser respetadas por cualquiera que las use. Quien no se atenga a ellas recibirá hasta dos (2) advertencias escritas del Postmaster de FreeBSD <postmaster@FreeBSD.org >, después de las cuales, a la tercera falta, el suscriptor será borrado de todas las listas de correo de FreeBSD y filtrado para evitar futuros envíos. Lamentamos que esas normas y medidas sean necesarias, pero la Internet de hoy es, según parece, un entorno bastante conflictivo, y mucha gente no se da cuenta de cuán frágiles son algunos de sus mecanismos.

Normas a respetar:

- El tema de cualquier envío debe atenerse al fin básico de la lista a la que se escribe, esto es, si la lista es sobre temas de debate técnico sus envíos deberían versar sobre temas técnicos. Enviar mensajes irrelevantes o insultos sólo sirve para deteriorar el valor de la lista de correo para sus miembros y no será tolerado. Para discusiones libres sin un tema en particular está la lista de correo freebsd-chat <freebsd-chat@FreeBSD.org >, que es libremente accesible y hecha para éste propósito.

- No se debería enviar el mismo mensaje a más de dos listas, y sólamente a 2 cuando exista una necesidad manifiesta de escribir a ambas listas. Hay una gran cantidad de personas suscritas a más de una lista y excepto para las mezclas más esotéricas (digamos "-stable

& -scsi") no hay razón para enviar un mensaje a más de una lista al mismo tiempo. Si le envían un mensaje en el que aparecen múltiples listas de correo en la línea "Cc" de la cabecera, dicha línea debe ser recortada antes de que envíe una respuesta. *Usted es el* responsable *de sus propios envíos cruzados, independientemente de quién fuese el remitente original.*

- No están admitidos los ataques personales ni la blasfemia (dentro del contexto o como argumento) y eso incluye tanto a usuarios como a desarrolladores. Violaciones graves de la netiqueta, como reenviar o extractar mensajes privados sin permiso ni visos de tenerlo, está mal visto, aunque no prohibido específicamente. Sin embargo, hay pocos casos en los que algo así encaje en la temática de una lista, por lo cual lo más probable es recibir una advertencia (o ser expulsado) tan sólo a causa de ello.

- El anuncio de productos o servicios no relacionados con FreeBSD están estrictamente prohibidas y conllevarán la inmediata expulsión de la lista si queda demostrado que el autor está practicando el “spam” o envío de correo no solicitado.

Normas de las listas individuales:

FREEBSD-AFS
: *Sistema de Ficheros Andrew*

  Ésta lista es para debates sobre el porte y uso de AFS, de CMU/Transarc

FREEBSD-ANNOUNCE
: *Sucesos importantes / hitos*

  Ésta es la lista de correo para gente interesada en recibir exclusivamente avisos de sucesos importantes dentro de FreeBSD. Ésto incluye anuncios sobre SNAPSHOTS y otras versiones. Puede incluír también peticiones de voluntarios, etc. Es una lista de bajo volumen y estrictamente moderada.

FREEBSD-ARCH
: *Debates sobre arquitectura y diseño*

  Ésta lista es para debates sobre la arquitectura de FreeBSD. Los mensajes deberían mantenerse dentro del ámbito técnico para el que fué creada la lista. Serían ejemplos de temas aptos para ésta lista:

  - Como reorganizar el sistema de construcción ("build") para poder tener varios procesos de construcción personalizados funcionando simultáneamente.

  - Qué se necesita arreglar en el VFS para que funcionen las capas de Heidemann.

  - Cómo cambiar el dispositivo de control de interfaces para que sea posible utilizar los mismos controladores directamente en la mayoría de los buses y arquitecturas.

- Cómo escribir un controlador de red.

FREEBSD-AUDIT
Proyecto de auditoría del código fuente

Ésta es la lista de correo del proyecto de auditoría del código fuente de FreeBSD. Aunque en principio fué puesta en marcha para cambios motivados por la seguridad su ámbito fué ampliado a la revisión de cualquier cambio en el código.

En ésta lista circula una gran cantidad de parches y probablemente no sea de interés para el típico usuario de FreeBSD. Las discusiones de seguridad que no estén relacionadas con una parte específica del código deben tener lugar en freebsd-security. Por otra parte se ruega a todos los desarrolladores que envíen sus parches a ésta lista para su revisión, especialmente si atañen a una parte del sistema donde un error pudiera afectar seriamente a la integridad del sistema.

FREEBSD-BINUP
Proyecto de Actualización Binaria de FreeBSD

Ésta lista existe para facilitar el debate sobre el sistema de actualizacion binaria o binup. Características de diseño, detalles de implementación, parches, informes de error, informes de estado, peticiones de características, "commit logs" y en general todo lo relacionado con binup es bienvenido.

FREEBSD-BUGS
Informe de errores

Ésta es la lista de correo para informar de errores en FreeBSD. Siempre que sea posible los errores deberían ser enviados mediante send-pr(1) o el interfaz WEB

FREEBSD-CHAT
Temas no técnicos relacionados con la comunidad FreeBSD

Ésta lista contiene todos los mensajes sobre información no técnica y social, contenidos que no tienen cabida en las demás listas. Eso incluye discusiones sobre si Julio Iglesias parece una gárgola, sobre si escribir o no en mayúsculas, quién está bebiendo demasiado café, dónde se elabora la mejor cerveza, quién está fabricando cerveza en su sótano y así sucesivamente. Pueden hacerse anuncios sobre actos importantes (como próximas fiestas, congresos, bodas, nacimientos, nuevos trabajos, etc.), pero las respuestas deben ser dirigidas a ésta misma lista.

FREEBSD-CORE
FreeBSD Core Team

Ésta es la lista de correo interna para uso de los miembros del Core Team. Los mensajes pueden ser enviados a ésta lista cuando un problema serio relacionado con FreeBSD necesite un estudio o arbitraje de alto nivel.

FREEBSD-CURRENT
Debates sobre el uso de FreeBSD-current

Ésta es la lista de correo para usuarios de freebsd-current. Esto incluye advertencias sobre nuevas características a ser incluídas en -current que afecten a todos los usuarios e instrucciones paso por paso que deben ser seguidas para mantener una instalación -current. Cualquier usuario de "current" debería suscribirse a ésta lista. Ésta es una lista de correo técnica en la que se esperan contenidos estrictamente técnicos.

FREEBSD-CURRENT-DIGEST
Debates sobre el uso de FreeBSD-current

Éste es el compendio de la lista freebsd-current. Consiste en que todos los mensajes enviados a freebsd-current son empaquetados y enviados periódicamente como un solo mensaje. Ésta lista es de *Sólo-Lectura* y no debería recibir correo.

FREEBSD-DOC
Proyecto de Documentación

Ésta lista de correo está destinada a discusiones relacionadas con cuestiones y proyectos relacionados con la creación de documentación de FreeBSD. Los miembros de ésta lista son llamados "El Proyecto de Documentación de FreeBSD". La lista es abierta; ¡suscríbase y contribuya!.

FREEBSD-FIREWIRE
Firewire (iLink, IEEE 1394)

Ésta lista de correo es para debates sobre diseño e implementación del subsistema Firewire(también conocido como IEEE o iLink) en FreeBSD. Los temas incluyen de modo específico los "standards", dispositivos de bus y sus protocolos, adaptación de placas base, tarjetas y chips y la arquitectura e implementación de código para soporte nativo.

FREEBSD-FS
Sistemas de ficheros

Debates acerca del sistema de ficheros de FreeBSD. Ésta es una lista de correo técnica en la que se espera un contenido estríctamente técnico.

FREEBSD-GNOME
GNOME

Debates acerca del Entorno de Escritorio GNOME para sistemas de ficheros FreeBSD. Ésta es una lista de correo técnica en la que se espera un contenido estrictamente técnico.

FREEBSD-IPFW
Cortafuegos IP

Éste es el foro de discusión técnica dedicado al rediseño del código del cortafuegos IP de FreeBSD. Ésta es una lista de correo técnica en la que se espera un contenido exclusivamente técnico.

FREEBSD-IA64
Porte de FreeBSD a IA64

Ésta es una lista de correo técnica para personas que están trabajando en el porte de FreeBSD a la plataforma IA-64 de Intel, para intercambiar problemas y soluciones alternativas. Cualquier persona interesada en seguir las discusiones técnicas es bienvenida.

FREEBSD-ISDN
Comunicaciones RDSI

Ésta es la lista de correo para quienes participan en el desarrollo del soporte RDSI para FreeBSD.

FREEBSD-JAVA
Desarrollo Java

Ésta es la lista de correo sobre el desarrollo de aplicaciones Java importantes para FreeBSD y el porte y mantenimiento de los JDK.

FREEBSD-HACKERS
Debates técnicos

Éste es un foro de debate técnico relacionado con FreeBSD. Ésta es la lista de correo técnica primaria. Es para personas que están trabajando en FreeBSD, solucionando problemas o para discutir soluciones alternativas. Las personas interesadas en seguir las discusiones técnicas también son bienvenidas. Ésta es una lista de correo técnica en la cual se espera un contenido estrictamente técnico.

FREEBSD-HACKERS-DIGEST
Technical discussions

Éste es el compendio de la lista de correo freebsd-hackers. Consiste en que todo el correo enviado a freebsd-hackers es empaquetado y enviado en un sólo mensaje. Ésta lista es de *Sólo Lectura* y no se debería enviar correo a ella.

FREEBSD-HARDWARE
Discusiones generales sobre hardware y FreeBSD

Discusiones generales sobre tipos de hardware que funciona en FreeBSD, diferentes problemas y sugerencias sobre qué comprar y qué no.

FREEBSD-HUBS
Réplicas

Avisos y discusiones para personas que administran sitios réplica.

FREEBSD-INSTALL
Discusiones sobre la instalación

Ésta lista de correo es para discusiones sobre el desarrollo de la instalación de FreeBSD en próximas versiones.

FREEBSD-ISP
Cuestiones de Proveedores de Servicios de Internet

Ésta lista de correo es para debates sobre temas relevantes para Proveedores de Servicios de Internet (ISP) que usan FreeBSD. Es una lista de correo técnica y en ella se esperan contenidos estríctamente técnicos.

FREEBSD-NEWBIES
Debates sobre actividades de los novatos

Cubrimos todas las actividades de los novatos que no quedan cubiertas por ninguna de las otras, incluyendo: aprendizaje autodidacta y técnicas de resolución de problemas, búsqueda y uso de recursos y peticiones de ayuda, cómo usar las listas de correo y qué lista usar, charla en general, meter la pata, jactarse, compartir ideas, historias, soporte moral (pero no técnico) e implicación en la comunidad FreeBSD. Usamos freebsd-questions para enviar nuestros problemas y peticiones de soporte y usamos freebsd-newbies para conocer a gente que está haciendo lo mismo que nosotros cuando éramos novatos.

FREEBSD-PLATFORMS
Porte a plataformas no Intel

Cuestiones sobre plataformas diversas, debates generales, y propuestas para portes de FreeBSD para plataformas no Intel. Es una lista de correo técnica y en ella se esperan contenidos estríctamente técnicos.

