

Guía del Usuario Avanzado Solaris

2550 Garcia Avenue
Mountain View, CA 94043
U.S.A.

Part No: 801-7484-10
Revision A, Agosto 1994



© 1994 Sun Microsystems, Inc.
2550 Garcia Avenue, Mountain View, California 94043-1100 U.S.A.

Todos los derechos reservados. Este producto y la documentación relacionada están protegidos por derecho de autor y distribuidos bajo licencias que restringen su uso, copia, distribución y descompilación. Ninguna parte de este producto o documentación relacionada puede ser reproducido en cualquier forma o medio sin previa autorización por escrito de Sun y sus concesionarios, si los hubiera.

Partes de este producto pueden derivarse de los sistemas UNIX® y Berkeley 4.3 BSD, bajo licencia de UNIX Systems Laboratories, Inc., un subsidiario perteneciente en su totalidad a Novell, Inc., y la Universidad de California respectivamente. El software de fuentes de terceros en este producto está protegido por derechos de autor y bajo licencia de los proveedores de fuentes de Sun.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the United States Government is subject to the restrictions set forth in DFARS 252.227-7013 (c)(1)(ii) and FAR 52.227-19.

El producto descrito en este manual puede estar protegido por una o más patentes de Estados Unidos, patentes extranjeras o solicitudes de patentes.

MARCAS REGISTRADAS

Sun, el logotipo de Sun, Sun Microsystems, Sun Microsystems Computer Corporation, SunSoft, el logotipo de SunSoft, Solaris, SunOS, OpenWindows, DeskSet, ONC, ONC+ y NFS son nombres comerciales o marcas registradas de Sun Microsystems, Inc. en EE.UU y otros países. UNIX y OPEN LOOK son marcas registradas de UNIX System Laboratories, Inc. una filial propiedad en su totalidad de Novell, Inc. PostScript y Display Postscripts son marcas registradas de Adobe Systems, Inc. Todos los demás nombres de productos aquí mencionados son las marcas registradas de sus respectivos propietarios.

Todas las marcas registradas de SPARC, incluyendo el logotipo de SCD Compliant, son nombres comerciales o marcas registradas de SPARC International, Inc. SPARCstation, SPARCserver, SPARCengine, SPARCstorage, SPARCware, SPARCcenter, SPARCclassic, SPARCcluster, SPARCdesign, SPARC811, SPARCprinter, UltraSPARC, microSPARC, SPARCworks y SPARCcompiler son licencias exclusivas para Sun Microsystems, Inc. Los productos que llevan las marcas registradas de SPARC están basados en una arquitectura desarrollada por Sun Microsystems, Inc.

Las interfaces de usuario gráficas OPEN LOOK® y Sun™ Graphical fueron desarrolladas por Sun Microsystems, Inc. para sus usuarios y concesionarios. Sun reconoce los esfuerzos pioneros de Xerox en la investigación y desarrollo del concepto de interfaces de usuarios visuales o gráficas para el sector informático. Sun mantiene una licencia no exclusiva de Xerox para la Interface de Usuario Gráfica Xerox, esta licencia cubre también a los concesionarios de Sun que implementen OPEN LOOK GUIs y que de cualquier otro modo, cumplan con los acuerdos de licencia escrita de Sun.

X Window System es un producto del Instituto de Tecnología de Massachusetts.

ESTA PUBLICACIÓN SE PROPORCIONA "TAL COMO ESTÁ" SIN GARANTÍA DE NINGUNA CLASE, EXPRESA O IMPLÍCITA, INCLUYENDO, PERO NO LIMITÁNDOSE, A LAS GARANTÍAS IMPLÍCITAS O DE COMERCIALIZACIÓN, Y LA IDONEIDAD PARA UN PROPÓSITO PARTICULAR O LA NO INFRACCIÓN.

ESTA PUBLICACIÓN PODRÍA INCLUIR INEXACTITUDES TÉCNICAS O ERRORES TIPOGRÁFICOS. LOS CAMBIOS SE AÑADEN PERIÓDICAMENTE A LA INFORMACIÓN AQUÍ CONTENIDA; ESTOS CAMBIOS SE INCORPORARÁN EN NUEVAS EDICIONES DE LA PUBLICACIÓN. SUN MICROSYSTEMS, INC. PUEDE REALIZAR MEJORAS Y/O CAMBIOS EN LOS PRODUCTOS Y/O LOS PROGRAMAS DESCRITOS EN ESTA PUBLICACIÓN EN CUALQUIER MOMENTO.



Papel
Reciclable

Contenido

Acerca de este libro	xiii
Convenciones propias de cada plataforma	xvi
1. Cómo entrar en SunOS y empezar OpenWindows.	1
Cómo entrar	1
Shell de entrada	2
Cómo salir.	3
Equivalencias del teclado	4
Consideraciones previas a la puesta en marcha de OpenWindows	5
La variable de entorno OPENWINHOME	6
Cómo utilizar el archivo de inicio correcto	7
Cómo empezar el entorno de usuario OpenWindows	9
Cómo ver el software de OpenWindows	9
Si OpenWindows no se ve	11
Cómo abandonar el entorno OpenWindows	12
Opciones de entrada especiales para OpenWindows	12

Cómo empezar con la seguridad de red limitada	13
Cómo empezar con varios tipos de monitores.	13
<code>SPARC</code> Cómo empezar con pantallas múltiples	14
2. Comandos básicos de SunOS	17
Indicador de comando	17
Cómo introducir comandos	18
Cómo corregir errores de escritura	18
Cómo introducir comandos múltiples y comandos largos	19
Cómo repetir comandos anteriores.	20
Cómo agregar opciones a los comandos.	21
Cómo redireccionar y enviar por el canal de comunicación la salida del comando	21
Cómo ejecutar comandos en el fondo.	22
Cómo obtener ayuda con los comandos del SO.	23
Cómo mostrar páginas de manual con <code>man</code>	24
Cómo mostrar un resumen de una línea con <code>what is</code> .	24
Búsqueda de una palabra clave con <code>apropos</code>	25
3. Trabajar con archivos y directorios	27
Conceptos de archivo	27
Uso de los comandos de archivo	28
Antes de comenzar	28
Crear un archivo de prueba	29
Listar archivos (<code>ls</code>)	29
Copiar archivos (<code>cp</code>)	29

Mover y renombrar archivos (<code>mv</code>)	30
Eliminar archivos (<code>rm</code>)	31
Mostrar el contenido del archivo (<code>more, cat</code>)	31
Mostrar el tipo de archivo (<code>file</code>)	32
Directorios y jerarquía.	32
Jerarquía de directorio	33
Imprimir el directorio en uso (<code>pwd</code>)	34
Su directorio de usuario	34
Cambiar el directorio en uso (<code>cd</code>)	34
Crear un directorio (<code>mkdir</code>)	37
Nombres de ruta de acceso relativos.	37
Mover y renombrar directorios	38
Copiar directorios	38
Eliminar directorios (<code>rmdir</code>)	39
Ver las diferencias entre directorios (<code>diff</code>)	39
Comparar tres archivos diferentes (<code>diff3</code>)	41
Uso de <code>bdiff</code> con archivos grandes	41
Buscar archivos (<code>find</code>)	41
Seguridad de archivos y directorios	44
Mostrar permisos y estado (<code>ls -l</code>)	45
Listar archivos “ocultos” (<code>ls -a</code>)	46
Cambiar permisos (<code>chmod</code>)	47
Establecer permisos absolutos	50
4. Cómo buscar archivos	55

Cómo buscar patrones con <code>grep</code>	55
<code>grep</code> como filtro	56
<code>grep</code> cadenas de palabras múltiples.....	58
Cómo buscar líneas sin una cadena de caracteres determinada.....	58
Más acerca de <code>grep</code>	59
Cómo buscar metacaracteres	60
Comillas dobles o sencillas en líneas de comando	62
5. Contraseñas, procesos y almacenamiento en disco	63
Uso de una contraseña	63
Cómo cambiar su contraseña.....	64
Caducidad de la contraseña.....	65
Procesos y números de procesos	66
Qué comandos se ejecutan actualmente (<code>ps</code>).....	66
Terminar procesos (<code>kill</code>)	67
Cómo administrar el almacenamiento en disco.....	68
Mostrar la utilización del disco (<code>df -k</code>).....	68
Mostrar la utilización del directorio (<code>du</code>).....	68
6. Uso del editor <code>vi</code>.....	71
Cómo empezar <code>vi</code>	72
Cómo crear un archivo	72
La línea de estado	73
Los dos modos de <code>vi</code>	73
Modo de entrada.....	73

Modo de comando.	74
Cómo finalizar una sesión	75
Cómo guardar los cambios y salir de vi	75
Cómo imprimir un archivo.	77
Comandos vi básicos	77
Cómo desplazarse por un archivo.	77
Cómo insertar texto.	81
Cómo cambiar texto	82
Cómo deshacer los cambios.	83
Cómo eliminar texto	84
Cómo copiar y mover texto -Tirar, eliminar y poner ..	85
Uso de un contador para repetir comandos.	87
Uso de los comandos ex.	87
Cómo activar y desactivar la numeración de las líneas	87
Cómo copiar líneas	88
Cómo mover líneas	89
Cómo eliminar líneas	90
Cómo buscar y reemplazar con vi	90
Cómo buscar una cadena de caracteres	90
Búsqueda más exacta	91
Cómo reemplazar una cadena de caracteres	92
Cómo ir a una línea específica	93
Cómo insertar un archivo en otro	94
Cómo editar archivos múltiples.	94

Cómo editar una serie de archivos	95
Cómo copiar líneas entre archivos	95
Cómo establecer parámetros vi	96
Recuperación tras una caída del sistema	96
Resumen de los comandos básicos de vi.....	97
7. Uso del correo electrónico	101
Datos básicos de mailx	101
Cómo empezar mailx	102
Cómo enviarse Vd. mismo una carta de prueba	102
Cómo leer su carta de prueba	104
Cómo salir de mailx	105
Cómo leer cartas	106
Cómo eliminar (y rehacer) cartas.....	107
Cómo imprimir cartas.....	109
Cómo enviar cartas	109
Cartas no entregables	111
Cómo cancelar una carta no enviada	112
Cómo agregar copias al carbón y borradores al carbón	112
Cómo insertar una copia de una carta o un archivo ...	113
Cómo responder a una carta	114
Cómo guardar y recuperar cartas	115
Cómo guardar y copiar cartas en archivos.....	115
Cómo leer cartas guardadas en archivos y carpetas ...	118
Uso de vi en mailx	119

Alias de correo electrónico	119
Cómo establecer alias de correo electrónico en <code>.mailrc</code>	120
Cómo establecer alias de correo electrónico en <code>/etc/aliases</code>	122
Comandos de tilde.....	125
Cómo obtener ayuda: otros comandos de <code>mailx</code>	126
8. Uso de las impresoras.....	129
Cómo encargar solicitudes de impresión.....	129
Cómo encargar solicitudes de impresión a la impresora predeterminada.....	130
Cómo encargar solicitudes de impresión utilizando un nombre de impresora	130
Cómo solicitar un aviso cuando la impresión esté completada.....	131
Cómo imprimir copias múltiples.....	132
Cuadro resumen de las opciones <code>lp</code>	132
Cómo determinar el estado de la impresora	133
Cómo comprobar el estado de las solicitudes de impresión	133
Cómo comprobar las impresoras disponibles	134
Cómo mostrar toda la información de estado	134
Cómo mostrar el estado de las impresoras	135
Mostrar las características de las impresoras.....	136
Tabla resumen de las opciones <code>lpstat</code>	137
Cómo cancelar una solicitud de impresión	138

Cómo cancelar una solicitud de impresión mediante el número de identificación	139
Cómo cancelar una solicitud de impresión mediante el nombre de la impresora	139
9. Uso de la red	141
Conceptos acerca de la red	142
Conectar con el sistema de forma remota (<code>rlogin</code>).....	142
<code>rlogin</code> sin un directorio de usuario	143
<code>rlogin</code> usando el nombre de otro usuario	144
<code>rlogin</code> con un sistema desconocido	144
Abandonar una conexión <code>rlogin</code>	145
Suspender una conexión <code>rlogin</code>	146
Verificar su posición (<code>who am i</code>)	146
Copiar archivos a distancia (<code>rcp</code>)	147
Copiar de otro sistema al suyo.	147
Copiar de su sistema a otro	148
Ejecutar comandos a distancia (<code>rsh</code>)	148
Ver información de usuario (<code>rusers</code>)	149
10. Personalizar su entorno de trabajo	151
Archivos de inicialización	151
Variables de entorno	152
El perfil del usuario.	153
Cómo establecer la variable <code>PATH</code>	155
Alias (Sólo C Shell)	156
Cómo cambiar el indicador de comando	157

Otras variables útiles.	158
Cómo establecer permisos de archivos predeterminados .	159
Cómo configurar las fuentes de OpenWindows	161
Cómo especificar el tipo de fuente y el tamaño del punto.	161
Cómo listar las fuentes disponibles.	164
A. Cómo cambiar a la versión 3.3 de OpenWindows, o posterior.	167
<u>SPARC</u> Cómo cambiar desde el entorno SunView	167
<u>SPARC</u> Los archivos .defaults y .xdefaults	167
Cómo cambiar desde una versión del entorno OpenWindows anterior a la 3.3	168
La variable de entorno OPENWINHOME	168
El archivo .xinitrc	169
Uso del archivo de inicio correcto.	170
Ventana de propiedades del área de trabajo.	171
Cómo personalizar el menú del área de trabajo	172
B. Cómo modificar el teclado.	173
Cómo activar/desactivar la tecla Componer	173
<u>SPARC</u> Remapeo del teclado para zurdos	174
<u>SPARC</u> Uso del archivo de comandos de remapeo	174
<u>SPARC</u> Cómo deshacer el remapeo del teclado	177
C. Cómo ejecutar aplicaciones en red.	181
Uso de rlogin para ejecutar una aplicación en red.	182
Más acerca de la seguridad.	183

Quién debe leer esta sección	183
Mecanismos de control de acceso	184
Protocolos de autorización	185
Manipulación del acceso al servidor	187
Cómo procesar clientes remota o localmente como otro usuario	191
D. SPARC - Interconexión de redes DECnet (DNI)	193
Cómo instalar la interconexión de redes DECnet	194
Cómo mostrar un cliente remoto en un sistema OpenWindows	195
Cómo mostrar un cliente remoto en un VAX	195
E. Configurar impresoras y módems	199
Configurar un módem para su sistema.	199
Para información adicional sobre módems	201
Configurar una impresora para usar en su sistema.	202
Para información adicional sobre impresoras	204
Índice	205

Acerca de este libro

Quién debe leer este libro

Este libro está dirigido a los usuarios del sistema de software Solaris™. El sistema de software Solaris está formado por SunOS™ y OpenWindows™.

Antes de leer este libro

Su sistema debe estar instalado y listo para usarse. Si no lo está, antes de continuar, consulte *SPARC: Installing Solaris Software*, *x86: Installing Solaris Software* o el manual de instalación correspondiente a su sistema.

Lectura adicional

Los Libros de respuesta del sistema de software Solaris dan acceso a una serie de libros acerca del software de Solaris, que están organizados en los siguientes grupos:

- **Conjunto para los administradores Sun**
Ofrece información detallada sobre la instalación y administración del sistema para una serie de configuraciones de sistema, incluyendo grandes redes de estaciones de trabajo Sun.
- **Conjunto para el personal de desarrollo Sun**
Contiene la información necesaria para el desarrollo de las aplicaciones utilizadas para escribir, depurar y mantener programas en el sistema.

-
- **Conjunto de manuales de referencia Sun**
Contiene la descripción de cada comando SunOS. También se le llama páginas de `manual`, y puede instalarse como documentación en línea.
 - **Conjunto para el usuario Sun**
Ofrece una descripción detallada de diversos aspectos del sistema SunOS, incluyendo el uso de los comandos SunOS, el funcionamiento de OpenWindows, la personalización de su entorno de trabajo, los problemas de manejo, la escritura de archivos de comandos shell, el uso del correo electrónico y el funcionamiento de la red.

Si desea más información acerca de los Libros de respuesta del sistema de software Solaris, consulte *Guía de Documentación*.

Convenciones tipográficas

En esta guía se utilizan las siguientes convenciones tipográficas:

- El tipo de letra *cursiva* se usa para resaltar palabras, en argumentos de comando, variables y títulos de libros. Por ejemplo:

Debe escribir el argumento *nombre_de_archivo* cómo se describe en *man Pages(1): User Commands*.

- El tipo de letra `Courier bold` indica lo que Vd. escribe en una sesión de prueba. Por ejemplo:

```
$ date
```

- El tipo de letra `Courier` se utiliza en listados de programas y en nombres de comandos, de programas y de sistemas. Por ejemplo:

Puede convertirse en un superusuario escribiendo `root` tras el indicador de entrada, o escribiendo el comando `su` tras el indicador de comando.

También se usa en textos que el sistema muestra en pantalla. Por ejemplo:

```
$ who  
bozo      console      Sep 11 15:36
```

-
- Los nombres de los botones del ratón se denominan según su función, no según su ubicación, y se escriben en mayúscula. Por ejemplo, se indica “Presione SELECCIONAR”, en vez de “Presione el botón izquierdo.”

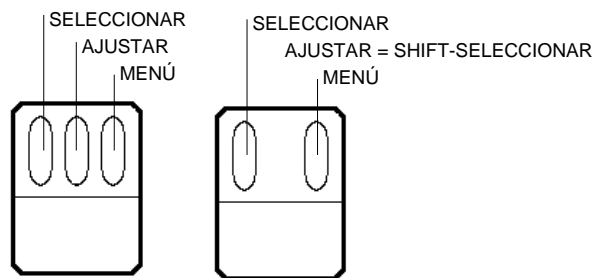


Ilustración P-1 Funciones predeterminadas de los botones del ratón

En un ratón de tres botones, las asignaciones de funciones predeterminadas son:

- SELECCIONAR = el botón izquierdo del ratón
- AJUSTAR = el botón central del ratón
- MENÚ = el botón derecho del ratón

En un ratón de dos botones, las asignaciones de funciones predeterminadas son:

- SELECCIONAR = el botón izquierdo del ratón
- MENÚ = el botón derecho del ratón

Para usar la función AJUSTAR en un ratón de dos botones, presione y mantenga pulsada la tecla Mayúscula del teclado y haga clic con el botón SELECCIONAR del ratón.

- Los nombres de las teclas del teclado empiezan por mayúscula (por ejemplo, Presione la tecla Mayúscula). La tecla llamada Retorno también se denomina Intro en algunos teclados.

Nota – Solaris soporta sistemas SPARC y sistemas basados en x86, que tienen teclados diferentes. Consecuentemente, algunas equivalencias del teclado de comandos existentes en los menús de OpenWindows son distintas, según cada teclado.

Para la mayoría de las equivalencias del teclado, Solaris soporta una combinación con la tecla Meta que puede usarse en las dos plataformas.

Consulte Capítulo 1, “Cómo entrar en SunOS y empezar OpenWindows” si desea una lista de las funciones de comandos y sus diferentes equivalencias del teclado.

- Los comandos de menú OpenWindows que se ejecutan utilizando equivalencias del teclado (también llamadas *aceleradores de teclado*), se representan en este manual mediante la siguiente convención: un guión (-) indica que debe presionar dos teclas a la vez.

Por ejemplo, Ctrl-Alt significa que debe presionar y mantener pulsadas las teclas Control y Alt a la vez; Ctrl-Alt-c significa que debe presionar y mantener pulsadas las teclas Control y Alt a la vez, y presionar simultáneamente c.

- Las equivalencias del teclado se obtienen presionando la tecla Meta y una o más teclas adicionales del teclado. En los teclados SPARC la tecla Meta es la tecla con el símbolo <> ; en los teclados x86 la tecla Meta se obtiene presionando las teclas Ctrl y Alt simultáneamente (Ctrl-Alt). Si desea más información acerca de las equivalencias del teclado, consulte el Capítulo 1, “Cómo entrar en SunOS y empezar OpenWindows”.
- Esta guía utiliza la siguiente notación para explicar cómo seleccionar una opción de menú. Por ejemplo: seleccione Área de trabajo ► Programas.
- Cuando se trate de un menú completo, o de un elemento de menú predeterminado, un *anillo predeterminado*, bordeará el elemento predeterminado del menú. Como valor predeterminado, si se hace clic SELECCIONAR en un botón de menú, aparecerá el menú completo.

Convenciones propias de cada plataforma

Los siguientes símbolos que preceden al título de una sección o procedimiento indican que dicha sección o procedimiento pertenecen sólo a esa determinada plataforma:

SPARC Indica una sección o procedimiento propios de la plataforma SPARC

x86 Indica una sección o procedimiento propios de la plataforma x86

Las notas particulares de cada plataforma se indican de este modo:

SPARC – Esta nota es particular de la plataforma SPARC.

x86 – Esta nota es particular de la plataforma x86.

Los puntos particulares de cada plataforma se indican de este modo:

- **SPARC**: Este punto es particular de la plataforma SPARC.
- **x86**: Este punto es particular de la plataforma x86.

Si una muestra de código, tabla o ilustración pertenece a una determinada plataforma, el nombre de dicha plataforma aparecerá al principio del título. Por ejemplo:

Tabla 1-1 SPARC: Título particular de la plataforma SPARC

Tabla 1-2 x86: Título particular de la plataforma x86

Cómo entrar en SunOS y empezar OpenWindows



El software de la versión de OpenWindows debe estar ya instalado en su disco duro o en un servidor accesible desde su sistema de archivos. Si no está seguro de ello, contacte con su administrador de sistemas, o consulte *SPARC: Installing Solaris Software*, *x86: Installing Solaris Software*.

En este capítulo se explica cómo entrar en el sistema, cómo usar un intérprete de comandos shell, cómo montar y empezar el entorno de usuario OpenWindows, cómo abandonar el sistema de ventanas, y cómo salir. También se explican algunos casos especiales, como la configuración del entorno OpenWindows para monitores duales.

Cómo entrar

Una *sesión de trabajo estándar* se define como el intervalo que transcurre entre el momento de entrada en el sistema y el de salida. El entorno multiusuario SunOS necesita que Vd. se identifique cada vez que quiera utilizar el sistema. Su *nombre de entrada* (también conocido como *nombre de usuario* o *cuenta de acceso*) le identifica para el sistema y para otros usuarios en el sistema. Su *contraseña* restringe el uso de su cuenta de acceso a aquellos que conocen dicha contraseña. Si todavía no tiene nombre de entrada ni contraseña, pida a la persona designada como *administrador de sistemas* que le asigne una contraseña. Una vez obtenida dicha información estará preparado para entrar.

Antes de que entre en el sistema, su pantalla deberá mostrar algo parecido a lo siguiente:

```
login:
```

Introduzca el nombre de entrada que le ha asignado el administrador de sistemas y presione la tecla Retorno. Si por ejemplo su nombre de entrada es `jose`, escriba:

```
login: jose
```

y presione Retorno. A continuación el sistema solicita así su contraseña:

```
login: jose
Password:
```

Escriba su contraseña tras el indicador y presione Retorno (si su cuenta de acceso no tiene asignada una contraseña, el sistema le introduce sin pedirle ninguna). Observe que a medida que Vd. escribe su contraseña, el sistema no la muestra (*no se produce eco*) en pantalla. Así se evita que alguien pueda averiguarla.

Shell de entrada

En los siguientes capítulos, empezará a introducir comandos de SunOS. Cuando Vd. transmite un comando al sistema, en realidad, envía información a un programa de interpretación de comandos llamado *shell*. Entonces, el programa shell lee la información que usted le ha transmitido y hace que la acción adecuada se realice en el sistema.

El shell predeterminado del software del sistema SunOS es el shell Bourne, pero también existen otros dos programas shell en el sistema operativo: el shell C y el shell Korn, cada uno de ellos con sus propias diferencias.

Nota- Podrá obtener información específica acerca de cualquier comando de SunOS, incluyendo cada uno de los shell disponibles, si consulta la página `man` (manual de referencia). Para una mayor información acerca de las páginas `man`, veáse “Cómo mostrar páginas de manual con `man`” del Capítulo 2, “Comandos básicos de SunOS”.

Cuando entra por primera vez en el sistema (o abre una nueva ventana de Utilidad de Comandos o de Utilidad Shell) y ve el indicador de comandos, éste le indica que ha empezado automáticamente un programa shell. A este shell se le llama *shell de entrada*. Si su shell de entrada no es el shell predeterminado de SunOS (el shell Bourne), se debe a que su administrador de sistemas ha especificado un shell diferente (el shell C o el shell Korn).

Como ya se ha mencionado, cada shell tiene sus propias diferencias. Algunos comandos o procedimientos que estarán a su disposición cuando use un shell pueden no estarlo cuando use otro diferente. Teniendo esto en cuenta, observe que siempre que se presente en este manual cualquier comando o procedimiento que no esté disponible en el shell predeterminado de SunOS (el shell Bourne), las secciones estarán marcadas claramente al efecto.

Cómo salir

Cuando haya acabado su sesión de trabajo y desee abandonar el sistema operativo, escriba la siguiente orden para salir:

```
$ exit
```

Pasado un momento, el sistema mostrará de nuevo el indicador de entrada:

```
$ exit  
login:
```

Si Vd. ve el indicador de entrada es que ha salido con éxito. Entonces el sistema estará preparado para que Vd. u otro usuario pueda entrar.

Nota- Con el sistema operativo SunOS, si desactiva su terminal o área de trabajo *no* significa necesariamente que vaya a salir del sistema. A no ser que salga explícitamente, permanecerá conectado con él.

Equivalencias del teclado

A pesar de que las operaciones que va a realizar no requieren un uso frecuente de los menús de OpenWindows y del ratón, en algunos casos puede agilizar dichas operaciones, utilizando una combinación de teclas, llamada *aceleradores de teclado*, que duplica las operaciones con el ratón y los menús, y de las teclas preconfiguradas del teclado.

La siguiente tabla lista una serie de operaciones de comandos y de equivalencias del teclado tanto para sistemas SPARC como para los basados en x86.

Nota- La tecla Meta es la tecla que tiene el símbolo <> en los teclados SPARC y que se obtiene en los teclados x86 presionando Ctrl-Alt.

Para realizar una operación con un acelerador de teclado, presione y mantenga pulsada la primera tecla (Meta, o Control-Alt simultáneamente) y presione la segunda tecla. Por ejemplo, para cortar texto seleccionado, presione y mantenga pulsada la tecla Meta y presione *x* en un sistema SPARC; en un sistema x86, presione y mantenga pulsado Control y Alt a la vez y presione *x* simultáneamente

Tabla 1-1 Aceleradores de teclado

Operación	Equivalencias del teclado	Acción
Repetir	Meta - a	Repite la operación anterior
Copiar	Meta - c	Copia la selección en el portapapeles
Cortar	Meta - x	Corta la selección y la coloca en el portapapeles
Buscar	Meta - f	Busca la selección a la derecha del indicador

Tabla 1-1 Aceleradores de teclado

Operación	Equivalencias del teclado	Acción
Ayuda	Ayuda o F1	Muestra una ventana de ayuda con ayuda contextual sobre el objeto en el que está situado el puntero.
Nuevo	Meta - n	Carga un archivo nuevo
Abrir (Archivo)	Meta - o	Abre un archivo (por ejemplo, si ha resaltado un icono de archivo en el Administrador de archivos)
Abrir (Ventana)	Meta - w	Abre un icono o cierra una ventana a un icono
Pegar	Meta - v	Copia la selección del potapapeles en el punto de inserción
Imprimir	Meta - p	Envía el archivo a la impresora (por ejemplo, si ha resaltado un icono de archivo en el Administrador de archivos)
Propiedades	Meta - i	Muestra la ventana de propiedades de la aplicación en la posición del puntero
Rehacer	Mayús-Meta - p	Rehace una operación de Deshacer
Guardar	Meta - s	Guarda el archivo actual
Parar	Parar o Esc	Para la operación actual
Deshacer	Meta - u	Deshace la operación anterior

Consideraciones previas a la puesta en marcha de OpenWindows

Antes de empezar el software de OpenWindows, lea las siguientes consideraciones previas. Si no le afecta ninguna de ellas, páselas por alto hasta llegar a la sección “Cómo ver el software de OpenWindows”, de este capítulo.

- Si es el responsable de la instalación de un servidor de OpenWindows en la red, consulte *SPARC: Installing Solaris Software*, o *x86: Installing Solaris Software*.
- Si actualmente se encuentra ejecutando el software de la versión 2 de OpenWindows, consulte el Apéndice A, “Cómo cambiar a la versión 3.3 de OpenWindows, o posteriores”.

- Si actualmente se encuentra ejecutando el entorno de usuario SunView, consulte el Apéndice A, “Cómo cambiar a la versión 3.3 de OpenWindows, o posteriores”.
- Si actualmente se encuentra ejecutando una versión de OpenWindows anterior a la 3.3, consulte las secciones “La variable de entorno OPENWINHOME” y “Cómo utilizar el archivo de inicio correcto”, de este capítulo.

x86 – Si cambia el tipo de ratón, teclado o tarjeta adaptadora de video de su sistema después de instalar Solaris, tendrá que ejecutar el programa `devconfig` para actualizar la versión del sistema. Consulte la página de manual `devconfig(1M)`.

La variable de entorno OPENWINHOME

Si en este momento se encuentra procesando el software de una versión de OpenWindows anterior a la 3.3, es posible que haya instalado ya su sistema para usar la variable de entorno `OPENWINHOME`. Sin embargo, en la actualidad ya no se recomienda que los usuarios establezcan la variable de entorno `OPENWINHOME`, ni manualmente, ni mediante un archivo de puesta en marcha.

Cuando Vd. ejecuta el comando `openwin`, éste establece automáticamente la variable de entorno `OPENWINHOME` como `/usr/openwin`, de modo que no es necesario que Vd. lo haga.

Si ha establecido la variable de entorno `OPENWINHOME`, bien en el archivo `.profile` o en el archivo `.cshrc` en su directorio de usuario, convierta la línea en comentario o elimínela por completo *antes* de ejecutar la versión 3.3. de OpenWindows.

Para eliminar, o convertir en comentario la variable de entorno `OPENWINHOME` en el archivo `.profile` o en el archivo `.cshrc`, siga estos pasos:

- 1. Abra el archivo `.profile` o el archivo `.cshrc` usando un editor de textos como `vi`.**
- 2. Escriba un signo (#) antes de la variable, como se indica a continuación o elimine la línea por completo.**

Si está trabajando en el archivo `.profile`, siga el paso a; si lo está haciendo en el archivo `.cshrc`, siga el paso b.

a. En el archivo .profile:

```
#OPENWINHOME=/usr/openwin
```

a. En el archivo .cshrc:

```
#setenv OPENWINHOME /usr/openwin
```

3. Grabe y salga del archivo.

4. Desactive la variable OPENWINHOME (o salga y vuelva a entrar en el sistema).

Si está ejecutando el shell Bourne o el shell Korn, siga el paso a. Si está procesando el shell C, siga el paso b.

a. En el shell Bourne o en el shell Korn, escriba:

```
$ unset OPENWINHOME
```

b. En el shell C, escriba:

```
example% unsetenv OPENWINHOME
```

Una vez que haya desestablecido la variable de entorno ya estará preparado para ejecutar el software de OpenWindows, como se describe a continuación en la sección “Cómo ver el software de OpenWindows”, de este capítulo.

Cómo utilizar el archivo de inicio correcto

Si está ejecutando en la actualidad el software de una versión de OpenWindows anterior a la 3.3, es importante determinar el estado de su archivo `.xinitrc`. El archivo `.xinitrc` es un archivo de inicio de OpenWindows que puede contener opciones elegibles por el usuario.

Para determinar el estado de su archivo `.xinitrc`, escriba:

```
$ cd
$ ls -a .xinitrc
```

Dependiendo de la salida de este comando, realice una de las siguientes acciones:

- Si no tiene un archivo `.xinitrc` (es decir, si el resultado del anterior comando `ls -a` no muestra un listado del archivo) no haga nada. Si no existe el archivo `.xinitrc` en su directorio de usuario, OpenWindows usa el archivo de inicio predeterminado del sistema.
- Si tiene un archivo `.xinitrc` (es decir, el resultado del anterior comando `ls -a` muestra un listado del archivo), pero si no ha realizado ningún cambio en el archivo o no quiere salvar los cambios que ha efectuado, siga el paso 1 de la sección “Procedimientos del archivo de inicio”.
- Si tiene un archivo `.xinitrc` (es decir, el resultado del anterior comando `ls -a` muestra un listado del archivo), y ha efectuado cambios en el archivo que quiere salvar, siga el paso 2 de la sección “Procedimientos del archivo de inicio”.

Procedimientos del archivo de inicio

1. Para eliminar el archivo `.xinitrc` de su directorio de usuario, escriba el siguiente comando:

```
$ rm .xinitrc
```

2. Para conservar los cambios de su archivo `.xinitrc`, siga estos pasos:

a. Mueva `.xinitrc` a `.xinitrc.save`:

```
$ mv .xinitrc .xinitrc.save
```

b. Copie `$OPENWINHOME/lib/Xinitrc` en `.xinitrc` en su directorio de usuario:

```
$ cp $OPENWINHOME/lib/Xinitrc $HOME/.xinitrc
```

c. Agregue las líneas que desee guardar del archivo `.xinitrc.save` a `.xinitrc`.