FREEBSD-POLICY
Decisiones de funcionamiento interno del Core Team

Es una lista de sólo lectura y bajo volumen destinada a la toma de decisiones de funcionamento interno del Core Team de FreeBSD.

FREEBSD-PORTS
Debates sobre "ports"

Debates acerca de la "colección de ports" (`/usr/ports`)de FreeBSD, propuestas de aplicaciones a portar, modificaciones a la infraestructura de ports y coordinación general de esfuerzos. Ésta es una lista de correo técnica en la cual se esperan contenidos exclusivamente técnicos.

FREEBSD-QUESTIONS
Preguntas de los usuarios

Ésta es la lista de correo para preguntas sobre FreeBSD. No debería enviar preguntas del estilo de "cómo hacer" a las listas técnicas salvo que el contenido sea claramente técnico.

FREEBSD-QUESTIONS-DIGEST
Preguntas de los usuarios

Éste es el compendio de la lista de correo freebsd-questions. Consiste en que todos los mensajes enviados a freebsd-questions son empaquetados y enviados en un único mensaje.

FREEBSD-SCSI
Subsistema SCSI

Ésta es la lista de correo para la gente que está trabajando en el subsistema SCSI de FREEBSD. Ésta es una lista de correo técnica en la cual se esperan contenidos puramente técnicos.

FREEBSD-SECURITY
Cuestiones de seguridad

Cuestiones de seguridad informática (DES, Kerberos, problemas de seguridad conocidos y sus soluciones, etc.) Ésta es una lista de correo técnica en la que se esperan contenidos puramente técnicos.

FREEBSD-SECURITY-NOTIFICATIONS
Avisos de seguridad

Avisos de problemas de seguridad en FreeBSD y sus soluciones. Ésta no es una lista de discusión. La lista de discusión es freebsd-security.

FREEBSD-SMALL
Uso de FreeBSD en aplicaciones embebidas

En ésta lista se debaten temas relacionados con instalaciones de FreeBSD inusualmente pequeñas y embebidas. Ésta es una lista de correo técnica en la cual se esperan contenidos estrictamente técnicos.

FREEBSD-STABLE
Debates sobre el uso de FreeBSD-stable

Ésta es la lista de correo para los usuarios de freebsd-stable. Incluye avisos sobre nuevas características a incluír en -stable que afectan a los usuarios e instrucciones paso por paso para permanecer usando la versión -stable. Cualquiera que utilice FreeBSD

"stable" debería suscribirse a ésta lista. Ésta es una lista técnica en la que se esperan contenidos puramente técnicos.

FREEBSD-STANDARDS
Cumplimiento de C99 & POSIX

Éste es el foro para debates técnicos relacionadas con el Cumplimiento de las normas C99 y POSIX en FreeBSD.

FREEBSD-USER-GROUPS
Lista de coordinación de de grupos de usuarios

Ésta es la lista de correo de los coordinadores de los grupos locales de usuarios para discutir cuestiones entre ellos o con personas elegidas del Core Team. Ésta lista de correo debería estar limitada a resúmenes de reuniones y coordinación de proyectos que atañen a los Grupos de Usuarios.

FREEBSD-VENDORS
VENDORS

Debates para la coordinación entre el Proyecto FreeBSD y Distribuidores de software y hardware para FreeBSD.

A.2. Grupos de noticias de Usenet

Además de los dos grupos de noticias específicos de FreeBSD hay muchos otros en los cuales se habla sobre FreeBSD o son de algún modo interesantes para usuarios de FreeBSD.Hay un archivo donde hacer búsquedas donde pueden encontrarse algunos de esos grupos de noticias por cortesía de Warren Toomey <wkt@cs.adfa.edu.au >.

A.2.1. Grupos de noticias específicos sobre BSD

- comp.unix.bsd.freebsd.announce
- comp.unix.bsd.freebsd.misc

A.2.2. Otros grupos de noticias interesantes sobre Unix

- comp.unix
- comp.unix.questions
- comp.unix.admin
- comp.unix.programmer
- comp.unix.shell

- comp.unix.user-friendly
- comp.security.unix
- comp.sources.unix
- comp.unix.advocacy
- comp.unix.misc
- comp.bugs.4bsd
- comp.bugs.4bsd.ucb-fixes
- comp.unix.bsd

A.2.3. Sistema X Window

- comp.windows.x.i386unix
- comp.windows.x
- comp.windows.x.apps
- comp.windows.x.announce
- comp.windows.x.intrinsics
- comp.windows.x.motif
- comp.windows.x.pex
- comp.emulators.ms-windows.wine

A.3. Servidores WWW

- http://www.FreeBSD.org/ — Servidor Central.
- http://www.au.FreeBSD.org/ — Australia/1.
- http://www2.au.FreeBSD.org/ — Australia/2.
- http://www3.au.FreeBSD.org/ — Australia/3.
- http://freebsd.itworks.com.au/ — Australia/4.
- http://www.br.FreeBSD.org/www.freebsd.org/ — Brasil/1.

- http://www2.br.FreeBSD.org/www.freebsd.org/ — Brasil/2.
- http://www3.br.FreeBSD.org/ — Brasil/3.
- http://www.bg.FreeBSD.org/ — Bulgaria.
- http://www.ca.FreeBSD.org/ — Canadá/1.
- http://www2.ca.FreeBSD.org/ — Canadá/2.
- http://www3.ca.FreeBSD.org/ — Canadá/3.
- http://www.cn.FreeBSD.org/ — China.
- http://www.cz.FreeBSD.org/ — República Checa.
- http://www.dk.FreeBSD.org/ — Dinamarca.
- http://www.ee.FreeBSD.org/ — Estonia.
- http://www.fi.FreeBSD.org/ — Finlandia.
- http://www.fr.FreeBSD.org/ — Francia.
- http://www.de.FreeBSD.org/ — Alemania/1.
- http://www1.de.FreeBSD.org/ — Alemania/2.
- http://www2.de.FreeBSD.org/ — Alemania/3.
- http://www.gr.FreeBSD.org/ — Grecia.
- http://www.hu.FreeBSD.org/ — Hungría.
- http://www.is.FreeBSD.org/ — Islandia.
- http://www.ie.FreeBSD.org/ — Irlanda.
- http://www.jp.FreeBSD.org/www.FreeBSD.org/ — Japón.
- http://www.kr.FreeBSD.org/ — Corea/1.
- http://www2.kr.FreeBSD.org/ — Corea/2.
- http://www.lv.FreeBSD.org/ — Letonia.
- http://rama.asiapac.net/freebsd/ — Malasia.
- http://www.nl.FreeBSD.org/ — Holanda/1.

- http://www2.nl.FreeBSD.org/ — Holanda/2.
- http://www.no.FreeBSD.org/ — Noruega.
- http://www.nz.FreeBSD.org/ — Nueva Zelanda.
- http://www.pl.FreeBSD.org/ — Polonia/1.
- http://www2.pl.FreeBSD.org/ — Polonia/2.
- http://www.pt.FreeBSD.org/ — Portugal/1.
- http://www2.pt.FreeBSD.org/ — Portugal/2.
- http://www3.pt.FreeBSD.org/ — Portugal/3.
- http://www.ro.FreeBSD.org/ — Rumanía.
- http://www.ru.FreeBSD.org/ — Rusia/1.
- http://www2.ru.FreeBSD.org/ — Rusia/2.
- http://www3.ru.FreeBSD.org/ — Rusia/3.
- http://www4.ru.FreeBSD.org/ — Rusia/4.
- http://freebsd.s1web.com/ — Singapur.
- http://www.sk.FreeBSD.org/ — República Eslovaca.
- http://www.si.FreeBSD.org/ — Eslovenia.
- http://www.es.FreeBSD.org/ — España.
- http://www.za.FreeBSD.org/ — Sudáfrica/1.
- http://www2.za.FreeBSD.org/ — Sudáfrica/2.
- http://www.se.FreeBSD.org/ — Suecia.
- http://www.ch.FreeBSD.org/ — Suiza.
- http://www.tw.FreeBSD.org/www.freebsd.org/data/ — Taiwan.
- http://www.tr.FreeBSD.org/ — Turquía.
- http://www.ua.FreeBSD.org/www.freebsd.org/ — Ucrania/1.
- http://www2.ua.FreeBSD.org/ — Ucrania/2.

- http://www4.ua.FreeBSD.org/ — Ucrania/Crimea.
- http://www.uk.FreeBSD.org/ — Reino Unido/1.
- http://www2.uk.FreeBSD.org/ — Reino Unido/2.
- http://www3.uk.FreeBSD.org/ — Reino Unido/3.
- http://www6.FreeBSD.org/ — USA/Oregón.
- http://www2.FreeBSD.org/ — USA/Tejas.

A.4. Direcciones de correo electrónico

Los siguientes grupos de usuarios de FreeBSD provéen a sus miembros de direcciones de correo. Dichos administradores se reservan el derecho de retirar el uso de la dirección si se abusa de ella de cualquier manera.

Dominio	Recursos que se suministran	Grupo de Usuarios	Administrador
ukug.uk.FreeBSD.org	Sólo redirección	<freebsd-users@uk.FreeBSD.org>	Lee Johnston <lee@uk.FreeBSD.org >

A.5. Cuentas shell

Los siguientes grupos de usuarios facilitan cuentas shell a gente que apoya activamente el proyecto FreeBSD. Sus respectivos administradores se reservan el derecho de cancelar la cuenta y si se abusa de ella de algún modo.

Servidor	Tipo de acceso	Servicios que se ofrecen	Administrador
storm.uk.FreeBSD.org	SSH only	CVS de sólo lectura, espacio web personal, correo electrónico	Brian Somers
dogma.freebsd-uk.eu.org	Telnet/FTP/SSH	Correo electrónico, espacio web, FTP Anónimo	Lee Johnston <lee@uk.FreeBSD.org >

Capítulo 32. PGP keys

En caso de que necesites verificar una firma o enviar un mail encriptado a alguno de los responsables o miembros del core team, aquí tienes una serie de claves para uso.