Atención- Cuando edite el archivo `.xinitrc`, no agregue una versión secundaria de `olwm`, no agregue `svenv`, ni elimine la línea que contenga `$OPENWINHOME/lib/openwin-sys`.

Cómo empezar el entorno de usuario OpenWindows

Para empezar el entorno de usuario OpenWindows se tienen que realizar los siguientes pasos generales:

1. Utilizando NFS, montar el software de OpenWindows desde el servidor en el que está instalado.

Para obtener información acerca de cómo montar el software desde un servidor, consulte *OpenWindows Reference Manual*, o contacte con su administrador de sistemas.

2. Empezar el software de OpenWindows con el comando `openwin`, agregar cualquier opción adicional para empezar cuando sea necesario.

Cómo ver el software de OpenWindows

Una vez que haya montado el software de OpenWindows desde un servidor y esté ejecutado la instalación del archivo de comandos de OpenWindows, ya estará preparado para empezar el software de OpenWindows.

Para empezar el software de OpenWindows, escriba `/usr/openwin/bin/openwin` tras el indicador de shell y presione Retorno.

```
$ /usr/openwin/bin/openwin
```

Así aparecerá la pantalla de la versión 3.3 de OpenWindows e instalará el entorno de trabajo de OpenWindows.

Cómo ver OpenWindows con rapidez

Una vez que Vd. haya empezado con éxito OpenWindows, ya puede configurar el sistema para usar un comando corto de forma que no necesite escribir cada vez la ruta de acceso de OpenWindows completa.

Si está usando el shell Bourne o el Korn conseguirá lo mismo situando una función shell en su archivo `.profile`. Si está usando el shell C pondrá un *alias* en su archivo `.cshrc`. Tanto el archivo `.profile` como el `.cshrc` se encuentran en su directorio de usuario.

Cuando haya situado el comando corto en el archivo apropiado de su shell, para empezar OpenWindows escriba simplemente:

```
$ openwin
```

En las siguientes secciones, “En el archivo `.profile`”, y “En el archivo `.cshrc`”, se explica cómo agregar el comando corto de OpenWindows.

En el archivo `.profile`

Para agregar el comando corto OpenWindows en su archivo `.profile`:

- 1. Abra el archivo `.profile` usando un editor de textos como `vi`.**
- 2. Agregue la siguiente función shell en el archivo de la misma forma que se indica,**

```
openwin () {  
    /usr/openwin/bin/openwin  
}
```

- 3. Guarde y salga del archivo.**

4. Guarde y vuelva a entrar para activar el comando corto, o escriba:

```
$ . .profile
```

Ahora, siempre que quiera empezar, simplemente tiene que escribir `openwin`.

En el archivo `.cshrc`

Para agregar el comando corto de OpenWindows en su archivo `.cshrc`:

1. Abra el archivo `.cshrc` usando un editor de textos como `vi`.
2. Agregue el siguiente comando alias al archivo, del modo que se indica a continuación:

```
alias openwin /usr/openwin/bin/openwin
```

3. Salve y abandone el archivo.

4. Salga y vuelva a entrar para activar el comando corto, o escriba:

```
example% source .cshrc
```

Ahora, siempre que quiera empezar OpenWindows, sólo tiene que escribir `openwin`.

Si OpenWindows no se ve

Cuando Vd. empieza el software de OpenWindows accede a él a través del directorio `/usr/pioneering`. El software de OpenWindows se instala en muchas aplicaciones, por ejemplo, el Calendario no puede cargarse a no ser que el software de OpenWindows esté instalado adecuadamente en el archivo `usr/openwin`.

Si OpenWindows no empieza cuando Vd. escribe el comando `/usr/openwin/bin/openwin`, bien no tiene el software de OpenWindows instalado, o bien está en un directorio diferente al `/usr/openwin`. Consulte , *OpenWindows Reference Manual*, o contacte con su administrador de sistemas.

Cómo abandonar el entorno OpenWindows

Una vez que haya visto el software de OpenWindows y esté trabajando en el entorno windows, ya no podrá salir de la línea de comando de la sesión estándar SunOS. Primero deberá abandonar el entorno ventana y después salir.

Si escribe `logout` tras un indicador de shell, verá el mensaje:

```
Not login shell.
```

Para salir del entorno OpenWindows, siga estos pasos:

- 1. Sitúe el ratón de manera que la flecha (*puntero*) no esté en el fondo de su pantalla (el *fondo de trabajo*).**
- 2. Presione el botón del ratón.**
El Fondo de trabajo aparecerá, presentando varias opciones.
- 3. Arrastre el puntero hacia abajo hasta que resalte el último elemento del menú: *Salir*.**
- 4. Suelte el botón del ratón.**
Aparecerá una ventana desplegable, solicitándole que confirme si desea abandonar el sistema de ventanas.
- 5. Sitúe el puntero en *Salir* y pulse el botón SELECCIONAR del ratón.**
Tras unos breves momentos todas las ventanas serán abandonadas y el indicador del sistema aparecerá en el borde inferior izquierdo de su pantalla.

Opciones de entrada especiales para OpenWindows

La mayoría de los usuarios pueden empezar el software de OpenWindows siguiendo los pasos descritos en “Cómo ver el software de OpenWindows”. Sin embargo, en algunos casos, tendrá que usar opciones adicionales para modificar el proceso de puesta en marcha de OpenWindows.

En esta sección se describen los siguientes casos especiales:

- Cómo empezar el software de OpenWindows con red de seguridad limitada.

- Cómo empezar el software de OpenWindows con varios tipos de monitores y memorias intermedias.
- Cómo empezar el software de OpenWindows en pantallas múltiples.

Para empezar el software de OpenWindows con opciones especiales, use el comando `openwin`:

```
$ openwin [ options ]
```

En el ejemplo anterior, *options* son las opciones de la línea de comando que le permiten confeccionar la configuración predeterminada del servidor. En las siguientes secciones se describen algunas de las opciones usadas más frecuentemente.

Cómo empezar con la seguridad de red limitada

Si está operando en un entorno en red abierto, y *no está interesado en la seguridad de red*, use la opción `-noauth` para que otros usuarios ejecuten aplicaciones en su sistema.

El siguiente comando modifica la propiedad de seguridad predeterminada, que le permite especificar otros usuarios que pueden acceder a su servidor de ventana:

```
$ openwin -noauth
```

Cómo empezar con varios tipos de monitores

Si posee un monitor de escala de gris (un monitor en blanco y negro) con una tarjeta para gráficos de 8 bits o más) use la opción `grayvis` cuando empiece el software del OpenWindows. Esto puede mejorar ciertos aspectos de la visualización de OpenWindows, pero no es necesario.

Para usar esta opción, escriba lo siguiente tras el indicador del sistema:

```
$ cd  
$ openwin -dev /dev/fb grayvis
```

SPARC *Cómo empezar con pantallas múltiples*

SPARC – Observe que las siguientes secciones pertenecientes a “Cómo empezar con pantallas múltiples” se refieren sólo a sistemas basados en SPARC. Dichas secciones requieren alguna experiencia en la administración de sistemas. Si nunca ha configurado un sistema, pida ayuda a su administrador de sistemas.

Para ejecutar el entorno OpenWindows en pantallas múltiples, debe informar al sistema de los dispositivos adicionales y los tipos de visualización que quiere ejecutar. Puede especificar las opciones del dispositivo o usar los valores predeterminados disponibles con el archivo de comandos `openwin` que pone en marcha el software de OpenWindows.

Se necesitan dos opciones con el comando `openwin` cuando empieza el software en monitores duales:

```
$ openwin [ [ -dev device ] [ deviceoptions ] ]
```

Los corchetes dobles indican que la combinación de [`-dev device`] [`deviceoptions`] puede entrar más de una vez en la línea de comando (es decir, una vez por mecanismo).

[`-dev device`]

La línea de opción `device` especifica el dispositivo de la tarjeta para gráficos o la pantalla que el servidor debe usar para la visualización.

Si la línea de comando no muestra esta opción, el servidor usará la opción predeterminada `/dev/fb`. La aparición múltiple de la opción `-dev` en la línea de comando indicará múltiples visualizaciones en el mismo servidor.

[*opciones de dispositivo*]

La opción de la línea de comando *opciones de dispositivo* es una lista de modificadores de dispositivo que cambia el comportamiento del dispositivo que viene especificado en la opción `-dev`.

Ejemplo de opción de dispositivo

En esta sección se incluyen ejemplos de posición de pantallas en línea y apiladas.

Nota- En todos los ejemplos, el orden de los dispositivos es importante. El primer dispositivo que se especifica debe ser la pantalla físicamente situada a la izquierda o parte superior del segundo dispositivo. El segundo dispositivo que se especifica debe ser la pantalla situada físicamente a la derecha o parte inferior del primer dispositivo.

[izquierda] [derecha]

La línea de comando siguiente da instrucciones al sistema para empezar dos pantallas. La pantalla izquierda es la tarjeta para gráficos predeterminada y la derecha es una pantalla monocroma. Esto le permite mover el cursor a izquierda y derecha entre las dos pantallas.

```
$ openwin -dev /dev/fb left -dev /dev/fbs/bwtwo0 right
```

El ejemplo siguiente equivale al anterior. Como valor predeterminado, el primer dispositivo está a la izquierda del segundo, que está en la línea de comando.

```
$ openwin -dev /dev/fb -dev /dev/fbs/bwtwo0
```

La siguiente línea de comando ordena al sistema empezar dos pantallas. La pantalla derecha es la tarjeta para gráficos y la izquierda es una pantalla monocroma. Esta configuración le permite mover el cursor a izquierda y derecha entre las dos pantallas.

```
$ openwin -dev /dev/fb right -dev /dev/fbs/bwtwo0 left
```

[superior] [inferior]

La siguiente línea de comandos ordena al sistema empezar dos despliegues. El despliegue superior es un CG6 y el botón de visualización es una pantalla monocroma. Esta configuración le permite mover el cursor de arriba y abajo entre las dos pantallas.

```
$ openwin -dev /dev/fbs/cgsix0 top -dev /dev/fbs/bwtwo0 bottom
```

El ejemplo siguiente *no* es equivalente al anterior. Como valor predeterminado, el primer dispositivo está a la *izquierda* del segundo listado en la línea de comando.

```
$ openwin -dev /dev/fbs/cgsix0 -dev /dev/fbs/bwtwo0
```

La siguiente línea de comandos ordena al servidor empezar dos pantallas. La pantalla del fondo es una CG6 y la superior es una pantalla monocroma. Esta configuración le permite mover el cursor arriba y abajo entre los dos despliegues.

```
$ openwin -dev /dev/fbs/cgsix0 bottom -dev /dev/fbs/bwtwo0 top
```

Notas Diversas

Las siguientes consideraciones son importantes si está ejecutando pantallas múltiples.

- Como valor predeterminado, `o1wm` controla todas las pantallas.
- No puede mover ventanas entre pantallas.

Comandos básicos de SunOS



Este capítulo ofrece una introducción a los comandos de usuario en el sistema operativo SunOS. Se describe cómo introducir comandos, cómo corregir errores de escritura, cómo introducir comandos largos o múltiples, cómo utilizar las opciones de los comandos, y otras informaciones útiles acerca de los comandos SunOS.

Para introducir comandos, utilice una ventana de la Herramienta de comandos o de la Herramienta del shell. Para mostrar estas ventanas, seleccione el submenú de Programas en el menú del Área de trabajo.

Indicador de comando

Una vez que haya entrado en el sistema, la pantalla o ventana estará vacía, tan solo con un indicador inicial. La naturaleza de este indicador variará dependiendo del shell que esté utilizando y de cómo lo haya configurado su administrador de sistemas. Debido a que el indicador de comando predeterminado para el software del sistema SunOS es el signo dolar (\$), este indicador se usará en la mayoría de los ejemplos de este manual.

Si más adelante deseara cambiar el indicador de comando, consulte “Cómo cambiar el indicador de comando” en el Capítulo 10, “Personalizar su entorno de trabajo” para conocer las instrucciones de cómo hacerlo.

Cómo introducir comandos

Cuando vea el indicador de comando, significará que el sistema espera que introduzca un comando. Intente introducir el comando `date` cuando aparezca el indicador tal como se muestra en este ejemplo (escriba `date` y presione la tecla Retorno):

```
$ date
Lun Feb 3 10:12:51 PST 1992
$
```

Como puede ver, este comando muestra la hora y fecha actuales. Ahora, intente introducir el mismo comando pero en mayúsculas:

```
$ Date
Date: Command not found.
$
```

Como puede ver, una `D` mayúscula no es equivalente a una `d` minúscula según la interpretación del sistema. Casi todos los comandos del sistema operativo SunOS se escriben en minúsculas.

Cómo corregir errores de escritura

Suponga que empieza a escribir `Date`, pero se da cuenta del error antes de presionar la tecla Retorno. El texto que escriba no se enviará al sistema hasta que se presione la tecla Retorno. Por tanto, aún es posible corregir su error. Tiene dos opciones:

- Presione la tecla Eliminar o la Tecla de retroceso para retroceder eliminando hasta el error; o
- escriba `Ctrl-U` para eliminar la línea entera y empezar de nuevo (mantenga presionada la tecla Control y presione “u”).

Intente ambos métodos y compruebe cómo funcionan. La tecla Eliminar/Tecla de retroceso varía en algunos sistemas. `Ctrl-U` debería funcionar en la mayoría de los sistemas.

Cómo introducir comandos múltiples y comandos largos

Puede introducir más de un comando en una única línea. Simplemente, escriba un punto y coma (;) entre los comandos, tal como se muestra a continuación con los comandos `date` y `logname`:

```
$ date; logname
Lun Feb 3 10:19:25 PST 1992
spanky
$
```

Como puede ver, se muestra la hora y fecha actual (con el comando `date`) y el nombre de entrada del usuario que está actualmente en el sistema (con el comando `logname`).

Si está escribiendo un comando muy largo, puede utilizar la barra inversa (\) para continuar escribiendo en la segunda línea. Por ejemplo:

```
$ date; \
logname
Lun Feb 3 10:23:25 PST 1992
hankw
$
```

Aunque los comandos `date` y `logname` no sean en realidad largos, se utilizan en este ejemplo para demostrar el concepto de continuar un conjunto de comandos en la siguiente línea de la forma más simple posible. Más tarde, cuando el comando que quiera utilizar sea mayor que el ancho de su pantalla, comprobará cómo la utilización del carácter puede ser muy útil.

Nota – Si está utilizando la ventana de la Herramienta del shell o de la Herramienta de comandos en el ambiente OpenWindows, no necesitará utilizar la barra inversa para continuar escribiendo comandos en la siguiente línea. Cuando alcance el final de una línea, los comandos que esté escribiendo cambiarán automáticamente de línea, y el sistema ejecutará todos los comandos cuando presione Retorno.

Cómo repetir comandos anteriores

Nota – La característica de repetición de comandos descrita en esta sección está disponible cuando se utiliza el shell C.

Una manera rápida de repetir el último comando que introdujo es escribir `!!` y presionar Retorno. El sistema guarda un *historial* de los comandos que ha escrito y es capaz de repetir los comandos anteriores. Por ejemplo, si el último comando que introdujo fue `date`:

```
example% !!
date
Lun Feb 3 10:26:20 PST 1992
example%
```

También puede repetir cualquier comando introducido con anterioridad escribiendo `!x`, donde `x` es el número correspondiente al comando deseado en la *lista del historial*. Para ver la lista del historial, escriba el comando `history` y presione Retorno. Lo siguiente es un ejemplo de lo que podría encontrar:

```
example% history
1  pwd
2  clear
3  ls -l
4  cd /usr/home/worker
5  logname
6  date
7  history
```

Otro método para repetir caracteres desde la lista de historia es continuar el símbolo `!` con un número negativo. Por ejemplo, para repetir el segundo comando empezando por el final de la lista de historia, escribiría lo siguiente:

```
example% !-2
logname
hankw
example%
```

Según el ejemplo de lista del historial anterior, se repetiría el comando `logname`.

Otro método es hacer seguir el símbolo `!` de los primeros caracteres de un comando anterior. Por ejemplo, si previamente ha introducido el comando `clear` para borrar su pantalla, puede escribir `!cl` para borrar su pantalla de nuevo. De cualquier forma, con este método de repetición de comandos, debe introducir los suficientes caracteres para distinguir el comando entre los de la lista de historia. En el caso de que escriba sólo un carácter después del símbolo `!`, el sistema repetirá el comando más reciente que empiece por dicha letra.