32.1. Responsables

32.1.1. Responsable de Seguridad de FreeBSD

<security-officer@freebsd.org >

```
FreeBSD Security Officer <security-officer@freebsd.org>
Fingerprint = 41 08 4E BB DB 41 60 71  F9 E5 0E 98 73 AF 3F 11

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3i

mQCNAzF7MY4AAAEEAK7qBgPuBejER5HQbQlsOldk3ZVWXlRj54raz3IbuAUrDrQL
h3g57T9QY++f3Mot2LAf5lDJbsMfWrtwPrPwCCFRYQd6XH778a+l4ju5axyjrt/L
Ciw9RrOC+WaPv3lIdLuqYge2QRC1LvKACIPNbIcgbnLeRGLovFUuHi5z0oilAAUR
tDdGcmVlQlNEIFNlY3VyaXR5IE9mZmljZXIgPHNlY3VyaXR5LW9mZmljZXJAZnJl
ZWJzZC5vcmc+iQCVAwUQMX6yrOJgpPLZnQjrAQHyowQA1Nv2AY8vJIrdp2ttV6RU
tZBYnI7gTO3sFC2bhIHsCvfVU3JphfqWQ7AnTXcD2yPjGcchUfc/EcL1tSlqW4y7
PMP4GHZp9vHog1NAsgLC9Y1P/1cOeuhZ0pDpZZ5zxTo6TQcCBjQA6KhiBFP4TJql
3olFfPBh3B/Tu3dqmEbSWpuJAJUDBRAxez3C9RVb+45ULV0BAak8A/9JIG/jRJaz
QbKom6wMw852C/Z0qBLJy7KdN30099zMjQYeC9PnlkZ0USjQ4TSpC8UerYv6IfhV
nNY6gyF2Hx4CbEFlopnfA1c4yxtXKti1kSN6wBy/ki3SmqtfDhPQ4Q31p63cSe5A
3aoHcjvWuqPLpW4ba2uHVKGP3g7SSt6AOYkAlQMFEDF8mz0ff6kIA1j8vQEBmZcD
/REaUPDRx6qr1XRQlMs6pfgNKEwnKmcUzQLCvKBnYYGmD5ydPLxCPSFnPcPthaUb
5zVgMTjfjS2fkEiRrua4duGRgqN4xY7VRAsIQeMSITBOZeBZZf2oa9Ntidr5PumS
9uQ9bvdfWMpsemk2MaRG9BSoy5Wvy8VxROYYUwpT8Cf2iQCVAwUQMXsyqWtaZ42B
sqd5AQHKjAQAvolI30Nyu3IyTfNeCb/DvOe9tlOn/o+VUDNJiE/PuBe1s2Y94a/P
BfcohpKC2kza3NiW6lLTp00OWQsuu0QAPc02vYOyseZWy4y3Phnw60pWzLcFdemT
0GiYS5Xm1o9nAhPFciybn9j1q8UadIlIq0wbqWgdInBT8YI/l4f5sf6JAJUDBRAx
ezKXVS4eLnPSiKUBAc5OBACIXTlKqQC3B53qt7bNMV46m81fuw1PhKaJEI033mCD
ovzyEFFQeOyRXeu25Jg9Bq0Sn37ynISucHSmt2tUD5W0+p1MUGyTqnfqejMUWBzO
v4Xhp6a8RtDdUMBOTtro16iulGiRrCKxzVgEl4i+9Z0ZiE6BWlg5AetoF5n3mGk1
lw==
=ipyA
-----END PGP PUBLIC KEY BLOCK-----
```

32.1.2. Warner Losh

```
Warner Losh <imp@village.org>
 aka <imp@freebsd.org>
Fingerprint = D4 31 FD B9 F7 90 17 E8 37 C5 E7 7F CF A6 C1 B9
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.2
```

```
mQCNAzDzTiAAAAEEAK8D7KWEbVFUrmlqhUEnAvphNIqHEbqqT8s+c5f5c2uHtlcH
V4mV2TlUaDSVBN4+/D70oHmZc4IgiQwMPCWRrSezg9z/MaKlWhaslc8YT6Xc1q+o
EP/fAdKUrq49H0QQbkQk6Ks5wKW6v9AOvdmsS6ZJEcet6d9G4dxynu/2qPVhAAUR
tCBNLiBXYXJuZXIgTG9zaCA8aW1wQHZpbGxhZ2Uub3JnPokAlQMFEDM/SK1VLh4u
c9KIpQEBFPsD/1n0YuuUPvD4CismZ9bx9M84y5sxLolgFEfP9Ux196ZSeaPpkA0g
C9YX/IyIy5VHh3372SDWN5iVSDYPwtCmZziwIV2YxzPtZw0nUu82P/Fn8ynlCSWB
5povLZmgrWijTJdnUWI0ApVBUTQoiW5MyrNN51H3HLWXGoXMgQFZXKWYiQCVAwUQ
MzmhkfUVW/uOVC1dAQG3+AP/T1HL/5EYF0ij0yQmNTzt1cLt0b1e3N3zN/wPFFWs
BfrQ+nsv1zw7cEgxLtktk73wBGM9jUIdJu8phgLtl5a0m9UjBq5oxrJaNJr6UTxN
a+sFkapTLT1g84UFUO/+8qRB12v+hZr2WeXMYjHAFUT18mp3xwjW9DUV+2fW1Wag
YDKJAJUDBRAzOYK1s1pi61mfMj0BARBbA/930CHswOF0HIr+4YYUs1ejDnZ2J3zn
icTZhl9uAfEQq++Xor1x476j67Z9fESxyHltUxCmwxsJ1uOJRwzjyEoMlyFrIN4C
dE0C8g8BF+sRTt7VLURLERvlBvFrVZueXSnXvmMoWFnqpSpt3EmN6TNaLe8Cm87a
k6EvQy0dpnkPKokAlQMFEDD9Lorccp7v9qj1YQEBrRUD/3N4cCMWjzsIFp2Vh9y+
RzUrblyF84tJyA7Rr1p+A7dxf7je3Zx5QMEXosWL1WGnS5vC9YH2WZwv6sCU61gU
rSy9z8KHlBEHh+Z6fdRMrjd9byPf+n3cktT0NhS23oXB1ZhNZcB2KKhVPlNctMqO
3gTYx+Nlo6xqjR+J2NnBYU8p =7fQV
-----END PGP PUBLIC KEY BLOCK-----
```

32.2. Miembros del Core Team

32.2.1. Satoshi Asami

```
Satoshi Asami <asami@cs.berkeley.edu>
          aka <asami@FreeBSD.ORG>
Fingerprint = EB 3C 68 9E FB 6C EB 3F  DB 2E 0F 10 8F CE 79 CA

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.2

mQCNAzPVyoQAAAEEAL7W+kipxB171Z4SVyyL9skaA7hG3eRsSOWk7lfvfUBLtPog
f3OKwrApoc/jwLf4+Qpdzv5DLEt/6Hd/clskhJ+q1gMNHyZ5ABmUxrTRRNvJMTrb
3fPU3oZj7sL/MyiFaT1zF8EaMP/iS2ZtcFsbYOqGeA8E/58uk4NA0SoeCNiJAAUR
tCVTYXRvc2hpIEFzYW1pIDxhc2FtaUBjcy5iZXJrZWxleS5lZHU+iQCVAwUQM/AT
+EqGN2HYnOMZAQF11QP/eSXb2FuTb1yX5yoo1Im8YnIk1SEgCGbyEbOMMBznVNDy
5g2TAD0ofLxPxy5Vodjg8rf+lfMVtO5amUH6aNcORXRncE83T10JmeM6JEp0T6jw
zOHKz8jRzygYLBayGsNIJ4BGxa4LeaGxJpO1ZEvRlNkPH/YEXK5oQmq9/DlrtYOJ
AEUDBRAz42JT8ng6GBbVvu0BAU8nAYCsJ8PiJpRUGlrz6rxjX8hqM1v3vqFHLcG+
G52nVMBSy+RZBgzsYIPwI5EZtWAKb22JAJUDBRAz4QBWdbtuOHaj97EBAaQPA/46
+NLUp+Wubl90JoonoXocwAg88tvAUVSzsxPXj0lvypAiSI2AJKsmn+5PuQ+/IoQy
lywRsxiQ5GD7C72SZ1yw2WI9DWFeAi+qa4b8n9fcLYrnHpyCY+zxEpu4pam8FJ7H
JocEUZz5HRoKKOLHErzXDiuTkkm72b1glmCqAQvnB4kAlQMFEDPZ3gyDQNEqHgjY
iQEBFfUEALu2C0uo+1Z7C5+xshWRYY5xNCzK20O6bANVJ+CO2fih96KhwsMof3lw
fDso5HJSwgFd8WT/sR+Wwzz6BAE5UtgsQq5GcsdYQuGI1yIlCYUpDp5sgswNm+OA
bX5a+r4F/ZJqrqT1J56Mer0VVsNfe5nIRsjd/rnFAFVfjcQtaQmjiQCVAwUQM9uV
mcdm8Q+/vPRJAQELHgP9GqNiMpLQlZig17fDnCJ73P0e5t/hRLFehZDlmEI2TK7j
Yeqbw078nZgyyuljZ7YsbstRIsWVCxobX5eH1kX+hIxuUqCAkCsWUY4abG89kHJr
XGQn6X1CX7xbZ+b6b9jLK+bJKFcLSfyqR3M2eCyscSiZYkWKQ5l3FYvbUzkeb6K0
IVNhdG9zaGkgQXNhbWkgPGFzYW1pQEZyZWVCU0QuT1JHPg==
```

```
=39SC
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.2. Jonathan M. Bresler

```
Jonathan M. Bresler <jmb@FreeBSD.org>
f16    Fingerprint16 = 31 57 41 56 06 C1 40 13  C5 1C E3 E5 DC 62 ↺
0E FB

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: PGPfreeware 5.0i for non-commercial use

mQCNAzG2GToAAAEEANI6+4SJAAgBpl53XcfEr1M9wZyBqC0tzpie7Zm4vhv3hO8s
o5BizSbcJheQimQiZAY4OnlrCpPxijMFSaihshs/VMAz1qbisUYAMqwGEO/T4QIB
nWNo0Q/qOniLMxUrxS1RpeW5vbghErHBKUX9GVhxbiVfbwc4wAHbXdKX5jjdAAUR
tCVKb25hdGhhbiBNLiBCcmVzbGVyIDxqbWJARnJlZUJTRC5PUkc+iQCVAwUQNbtI
gAHbXdKX5jjdAQHamQP+OQr10QRknamIPmuHmFYJZ0jU9XPIvTTMuOiUYLcXlTdn
GyTUuzhbEywgtOldW2V5iA8platXThtqC68NsnN/xQfHA5xmFXVbayNKn8H5stDY
2s/4+CZ06mmJfqYmONF1RCbUk/M84rVT3Gn2tydsxFh4Pm32lf4WREZWRiLqmw+J
AJUDBRA0DfF99RVb+45ULV0BAcZ0BACCydiSUG1VR0a5DBcHdtin2iZMPsJUPRqJ
tWvP6VeI8OFpNWQ4LW6ETAvn35HxV2kCcQMyht1kMD+KEJz7r8Vb94TS7KtZnNvk
2D1XUx8Locj6xel5c/Lnzlnnp7Bp1XbJj2u/NzCaZQ0eYBdP/k7RLYBYHQQln5x7
BOuiRJNVU4kAlQMFEDQLcShVLh4uc9KIpQEBJv4D/3mDrD0MM9EYOVuyXik3UGVI
8quYNA9ErVcLdt10NjYc16VI2HOnYVgPRag3Wt7W8wlXShpokfC/vCNt7f5JgRf8
h2a1/MjQxtlD+4/Js8k7GLa53oLon6YQYk32IEKexoLPwIRO4L2BHWa3GzHJJSP2
aTR/Ep90/pLdAOu/oJDUiQCVAwUQMqyL0LNaYutZnzI9AQF25QP9GFXhBrz2tiWz
2+0gWbpcGNnyZbfsVjF6ojGDdmsjJMyWCGw49XR/vPKYIJY9EYo4t49GIajRkISQ
NNiIz22fBAjT2uY9YlvnTJ9NJleMfHr4dybo7oEKYMWWijQzGjqf2m8wf9OaaofE
KwBX6nxcRbKsxm/BVLKczGYl3XtjkcuJAJUDBRA1ol5TZWCprDT5+dUBATzXA/9h
/ZUuhoRKTWViaistGJfWi26FB/Km5nDQBr/Erw3XksQCMwTLyEugg6dahQ1u9Y5E
5tKPxbB69eF+7JXVHE/z3zizR6VL3sdRx74TPacPsdhZRjChEQc0htLLYAPkJrFP
VAzAlSlm7qd+MXf8fJovQs6xPtZJXukQukPNlhqZ94kAPwMFEDSH/kF4tXKgazlt
bxECfk4AoO+VaFVfguUkWX10pPSSfvPyPKqiAJ4xn8RSIe1ttmnqkkDMhLh00mKj
lLQuSm9uYXRoYW4gTS4gQnJlc2xlciA8Sm9uYXRoYW4uQnJlc2xlckBVU2kubmV0
PokAlQMFEDXbdSkB213Sl+Y43QEBV/4D/RLJNTrtAqJ1ATxXWv9g8Cr3/YF0GTmx
5dIrJOpBup7eSSmiM/BL9Is4YMsoVbXCI/8TqA67TMICvq35PZU4wboQB8DqBAr+
gQ8578M7Ekw1OAF6JXY6AF2P8k7hMcVBcVOACELPT/NyPNByG5QRDoNmlsokJaWU
/2ls4QSBZZlb
=zbCw
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.3. Andrey A. Chernov

```
Andrey A. Chernov <ache@FreeBSD.org>
       aka <ache@nagual.pp.ru>
Key fingerprint = 33 03 9F 48 33 7B 4A 15  63 48 88 0A C4 97 FD 49

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3ia

mQCNAiqUMGQAAAEEAPGhcD6A2Buey5LYz0sphDLpVgOZc/bb9UHAbaGKUAGXmafs
```

```
Dcb2HnsuYGgX/zrQXuCi/wIGtXcZWB97APtKOhFsZnPinDR5n/dde/mw9FnuhwqD
m+rKSL1HlN0z/Msa5y7g16760wHhSR6NoBSEG5wQAHIMMq7Q0uJgpPLZnQjrAAUT
tCVBbmRyZXkgQS4gQ2hlcm5vdiA8YWNoZUBuYWd1YWwucHAucnU+iQCVAwUQM2Ez
u+JgpPLZnQjrAQEyugP8DPnS8ixJ5OeuYgPFQf5sy6l+LrB6hyaS+lgsUPahWjNY
cnaDmfda/q/BV5d4+y5rlQe/pjnYG7/yQuAR3jhlXz8XDrqlBOnW9AtYjDt5rMfJ
aGFTGXAPGZ6k6zQZE0/YurT8ia3qjvuZm3Fw4NJrHRx7ETHRvVJDvxA6Ggsvmr20
JEFuZHJleSBBLiBDaGVybm92IDxhY2hlQEZyZWVCU0Qub3JnPokAlQMFEDR5uVbi
YKTy2Z0I6wEBLgED/2mn+hw4/3peLx0Sb9LNx//NfCCkVefSf2G9Qwhx6dvwbX7h
mFca97h7BQN4GubU1Z5Ffs6TeamSBrotBYGmOCwvJ6S9WigF9YHQIQ3B4LEjskAt
pcjU583y42zM11kkvEuQU2Gde61daIylJyOxsgpjSWpkxq50fgY2kLMfgl/ftCZB
bmRyZXkgQS4gQ2hlcm5vdiA8YWNoZUBuaWV0enNjaGUubmV0PokAlQMFEDR5svDi
YKTy2Z0I6wEBOTQD/0OTCAXIjuak363mjERvzSkVsNtIH9hA1l0w6Z95+iH0fHrW
xXKT0vBZE0y0Em+S3cotLL0bMmVE3F3D3GyxhBVmgzjyx0NYNoiQjYdi+6g/PV30
Cn4vOO6hBBpSyI6vY6qGNqcsawuRtHNvK/53MpOfKwSlICEBYQimcZhkci+EtCJB
bmRyZXkgQS4gQ2hlcm5vdiA8YWNoZUBuYWd1YWwucnU+iQCVAwUQMcm5HeJgpPLZ
nQjrAQHwvQP9GdmAf1gdcuayHEgNkc11macPH11cwWjYjzA2YoecFMGV7iqKK8QY
rr1MjbGXf8DAG8Ubfm0QbI8Lj8iG3NgqIru0c72UuHGSn/APfGGG0AtPX5UK/k7B
gI0Ca2po6NA5nrSp8tDsdEz/4gyea84RXl2prtTf5Jj07hflbRstGXK0MkFuZHJl
eSBBLiBDaGVybm92LCBCbGFjayBNYWdlIDxhY2hlQGFzdHJhbC5tc2suc3U+iQCV
AwUQMCsAo5/rGryoL8h3AQHq1QQAidyNFqA9hvrmMcjpY7csJVFlGvj574Wj4GPa
o3pZeuQaMBmsWqaXLYnWU/Aldb6kTz6+nRcQX50zFH0THSPfApwEW7yybSTI5apJ
mWT3qhKN2vmLNg2yNzhqLTzHLD1lH3i1pfQq8WevrNfjLUco5S/VuekTma/osnzC
Cw7fQzCJAJUDBRAwKvwoa1pnjYGyp3kBARihBACoXr3qfG65hFCyKJISmjOvaoGr
anxUIkeDS0yQdTHzhQ+dwB1OhhK15E0Nwr0MKajLMm90n6+Zdb5y/FIjpPriu8dI
rlHrWZlewa88eEDM+Q/NxT1iYg+HaKDAE171jmLpSpCL0MiJtO0i36L3ekVD7Hv8
vffOZHPSHirIzJOZTYkAlQMFEDAau6zFLUdtDb+QbQEBQX8D/AxwkYeFaYxZYMFO
DHIvSk23hAsjCmUA2Uil1FeWAusb+o8xRfPDc7TnosrIifJqbF5+fcHCG5VSTGlh
Bhd18YWUeabf/h9O2BsQX55yWRuB2x3diJ1xI/VVdG+rxlMCmE4ZR1Tl9x+Mtun9
KqKVpB39VlkCBYQ3hlgNt/TJUY4riQCVAwUQMBHMmyJRltlmbQBRAQFQkwP/YC3a
hs3ZMMoriOlt3ZxGNUUPTF7rIER3j+c7mqGG46dEnDB5sUrkzacpoLX5sj1tGR3b
vz9a4vmk1Av3KFNNvrZZ3/BZFGpq3mCTiAC9zsyNYQ8L0AfGIUO5goCIjqwOTNQI
AOpNsJ5S+nMAkQB4YmmNlI6GTb3D18zfhPZ6uciJAJUCBRAwD0sl4uW74fteFRkB
AWsAA/9NYqBRBKbmltQDpyK4+jBAYjkXBJmARFXKJYTlnTgOHMpZqoVyW96xnaa5
MzxEiu7ZWm5oL10QDIp1krkBP2KcmvfSMMHb5aGCCQc2/P8NlfXAuHtNGzYiI0UA
Iwi8ih/S1liVfvnqF9uV3d3koE7VsQ9OA4Qo0ZL2ggW+/gEaYIkAlQMFEDAOz6qx
/IyHe3rl4QEBIvYD/jIr8Xqo/2I5gncghSeFR01n0vELFIvaF4cHofGzyzBpYsfA
+6pgFI1IM+LUF3kbUkAY/2uSf9U5ECcaMCTWCwVgJVO+oG075SHEM4buhrzutZiM
1dTyTaepaPpTyRMUUx9ZMMYJs7sbqLId1eDwrJxUPhrBNvf/w2W2sYHSY8cdiQCV
AwUQMAzqgHcdkq6JcsfBAQGTxwQAtgeLFi2rhSOdllpDXUwz+SS6bEjFTWgRsWFM
y9QnOcqryw7LyuFmWein4jasjY033JsODfWQPiPVNA3UEnXVg9+n8AvNMPO8JkRv
Cn1eNg0VaJy9J368uArio93agd2Yf/R5r+QEuPjIssVk8hdcy/luEhSiXWf6bLMV
HEA0J+OJAJUDBRAwDUi+4mCk8tmdCOsBAatBBACHB+qtW880seRCDZLjl/bT1b14
5po60U7u6a3PEBkY0NA72tWDQuRPF/Cn/0+VdFNxQUsgkrbwaJWOoi0KQsvlOm3R
rsxKbn9uvEKLxExyKH3pxp76kvz/lEWwEeKvBK+84Pb1lzpG3W7u2XDfi3VQPTi3
5SZMAHc6C0Ct/mjNlYkAlQMFEDAMrPD7wj+NsTMUOQEBJckD/ik4WsZzm2qOx9Fw
erGq7Zwchc+Jq1YeN5PxpzqSf4AG7+7dFIn+oe6X2FcIzgbYY+IfmgJIHEVjDHH5
+uAXyb6l4iKc89eQawO3t88pfHLJWbTzmnvgz2cMrxt94HRvgkHfvcpGEgbyldq6
EB33OunazFcfZFRIcXk1sfyLDvYE
=1ahV
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.4. Jordan K. Hubbard

```
Jordan K. Hubbard <jkh@FreeBSD.org>
Fingerprint = 3C F2 27 7E 4A 6C 09 0A  4B C9 47 CD 4F 4D 0B 20