Cómo agregar opciones a los comandos

Muchos comandos tienen *opciones* que invocan características especiales del comando. Por ejemplo, el comando `date` tiene la opción `-u`, con la que la fecha se expresará según el meridiano de Greenwich en vez de la hora local:

```
$ date -u
Lun Feb 3 11:06:51 GMT 1993
$
```

La mayoría de las opciones se expresan como un carácter precedido por un guión (`-`). No todos los comandos tienen opciones. Otros tienen más de una. Si utiliza más de una opción en un comando, puede escribirlas separadamente (`-a -b`) o agrupadas (`-ab`).

Cómo redireccionar y enviar por el canal de comunicación la salida del comando

A no ser que se indique otra cosa, los comandos mostrarán sus resultados en la pantalla. Hay símbolos especiales que le permiten *redirigir* la salida del comando. Por ejemplo, podría querer que la salida se guardase en un archivo en vez de que se mostrase por pantalla. El ejemplo siguiente muestra el uso del símbolo de redirección (`>`):

```
$ date > sample.file
$
```

En este ejemplo, la salida del comando `date` se redirecciona a un nuevo archivo llamado `sample.file`. Después, el contenido de `sample.file` se ve mediante el comando `more`:

```
$ more sample.file
Lun Feb 3 12:56:26 PST 1993
$
```

Como puede ver, el contenido de `sample.file` es ahora la salida del comando `date`. (consulte el Capítulo 3, “Trabajar con archivos y directorios,” si desea información acerca del comando `more`).

Hay ocasiones en las que le podría interesar enviar la salida de un comando como la entrada de otro. Un conjunto de comandos enlazados de esta manera se llama *canal de comunicación*. El símbolo para este tipo de redirección es una barra vertical (`|`) llamada *canal de comunicación*.

Por ejemplo, en vez de guardar la salida de un comando en un archivo, podría direccionarlo como entrada del comando de impresión (`lp`) utilizando el símbolo de canal de comunicación (`|`). Para enviar directamente la salida del comando `date` a la impresora, debe escribir lo siguiente:

```
⌘ date | lp
⌘
```

Esto imprimirá el resultado del comando `date`. Consulte “Cómo encargar solicitudes de impresión a la impresora predeterminada” en el Capítulo 8, “Uso de las impresoras,” si desea información acerca de la utilización del comando `lp` para imprimir archivos.

Los ejemplos de redirección de comandos mostrados aquí son muy simples, pero cuando conozca más comandos avanzados, encontrará que hay muchos usos de la redirección y el envío por el canal de comunicación.

Cómo ejecutar comandos en el fondo

A menudo es conveniente iniciar un comando en el indicador de comando y después situar ese comando en el *fondo*. Cuando un comando no está situado en el fondo, el siguiente indicador no aparecerá hasta que el comando comple-

te su tarea. De cualquier modo, algunos comandos tardan mucho en acabar, mientras que quizás Vd. desearía introducir otros comandos durante ese tiempo.

Si su intención es escribir un comando en el fondo, escriba un símbolo (&) después del comando, tal como se muestra más adelante. El número que aparece a continuación es el identificador del proceso:

```
$ bigjob &
[1] 21414
$
```

El comando `bigjob` se ejecutara en el fondo, y usted puede continuar escribiendo otros comandos. Después de que el trabajo se complete, la próxima vez que introduzca otro comando verá un mensaje parecido al siguiente, (en este caso el comando siguiente es `date`):

```
$ date
Lun Feb 3 10:23:25 PST 1992
[1] + Done    bigjob
$
```

Si es probable que Vd. salga del sistema antes de que se complete el trabajo de fondo, utilice el comando `nohup` (abreviatura de `no hangup`) para permitir que el trabajo se complete, tal como se ve en el ejemplo. En otro caso, el trabajo de fondo se terminará cuando salga:

```
$ nohup bigjob &
[1] 21414
$
```

Cómo obtener ayuda con los comandos del SO

Esta sección describe varias características de ayuda en línea. Estas características le permitirán ver la información de referencia desde su estación de trabajo o terminal.

Nota – Las características que se describen aquí *se suman* a las facilidades de ayuda de OpenWindows.

Cómo mostrar páginas de manual con `man`

Si conoce el nombre de un comando, pero no está seguro de lo que hace, el comando `man` puede serle útil. escriba lo siguiente para averiguar más acerca de este comando:

```
$ man man
```

Este comando muestra la primera parte de una página del manual de referencia de SunOS en el área de muestra de la ventana. Presione la Barra espaciadora para ver la pantalla siguiente, o presione la tecla `Q` para salir y volver al indicador de comando. Utilice el comando `man` para ver todas las opciones disponibles y la sintaxis propia del comando en cuestión. Las páginas del manual de referencia ofrecen a menudo ejemplos que ilustran los distintos usos del comando.

Cómo mostrar un resumen de una línea con `whatis`

Si sólo quiere un resumen en una línea de la función que realiza el comando, utilice el comando `whatis`, como se muestra aquí:

```
$ whatis date
date (1)          -display or set the date
$
```

Observe en el ejemplo de arriba el número entre paréntesis que sigue al nombre del comando. Este número indica la sección a la que pertenece este comando. Los comandos se agrupan en varias categorías según su función. La mayoría de los comandos de usuario están en la sección 1. Por convenio, el número de sección aparece entre paréntesis después del nombre del comando. Si busca un comando en el manual de referencia impreso, lo encontrará por orden alfabético dentro de su grupo.

Búsqueda de una palabra clave con apropos

Suponga que sabe lo que quiere hacer, pero no está seguro del comando a utilizar. Para esto es adecuado el comando `apropos`. Este comando localiza un comando mediante una búsqueda de palabras clave. El comando `apropos` listará todos los comandos cuyos resúmenes de una línea contengan cualquiera de las palabras clave que le fueron proporcionadas. Esto puede dar lugar a una visualización muy larga, ya que algunas palabras clave podrían aparecer en muchos sitios.

Como muestra de la salida de `apropos`, intente introducir todos o algunos de estos ejemplos:

- `apropos who`
- `apropos execute`
- `apropos apropos`

Si introduce una palabra clave que genera una visualización demasiado larga, presionando `Ctrl-C` se interrumpirá la muestra y se volverá al indicador de comando (mantenga presionada la tecla `Control` y presione “`c`”).

Trabajar con archivos y directorios



La línea de comando de SunOS se utiliza para manipular archivos y directorios. Para llevar a cabo operaciones específicas es necesario escribir el nombre del archivo y del directorio junto con los comandos de SunOS. Es un sistema diferente al de OpenWindows, en el que los archivos se muestran como iconos que se pueden mover y hacer clic sobre ellos, y los comandos se seleccionan en los menús.

Este capítulo le presenta los conceptos y procedimientos que se utilizan para trabajar con archivos y directorios desde la línea de comando SunOS. Dichas operaciones son aplicables para cualquier línea de comando SunOS, independientemente de que esté usando una Herramienta del shell o una Herramienta de comandos en el ambiente OpenWindows o esté conectado a un terminal remoto. Para hacer un uso completo del sistema operativo SunOS es esencial que entienda los conceptos que se presentan en este capítulo.

Conceptos de archivo

El *archivo* es la unidad básica del sistema operativo SunOS. Casi todo es tratado como si fuese un archivo, incluyendo:

- **Documentos:** Estos incluyen a su vez archivos de texto, como por ejemplo cartas o informes, códigos de fuente del computador o cualquier cosa que escriba y quiera salvar.

- **Comandos:** La mayoría de los comandos son archivos *ejecutables*; es decir, son archivos que puede ejecutar para procesar un determinado programa. Por ejemplo, el comando `date` que ya vio en el capítulo anterior, que ejecuta un programa que proporciona la fecha actual, es un archivo ejecutable.
- **Dispositivos:** Su terminal, impresora y unidad/es de disco son tratados como archivos.
- **Directorios:** Un directorio es simplemente un archivo que contiene otros archivos.

Las siguientes secciones explican los comandos utilizados para crear, listar, copiar, mover y eliminar archivos. También podrá ver cómo listar el contenido de un archivo y cómo determinar su naturaleza.

Uso de los comandos de archivo

Cada uno de los comandos que se presentan en esta sección incluye un ejemplo de su uso. Realice los ejemplos según vaya leyendo el texto. Esta práctica hará que los comandos y sus respectivos conceptos sean más fáciles de entender y recordar.

Antes de comenzar

Antes de que empiece a experimentar con archivos, asegúrese de que está en su directorio *de usuario*. Este es un directorio que le estableció su administrador de sistemas cuando creó su autorización de acceso. Si ejecuta las tareas que se muestran en los siguientes ejemplos desde su directorio de usuario, le será más difícil crear, copiar, mover o (lo peor de todo) eliminar archivos de partes del sistema que otros usuarios esperan que permanezcan inalteradas.

Para cerciorarse de que se encuentra en su directorio de usuario, escriba solamente el comando `cd` (abreviatura de *change directory*). Esto le llevará a su directorio de usuario (predeterminado). Después escriba el comando `pwd` (*print working directory*) para mostrar su posición actual dentro del sistema de archivos. El directorio que aparezca será su directorio de usuario.

```
$ cd
$ pwd
/export/home/nombre_del_usuario
```

En este ejemplo, el directorio de usuario es `/export/home/nombre_del_usuario`, en donde *nombre_del_usuario* es el nombre del usuario al que pertenece el directorio de usuario.

Crear un archivo de prueba

Utilice el comando `touch` para crear un archivo vacío. Si no existe un archivo con el nombre que haya especificado, el comando `touch` creará un archivo vacío (si dicho archivo ya existe, `touch` actualizará la hora del último acceso al archivo).

```
$ touch archivoprueba
$
```

Listar archivos (`ls`)

Ahora liste el archivo con el comando `ls` para verificar que lo ha creado:

```
$ ls archivoprueba
tempfile
```

Si introduce el comando `ls` solamente, éste listará todos los archivos situados en su posición actual. Si introduce el comando `ls` con un nombre de archivo determinado, listará sólo dicho archivo, si es que existe.

Si desea más información sobre el comando `ls`, consulte *man Pages(1): User Commands*.

Copiar archivos (`cp`)

Utilice el comando `cp` para copiar `tempfile` en un archivo llamado `copyfile`:

```
$ cp tempfile copyfile
$
```

Ahora intente listar ambos archivos. Observe que los dos nombres acaban con los caracteres “file”. Puede utilizar el *carácter comodín* (*), para sustituir a cualquier carácter o secuencia de caracteres. Por lo tanto, el comando `ls *file` debe listar tanto `tempfile` como `copyfile` (y cualquier archivo de ese directorio cuyo nombre termine en `file`):

```
$ ls *file
copyfile  tempfile
```

Observe que `copyfile` se lista primero. Los archivos se listan por orden alfabético, (las letras mayúsculas y los números preceden a las letras minúsculas).

Si desea una información detallada acerca del comando `cp`, consulte *man Pages(1): User Commands*.

Mover y renombrar archivos (mv)

Puede mover y renombrar archivos utilizando el mismo comando: `mv` (move). En este ejemplo, use el comando `mv` para renombrar `tempfile` como `emptyfile`:

```
$ mv tempfile emptyfile
$
```

Ahora liste de nuevo ambos archivos para verificar el cambio:

```
$ ls *file
copyfile  emptyfile
```

Como ve, `tempfile` ha sido reemplazado por `emptyfile`.

Si desea más información acerca del comando `mv`, consulte *man Pages(1): User Commands*.

Eliminar archivos (rm)

Finalmente, use el comando `rm` (remove) para eliminar `copyfile` y compruebe el resultado con el comando `ls`:

```
$ rm copyfile
$ ls *file
emptyfile
```



Atención– Una vez que elimine un archivo, éste desaparecerá para siempre. A no ser que exista una copia de seguridad, no podrá restablecer el archivo. Tenga cuidado al usar el comando `rm`, y una especial precaución si lo usa junto al carácter comodín (`*`). Los archivos eliminados con `rm` no podrán recuperarse.

Si desea una información más detallada acerca del comando `rm`, consulte *man Pages(1): User Commands*.

Mostrar el contenido del archivo (more, cat)

Use el comando `more` para mostrar el contenido de un archivo. Escriba `more` seguido del nombre del archivo que se va a mostrar. El contenido del archivo se desplazará hacia abajo en la pantalla. Si el archivo es más largo que una pantalla, aparecerá el siguiente mensaje:

```
--More--(nn%) [Press space to continue, 'q' to quit.]
```

donde *nn* es el porcentaje de archivo ya mostrado.

También puede usar el comando `cat` para mostrar el contenido de un archivo, pero éste muestra de una vez el archivo completo sin pararse. El comando `cat` (concatenate) se usa más frecuentemente para unir dos o más archivos y formar otro más grande, como se ve en el ejemplo siguiente:

```
$ cat file1 file2 file3 > bigfile
$ ls *file
bigfile
file1
file2
file3
$
```

Si desea más información acerca de los comandos `more` o `cat`, consulte *man Pages(1): User Commands*.

Mostrar el tipo de archivo (`file`)

Algunos archivos como los binarios o los ejecutables no se pueden imprimir ni mostrar en pantalla. El comando `file` puede resultarle útil si no está seguro del tipo de archivo.

Use el comando `file` para mostrar el tipo de archivo:

```
$ file copyfile
copyfile:  ascii text
```

Directorios y jerarquía

Hasta ahora ha aprendido a listar, copiar, renombrar y eliminar archivos. Sin embargo, se estará preguntando acerca de aspectos más complicados, como por ejemplo ¿dónde están situados dichos archivos? Esta sección explica la jerarquía del directorio. Lea con atención el siguiente texto, y después realice los ejemplos de las secciones siguientes.

Jerarquía de directorio

Los archivos se agrupan en directorios, que a su vez se organizan en una jerarquía. En la cima de la jerarquía está el directorio “raíz”, representado por “/”.

Como se muestra en el ejemplo de la Ilustración 3-1, cada directorio del sistema de archivos puede incluir muchos otros directorios. La convención consiste en distinguir niveles de directorio mediante el carácter /. Teniendo esto en cuenta, observe que el directorio / (raíz) contiene entre otros los subdirectorios /usr, /bin, /home y /lib. El subdirectorio /home contiene user1, user2, y user3.

Los directorios (y los archivos que hay en su interior) se especifican incluyendo el nombre de los directorios en los que están. A esto se le llama *nombre de ruta de acceso*. Por ejemplo, el nombre de la ruta de acceso del directorio user3 de la próxima ilustración es /home/user3.

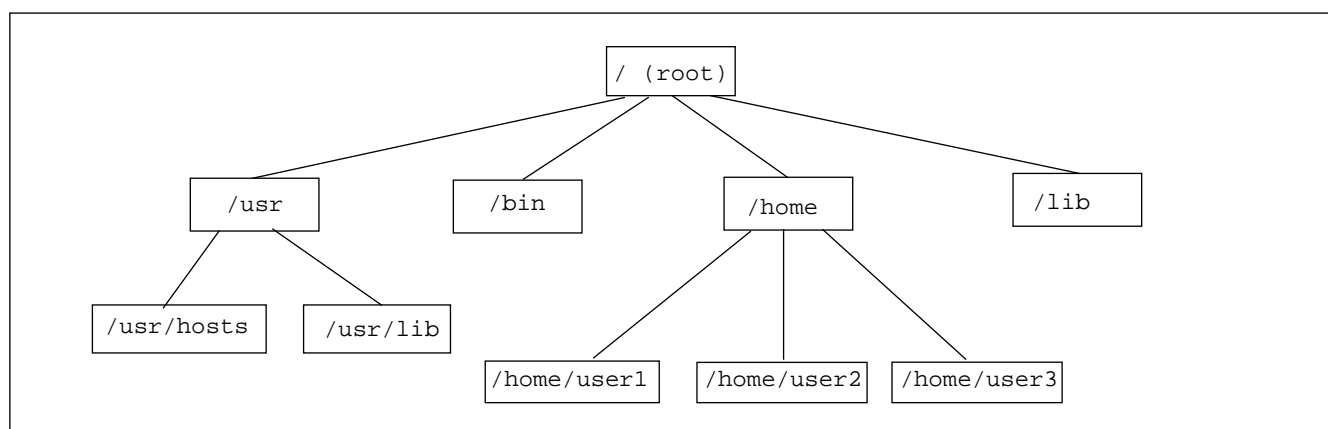


Ilustración 3-1 Jerarquía del sistema de archivos

Todos los subdirectorios y nombres de archivo que incluya un directorio deben ser únicos. Sin embargo, los nombres pertenecientes a directorios diferentes pueden ser iguales. Por ejemplo, el directorio /usr contiene el subdirectorio /usr/lib. No hay ningún conflicto entre /usr/lib y /lib porque los nombres de ruta de acceso son diferentes.