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3ia

mQCNAzFjX0IAAAEEAML+nm9/kDNPp43ZUZGjYkm2QLtoC1Wxr8JulZXqk7qmhYcQ
jvX+fyoriJ6/7ZlnLe2oG5j9tZOnRLPvMaz0g9CpW6Dz3nkXrNPkmOFV9B8D94Mk
tyFeRJFqnkCuqBj6D+H8FtBwEeeTecSh2tJ0bZZTXnAMhxeOdvUVW/uOVC1dAAUR
tCNKb3JkYW4gSy4gSHViYmFyZCA8amtoQEZyZWVCU0Qub3JnPokBFQMFEDXCTXQM
j46yp4IfPQEBwO8IAIN0J09AXBf86dFUTFGcAMrEQqOF5IL+KGorAjzuYxERhKfD
ZV7jA+sCQqxkWfcVcE20kVyVYqzZIkio9a5zXP6TwA247JkPt54S1PmMDYHNlRIY
laXlNoji+4q3HP2DfHqXRT2859rYpm/fG/v6pWkos5voPKcZ2OFEp9W+Ap88oqw+
5rx4VetZNJq1Epmis4INj6XqNqj85+MOOIYE+f445ohDM6B/Mxazd6cHFGGIR+az
VjZ6lCDMLjzhB5+FqfrDLYuMjqkMTR5z9DL+psUvPlCkYbQ11NEWtEmiIWjUcNJN
GCxGzv5bXk0XPu3ADwbPkFE2usW1cSM7AQFiwuyJAJUDBRAxe+Q9a1pnjYGyp3kB
AV7XA/oCSL/Cc2USpQ2ckwkGpyvIkYBPszIcabSNJAzm2hsU9Qa6WOPxD8olDddB
uJNiW/gznPC4NsQ0N8Zr4IqRX/TTDVf04WhLmd8AN9SOrVv2q0BKgU6fLuk979tJ
utrewH6PR2qBOjAaR0FJNk4pcYAHeT+e7KaKy96YFvWKIyDvc4kAlQMFEDF8ldof
f6kIA1j8vQEBDH4D/0Zm0oNlpXrAE1EOFrmp43HURHbij8n0Gra1w9sbfo4PV+/H
U8ojTdWLy6r0+prH7NODCkgtIQNpqLuqM8PF2pPtUJj9HwTmSqfaT/LMztfPA6PQ
csyT7xxdXl0+4xTDl1avGSJfYsI8XCAy85cTs+PQwuyzugE/iykJO1Bnj/paiQCV
AwUQMXvlBvUVW/uOVC1dAQF2fQP/RfYC6RrpFTZHjo2qsUHSRk0vmsYfwG5NHP5y
oQBMsaQJeSckN4n2JOgR4T75U4vS62aFxgPLJP3lOHkU2Vc7xhAuBvsbGr5RP8c5
LvPOeUEyz6ZArp1KUHrtcM2iK1FBOmY4dOYphWyWMkDgYExabqlrAq7FKZftpq/C
BiMRuaw=
=C/Jw
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.5. Poul-Henning Kamp

```
Poul-Henning Kamp <phk@FreeBSD.org>
Fingerprint = A3 F3 88 28 2F 9B 99 A2  49 F4 E2 FA 5A 78 8B 3E

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3ia

mQCNAzAdpMIAAAEEALHDgrFUwhZtb7PbXg3upELoDVEUPFRwnmpJH1rRqyROUGcI
ooVe7u+FQlIs5OsXK8ECs/5Wpe2UrZSzHvjwBYOND5H42YtI5UULZLRCo5bFfTVA
K9Rpo5icfTsYihrzU2nmnycwFMk+jYXyT/ZDYWDP/BM9iLjj0x9/qQgDWPy9AAUR
tCNQb3VsLUhlbm5pbmcgS2FtcCA8cGhrQEZyZWVCU0Qub3JnPokAlQMFEDQQ0aZ1
u244dqP3sQEBu4ID/jXFFeJgs2MdTDNOZM/FbfDhI4qxAbYUsqS3+Ra16yd8Wd/A
jV+IHJE2NomFWl8UrUjCGinXiwzPgK1OfFJrS9Og1wQLvAl0X84BA8MTP9BQr4w7
6I/RbksgUSrVCIO8MJwlydjSPocWGBeXlVjbZxXzyuJk7H+TG+zuI5BuBcNIiQCV
AwUQMwYr2rNaYutZnzI9AQHiIQP/XxtBWFXaBRgVLEhRNpS07YdU+LsZGlLOZehN
9L4UnJFHQQPNOpMey2gF7Y95aBOw5/1xS5vlQpwmRFCntWsm/gqdzK6rulfr1r5A
y94LO5TAC6ucNu396Y4vo1TyD1STnRC466KlvmtQtAtFGgXlORWLL9URLzcRFd1h
D0yXd9aJAJUDBRAxfo19a1pnjYGyp3kBAQqyA/4v64vP3l1F0Sadn6ias761hkz/
SMdTuLzILmofSCC4o4KWMjiWJHs2Soo41QlZi1+xMHzV32JKiwFlGtPHqL+EHyXy
```

```
Q4H3vmf9/1KF+0XCaMtgI0wWUMziPSTJK8xXbRRmMDK/0F4TnVVaUhnmf+h5K7O6
XdmejDTa0X/NWcicmIkAlQMFEDF8lef1FVv7jlQtXQEBcnwD/0ro1PpUtlkLmreD
tsGTkNa7MFLegrYRvDDrHOwPZH152W2jPUncY+eArQJakeHiTDmJNpFagLZglhE0
bqJyca+UwCXX+6upAclWHEBMg2byiWMMqyPVEEnpUoHM1sIkgdNWlfQAmipRBfYh
2LyCgWvR8CbtwPYIFvUmGgB3MR87iQCVAwUQMUseXB9/qQgDWPy9AQGPkwP/WEDy
El2Gkvua9COtMAifot2vTwuvWWpNopIEx0Ivey4aVbRLD90gGCJw8OGDEtqFPcNV
8aIiy3fYVKXGZZjvCKd7zRfhNmQn0eLDcymq2OX3aPrMc2rRlkT4Jx425ukR1gsO
qiQAgw91aWhY8dlw/EKzk8ojm52x4VgXaBACMjaJAJUDBRAxOUOg72G56RHVjtUB
AbL4A/9HOn5Qa0lq9tKI/HkSdc5fGQD/66VdCBAb292RbB7CS/EM07MdbcqRRYIa
0+0gwQ3OdsWPdCVgH5RIhp/WiC+UPkR1cY8N9Mg2kTwJfZZfNqN+BgWlgRMPN27C
OhYNl8Q33Nl9CpBLrZWABF44jPeT0EvvTzP/5ZQ7T75EsYKYiYkAlQMFEDDmryQA
8tkJ67sbQQEBPdsEALCj6v1OBuJLLJTlxmmrkqAZPVzt5QdeO3Eqa2tcPWcU0nqP
vHYMzZcZ7oFg58NZsWrhSQQDIB5e+K65Q/h6dC7W/aDskZd64jxtEznX2kt0/MOr
8OdsDis1K2f9KQftrAx81KmVwW4Tqtzl7NWTDXt44fMOtibCwVq8v2DFkTJy
=JKbP
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.6. Rich Murphey

```
Rich Murphey <rich@FreeBSD.org>
fingerprint = AF A0 60 C4 84 D6 0C 73  D1 EF C0 E9 9D 21 DB E4

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.2

mQCNAy97V+MAAAEEALiNM3FCwm3qrCe81E20UOSlNclOWfZHNAyOyj1ahHeINvo1
FBF2Gd5Lbj0y8SLMno5yJ6P4F4r+x3jwHZrzAIwMs/lxDXRtB0VeVWnlj6a3Rezs
wbfaTeSVyh5JohEcKdoYiMG5wjATOwK/NAwIPthB1RzRjnEeer3HI3ZYNEOpAAUR
tCRSaWNoIE11cnBoZXkgPHJpY2hAbGFtcHJleS51dG1iLmVkdT6JAJUDBRAve15W
vccjdlg0Q6kBAZTZBACcNd/LiVnMFURPrO4pVRn1sVQeokVX7izeWQ7siE31Iy7g
Sb97WRLEYDi686osaGfsuKNA87Rm+q5F+jxeUV4w4szoqp60gGvCbD0KCB2hWraP
/2s2qdVAxhfcoTin/Qp1ZWvXxFF7imGA/IjYIfB42VkaRYu6BwLEm3YAGfGcSw==
=QoiM
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.7. John Polstra

```
John D. Polstra <jdp@polstra.com>
Fingerprint = 54 3A 90 59 6B A4 9D 61  BF 1D 03 09 35 8D F6 0D

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.2

mQCNAzMElMEAAAEEALizp6ZW9QifQgWoFmG3cXhzQ1+Gt+a4S1adC/TdHdBvw1M/
I6Ok7TC0dKF8blW3VRgeHo4F3XhGn+n9MqIdboh4HJC5Iiy63m98sVLJSwyGO4oM
dkEGyyCLxqP6h/DU/tzNBdqFzetGtYvU4ftt3RO0a506cr2CHcdm8Q+/vPRJAAUR
tCFKb2huIEQuIFBvbHN0cmEgPGpkcEBwb2xzdHJhLmNvbT6JAJUDBRAzBNBE9RVb
+45ULV0BAWgiA/0WWO3+c3qlptPCHJ3DFm6gG/qNKsY94agL/mHOr0fxMP5l2qKX
O6a1bWkvGoYq0EwoKGFfn0QeHiCl6jVi3CdBX+W7bObMcoi+foqZ6zluOWBC1Jdk
WQ5/DeqQGYXqbYjqO8voCScTAPge3XlMwVpMZTv24u+nYxtLkE0ZcwtY9IkAlQMF
EDMEt/DHZvEPv7z0SQEBXh8D/2egM5ckIRpGz9kcFTDClgdWWtlgwC1iI2p9gEhq
```

```
aufy+FUJlZS4GSQLWB0BlrTmDC9HuyQ+KZqKFRbVZLyzkH7WFs4zDmwQryLV5wkN
C4BRRBXZfWy8s4+zT2WQD1aPO+ZsgRauYLkJgTvXTPU2JCN62Nsd8R7bJS5tuHEm
7HGmiQCVAwUQMwSvHB9/qQgDWPy9AQFAhAQAgJ1AlbKITrEoJ0+pLIsov3eQ348m
SVHEBGIkU3Xznjr8NzT9aYtq4TIzt8jplqP3QoV1ka1yYpZf0NjvfZ+ffYp/sIaU
wPbEpgtmHnVWJAebMbNs/Ad1w8GDvxEt9IaCbMJGZnHmfnEqOBIxF7VBDPHHoJxM
V31K/PIoYsHAy5w=
=cHFa
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.8. Guido van Rooij

```
Guido van Rooij <guido@gvr.win.tue.nl>
Fingerprint = 16 79 09 F3 C0 E4 28 A7  32 62 FA F6 60 31 C0 ED