Los nombres de ruta de acceso de los archivos trabajan exactamente igual que los nombres de ruta de acceso de los directorios. El nombre de ruta de acceso de un archivo describe su posición dentro de la jerarquía del sistema de archivos. Por ejemplo, si el directorio `/home/user2` contiene un archivo llamado `report5`, su nombre de ruta de acceso será `/home/user2/report5`. Esto indica que el archivo `report5` está dentro del directorio `user2`, que a su vez está incluido en el directorio `home`, que se encuentra en el directorio raíz (`/`).

Los directorios pueden contener sólo directorios, sólo archivos, o ambos.

Imprimir el directorio en uso (pwd)

El comando `pwd` (print working directory) le indica donde se encuentra dentro de la jerarquía del sistema de archivos:

```
$ pwd
/home/user1
```

El resultado que obtenga será algo diferente al del ejemplo, ya que la estructura de su directorio es diferente. Recuerde que el directorio en uso es su posición actual dentro de la jerarquía del sistema de archivos.

Su directorio de usuario

Cada usuario tiene un directorio *de usuario*. Cuando abre por primera vez la ventana de la Herramienta de comandos o la Herramienta del shell en el entorno OpenWindows, su posición inicial (directorio en uso) es su directorio de usuario. Dicho directorio lo estableció su administrador de sistemas cuando creó su cuenta de acceso.

Cambiar el directorio en uso (cd)

El comando `cd` (change directory) le permite moverse dentro de la jerarquía del sistema de archivos:

```
$ cd /usr/lib
$ pwd
/usr/lib
```

Si escribe el comando `cd` solamente, vuelve a su directorio de usuario. Por ejemplo, si su directorio de usuario era `/home/user1`:

```
$ cd
$ pwd
/home/user1
```

En el shell C, el signo tilde (`~`) se usa como un método abreviado para especificar su directorio de usuario. Por ejemplo, para cambiar el subdirectorio `music` debería escribir lo siguiente dentro de su directorio de usuari

```
example% cd ~/music
```

También puede usar dicho método abreviado para especificar el directorio de usuario de otro usuario. Por ejemplo:

```
example% cd ~nombre_del_usuario
```

donde *nombre_del_usuario* es el nombre de entrada de otro usuario, y cambiaría al directorio de usuario de dicho usuario.

Nota – Si está utilizando el shell Bourne, el método abreviado `~` no funcionará.

Si está usando el shell Bourne, es posible que su administrador de sistemas lo haya configurado para que Vd. pueda escribir `$home` para especificar su directorio de usuario. Si este es el caso escriba:

```
$ $home/music
```

esto le trasladará al subdirectorio `music` de su directorio de usuario. Del mismo modo, si escribe:

```
$ $homenombre_del_usuario
```

se trasladará al directorio de usuario del usuario que haya especificado, donde *username* representa el nombre de entrada del otro usuario.

El directorio situado justo “por encima” de un subdirectorio se denomina *directorio maestro*. En el ejemplo anterior, /home es el directorio maestro de /home/user1. El símbolo .. (“punto, punto”) representa el directorio maestro. Por lo tanto, el comando `cd ..` cambia el directorio en uso al directorio maestro, como muestra este ejemplo:

```
$ pwd
/home/user1
$ cd ..
$ pwd
/home
```

Suponga que su directorio en uso actual es /home/user1 y desea trabajar con varios archivos en /home/user2. El siguiente método abreviado podrá serle útil:

```
$ pwd
/home/user1
$ cd ../user2
$ pwd
/home/user2
```

`../user2` ordena al sistema buscar `user2` en el directorio maestro. Como puede observar, esto resulta más sencillo que escribir el nombre de ruta de acceso /home/user2 entero.

Crear un directorio (`mkdir`)

Crear un directorio nuevo es fácil. Escriba el comando `mkdir` seguido del nombre del nuevo directorio:

```
$ mkdir veggies
$ cd veggies
$ mkdir broccoli
$ cd broccoli
$ pwd
/home/user2/veggies/broccoli
```

Nombres de ruta de acceso relativos

El nombre de ruta de acceso completo de un directorio o de un archivo comienza por un signo (`/`) y describe toda la estructura del directorio comprendida entre dicho archivo (o directorio) y el directorio raíz. Sin embargo, a menudo puede usar un nombre mucho más corto que define el archivo o directorio *en relación con* el directorio actual en uso.

Cuando se encuentre en el directorio maestro, puede desplazarse a un subdirectorío utilizando sólo el nombre del directorio y no el nombre de ruta de acceso completo. En el ejemplo anterior, el comando `cd veggies` usa el nombre de ruta de acceso relativo del directorio `veggies`. Si el directorio en uso actual es `/home/user2`, el nombre de ruta de acceso completo de este directorio es `/home/user2/veggies`.

Intente crear varios subdirectoríos diferentes y después muévase por ellos dentro de la estructura del directorio. Use los nombres de ruta de acceso completos y los nombres de ruta de acceso relativos y confirme su posición con el comando `pwd`.

Mover y renombrar directorios

Un directorio se renombra moviéndolo a un nombre diferente. Use el comando `mv` para renombrar archivos:

```
$ pwd
/home/user2/veggies
$ ls
broccoli
$ mv broccoli carrots
$ ls
carrots
```

También puede usar `mv` para mover un directorio a una posición de otro directorio:

```
$ pwd
/home/user2/veggies
$ ls
carrots
$ mv carrots ../veggies2
$ ls ../veggies2
carrots
```

En este ejemplo, el directorio `carrots` se mueve de `veggies` a `veggies2` con el comando `mv`.

Copiar directorios

Use el comando `cp -r` para copiar directorios y los archivos que contengan:

```
$ cp -r veggies veggies3
$
```

Este comando copia todos los archivos y directorios dentro del directorio `veggies` en un nuevo directorio `veggies3`. Esta es una copia *recursiva*, como designa la opción `-r`. Si intenta copiar un directorio sin usar esta opción, aparecerá un mensaje de error.

Eliminar directorios (`rmdir`)

Para eliminar un directorio vacío utilice el comando `rmdir` como sigue:

```
$ rmdir veggies3
$
```

Si el directorio contiene todavía archivos o subdirectorios, el comando `rmdir` no lo eliminará.

Use `rm -r` (agregando la opción *recursiva* `-r` al comando `rm`) para eliminar un directorio y su contenido, incluyendo cualquier subdirectorio y sus archivos, del modo siguiente:

```
$ rm -r veggies3
$
```



Atención– Los directorios eliminados con el comando `rmdir` *no* pueden recuperarse, igual que sucede con los directorios y su contenido eliminados con el comando `rm -r`.

Ver las diferencias entre directorios (`diff`)

Ocurre a menudo que diversas personas con acceso a un archivo hacen copias de él que después editan. El comando `diff` le muestra las diferencias existentes entre las versiones de un archivo ASCII. El comando:

```
$ diff leftfile rightfile
```

explora cada línea de los archivos `leftfile` y `rightfile` para encontrar las diferencias. Cuando encuentra una línea o líneas diferente/s, determina si las diferencias son resultado de una acción de agregar, eliminar, o cambiar una línea, y cuántas líneas han sido afectadas. También le indica el correspondiente número o números de línea de cada archivo, seguido del texto pertinente.

Si la diferencia es el resultado de una acción de agregar, `diff` muestra una línea con el siguiente formato:

`l[,l]` a `r[,r]`

donde *l* es un número de línea de `leftfile` y *r* es un número de línea de `rightfile`.

Si la diferencia es el resultado de una eliminación, `diff` utiliza una `d` en lugar de una `a`; si es el resultado de un cambio de línea, `diff` usa una `c`.

Las líneas correspondientes a ambos archivos siguen inmediatamente a la información de número de línea. El texto de `leftfile` va precedido por el signo (<). El texto de `rightfile` va precedido por un signo (>).

A continuación se muestran dos ejemplos de archivos, seguidos del resultado de aplicarles el comando `diff`:

```

$ cat sched.7.15
Week of 7/15

Day:  Time:          Action Item:          Details:

T     10:00          Hardware mtg.         every other week
W     1:30            Software mtg.
T     3:00            Docs. mtg.
F     1:00            Interview

$ cat sched.7.22
Week of 7/22

Day:  Time:          Action Item:          Details:

M     8:30            Staff mtg.            all day
T     10:00          Hardware mtg.         every other week
W     1:30            Software mtg.
T     3:00            Docs. mtg.

$ diff sched.7.15 sched.7.22
1c1
< Week of 7/15
---
> Week of 7/22
4a5
> M     8:30            Staff mtg.            all day
8d8
< F     1:00            Interview

```

Si los dos archivos que se comparan son idénticos, el comando `diff` no producirá ningún resultado.

El comando `diff` tiene muchas más opciones de las que se explican aquí. Si desea más información, consulte *man Pages(1): User Commands*.

Comparar tres archivos diferentes (diff3)

Si tiene tres versiones de un archivo que quiere comparar a la vez, use el comando `diff3` como sigue:

```
$ diff3 file1 file2 file3
```

El comando `diff3` compara las tres versiones de un archivo y publica series de texto diferenciado, señaladas con estos códigos:

```
==== los tres archivos son diferentes
====1 file1 es diferente
====2 file2 es diferente
====3 file3 es diferente
```

Uso de bdiff con archivos grandes

Si compara archivos muy grandes, use `bdiff` en lugar de `diff`. Ambos programas trabajan de una forma similar:

```
$ bdiff leftfile rightfile
```

Use `bdiff` en vez de `diff` para archivos que tengan más de 3500 líneas.

Buscar archivos (find)

El comando `find` busca archivos que cumplan las condiciones que Vd. especifique, comenzando por el directorio que nombre. Por ejemplo, si quiere buscar nombres de archivos que concuerden con determinado patrón o que hayan sido modificados durante un periodo de tiempo determinado.

Al contrario que la mayoría de los comandos, las opciones de `find` tienen una longitud de varios caracteres, y el nombre del directorio inicial debe ir antes que éstos en la línea de comando, como se indica a continuación:

```
$ find directorio opciones
```

donde *directorio* es el nombre del directorio inicial y *opciones* representa las opciones del comando `find`.

Cada opción describe un criterio de selección de archivos. Un archivo debe cumplir todos los criterios para ser seleccionado. De este modo, cuántas más opciones aplique, más limitado será el campo. La opción `-print` indica que Vd. desea que se muestre el resultado, (como se describirá más adelante, puede utilizar `find` para ejecutar comandos; en ese caso, deberá ordenarle a `find` que omita la visualización de los archivos seleccionados).

La opción `-name nombre_de_archivo` le indica a `find` que seleccione archivos que concuerden con *nombre_de_archivo*. Aquí, *nombre_de_archivo* se considera como el componente situado más a la derecha del nombre de ruta de acceso completo de un archivo. Por ejemplo, el componente situado más a la derecha de `/usr/lib/calendar` sería `calendar`. A esta parte del nombre de un archivo se la denomina a menudo *nombre base*.

Por ejemplo, para ver que archivos del directorio en uso y sus subdirectorios terminan en `s`, escriba:

```
$ find . -name '*s' -print
./programs
./programs/graphics
./programs/graphics/gks
./src/gks
$
```

Otras opciones incluyen:

`-name nombre_de_archivo`

Selecciona archivos cuyo elemento situado más a la derecha concuerda con *nombre_de_archivo*. Escriba *nombre_de_archivo* entre comillas si éste incluye patrones de sustitución de nombre de archivo.

`-user id_usuario`

Selecciona archivos pertenecientes a *id_usuario*. *id_usuario* puede ser un nombre de entrada o un número de identificación de usuario.

`-group grupo`

Selecciona archivos pertenecientes a *grupo*.

`-mtime n`

Selecciona archivos que han sido modificados hace *n* días.

`-newer checkfile`

Selecciona archivos que han sido modificados más recientemente que *checkfile*.

Puede especificar un orden de preferencia, combinando opciones entre paréntesis (precedidos por una barra inclinada), (por ejemplo, `\(options \)`). Dentro de este tipo de paréntesis se puede usar el indicador `-o` entre las opciones para indicar que `find` seleccione los archivos que cumplan esta categoría, en vez de aquellos que cumplan ambas.:

```
$ find . \( -name AAA -o -name BBB \) -print
./AAA
./BBB
```

Puede invertir el sentido de una opción situando de lante de ella un signo de admiración precedido por una barra inclinada. Entonces, `find` seleccionará archivos a los que *no* afecte dicha opción:

```
$ find . \!-name BBB -print
./AAA
```

También puede usar `find` para aplicar comandos a los archivos que éste seleccione con la opción:

`-exec command ' {} ' \;`

Dicha opción termina con un punto y coma precedido por una barra inclinada (`\;`). Las llaves entre comillas se reemplazan por los nombres de archivo que `find` selecciona.

Como ejemplo, puede usar `find` para eliminar automáticamente archivos de trabajo temporales. Si nombra coherentemente sus archivos temporales, puede utilizar `find` para buscarlos y destruirlos siempre que le estorben. Por ejemplo, si llama a sus archivos temporales `junk` o `dummy`, el comando los encontrará y eliminará

```
$ find . \( -name junk -o -name dummy \) -exec rm '{}' \;
```

Si desea más información, consulte *man Pages(1): User Commands*.

Seguridad de archivos y directorios

Nota – Lea esta sección atentamente. Un buen entendimiento de los permisos de archivo es siempre importante para el trabajo cotidiano.

Los permisos de archivo protegen a los archivos y los directorios para que no se puedan leer ni escribir sin autorización. A menudo tendrá archivos que quiera que otras personas puedan leer pero no cambiar. En otras ocasiones, deberá compartir archivos ejecutables (programas). Los permisos de archivo le permiten controlar el acceso a sus archivos.

Estos son los tipos básicos de acceso de archivos y directorios:

- `r` – permiso de *lectura*. Un archivo debe poderse leer si se quiere examinar o copiar. Un directorio debe poderse leer si se quiere listar su contenido.
- `w` – permiso de *escritura*. Un archivo debe poder escribirse si desea modificarlo, eliminarlo o renombrarlo. Un directorio debe poder escribirse para agregar o eliminar archivos en él.
- `x` – permiso de *ejecución*. Un archivo con permisos ejecutables es aquel que Vd. puede procesar, como por ejemplo un programa. Un directorio debe ser ejecutable si quiere tener acceso a cualquiera de sus subdirectorios.

Existen tres categorías de usuarios para los que puede establecer permisos:

- Uno mismo – El usuario

- Grupo – Otros usuarios pertenecientes al mismo grupo del usuario (por ejemplo, todos los usuarios con cuenta de acceso). Los grupos los establece y mantiene su administrador de sistemas.
- Otros – Todos los demás

Mostrar permisos y estado (`ls -l`)

Ya ha utilizado el comando `ls` para listar archivos. El comando `ls` dispone de muchas opciones. Use la opción `-l` para mostrar una lista con formato *largo*. Los archivos y directorios se listan por orden alfabético. La Ilustración 3-2 ejemplifica este método para mostrar archivos:

```

$ pwd
/home/hostname/user2
$ ls -l
total 8
drwxr-xr-x  2 user2          1024 Feb  9 14:22 directory1
-rw-r--r--  1 user2             0 Feb 10 10:20 emptyfile
-rw-r--r--  1 user2       104357 Feb  5 08:20 large-file
drwxr-xr-x  3 user2          1024 Feb 10 11:13 veggies2

```

Permisos
Enlaces
Dueño
Tamaño
Fecha . Hora
Nombre arch./direct.

Ilustración 3-2Mostrar permisos y estado

El primer carácter de la línea indica el tipo de archivo. Un guión (-) es un archivo corriente; una *d* indica un directorio, y otros caracteres pueden indicar tipos de archivos especiales.

Los nueve caracteres siguientes indican los permisos del archivo o el directorio. Dichos caracteres están formados por tres grupos de tres elementos, que indican los permisos del propietario del archivo, del grupo de dicho propietario y del mundo respectivamente. Los permisos para `emptyfile` son `rw-r--r--`, que indican que el propietario del archivo puede leerlo y escribir en él, que todos pueden leerlo y nadie puede ejecutarlo. Los permisos del directorio `veggies2` son `drwxr-xr-x`, que indican que todos tienen permiso para leerlo y ejecutarlo, pero que sólo su dueño puede escribir en él.

Además de los permisos de archivo, la pantalla muestra la siguiente información:

- El número de enlaces con dicho archivo o directorio.
- El nombre del propietario (`user2` en este caso).
- El número de bytes (caracteres) del archivo.
- Fecha y hora en la que el archivo o el directorio fue actualizado la última vez.
- Nombre del archivo o directorio.

Use el comando `cd` para desplazarse a su directorio de usuario y pruebe el comando `ls -l`. Por supuesto, el resultado que obtenga será diferente al del ejemplo.

Ahora intente escribir un comando como este:

```
$ ls -l nombre_de_directorio
```

donde *nombre_de_directorio* es el nombre de un directorio existente en su sistema de archivos. Cuando le pone nombre a un directorio, el comando `ls -l` imprime información relativa a todos los archivos y directorios (si los hay) de dicho directorio.

Listar archivos “ocultos” (`ls -a`)

Hay algunos archivos que no son listados por el comando `ls`. Dichos archivos comienzan por el carácter `.` (llamado “punto”), como por ejemplo `.cshrc`, `.login` o `.profile`. Use el comando `ls -a` para listar los citados archivos punto:

```
$ ls -a
.
..
.cshrc
.login
.profile
emptyfile
```

Observe que los archivos que comienzan por `.` son listados antes que los demás. Hay dos archivos especiales en este listado: el archivo `.` es la referencia del directorio en uso, y el archivo `..` es la referencia del directorio maestro.

Hablando en general, los archivos que comienzan por `.` son usados por utilidades del sistema y el usuario normalmente no los modifica. Existen algunas excepciones al respecto.

Cambiar permisos (chmod)

Use el comando `chmod` para cambiar los permisos de un archivo o un directorio. Vd. debe ser el propietario del archivo o del directorio, o tener acceso al directorio raíz para cambiar sus permisos. La forma general del comando `chmod` es:

```
chmod permisos nombre
```

donde `permisos` indica los permisos que se quieren cambiar y `nombre` es el nombre del archivo o directorio afectado.

Los permisos pueden especificarse de varias formas. A continuación se cita uno de los métodos más sencillos:

1. Usar una o más letras para indicar los usuarios implicados:
 - `u` (para el *usuario*)
 - `g` (para el *grupo*)
 - `o` (para *otros*)
 - `a` (*all*; para todas las categorías anteriores)
2. Indicar si los permisos se van a agregar (+) o eliminar (-).
3. Utilizar una o más letras para indicar los permisos:
 - `r` (*read*; de lectura)
 - `w` (*write*; de escritura)
 - `x` (*execute*; de ejecución)

En el ejemplo siguiente, se agrega un permiso de lectura al directorio `carrots` para los usuarios que pertenezcan al mismo grupo (de esta forma, *permisos* es `g+w` y *nombre* es `carrots`):

```
$ ls -l carrots
drwxr-xr-x  3 user2          1024 Feb 10 11:15 carrots
$ chmod g+w carrots
$ ls -l carrots
drwxrwxr-x  3 user2          1024 Feb 10 11:15 carrots
$
```

Como puede ver, el guión (-) del conjunto de caracteres de grupo se cambia por una `w` como resultado del uso del comando.

Para que otros usuarios que no pertenezcan al grupo no puedan leer ni ejecutar este mismo directorio (*permisos* es `o-rx`), tendría que escribir lo siguiente:

```
$ ls -l carrots
drwxrwxr-x  3 user2          1024 Feb 10 11:15 carrots
$ chmod o-rx carrots
$ ls -l carrots
drwxrwx---  3 user2          1024 Feb 10 11:15 carrots
$
```

Ahora, la `r` (de read) y la `x` (de execute) del conjunto de caracteres correspondiente a otros usuarios se cambian por guiones (-).

Cuando cree un archivo o un directorio nuevo, el sistema le asignará automáticamente permisos.

En general, la configuración predeterminada para los archivos nuevos es:

```
-rw-r--r--
```

y para los directorios nuevos es:

```
drwxr-xr-x
```

De forma que para hacer ejecutable un archivo nuevo como `turnip`, su propietario (`user2`), debería introducir lo siguiente:

```
$ ls -l turnip
-rw-r--r--  3 user2          1024 Feb 10 12:27 turnip
$ chmod u+x turnip
$ ls -l turnip
-rwxr--r--  3 user2          1024 Feb 10 12:27 turnip
$
```

Si quiere afectar a las tres categorías de usuarios a la vez, use la opción `a`. Si quiere hacer que un archivo nuevo como `garlic` sea ejecutable por todos, tendría que introducir lo siguiente:

```
$ ls -l garlic
-rw-r--r--  3 user2          1024 Feb 10 11:31 garlic
$ chmod a+x garlic
$ ls -l garlic
-rwxr-xr-x  3 user2          1024 Feb 10 11:31 garlic
$
```

Como resultado de esto, el indicador `x` aparecerá en las tres categorías.

También puede cambiar los permisos de grupos de archivos o directorios utilizando carácter comodín *. Por ejemplo, para cambiar los permisos de todos los archivos del directorio actual `veggies` para que sólo Vd. pueda escribir en ellos, tendría que escribir lo siguiente:

```

$ pwd
/home/user2/veggies
$ ls -l
-rwxrwxrwx  3 user2      21032 Feb 12 10:31 beats
-rwxrwxrwx  2 user2         68 Feb 10 11:09 corn
-rwxrwxrwx  3 user2    12675 Feb 08 09:31 garlic
-rwxrwxrwx  1 user2     1024 Feb 14 16:38 onions
$ chmod go-w *
$ ls -l
-rwxr-xr-x  3 user2      21032 Feb 12 10:31 beats
-rwxr-xr-x  2 user2         68 Feb 10 11:09 corn
-rwxr-xr-x  3 user2    12675 Feb 08 09:31 garlic
-rwxr-xr-x  1 user2     1024 Feb 14 16:38 onions
$

```

En este ejemplo se incluye el comando `pwd` para mostrar que el directorio en el que realiza la operación `chmod` debe ser el actual.

Establecer permisos absolutos

Hasta el momento, en esta presentación de los permisos sólo se ha incluido el comando `chmod` para cambiar permisos *relativos* a su configuración actual. Si utiliza una forma diferente del comando `chmod`, que aplica códigos numéricos para especificar permisos, puede establecer el permiso de un archivo o un directorio *de manera absoluta*.

La sintaxis de este modo de utilizar el comando `chmod` es:

```
chmod numcode name
```

donde *numcode* es el código numérico y *name* es el nombre del archivo o del directorio al que está cambiando los permisos.

El código numérico completo consta de tres números. Se usa un número para cada una de las tres categorías: usuario, grupo y otros. Por ejemplo, el siguiente comando establece permisos absolutos de lectura, escritura y ejecución para el usuario y el grupo, y sólo permiso de ejecución para otros usuarios:

```
$ chmod 771 garlic
```

La Tabla 3-1 ilustra cómo el código 771 representa los permisos asignados a `garlic`.

Table 3-1 Permisos de `garlic`

Permiso	Usuario	Grupo	Otros
Lectura	4	4	0
Escritura	2	2	0
Ejecución	1	1	1
Total	7	7	1

Cada columna de la Tabla 3-1 representa una de las categorías: usuario, grupo y otros. Para establecer permisos de lectura se agrega un 4 en la columna apropiada, para establecer permisos de escritura un 2 y para permisos de ejecución un 1. El total de las tres columnas, en la última hilera de la tabla, es el código numérico completo.

A continuación se muestra otro ejemplo del uso de este método de establecer permisos absolutos, incluyendo el comando `ls -l` para demostrar el resultado:

```
$ ls -l onion
-rw-r--r-- 3 user2          1024 Feb 10 11:46 onion
$ chmod 755 onion
$ ls -l onion
-rwxr-xr-x 3 user2          1024 Feb 10 11:48 onion
$
```

Se establecen los permisos del archivo `onion` para que el usuario pueda leerlo, escribir en él o ejecutarlo; los miembros del grupo puedan leerlo y ejecutarlo y otros usuarios puedan también leerlo y ejecutarlo. La Tabla 3-2 muestra el desglose de los códigos numéricos utilizados para establecer los permisos de `onion`.

Table 3-2 Permisos de `onion`

Permiso	Usuario	Grupo	Otros
Lectura	4	4	4
Escritura	2	0	0
Ejecución	1	1	1
Total	7	5	5

Por supuesto, para conceder permisos de lectura, escritura y ejecución a Vd. mismo, a su grupo o a otros usuarios respecto al archivo `cabbage`, escribiría lo siguiente:

```

$ ls -l cabbage
-rw-r--r-- 3 user2          1024 Feb 10 11:51 cabbage
$ chmod 777 cabbage
$ ls -l cabbage
-rwxrwxrwx 3 user2          1024 Feb 10 11:53 cabbage
$

```

La Tabla 3-3 muestra el desglose de este ejemplo.

Table 3-3 Permisos de `cabbage`

Permiso	Usuario	Grupo	Otros
Lectura	4	4	4
Escritura	2	2	2
Ejecución	1	1	1
Total	7	7	7

El código numérico `777` representa el nivel de permisos máximo que puede conceder.

Se puede utilizar el carácter comodín `*` para establecer permisos absolutos para todos los archivos del directorio, del mismo modo que se cambian los permisos relativos. Por ejemplo, para establecer permisos absolutos de todos los archivos del directorio `veggies` de modo que Vd. pueda leer, escribir y ejecutar dichos permisos, su grupo pueda leerlos y ejecutarlos, y los demás usuarios puedan sólo ejecutarlos, tendría que escribir lo siguiente:

```
$ pwd
/home/user2/veggies
$ ls -l
-rwxrwxrwx  3 user2      21032 Feb 12 10:31 beats
-rwxrwxrwx  2 user2         68 Feb 10 11:09 corn
-rwxrwxrwx  3 user2    12675 Feb 08 09:31 garlic
-rwxrwxrwx  1 user2     1024 Feb 14 16:38 onions
$ chmod 751 *
$ ls -l
-rwxr-x--x  3 user2      21032 Feb 12 10:31 beats
-rwxr-x--x  2 user2         68 Feb 10 11:09 corn
-rwxr-x--x  3 user2    12675 Feb 08 09:31 garlic
-rwxr-x--x  1 user2     1024 Feb 14 16:38 onions
$
```

En este ejemplo se incluye el comando `pwd` para ilustrar que el directorio en el que realiza esta operación debe ser el directorio actual. El comando `ls -l` se muestra sólo para ilustrar los cambios que han sufrido los permisos. Cuando se establecen permisos absolutos no es necesario saber qué permisos están en activo.

Si desea más información acerca del comando `chmod`, consulte *man Pages(1): User Commands*.

Cómo buscar archivos



Este capítulo describe cómo buscar archivos y directorios para palabras clave y cadenas de caracteres utilizando el comando `grep` de SunOS.

Cómo buscar patrones con `grep`

Para buscar una determinada cadena de caracteres en un archivo, use el comando `grep`. La sintaxis básica del comando `grep` es:

```
$ grep cadena de caracteres archivo
```

donde *cadena de caracteres* es la palabra o frase que quiere encontrar, y *archivo* es el archivo que quiere buscar.

Nota – Una *cadena de caracteres* está formada por uno o más caracteres, una única letra, una palabra o una frase. Las cadenas de caracteres pueden incluir “espacios en blanco”, signos de puntuación, y caracteres invisibles (de control).

Por ejemplo, para buscar el teléfono de Edgar Allan Poe, escriba `grep` seguido de parte o de la totalidad de su nombre y del archivo que contenga la información:

```
$ grep Poe extensions
Edgar Allan Poe      x72836
$
```

Observe que puede haber más de una línea que cumpla con los patrones especificados:

```
$ grep Allan extensions
David Allan          x76438
Edgar Allan Poe      x72836
$ grep Al extensions
Louisa May Alcott    x74236
David Allan          x76438
Edgar Allan Poe      x72836
$
```

`grep` distingue entre mayúsculas y minúsculas; es decir, debe especificar si las entradas van en mayúsculas o minúsculas:

```
$ grep allan extensions
$ grep Allan extensions
David Allan          x76438
Edgar Allan Poe      x72836
$
```

Observe que `grep` no funcionó en el primer intento porque ninguna de las entradas comenzaba por minúscula.

grep como filtro

`grep` se utiliza muy a menudo como “filtro” con otros comandos. Esto le permite filtrar y eliminar la información inútil que se produce tras ejecutar un comando. Para usar `grep` como filtro, debe enviar el resultado del comando mediante `grep` por el canal de comunicación. El símbolo para el canal de comunicación es “|”.

El siguiente ejemplo muestra archivos que acaban por “.ps” creados en el mes de mayo:

```
$ ls -l *.ps | grep May
```

La primera parte de esta línea de comando,

```
ls -l *.ps
```

muestra una lista de archivos:

```
$ ls -l *.ps
-rw-r--r-- 1 elvis      7228 Apr 22 15:07 change.ps
-rw-r--r-- 1 elvis     2356 May 22 12:56 clock.ps
-rw-r--r-- 1 elvis     1567 Jun 22 12:56 cmdtool.ps
-rw-r--r-- 1 elvis    10198 Jun 22 15:07 command.ps
-rw-r--r-- 1 elvis     5644 May 22 15:07 buttons.ps
$
```

La segunda parte,

```
| grep May
```

envía la lista por medio de `grep`, buscando el patrón mayo.:

```
$ ls -l *.ps | grep May
-rw-r--r-- 1 elvis     2356 May 22 12:56 clock.ps
-rw-r--r-- 1 elvis     5644 May 22 15:07 buttons.ps
$
```

grep *cadena de palabras múltiples*

Para buscar un patrón que tenga más de una palabra, escriba la cadena entre comillas dobles o sencillas:

```
$ grep "Louisa May" extensions
Louisa May Alcott      x74236
$
```

grep puede buscar una cadena de caracteres en un grupo de archivos. Cuando encuentre un patrón que cumple con las pautas en más de un archivo, imprime el nombre del archivo, seguido de una coma y de la línea que concuerda con el patrón:

```
$ grep es *
actores:Humphrey Bogart
alaska:Alaska es el estado más extenso de los Estados Unidos.
wilde:libro. Los libros están bien o mal escritos.
$
```

Cómo buscar líneas sin una cadena de caracteres determinada

Para buscar todas las líneas que *no* contengan una determinada cadena de caracteres, use la opción `-v` para `grep`. El siguiente ejemplo explica cómo buscar todas las líneas de los archivos del directorio personal `medici` que no contengan la letra `e`:

```
$ ls
actores  alaska  interior  tutores  wilde
$ grep -v e *
actores:Lun Mar 14 10:00 PST 1936
wilde:Eso es todo.
$
```

Más acerca de `grep`

También puede usar el comando `grep` para buscar destinos definidos como patrón usando *expresiones regulares*. Las expresiones regulares están formadas por letras y números, así como por caracteres que tienen un significado especial para `grep`. Dichos caracteres especiales, llamados *metacaracteres*, también tienen un significado especial para el sistema y tienen que ir entre comillas o entre paréntesis precedidos por una barra inversa. Siempre que use una expresión regular de `grep` tras el indicador de comando, escribala entre comillas, o precedida de metacaracteres de escape (como `&` `!` `.` `*` `$` `?` y `\`) con una barra inversa (`\`).

- Un signo de intercalación (^) indica el inicio de línea. De modo que el comando:

```
$ grep '^b' list
```

busca cualquier línea del archivo `list` que empiece por “b.”

- Un signo de dolar (\$) indica el fin de línea. El comando:

```
$ grep 'b$' list
```

muestra cualquier línea en la que “b” sea su último carácter. Y el comando:

```
$ grep '^b$' list
```

muestra cualquier línea de `list` en la que “b” es el *único* carácter de la línea.

- En una expresión regular, el punto (.) busca cualquier carácter único. De forma que el comando:

```
$ grep 'an.' list
```

busca cualquier cadena de tres caracteres que contenga “an” como sus dos primeros, incluyendo “antes” “planta”, “finanzas” y “plan” (ya que los espacios también también se tienen en cuenta).

- Cuando un asterisco (*) va detrás de un carácter, `grep` lo interpreta como “cero o más apariciones de dicho carácter”. Cuando el asterisco va detrás de una expresión regular, `grep` lo interpreta como “cero o más apariciones de caracteres que concuerden con el patrón.”

Como incluye cero apariciones, el uso del asterisco no es nada intuitivo. Suponga que desea encontrar todas las letras que contengan una “qu”. Si escribe:

```
$ grep 'qu*' list
```

funcionará según lo esperaba. Sin embargo, si quiere buscar todas las palabras que contengan la letra “n” tendría que escribir:

```
$ grep 'nn*' list
```

Si quiere buscar todas las palabras que contengan el modelo “nn”, tendría que introducir:

```
$ grep 'nnn*' list
```

Puede intentar ver lo que ocurre al contrario.

- Para concordar cero o más apariciones de *cualquier* carácter de `list`, escriba:

```
$ grep .* list
```

Cómo buscar metacaracteres

Suponga que quiera buscar las líneas de un texto que tenga el símbolo (\$). Si precede el signo de dólar de la expresión regular con una barra invertida (\), indicará a `grep` que ignore (escape) su significado especial. Esto es válido también para los demás metacaracteres (& ! . * ? y el propio \).

Por ejemplo, la expresión

```
$ grep ^\.
```

concuerta líneas que empiecen con un punto, y resulta especialmente útil cuando busque peticiones de formateo de `nroff` o `troff` (que empiecen con un punto).