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.2

mQCNAzGeO84AAAEEAKKAY91Na//DXwlUusr9GVESSlVwVP6DyH1wcZXhfN1fyZHq
SwhMCEdHYoojQds+VqD1iiZQvv1RLByBgj622PDAPN4+Z49HjGs7YbZsUNuQqPPU
wRPpP6ty69x1hPKq1sQIB5MS4radpCM+4wbZbhxv7l4rP3RWUbNaYutZnzI9AAUR
tCZHdWlkbyB2YW4gUm9vaWogPGd1aWRvQGd2ci53aW4udHVlLm5sPokAlQMFEDMG
Hcgff6kIA1j8vQEBbYgD/jm9xHuUuY+iXDkOzpCXBYACYEZDV913MjtyBAmaVqYo
Rh5HFimkGXe+rCo78Aau0hc57fFMTsJqnuWEqVt3GRq28hSK1FOZ7ni9/XibHcmN
rt2yugl3hYpClijo4nrDL1NxibbamkGW/vFGcljS0jqXz6NDVbGx5Oo7HBByxByz
iQCVAwUQMhmtVjt/x7zOdmsfAQFuVQQApsVUTigT5YWjQA9Nd5Z0+a/oVtZpyw5Z
OljLJP3vqJdMa6TidhfcatjHbFTve5x1dmjFgMX/MQTd8zf/+Xccy/PX4+lnKNpP
eSf1Y4aK+E8KHmBGd6GzX6CIboyGYLS9e3kGnN06F2AQtaLyJFgQ71wRaGuyKmQG
FwTn7jiKb1aJAJUDBRAyEOLXPt3iN6QQUSEBATwQA/9jqu0Nbk154+Pn+9mJX/YT
fYR2UqK/5FKCqgL5Nt/Deg2re0zMD1f8F9Dj6vuAAxq8hnOkIHKlWolMjkRKkzJi
mSPEWl3AuHJ31k948J8it4f8kq/o44usIA2KKVMlI63Q/rmNdfWCyiYQEVGcRbTm
GTdZIHYCOgV5dOo4ebFqgYkAlQMFEDIE1nMEJn15jgpJ0QEBW6kEAKqN8XSgzTqf
CrxFXT07MlHhfdbKUTNUoboxCGCLNW05vf1A8F5fdE5i14LiwkldWIzPxWD+Sa3L
fNPCfCZTaCiyGcLyTzVfBHA18MBAOOX6JiTpdcm22jLGUWBf/aJK3yz/nfbWntd/
LRHysIdVp29lP5BF+J9/Lzbb/9LxP1taiQCVAwUQMgRXZ44CzbsJWQz9AQFf7gP/
Qa2FS5S6RYKG3rYanWADVe/ikFV2lxuM1azlWbsmljXvKVWGe6cV693nS5lGGAjx
lbd2ADwXjlkNhv45HLWFm9PEveO9Jjr6tMuXVt8N2pxiX+1PLUN9CtphTIU7Yfjn
s6ryZZfwGHSfIxNGi5ua2SoXhg0svaYnxHxXmOtH24iJAJUDBRAyAkpV8qaAEa3W
TBkBARfQBAC+S3kbulEAN3SI7/A+A/dtl9DfZezT9C4SRBGsl2clQFMGIXmMQ/7v
7lLXrKQ7U2zVbgNfU8smw5h2vBIL6f1PyexSmc3mz9JY4er8KeZpcf6H0rSkHl+i
d7TF0GvuTdNPFO8hc9En+GG6QHOqbkB4NRZ6cwtfwUMhk2FHXBnjF4kAlQMFEDH5
FFukUJAsCdPmTQEBe74EAMBsxDnbD9cuI5MfF/QeTNEG4BIVUZtAkDme4Eg7zvsP
d3DeJKCGeNjiCWYrRTCGwaCWzMQk+/+MOmdkI6Oml+AIurJLoHceHS9jP1izdP7f
N2jkdeJSBsixunbQWtUElSgOQQ4iF5kqwBhxtOfEP/L9QsoydRMR1yB6WPD75H7V
iQCVAwUQMZ9YNGtaZ42Bsqd5AQH0PAQAhpVlAc3ZM/KOTywBSh8zWKVlSk3q/zGn
k7hJmFThnlhH1723+WmXE8aAPJi+VXOWJUFQgwELJ6R8jSU2qvk2m1VWyYSqRKvc
VRQMqT2wjss0GE1Ngg7tMrkRHT0il7E2xxIb8vMrIwmdkbTfYqBUhhGnsWPHZHq7
MoA1/b+rK7CJAJUDBRAxnvXh3IDyptUyfLkBAYTDA/4mEKlIP/EUX2Zmxgrd/JQB
hqcQlkTrBAaDOnOqe/4oewMKR7yaMpztYhJs97i03Vu3fgoLhDspE55ooEeHj0r4
cOdiWfYDsjSFUYSPNVhW4OSruMA3c29ynMqNHD7hpr3rcCPUi7J2RncocOcCjjK2
BQb/9IAUNeK4C9gPxMEZLokAlQMFEDGeO86zWmLrWZ8yPQEBEEID/2fPEUrSX3Yk
j5TJPFZ9MNX0lEo7AHYjnJgEbNI4pYm6C3PnMlsYfCSQDHuXmRQHAOWSdwOLvCkN
F8eDaF3M6u0urgeVJ+KVUnTz2+LZoZs12XSZKCte0HxjbvPpWMTTrYyimGezH79C
```

```
mgDVjsHaYOx3EXF0nnDmtXurGioEmW1J
=mSvM
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.9. Peter Wemm

```
Peter Wemm <peter@FreeBSD.org>
       aka <peter@spinner.dialix.com>
       aka <peter@haywire.dialix.com>
       aka <peter@perth.dialix.oz.au>
Key fingerprint = 47 05 04 CA 4C EE F8 93  F6 DB 02 92 6D F5 58 8A

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3ia

mQCNAy9/FJwAAAEEALxs9dE9tFd0Ru1TXdq301KfEoe5uYKKuldHRBOacG2Wny6/
W3Ill57hOi2+xmq5X/mHkapywxvy4cyLdt31i4GEKDvxpDvEzAYcy2n9dIup/eg2
kEhRBX9G5k/LKM4NQsRIieaIEGGgCZRm0lINqw495aZYrPpO4EqGN2HYnOMZAAUT
tCVQZXRlciBXZW1tIDxwZXRlckBoYXl3aXJlLmRpYWxpeC5jb20+iQCVAwUQMwWT
cXW7bjh2o/exAQEFkQP+LIx5zKlYp1uR24xGApMFNrNtjh+iDIWnxxb2M2Kb6x4G
9z6OmbUCoDTGrX9SSL2Usm2RD0BZfyv9D9QRWC2TSOPkPRqQgIycc11vgbLolJJN
eixqsxlFeKLGEx9eRQCCbo3dQIUjc2yaOe484QamhsK1nL5xpoNWI1P9zIOpDiGJ
AJUDBRAxsRPqSoY3Ydic4xkBAbWLA/9q1Fdnnk4unpGQsG31Qbtr4AzaQD5m/JHI
4gRmSmbj6luJMgNG3fpO06Gd/Z7uxyCJB8pTst2a8C/ljOYZxWT+5uSzkQXeMi5c
YcI1sZbUpkHtmqPW623hr1PB3ZLA1TIcTbQW+NzJsxQ1Pc6XG9fGkT9WXQW3Xhet
AP+juVTAhLQlUGV0ZXIgV2VtbSA8cGV0ZXJAcGVydGguZGlhbGl4Lm96LmF1PokA
lQMFEDGxFCFKhjdh2JzjGQEB6XkD/2HOwfuFrnQUtdwFPUkgtEqNeSr64jQ3Maz8
xgEtbaw/ym1PbhbCk311UWQq4+izZE2xktHTFClJfaMnxVIfboPyuiSF99KHiWnf
/Gspet0S7m/+RXIwZi1qSqvAanxMiA7kKgFSCmchzas8TQcyyXHtn/gl9v0khJkb
/fv3R20btB5QZXRlciBXZW1tIDxwZXRlckBGcmVlQlNELm9yZz6JAJUDBRAxsRJd
SoY3Ydic4xkBAZJUA/4i/NWHz5LIH/R4IF/3V3LleFyMFr5EPFY0/4mcv2v+ju9g
brOEM/xd4LlPrx1XqPeZ74JQ6K9mHR64RhKR7ZJJ9A+12yr5dVqihe911KyLKab9
4qZUHYi36WQu2VtLGnw/t8Jg44fQSzbBF5q9iTzcfNOYhRkSD3BdDrC3llywO7Ql
UGV0ZXIgV2VtbSA8cGV0ZXJAc3Bpbm5lci5kaWFsaXguY29tPokAlQMFEDGxEi1K
hjdh2JzjGQEBdA4EAKmNFlj8RF9HQsoI3UabnvYqAWN5wCwEB4u+Zf8zq6OHic23
TzoK1SPlmSdBE1dXXQGS6aiDkLT+xOdeewNs7nfUIcH/DBjSuklAOJzKliXPQW7E
kuKNwy4eq5bl+j3HB27i+WBXhn6OaNNQY674LGaR41EGq44Wo5ATcIicig/z
=gv+h
-----END PGP PUBLIC KEY BLOCK-----
```

32.2.10. Jörg Wunsch

```
Type Bits/KeyID    Date       User ID
pub  1024/76A3F7B1 1996/04/27 Joerg Wunsch <joerg_wunsch@uriah.heep.↺
sax.de>
           Key fingerprint = DC 47 E6 E4 FF A6 E9 8F  93 21 E0 7D ↺
F9 12 D6 4E
                             Joerg Wunsch <joerg_wunsch@interface-
business.de>
                             Joerg Wunsch <j@uriah.heep.sax.de>
                             Joerg Wunsch <j@interface-business.de>
```