La siguiente tabla, Tabla 4-1, proporciona una lista de los patrones de búsqueda de modelos que más frecuentemente puede usar con `grep`.

Tabla 4-1 Elementos del modelo de búsqueda de `grep`

Carácter	Concuerta
<code>^</code>	El comienzo de una línea de texto
<code>\$</code>	El final de una línea de texto
<code>.</code>	Cualquier carácter único
<code>[...]</code>	Cualquier carácter único de la lista o rango entre paréntesis
<code>[^...]</code>	Cualquier carácter que no esté en la lista o el rango
<code>*</code>	Cero o más apariciones del carácter precedente o de la expresión regular
<code>.*</code>	Cero o más apariciones de cualquier carácter único
<code>\</code>	Ignora el significado especial del próximo carácter

Observe que estos caracteres de búsqueda también se pueden usar en las búsquedas del editor de textos `vi`.

Comillas dobles o sencillas en líneas de comando

Como se explicó anteriormente, las comillas se utilizan para delimitar el texto que desee que sea interpretado como una palabra. Por ejemplo, para que `grep` busque todos los archivos que contengan la frase “a la carga, mis valientes”, tendría que introducir lo siguiente:

```
$ grep "a la carga, mis valientes" *
```

Las comillas simples (') también se pueden usar para agrupar frases con palabras múltiples formando unidades únicas, o para asegurarse de que determinados caracteres como por ejemplo `$` sean interpretados literalmente (el metacarácter `history !` se interpreta siempre como tal, incluso si está entre comillas, salvo que lo haga preceder por una barra inversa). En cualquier caso, es una buena idea escribir caracteres como `& ! $? . ; y \` precedidos de una barra inversa, si quiere que se interpreten como caracteres tipográficos normales.

Si escribe por ejemplo:

```
$ grep $ list
```

verá *todas* las líneas de `list`. Sin embargo, si introduce:

```
$ grep '\$' list
```

sólo verá las líneas que contengan el carácter “\$”.

Si desea más información acerca del comando `grep`, consulte *man Pages(1): User Commands*.

Contraseñas, procesos y almacenamiento en disco



SunOS ofrece una amplia gama de comandos para realizar varias tareas del sistema desde la línea de comandos. Este capítulo describe cómo establecer una contraseña, cómo listar los procesos que en ese momento se ejecuten en su sistema, cómo cortar los procesos no deseados, y cómo mostrar la cantidad de espacio de almacenamiento usado en disco.

Uso de una contraseña

Teniendo en cuenta la seguridad de su sistema, SunOS requiere la utilización de una contraseña para su sistema. El cambiar su contraseña varias veces al año le ayudará a garantizar que Vd. será el único usuario que accederá fácilmente a su cuenta de acceso. Si sospecha que alguien ha utilizado su cuenta de acceso sin su permiso, cambie inmediatamente su contraseña.

Al escoger una contraseña, recuerde lo siguiente:

- Seleccione una contraseña que pueda recordar sin necesidad de escribirla. Una contraseña que no pueda recordar es peor que otra fácil de averiguar.
- Seleccione una contraseña que tenga como mínimo seis caracteres de longitud y contenga al menos un número.
- No utilice su propio nombre o iniciales ni el nombre o iniciales de su esposa.
- No utilice los nombres de mascotas u objetos públicos que sean de su interés.
- No utilice todas las letras en mayúscula.

- Si tiene más de una cuenta de acceso, no utilice la misma contraseña para cada cuenta de acceso.
- Aunque puede utilizar cualquier carácter para su contraseña, algunos caracteres, como por ejemplo Ctrl-C, Ctrl-Z, Ctrl-U, Ctrl-S, Escape, Tabulador, y en algunos casos # y @ pueden ser interpretados por el terminal como señales. Deben evitarse estos caracteres. El terminal puede interpretarlos como señales en vez de caracteres de texto, y esto le impedirá escribir correctamente su contraseña.

Cómo cambiar su contraseña

Para cambiar su contraseña personal, escriba el comando `passwd`:

```
$ passwd
Changing password for hankw on worker
Old password:
New password:
Retype new password:
$
```

- 1. Cuando el sistema le indique `Old Password:`, introduzca su contraseña actual.**
(Si su cuenta de acceso no tiene asignada una contraseña actualmente, el sistema omitirá el indicador `Old Password:`). Observe que el sistema no muestra su contraseña en la pantalla (no produce eco). Esto evitará que otros usuarios descubran su contraseña.
- 2. Cuando el sistema le indique `New Password:`, introduzca la contraseña que ha decidido asignar a su cuenta de acceso.**
De nuevo, la contraseña que escriba no aparecerá en la pantalla.
- 3. En el indicador final, `Retype new password:`, introduzca su nueva contraseña por segunda vez.**
Esto verificará que lo que escribió es exactamente lo que quería escribir.

Si no introduce su contraseña exactamente igual a como lo hizo en el indicador anterior, el sistema rechazará el cambio de su contraseña y responderá `Lo siento (Sorry)`. Si esto ocurre repetidas veces, contacte con su administrador de sistemas para conseguir una nueva contraseña.

Nota – No se permiten claves de acceso con menos de seis caracteres. Y además, una contraseña nueva debe diferenciarse de la anterior en al menos tres caracteres.

Caducidad de la contraseña

Si su sistema usa la caducidad de la contraseña (implementado mediante opciones del comando `passwd`), su contraseña podría tener una duración máxima, o bien una duración máxima y mínima. La duración de su contraseña la establece su administrador de sistemas.

Cuando su contraseña alcance la caducidad (o edad máxima), se le indicará que cambie su contraseña. Esto ocurrirá cuando acceda al sistema y se le mostrará lo siguiente:

```
Su contraseña ha caducado. Seleccione una nueva.
```

El sistema ejecutará automáticamente el programa `passwd` y le pedirá una nueva contraseña.

Si por ejemplo, el tiempo *mínimo* de caducidad para su contraseña ha sido establecido en dos semanas e intenta cambiar su contraseña antes de cumplirse ese tiempo, se le mostrará lo siguiente:

```
Lo siento, menos de dos semanas desde el último cambio.
```

Para ver la información sobre el tiempo de caducidad de su contraseña, utilice la opción `-d` del comando `passwd`:

```
$ passwd -d
username 2-14-92 14 60
```

La pantalla mostrará, por orden, la fecha en que se creó la contraseña actual, el tiempo mínimo y el tiempo máximo de caducidad (esta información sólo aparecerá si se implementó el tiempo de caducidad para las contraseñas).

Si desea más información sobre las contraseñas y el tiempo de caducidad de las mismas, consulte *man Pages(1): User Commands*.

Procesos y números de procesos

Después de que cada comando sea interpretado por el sistema, se creará un *proceso* independiente, con un único número de indentificación de proceso (PID), para ejecutar el comando. El sistema utiliza el PID para averiguar el estado actual de cada proceso.

Qué comandos se ejecutan actualmente (ps)

Utilice el comando `ps` para conocer qué procesos se ejecutan actualmente. Además de mostrar el *número de indentificación del proceso* (listado como `PID`) para cada uno de sus procesos (creado como resultado de un comando que introdujo), `ps` le mostrará también el *terminal* desde el que se empezó (`TTY`), el *tiempo de cpu* que utilizó hasta ahora (`TIME`), y el *comando* que está actuando (`COMMAND`).

Agregando la opción `-l` al comando `ps` se mostrarán distintas informaciones acerca de los procesos actualmente en marcha, incluyendo el *estado* de cada proceso (listado bajo `S`). Los códigos utilizados para mostrar esto son los siguientes:

- `O` - El proceso está ejecutándose en un procesador.
- `S` - Durmiendo: el proceso espera a que ocurra un acontecimiento o suceso.
- `R` - Ejecutable: el proceso está en la cola de ejecución.
- `I` - Ocioso: el proceso está siendo creado.
- `Z` - Estado autómeta: el proceso ha terminado y no hay proceso maestro esperando.
- `T` - Rastreado: proceso detenido por una señal porque el proceso maestro lo está rastreando.
- `X` - Estado `SXBRK`: el proceso está esperando más memoria primaria.

Observe que mientras se procesa `ps`, las cosas pueden cambiar. Ya que el comando `ps` le ofrece sólo una foto de lo que está ocurriendo, es la situación un segundo después de escribir el comando. La información puede no estar en concordancia con el momento en que la ve.

El comando `ps` tiene más opciones que las explicadas aquí. Consulte *man Pages(1): User Commands*.

Terminar procesos (`kill`)

El comando `kill` le ofrece una forma directa de parar el procesamiento de los comandos que ya no quiere. Esto es particularmente útil si se equivocó al escribir un comando que tarda mucho tiempo en ejecutarse..

Para terminar un proceso:

1. Escriba `ps` para averiguar el PID(s) del proceso(s).
2. Escriba `kill` seguido del PID(s).

El ejemplo siguiente ilustra este procedimiento:

```
$ ps
PID   TTY    TIME    COMMAND
1291  co     0:12    -bin/csh (csh)
3250  p0     0:00    ps
1286  p1     0:05    -bin/csh (csh)
3248  p1     0:05    vi commands
$ kill 1291
[1] Terminated -bin/csh/ (csh)
$
```

Observe que una forma más rápida de determinar el PID correcto es enviar la salida de `ps` por el canal de comunicación hacia `grep` como se ve a continuación:

```
$ ps | grep nombre_de_comando
```

donde *nombre_de_comando* es el nombre del proceso del comando que quiere parar.

Si necesita forzar la terminación de un proceso, puede utilizar la opción `-9` del comando `ps` tal como se muestra a continuación:

```
$ kill -9 PID#
```

donde *PID#* es el número de identificación del proceso que quiere parar.

Cómo administrar el almacenamiento en disco

Ya que el espacio de almacenamiento en disco es un recurso limitado, es conveniente ir comprobando el espacio actualmente ocupado.

Mostrar la utilización del disco (*df -k*)

df -k le muestra la cantidad de espacio actualmente ocupado en cada disco que esté montado (sea accesible directamente) en su sistema. Sólo escriba:

```
$ df -k
```

para conocer la ocupación de cada disco montado en su sistema, la cantidad disponible, y el porcentaje de uso actual.

Los sistemas de archivos por encima del 90 por ciento de ocupación no debeten archivos innecesarios. Puede hacerlo moviéndolos a un disco o cinta menos ocupada, utilizando *cp* para copiarlos y *rm* para quitarlos, o simplemente puede eliminarlos por completo. Por supuesto, sólo debe llevar a cabo estas tareas de “limpieza doméstica” en archivos que conozca.

Mostrar la utilización del directorio (*du*)

Puede utilizar *du* para mostrar la ocupación de un directorio y de todos sus subdirectorios en bloques de 512 bytes; es decir, unidades de 512 bytes o caracteres.

du le muestra la ocupación en disco de cada subdirectorio. Para obtener la lista de subdirectorios en un sistema de archivos, vaya con *cd* al nombre de ruta de acceso asociado con ese sistema de archivos, y ejecute el siguiente canal de comunicación:

```
$ du | sort -r -n
```

Este canal de comunicación, que utiliza las opciones *-r (reverse)* y *-n (numeric)* del comando `sort`, señalará con toda precisión los directorios grandes. Utilice `ls -l` para examinar el tamaño (en bytes) y las fechas de modificación de los archivos de cada directorio. Los archivos viejos, o los archivos de texto de más de 100 Kbytes, a menudo garantizan la desconexión del almacenamiento.

Uso del editor `vi`



`vi` (pronunciado “vi-ai,” abreviatura de visual display editor) es el editor de texto estándar de SunOS. Puesto que `vi` no está basado en ventanas, este editor multipropósito puede utilizarse en cualquier tipo de terminal para editar un amplio rango de tipos de archivos.

Puede introducir y editar texto con `vi`, pero no es un procesador de textos. No se creó para procesar texto formateado como lo haría un procesador de textos comercial. Para emitir impresiones formateadas, `vi` necesita un programa de emulación de ajuste de escritura, como por ejemplo `nroff`, `troff`, o `ditroff`. Estos programas le permiten formatear texto `vi` insertando códigos que serán interpretados por el emulador.

`vi` contiene una amplia lista de comandos, muchos de los cuales tienen funciones solapadas. Al principio es muy normal que los nuevos usuarios se sientan sobrecargados por ello. El propósito de este capítulo, de cualquier forma, es ofrecer una visión de los comandos `vi` básicos. Al comenzar a utilizar `vi`, encontrará que es un editor de texto extremadamente poderoso, y que puede llevarle algún tiempo el dominarlo.

Observe que existe una versión de `vi` únicamente para leer llamada `view`. Cuando abra un archivo con `view`, puede utilizar comandos `vi`, pero no puede escribir (ni guardar) sus cambios. Esto permite que Vd. u otra persona lean el archivo `vi` sin modificarlo accidentalmente.

Cómo empezar vi

En las subsecciones que siguen, aprenderá como empezar vi, introducir texto en un archivo, guardar (escribir) el archivo, y salir de vi. También creará un archivo de prácticas que utilizará para el resto del capítulo.

Cómo crear un archivo

Empiece vi y edite el archivo `texto` como se muestra en este ejemplo:

```
$ vi texto
```

Si `texto` ya existe, vi lo abrirá, si es un archivo nuevo, vi lo creará. Para el propósito de este ejemplo, `texto` debería ser un archivo nuevo.

En un momento aparecerá la pantalla de edición de vi:

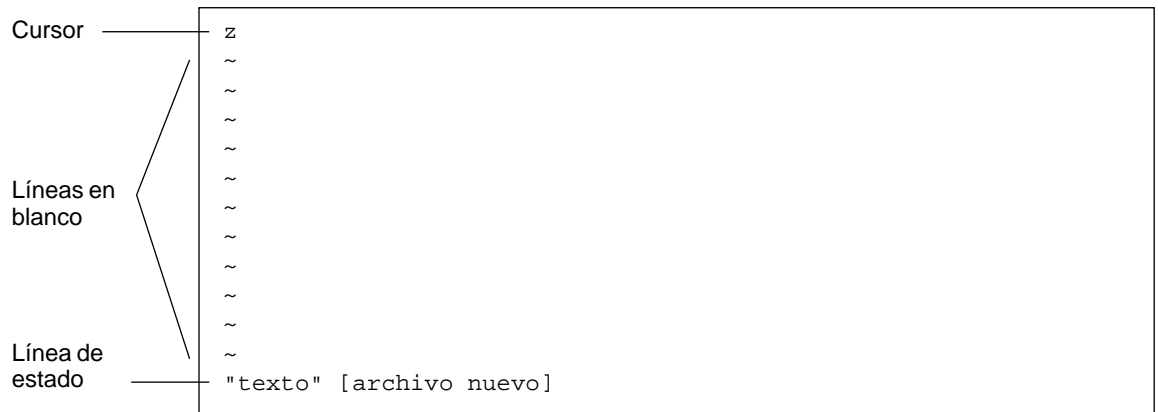


Ilustración 6-1 La pantalla de edición de vi

El cursor aparecerá en la esquina superior izquierda de la pantalla. Las líneas en blanco se indican con una serie vertical de tildes (~).

Observe que también puede empezar vi sin especificar un nombre de archivo introduciendo únicamente vi. Puede nombrar el archivo más tarde, al salir de vi.

La línea de estado

La última línea de la pantalla, llamada *línea de estado*, muestra el nombre del archivo y el número de líneas y caracteres del mismo. Cuando cree un archivo nuevo, como ocurre en este ejemplo, la línea de estado indicará que es un nuevo archivo.

Los dos modos de `vi`

Hay dos modos de operar en `vi`: modo de entrada y modo de comando. Utilice el *modo de entrada* para introducir texto en un archivo, mientras que el *modo de comando* se utiliza para introducir comandos que llevan a cabo funciones específicas de `vi` específicas. El modo de comando es el modo estándar para `vi`.

Ya que `vi` no indica el modo en el que está actualmente, el distinguir entre el modo de comando y el modo de entrada es probablemente la única causa mayor de confusión de los usuarios nuevos de `vi`. De cualquier modo, si recuerda sólo unos pocos conceptos básicos desde el principio, será capaz de evitar todo el “estrés de `vi`”.

Recién abierto el archivo con `vi`, éste estará siempre en modo de comando. Antes de que pueda introducir texto en el archivo, debe escribir uno de los comandos de entrada de `vi`, como por ejemplo `i` (“insertar”), para insertar texto *en* la posición actual del cursor, o bien `a` (“agregar”), para agregar texto *después* de la posición actual del cursor (estos y otros comandos de entrada de `vi` se explican con detalle más adelante en este capítulo).