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3ia

mQCNAzGCFeAAAAEEAKmRBU2Nvc7nZy1Ouid61HunA/5hF4O91cXm71/KPaT7dskz
q5sFXvPJPpawwvqHPHfEbAK42ZaywyFp59L1GaYj87Pda+PlAYRJyY2DJl5/7JPe
ziq+7B8MdvbX6D526sdmcR+jPXPbHznASjkx9DPmK+7TgFujyXW7bjh2o/exAAUR
tC1Kb2VyZyBXdW5zY2ggPGpvZXJnX3d1bnNjaEB1cmlhaC5oZWVwLnNheC5kZT6J
AJUDBRA0FFkBs1pi61mfMj0BAfDCA/oCfkjrhvRwRCpSL8klJ1YDoUJdmw+v4nJc
pw3OpYXbwKOPLClsE7K3KCQscHel7auf91nrekAwbrXv9Clp0TegYeAQNjw5vZ9f
L6UZ5l3fH8E2GGA7+kqgNWs1KxAnG5GdUvJ9viyrWm8dqWRGo+loDWlZ12L2OgAD
fp7jVZTI1okAlQMFEDQPrLoff6kIA1j8vQEB2XQEAK/+SsQPCT/X4RB/PBbxUr28
GpGJMn3AafAaA3plYw3nb4ONbqEw9tJtofAn4UeGraiWw8nHYR2DAzoAjR6OzuX3
TtUV+57BIzrTPHcNkb6h8fPuHU+dFzR+LNoPaGJsFeov6w+Ug6qS9wa5FGDAgaRo
LHSyBxcRVoCbOEaS5S5EiQCVAwUQM5BktWVgqaw0+fnVAQGKPwP+OiWho3Zm2GKp
lEjiZ5zx3y8upzb+r1Qutb08jr2Ewja04hLg0fCrt6Ad3DoVqxe4POghIpmHM4O4
tcW92THQil70CLzfCxtfUc6eDzoP3krD1/Gwpm2hGrmYA9b/ez9+r2vKBbnUhPmC
glx5pf1IzHU9R2XyQz9Xu7FI2baOSZqJAJUDBRAyCIWZdbtuOHaj97EBAVMzA/41
VIph36l+yO9WGKkEB+NYbYOz2W/kyi74kXLvLdTXcRYFaCSZORSsQKPGNMrPZUoL
oAKxE25AoCgl5towqr/sCcu0A0MMvJddUvlQ2T+ylSpGmWchqoXCN7FdGyxrZ5zz
xzLIvtcio6kaHd76XxyJpltCASupdD53nEtxnu8sRrQxSm9lcmcgV3Vuc2NoIDxq
b2VyZ193dW5zY2hAaW50ZXJmYWNlLWJ1c2luZXNzLmRlPokAlQMFEDIIhfR1u244
dqP3sQEBWoID/RhBm+qtW+hu2fqAj9d8CVgEKJugrxZIpXuCKFvO+bCgQtogt9EX
+TJh4s8UUdcFkyEIu8CT2C3Rrr1grvckfxvrTgzSzvtYyv1072X3GkVY+SlUMBMA
rdl1qNW23oT7Q558ajnsaL065XJ5m7HacgTTikiofYG8i1s7TrsEeq6PtCJKb2Vy
ZyBXdW5zY2ggPGpAdXJpYWguaGVlcC5zYXguZGU+iQCVAwUQMaS91D4gHQUlG9CZ
AQGYOwQAhPpiobK3d/fz+jWrbQgjkoO+j39glYGXb22+6iuEprFRs/ufKYtjljNT
NK3B4DWSkyIPawcuO4Lotijp6jke2bsjFSSashGWcsJlpnwsv7EeFItT3oWTTTQQ
ItPbtNyLW6M6xB+jLGtaAvJqfOlzgO9BLfHuA2LY+WvbVW447SWJAJUDBRAxqWRs
dbtuOHaj97EBAXDBA/49rzZB5akkTSbt/gNd38OJgC+H8N5da25vV9dD3KoAvXfW
fw7OxIsxvQ/Ab+rJmukrrWxPdsC+1WU1+1rGa4PvJp/VJRDes2awGrn+iO7/cQoS
IVziC27JpcbvjLvLVcBIiy1yT/RvJ+87a3jPRHt3VFGcpFh4KykxxSNiyGygl4kA
lQMFEDGCUB31FVv7jlQtXQEB5KgD/iIJZe5lFkPr2B/Cr7BKMVBot1/JSu05NsHg
JZ3uK15w4mVtNPZcFi/dKbn+qRM6LKDFe/GF0HZD/ZD1FJt8yQjzF2w340B+F2GG
EOwnClqZDtEAqnIBzM/ECQQqH+6Bi8gpkFZrFgg5eON7ikqmusDnOlYStM/CBfgp
SbR8kDmFtCZKb2VyZyBXdW5zY2ggPGpAaW50ZXJmYWNlLWJ1c2luZXNzLmRlPokA
lQMFEDHioSdlYKmsNPn51QEByz8D/10uMrwP7MdaXnptd1XNFhpaAPYTVAOcaKlY
OGI/LLR9PiU3FbqXO+7INhaxFjBxa0Tw/p4au5Lq1+Mx81edHniJZNS8tz3I3goi
jIC3+jn2gnVAWnK5UZUTUVUn/JLVk/oSaIJNIMMDaw4J9xPVVkb+Fh1A+XqtPsVa
YESrNp0+iQCVAwUQMwXkzcdm8Q+/vPRJAQEA4QQAgNNX1HFgXrMetDb+w6yEGQDk
JCDAY9b6mA2HNeKLQAhsoZl4HwA1+iuQaCgo3lyFC+1Sf097OUTs74z5X1vCedqV
oFw9CxI3xuctt3pJCbbN68flOlnq0WdYouWWGlFwLlh5PEy//VtwX9lqgsizlhzi
t+fX6BT4BgKi5baDhrWJAJUDBRAyCKveD9eCJxX4hUkBAebMA/9mRPy6K6i7TX2R
jUKSl2p5oYrXPk12Zsw4ijuktslxzQhOCyMSCGK2UEC4UM9MXp1H1JZQxN/DcfnM
7VaUt+Ve0wZ6DC9gBSHJ1hKVxHe5XTj26mIr4rcXNy2XEDMK9QsnBxIAZnBVTjSO
LdhqqSMp3ULLOpBlRL2RYrqi27IXr4kAlQMFEDGpbnd1u244dqP3sQEBJnQD/RVS
Azgf4uorv3fpbosI0LE3LUufAYGBSJNJnskeKyudZkNkI5zGGDwVneH/cSkKT4OR
ooeqcTBxKeMaMuXPVl30QahgNwWjfuTvl5OZ8orsQGGWIn5FhqYXsKkjEGxIOBOf
vvlVQ0UbcR0N2+5F6Mb5GqrXZpIesn7jFJpkQKPU
=97h7
-----END PGP PUBLIC KEY BLOCK-----
```

Índice

Símbolos

.k5login, 382
.k5users, 382
.rhosts, 470
/boot/kernel.old, 244
/etc/gettytab, 547
/etc/group, 342
/etc/login.conf, 339
/etc/mail/access, 619
/etc/mail/aliases, 619
/etc/mail/local-host-names, 619
/etc/mail/mailer.conf, 619
/etc/mail/mailertable, 619
/etc/mail/sendmail.cf, 619
/etc/mail/virtusertable, 619
/etc/remote, 554
/etc/ttys, 549
/usr/bin/login, 546
/usr/share/skel, 333
10 base 2, 703
10 base T, 704
386BSD, 9, 10, 14
386BSD Patchkit, 9
4.3BSD-Lite, 10
4.4BSD-Lite, 5, 7
802.11 (ver Redes sin cables)

A

Abacus, 209
AbiWord, 204
ACL, 411
Acrobat Reader, 207
address redirection, 755
adduser, 333, 517
Advertencias de seguridad en FreeBSD, 414
Agentes de Correo de Usuario, 640
AIX, 705
Alpha, 18, 23, 30, 96
Alpha BIOS, 18
Amanda, 472
amd, 686
anti-aliased fonts, 183
Apache, 8
apagado, 328
aplicaciones
  Maple, 274
  Mathematica, 272
  MATLAB, 276
  Oracle, 280
  SAP R/3, 284
APM, 253
ARC, 18
ASCII, 515
AT&T, 10
Ataques de denegación de servicio (DoS), 354
Ataques DoS (ver Denegación de servicio (DoS))
AUDIT, 435
Auditoría de eventos de seguridad (ver MAC)
automatic mounter daemon, 686
AutoPPP, 581

B

backup software
  dump / restore, 469
BGP, 658
bibliotecas compartidas, 269
binary compatibility
  BSD/OS,
  Linux, 6
  NetBSD,
  SCO,
  SVR4,
BIND, 626, 733
  caching name server, 742
  configuration files, 736
  running in a sandbox, 743
  starting, 735
BIOS, 41
  Sistema Básico de Entrada/Salida, 320
bits-por-second, 533
Blue Mountain Arts, 8
Bluetooth, 668
BOOTP
  diskless operation, 693
bridge, 679

BSD Copyright, 11

C

cajas de arena sandboxes, 350
cargador, 323
cargador de clase en ejecución, 308
cargador-de-arranque (boot-loader), 323
CDROM
  creación, 449
  grabar, 451
CDROMs
  creación cd CD arrancables, 450
  grabar, 452
CHAP, , 576, 582
Chino tradicional
  codificación BIG-5, 516
chpass, 336
chroot, 743
Cisco, 587
clase de sesión, 517
clases de sesión, 516
Coda, 478
codificaciones, 515
códigos de idioma, 514
códigos de país, 514
Colaboradores, 13
colección de ports, 268
committers, 12
compatibilidad binaria
  Linux, 267
compatibilidad binaria con Linux, 267
compilers
  C, 6
  C++,
  Fortran,
compresión, 468
Computer Systems Research Group (CSRG), 7, 14
comsat, 350
concatenación de discos, 503
Concurrent Versions System (ver CVS)
configuración del cargador, 323
consola, 110, 327
consola virtual, 109
contabilidad
  espacio en disco, 484
Contabilidad de procesos, 417
Contraseñas de un solo uso, 358
core team, 12
coredumpsize, 339
correo
  recepción, 617
correo electónico, 615
cortafuegos, 613
cpio, 471
cputime, 340
crypt, 357
cu, 102
cuaa, 538
cuentas
  añadir, 333
  cambiar contraseña, 337
  daemon, 332
  eliminar, 335
  grupos, 342
  limitar, 339
  modificar, 332
  nobody, 332
  operator, 332
  sistema, 332
  superusuario (root), 331
  usuario, 332
cuotas, 339
cuotas de disco, 339, 484
  límites, 486
  revisando, 487
  revisión, 485
CVS
  repository, 12

D

DCE, 534
default route, 601, 654, 656
Denegacion de servicio (DoS), 346
DES, 357
device nodes, 215
device.hints, 326
DGA, 220
DHCP
  configuration files, 729, 732
  dhcpd.conf, 731
  diskless operation, 692

installation, 730
requirements, 728
server, 729
dial-in service, 545
dial-out service, 553
Dirección IP estática, 570
direcciones estáticas, 610
directorio esqueleto, 333
directorios, 114
discos
añadir, 438
cifrado, 488
desconectar un disco de memoria, 483
file-backed (4.X), 478
file-backed (5.X), 479
memoria, 478
sistemas de ficheros en memoria (4.X), 481
sistemas de ficheros en memoria (5.X), 482
virtuales, 478
discos flexibles, 468
discos virtuales, 478
Disk Mirroring, 497
disklabel, 473, 477
diskless operation, 689
/usr read-only, 699
kernel configuration, 696
diskless workstation, 689
DNS, 582, 616, 632, 733
records, 740
DNS Server,
domain (nombre de dominio), 601
DOS, 27, 40, 521
DSL, 680
DSP, 215
DTE, 533
dual homed hosts, 657
dump, 469
DVD
DVD+RW, 459
DVD-RW, 460
DVD-Video, 458
grabar, 456
Dynamic Host Configuration Protocol (ver DHCP)

E

editores, 142
editores de texto, 142
editors
ee, 142
emacs, 142
vi, 142
ee, 142
electronic mail (ver email)
ELF, 308
marcado, 309
emacs, 142
email, 615
cambio de mta, 622
configuration, 630
Depuración de problemas, 626
enlaces simbólicos, 269
Etherboot, 694
Ethernet, 607
dirección MAC, 608
MAC address, 273, 654

F

fdisk, 438
ficheros rc, 328
filesize, 340
finger, 350
Firefox, 201
firewall, 7, 679, 680
fix-it floppies, 473
fonts, 274
anti-aliased, 183
spacing, 184
TrueType, 182
Fonts
LCD screen, 185
Free Software Foundation, 10, 14, 147
FreeBSD Project
development model, 12
goals, 11
history, 9
FreshMeat, 152
FreshPorts, 152
FTP
anonymous, 68, 89
modo pasivo, 60

via a HTTP proxy, 60
FTP servers, 7

G

gated, 607, 610
gateway, 654
GEOM, 495, 496, 497
GEOM Disk Framework (ver GEOM)
getty, 546
GNOME, 189
anti-aliased fonts, 190
GNU Compiler Collection, 8
GNU General Public License (GPL), 11
GNU Lesser General Public License (LGPL), 11
GNU toolchain, 271
GnuCash, 208
Gnumeric, 209
GQview, 208
grabadora de CD
ATAPI, 449
controlador ATAPI/CAM, 450
Grabadora de CD
controlador ATAPI/CAM, 455
Greenman, David, 9
Grimes, Rod, 9
grupos, 342
gv, 207
gzip, 468

H

HCI, 670
horizontal scan rate, 176
hostname, 572
hosts, 572
HP-UX, 705
Hubbard, Jordan, 9

I

I/O port, 215
ICMP_BANDLIM, 355
IEEE, 471
ifconfig, 668
IKE, 398
image scanners, 230
IMAP, 616, 617
impresoras, 523
imprimir, 263
inicio, 319
init, 320, 327
instalación, 17
installation
floppies, 106
from MS-DOS, 107
from QIC/SCSI Tape, 107
headless (serial console), 101
network
Ethernet, 108
FTP, 60, 105
NFS, 108
parallel (PLIP), 108
serial (SLIP or PPP), 108
troubleshooting, 97
Intel i810 graphic chipset, 179
internacionalización (ver localización)
Internet connection sharing, 751
Internet Software Consortium (ISC), 727
IP masquerading, , 751
IP subnet, 679
IPCP, 577
IPsec, 390
AH, 391
ESP, 391
políticas de seguridad, 398
IPX/SPX, 705
IRQ, 215
ISA, 212
ISDN, 680
cards, 700
stand-alone bridges/routers, 703
ISO 9660, 449
ISP, 570, 576

J

jails, 419
jerarquía de directorios, 119
JMA Wired, 9
Jolitz, Bill, 9

K

KDE, 191
display manager, 192

Kerberos5
  Centro de distribución de llaves, 377
  configurar clientes, 381
  habilitación de servicios, 380
  historia, 377
  limitaciones y deficiencias, 385
  recursos externos, 386
  solución de problemas, 383
KerberosIV, 350, 356
  encendido inicial, 370
  instalación, 368
kermit, 587, 588
kernel, 320
  compilación / instalación, 240
  Compilar un kernel a medida, 237
  configuración, 605
  configuration, 212, 752
  configuration file, 244, 245
  drivers / modules / subsystems, 240
  interacción de arranque, 326
  NOTES, 244
  opciones de arranque, 326
kernel options
  cpu, 245
  ident, 245
  LINUX, 268
  machine, 245
  MSDOSFS, 247
  NFS, 247
  NFS_ROOT, 247
  options BRIDGE, 680
  SMP, 249
kernel.old, 325
KLD (objeto cargable del kernel), 268
KOffice, 203
Konqueror, 203

L

L2CAP, 671
LCD screen, 185
LCP, 581
limitar a los usuarios, 339
  coredumpsize, 339
  cputime, 340
  cuotas, 339
  filesize, 340
  maxproc, 340
  memorylocked, 340
  memoryuse, 341
  openfiles, 341
  sbsize, 341
  stacksize, 341
límite duro, 486
límite suave, 486
línea de órdenes, 139
Linux, 705
  binarios ELF, 271
  instalación de bibliotecas Linux, 268
LISA, 472
locale, 514, 518
locales, 292, 516
localización, 513
  Alemán, 525
  Chino tradicional, 525
  Coreano, 525
  Japonés, 525
  Ruso, 522
login name,
loopback device, 654
ls, 114

M

MAC, 433
MacOS, 362
MAKEDEV, 539
Mandatory Access Control (ver MAC)
mapa de pantalla, 519
mapa de teclado, 519
maxproc, 340
MD5, 357
mecanismo de arranque (bootstrap), 319
medios de cinta, 464
  AIT, 467
  cintas DDS (4mm), 464
  cintas Exabyte (8mm), 465
  cintas QIC, 465
  DLT, 466
  QIC-150, 466
memory protection, 6
memorylocked, 340
memoryuse, 341
mencoder, 226

Método de introducción X11 (XIM), 521
mgetty, 581
MiContraseña, 583
Microsoft Windows, 40
MIME, 516, 518
MIT, 368
modem, 545, 587, 701
módem, 604
modo de arranque mono-usuario, 325
modo mono-usuario, 327
modo multi-usuario, 328
mount, 102, 477
mountd, 682
moused, 519
Mozilla, 186, 200
MPlayer
  making, 224
  use, 225
MS-DOS, 362
multi-user facilities, 5
multicast
  options MROUTING, 661
MX record, 627, 628, 631, 742
MySQL, 522

N

nameserver, 601
natd, 751
Net/2, 10, 10
NetBIOS, 582
NetBSD, 14, 705
Netcraft, 8
netgroups, 718, 720
network address translation, 679
newfs, 477
NFS, 478, 488, 681
  configuration, 683
  diskless operation, 695
  export examples, 683
  mounting, 685
  server, 682
  uses, 685
nfsd, 682
NIS, 705
  client, 708
  client configuration, 715
  domainname, 709
  domains, 706
  maps, 711
  master server, 707
  password formats, 726
  server configuration, 710
  slave server, 708, 712
NOTES, 244
Novell, 10
ntalk, 350
NTP, 748
  choosing servers, 748
  configuration, 748
  ntp.conf, 749
  ntpd, 748
ntpdate, 748
null modem cable, 102
null-modem cable, 534, 558

O

OBEX, 677
office suite
  KOffice, 203
  OpenOffice.org, 205
opciones de kernel
  FAST_IPSEC, 390
  IPSEC, 391, 398
  IPSEC_DEBUG, 391
  IPSEC_ESP, 391
OpenBSD, 14, 705
openfiles, 341
OpenOffice.org, 205
OpenSSH, 405
  cliente, 405
  configuración, 406
  copia segura, 406
  habilitar, 405
  túneles, 408
OpenSSL
  generación de certificados, 387
Opera, 202
OS/2, 440
OSPF, 658

P

packages, 149

deleting, 155
installing, 153
páginas de manual, 145
Pair Networks, 8
pairing, 673
PAP, 570, 576, 582
Parallel Line IP, 761
partición raíz, 477
particiones, 129, 438
particiones BSD, 439
passwd, 337
password,
pax, 471
PCI, 212
PDF
viewing, 207, 207, 207
peligrosamente dedicadas, 129
periodo de gracia, 488
permisos, 114
symbolic, 115
permisos de fichero, 114
personalización del kernel, 281
Personalización del kernel, 293
Physical Address Extensions (PAE)
large memory, 259
pkg_add, 153, 154
pkg_delete, 155
PLIP, 761
POP, 616, 617
Portaudit, 413
portmanager, 165
portmap, 682, 706
portmaster, 165
ports, 149
disk-space, 167
installing, 158
removing, 163
upgrading, 164
portupgrade, 164
POSIX, 471, 516
PostScript
viewing, 207
PPP, 569, , 627, 702
cliente, 587
con dirección IP fija, 574
con direcciones IP dinámica, 577
configuración, 573, 584
Extensiones Microsoft, 582
kernel PPP, 586
recibiendo llamdas externas, 578
servidor, 586
sobre ATM, 596
sobre Ethernet, 594
user PPP, 575
PPP shells, 579, 580
PPPoA (ver PPP, sobre ATM)
PPPoE (ver PPP, sobre Ethernet)
preemptive multitasking, 5
pw, 338, 518

R

racoon, 399
RAID
CCD, 441
hardware, 445
software, 441, 445
Software, 503
Vinum, 445
RAID-1, 504
rc files
rc.serial, 539, 550
RDSI, 700
Red privada virtual (ver VPN)
Redes por TCP/IP, 607
redes TCP/IP, 603
Registro Maestro de Arranque (RMA), 321
registro MX, 617
Réplica de discos, 504
resolver, 572, 734
respaldo en discos flexibles, 468
restore, 469
reverse DNS, 734
RFCOMM, 673
RIP, 611, 658
rlogind, 350
rmuser, 335
root file system
diskless operation, 697
root zone, 734
routed, 584
router, 8, 658, 680
routing, 654

routing propagation, 660
RPM, 289
RS-232C cables, 534, 535
rshd, 350

S

sbsize, 341
scp, 406
scripts de inicio, 111
SCSI, 41
SDL, 220
SDP, 674
seguridad, 345
  asegurar FreeBSD, 348
  Ataques DoS (ver Denegación de servicios (DoS))
  compromiso de cuentas, 347
  Contraseñas de un solo uso, 358
  cortafuegos, 613
  crypt, 357
  OpenSSH, 405
  OpenSSL, 386
  puertas traseras, 347
sendmail, 351, 585, 619
serial communications, 533
serial console, 101, 558
servidor de correo, 618
Servidor de tipos de letra True Type de X11, 521
Servidor DNS, 570
setkey, 398
shells, 139
Shells Bourne, 141
Sistema de cola LPD, 263
sistema de ficheros
  ISO 9660, 449
sistema de ficheros raíz, 131
sistemas de ficheros
  desmontar, 134
  HFS, 450
  ISO 9660, 450
  Joliet, 450
  montaje, 132
  montaje con fstab, 131
  snapshots, 483
slices, 129, 438
SLIP, 569, 599, 606, 607
  cliente, 599
  conectando con, 601
  enrutamiento, 609
  servidor, 603
SMTP, 585, 630
software de respaldo
  Amanda, 472
  cpio, 471
  pax, 471
  tar, 470
Solaris, 308, 705
Sony Japan, 8
Sophos Anti-Virus, 9
sound cards, 212
source code, 6
spreadsheet
  Abacus, 209
  Gnumeric, 209
SRM, 18
SSH
  sshd, 68
ssh, 356
sshd, 350
stacksize, 341
Striping, 496
striping de discos, 503
su, 348, 439
subnet, 654, 655
Subversion (ver SVN)
SunOS, 709
Supervalu, 8
SVN
  repository, 12
Symmetric Multi-Processing (SMP), 6
sysctl, 352
sysinstall, 519, 728
  añadir discos, 439
sysutils/cdrtools, 449

T

tar, 468, 471
TCP Wrappers, 364
TCP/IP networking, 6
tcpwrapper, 717
TELEHOUSE America, 9

telnetd, 350
terminales, 109
terminals, 539
TeX, 265
TFTP
  diskless operation, 695
The GIMP, 204
timeout, 576
Tipos TrueType, 182
trabajos de impresión, 264
traceroute, 661
transporte
  exim, 616
  postfix, 616
  qmail, 616
  sendmail, 616
Tru64 UNIX, 711
ttyd, 538
TV cards, 229

U

U.C. Berkeley, 7, 10, 10, 14
UDP, 727
Unicode, 521
UNIX, 113
Unix,
USB
  discos, 447
USENET, 7
users
  large sites running FreeBSD, 8
Uso de fetchmail, 648
Uso de procmail, 649
UUCP, 627

V

variables de entorno, 140, 140
vertical scan rate, 176
vi, 142
video packages, 223
video ports, 223
Vinum, 503
  concatenación, 503
  réplicas, 504
  striping, 503
vipw, 517
virtual memory, 6
VPN, 391
  creación, 392

W

Walnut Creek CDROM, 9
Weathernews, 8
web browsers
  Mozilla (ver Mozilla)
web servers, 7
WEP, 667
wheel, 349
widescreen flatpanel configuration, 180
Williams, Nate, 9
Windows, 362
Windows NT, 706
wireless networking, 662
  encryption, 666
  Punto de acceso, 663

X

X Display Manager, 186
X Window System, 6
  (ver también XFree86)
  Accelerated-X,
  XFree86, 8
X11, 176
X11 tuning, 177
XML, 183
Xorg, 176
xorg.conf, 178
Xpdf, 207
XVideo, 220

Y

Yahoo!, 8
yellow pages (ver NIS)

Z

zip drive, 255
zones
  examples, 734

Colofón

Este libro es el resultado del trabajo de cientos de colaboradores del “Proyecto de Documentación de FreeBSD”. El texto esta escrito en SGML de acuerdo al DTD DocBook y es formateado automáticamente en diferentes formatos de presentación usando Jade, una herramienta DSSSL de software libre. Se han usado las hojas de estilo DSSSL de Norm Walsh con personalización de niveles para las instrucciones de presentación de Jade. La versión impresa de este documento no hubiese sido posible sin los programas TeX de Donald Knuth, LaTeX de Leslie Lamport y JadeTeX de Sebastian Rahtz.