Siempre que quiera volver al modo de comando de `vi`, presione `Escape`. Si no está seguro del modo en el que se encuentra `vi`, presione simplemente `Escape` para asegurarse de que está en modo de comando y continúe. Si presiona `Escape` cuando `vi` ya se encuentra en modo de comando, el sistema emitirá una señal acústica y la pantalla destelleará sin que esto produzca daño alguno.

Modo de entrada

Para introducir texto en el archivo de ejemplo `texto`, introduzca el comando `i` “insertar” de `vi`. Esto hace que `vi` abandone el modo de comando por el modo de entrada.

Ahora introduzca unas pocas líneas breves, finalizando cada una con un Retorno. Los caracteres que introduzca aparecerán a la izquierda del cursor y desplazarán los caracteres existentes a la derecha. Por el momento, puede corregir los errores retrocediendo con tecla de retroceso y escribiendo de nuevo la línea antes de presionar Retorno. Más tarde aprenderá a editar el texto introducido.

Cuando acabe de introducir texto en `texto`, presione `Escape` para volver al modo de comando. El cursor retrocederá hasta el último carácter introducido. Ahora puede introducir más comandos `vi`.

Si `vi` parece que actúa de forma imprevisible, asegúrese de que no está en modo “Mayúsculas”, lo que haría que todas sus entradas fuesen letras mayúsculas. En algunos sistemas, la tecla `F1` (que normalmente está colocada al lado de la tecla `Escape`) actúa como Bloqueo de mayúsculas. Presionar esta tecla en vez de la tecla `Escape` es un error muy frecuente.

Nota – En ocasiones necesitará pedir a `vi` que borre o vuelva a trazar la pantalla para eliminar, por ejemplo, mensajes ajenos del sistema. Para volver a trazar la pantalla, entre en el modo de comando y presione `Ctrl-L`. Esto es parecido al comando Actualizar de `OpenWindows`.

Modo de comando

Cuando abra un archivo con `vi`, estará en modo de comando. En este modo, puede introducir comandos que implementan un amplio rango de funciones. La mayoría de los comandos `vi` se componen de una o dos letras y un número opcional. Normalmente, hay versiones en minúsculas y en mayúsculas de los comandos que ejecutan funciones relacionadas pero distintas. Como ejemplo, al introducir `a` se agregará el texto a la derecha del cursor, mientras que al escribir `A` se añadirá el texto *al final* de la línea.

La mayoría de los comandos `vi` no precisan que Vd. escriba Retorno para ejecutarlos. Sin embargo, los comandos que empiezan con los dos puntos (`:`) si que requieren que presione Retorno a continuación del comando. Algunos tratados acerca del editor `vi` se refieren a los comandos precedidos por los dos puntos como un tercer modo por separado de `vi`, el *modo de última línea*. Esto se debe a que cuando escribe los dos puntos mientras está en modo de comando, los dos puntos y el resto de lo escrito aparece en la parte inferior de la pantalla. De cualquier manera, en este manual, todos los comandos `vi` se inician desde el modo de comando.

Los comandos precedidos por dos puntos son de hecho comandos *ex*. *vi* y *ex* son dos interfaces separadas para el mismo programa de edición. Mientras que *vi* es una interfaz orientada en pantalla, *ex* es una interfaz orientada en línea. El conjunto completo de comandos *ex* está disponible desde *vi*. Cuando presione los dos puntos, de hecho está cambiando a la interfaz *ex* orientada en línea. Esto le permitirá ejecutar muchos comandos de archivo sin tener que abandonar *vi*. Consulte “Uso de los comandos *ex*” de este capítulo, si desea más información.

Cómo finalizar una sesión

Cuando edita un archivo con *vi*, sus cambios no se hacen directamente en el archivo. En realidad se aplican a una copia del archivo que *vi* crea en un espacio de memoria temporal llamado *buffer* o memoria intermedia. La copia permanente en disco del archivo se modificará sólo cuando *escriba* (guarde) el contenido de la memoria intermedia.

Esta disposición tiene sus aspectos buenos y malos. Por un lado, significa que puede salir de un archivo y descartar todos los cambios que ha hecho durante una sesión de edición, dejando la copia en disco intacta. Por otro lado, podría perder el contenido (no guardado) de la memoria intermedia de trabajo si el sistema tiene una avería o fallo (las personas en terminales remotos conectados por líneas de teléfono son especialmente vulnerables frente a interrupciones no planificadas).

Lo más conveniente es guardar su trabajo frecuentemente, especialmente cuando hace cambios sustanciales.



Atención– Aunque es posible ejecutar sesiones *vi* múltiples y simultáneas de un mismo archivo, no es buena idea. Puede dar lugar a confusión cuando intente averiguar qué cambios se han escrito en el archivo y cuales se han sobrescrito provenientes de otra sesión simultánea.

Cómo guardar los cambios y salir de *vi*

Abundan en *vi* comandos más o menos sinónimos que controlan la copia del contenido de la memoria intermedia en un archivo y la salida de *vi*. Estos comandos le ofrecen las opciones de guardar, guardar y salir, o salir sin guardar.

Cómo guardar

Guarde el contenido de la memoria intermedia (escriba la memoria intermedia en el archivo en disco) escribiendo:

```
:w
```

seguido de Retorno.

Cómo guardar y salir

Guarde y salga escribiendo:

```
:wq
```

seguido de Retorno. Otra forma de hacer esto es introduciendo ZZ.

Observe que el comando ZZ no va precedido por los dos puntos ni seguido de Retorno.

Cómo salir sin guardar

Cuando no ha hecho cambios en el archivo y simplemente quiere salir, escriba:

```
:q
```

seguido de Retorno. Si hizo cambios, vi no le permitirá salir con :q. Mostrará el mensaje `No write since last change (:quit! overrides)`.

Si no quiere guardar los cambios que haya hecho, escriba:

```
:q!
```

seguido de Retorno.

Cómo imprimir un archivo

Una vez que ha salido de un archivo `vi`, puede imprimir el archivo con el comando siguiente:

```
$ lp nombre_de_archivo
```

donde *nombre_de_archivo* es el nombre del archivo `vi` a imprimir. Este comando imprime el archivo en su impresora predeterminada. El archivo se imprime sin ningún formato, línea tras línea, tal como aparece en la pantalla. Consulte el Capítulo 8, “Uso de las impresoras,” si desea más información acerca de los comandos de impresión.

Comandos `vi` básicos

Las secciones siguientes explican varias categorías de comandos `vi`. Estas incluyen:

- Cómo desplazarse por un archivo
- Cómo insertar texto
- Cómo cambiar y sustituir texto
- Cómo deshacer los cambios de texto
- Cómo eliminar texto
- Cómo copiar y mover texto
- Cómo repetir comandos

Cómo desplazarse por un archivo

En las secciones previas ha aprendido a crear, guardar, imprimir, y salir de un archivo `vi`. Ahora que ha creado un archivo, necesitará entender los conceptos necesarios para desplazarse por él. Abra su archivo de prácticas ahora, e intente probar cada uno de los comandos discutidos en esta sección.

Cómo mover el cursor

Cuando empieza `vi`, el cursor estará en la esquina superior izquierda de la pantalla de `vi`. En modo de comando, puede mover el cursor con varios comandos de teclado. Ciertas teclas de letras, las teclas de dirección, y la tecla Retorno, la tecla de retroceso (o Eliminar), y la Barra Espaciadora pueden utilizarse para mover el cursor cuando esté en modo de comando.

Nota – La mayoría de los comandos `vi` distinguen entre letras mayúsculas y minúsculas, el “mismo” comando escrito en mayúscula y en minúscula puede tener efectos radicalmente diferentes.

Cómo moverse con las teclas de dirección

Si su sistema está equipado con teclas de dirección, pruébelas ahora. Debe ser capaz de mover el cursor libremente por la pantalla utilizando combinadas las teclas de flecha arriba, abajo, derecha, e izquierda. Observe que sólo puede mover el cursor sobre el texto ya existente o sobre los espacios de entrada.

Si usa `vi` desde un terminal remoto, las teclas de dirección podrían no funcionar correctamente. Esto dependerá de su emulador de terminal. Si en su caso las teclas de dirección no funcionan, puede sustituirlas con las siguientes:

- Para moverse a la izquierda, presione `h`.
- Para moverse a la derecha, presione `l`.
- Para moverse hacia abajo, presione `j`.
- Para moverse hacia arriba, presione `k`.

Cómo moverse una palabra

Presione `w` (“word”, palabra) para mover el cursor una palabra hacia la derecha.

Presione `b` (“back”, hacia atrás) para mover el cursor una palabra hacia la izquierda.

Presione `W` o `B` para mover el cursor después del carácter de puntuación contiguo hasta el espacio en blanco siguiente o anterior.

Presione `e` (“end”, al final) para mover el cursor hasta el último carácter de la palabra actual.

Cómo moverse al inicio o al final de una línea

Presione ^ para mover el cursor al inicio de la línea actual.

Presione \$ para mover el cursor al final de la línea actual.

Cómo moverse una línea hacia abajo

Presione la tecla Retorno para mover el cursor al principio de la línea siguiente.

Cómo moverse hacia la izquierda

Presione la tecla de retroceso para mover el cursor un carácter hacia la izquierda.

Cómo moverse hacia la derecha

Presione la Barra espaciadora para mover el cursor un carácter hacia la derecha.

Cómo moverse hasta arriba

Presione H (“high”, arriba) para mover el cursor hasta la parte de arriba de la pantalla.

Cómo moverse al centro

Presione M (“middle”, centro) para mover el cursor hasta el centro de la pantalla.

Cómo moverse hasta la parte inferior

Presione L (“low”, inferior) para mover el cursor a la parte inferior de la pantalla.

Cómo paginar y desplazarse

Si se mueve hacia abajo cuando el cursor está en la parte inferior de la pantalla, o hacia arriba cuando el cursor está en la parte superior de la pantalla, verá el texto desplazarse hacia arriba o hacia abajo. Esta puede ser una manera efectiva de mostrar más texto en un archivo muy corto, pero puede resultar tedioso el moverse de esta manera por un archivo largo.

Puede haber observado que moviendo el cursor hacia abajo una vez que se ha alcanzado la parte inferior de la pantalla, o bien hacia arriba una vez alcanzada la parte superior de la pantalla tiene el mismo efecto que desplazar texto hacia arriba o hacia abajo. Esto funcionará en un archivo corto, pero es una manera tediosa de moverse por un archivo largo.

Puede cambiar de página o desplazarse hacia adelante o hacia atrás a través de un archivo, avanzando o retrocediendo una o media pantalla cada vez (para intentar este comando en `texto`, deberá agregar el texto necesario para hacer el archivo lo suficientemente grande para que el comando tenga efecto).

Observe que hay una diferencia fundamental entre cambiar de página y desplazar. Desplazar, desplaza realmente el cursor hacia arriba o hacia abajo en el texto *de línea en línea*, como si se desplazase sobre papel. Pagarinar mueve el cursor hacia arriba o hacia abajo en el texto *de pantalla en pantalla*. En un sistema rápido, podría no notar la diferencia. De cualquier modo, si está trabajando en un terminal remoto o en alguna otra situación en la que su sistema trabaja más lento de lo habitual, esta diferencia se convertirá lamentablemente en evidente.

Cómo paginar hacia adelante una pantalla

Para desplazarse hacia adelante (moverse hacia abajo) una pantalla completa, presione Ctrl-F (mantenga presionada la tecla Control y presione la tecla F). El cursor se moverá a la esquina superior izquierda de la pantalla nueva.

Cómo desplazarse media pantalla hacia adelante

Para desplazarse hacia adelante media pantalla, presione Ctrl-D.

Cómo paginar hacia atrás una pantalla

Para desplazarse hacia atrás (es decir, moverse hacia arriba) una pantalla completa, presione Ctrl-B.

Cómo desplazarse media pantalla hacia atrás

Para desplazarse hacia atrás media pantalla, presione Ctrl-U.

Cómo insertar texto

`vi` ofrece muchos comandos para insertar texto. Esta sección le presenta los más útiles. Observe que cada uno de estos comandos sitúa a `vi` en modo de entrada. Para utilizar cualquiera de estos comandos, primero debe estar en modo de comando. Acuérdesse de presionar Escape para asegurarse de que está en modo de comando.

Cómo agregar

Escriba `a` (agregar) para insertar texto a la *derecha* del cursor. Compruébelo moviendo el cursor a cualquier parte de una línea y escribiendo `a`, seguido del texto que quiera agregar. Presione Escape cuando haya terminado.

Escriba `A` para agregar texto al *final* de una línea. Para ver como funciona, sitúe el cursor en cualquier parte de una línea y escriba `A`. El cursor se moverá al final de la línea, donde podrá escribir el texto a agregar. Presione Escape cuando lo haya hecho.

Cómo insertar

Inserte texto a la izquierda del cursor escribiendo `i` desde el modo de comando.

Escriba `I` para insertar texto al principio de una línea (el comando moverá el cursor desde cualquier posición en la línea.). De nuevo, y al igual que todos los comandos en esta sección, presione Escape para volver al modo de comando después de escribir el texto deseado.

Cómo abrir línea

Use estos comandos para abrir líneas nuevas, bien encima o debajo de la posición del cursor actual.

Escriba `o` para abrir una línea *debajo* de la posición del cursor actual. Para probarlo, escriba `o` seguido de un poco de texto. Puede escribir varias líneas de texto si lo desea. Presione Escape cuando haya terminado.

Escriba `O` para abrir una línea *encima* de la posición del cursor actual.

Cómo cambiar texto

Cambiar texto lleva consigo sustituir una parte de texto por otra. `vi` tiene varias formas de hacerlo, dependiendo de las circunstancias.

Cómo cambiar una palabra

Para reemplazar una palabra, sitúe el cursor al principio de la palabra que va a ser reemplazada. escriba `cw`, seguido de la palabra nueva. Para terminar, presione Escape.

Para cambiar *parte* de una palabra, sitúe el cursor en la palabra, a la *derecha* de la parte a guardar. Escriba `cw`, introduzca la corrección, y presione Escape.

Cómo cambiar una línea

Para reemplazar una línea, sitúe el cursor en cualquier lugar de la línea y escriba `cc`. La línea desaparecerá, dejando una línea en blanco para el texto nuevo (que puede ser de cualquier longitud). Presione Escape para terminar.

Cómo cambiar parte de una línea

Para reemplazar parte de una línea, sitúe el cursor a la *derecha* de la parte a guardar. escriba `C`, introduzca la corrección, y presione Escape. Esto cambiará la parte de la línea desde la posición actual del cursor hasta el final de la línea.

Cómo sustituir uno o varios caracteres

Para sustituir uno o más caracteres a partir del carácter bajo el cursor, escriba `s`, seguido por el texto nuevo. Presione Escape para volver al modo de comando.

Cómo reemplazar un carácter

Use este comando que sirve para reemplazar el carácter resaltado por el cursor por otro carácter. Sitúe el cursor sobre el carácter y escriba `r`, seguido de un único carácter de reemplazo. Después de la sustitución, `vi` volverá automáticamente al modo de comando (no es necesario presionar Escape).

Cómo transponer caracteres

Para corregir los caracteres transpuestos en `vi` sólo se necesita escribir dos veces. Suponga que escribió "lso" cuando lo que quería era "los". Haga la correspondiente corrección colocando el cursor sobre la primera letra que se quiere mover (en este caso, e), y escribiendo a continuación `xp`. La e y la h intercambiarán sus lugares – y `vi` volverá automáticamente al modo de comando.

Cómo partir o unir líneas

Para partir una línea sin que afecte al texto, mueva el cursor al espacio donde quiera partir la línea y escriba `r` (para “reemplazar”) seguido de Retorno. Observe que si escribe `r` con el cursor sobre un carácter y a continuación presiona Retorno, ese carácter se reemplazará por Retorno.

Para unir dos líneas, sitúe el cursor en la línea superior y escriba una `J` mayúscula (no es necesario presionar Escape después de escribir `J`.)

Cómo deshacer los cambios

Cuando edite texto y haga cambios en un archivo `vi`, podría ocurrir que en algún momento desee no haber hecho un cambio en el texto. Los comandos `undo` (deshacer) de `vi` le permitirán anular los efectos de una operación y continuar la edición desde el momento anterior.

Cómo deshacer los comandos anteriores

Si se equivocó en `vi` o si cambió de idea una vez que una operación se completó, puede deshacer el último comando presionando `u` inmediatamente después del comando (no es necesario presionar Escape después de escribir `u`). Si presiona `u` por segunda vez deshace la operación de deshacer.

Cómo deshacer cambios en una línea

Escriba `U` para deshacer todos los cambios que hizo en una línea. Este comando funcionará únicamente antes de mover el cursor a otra línea (no es necesario presionar Escape después de escribir `U`).

Cómo eliminar texto

Estos comandos `vi` eliminan el carácter, la palabra o la línea que Vd. indique. `vi` seguirá en modo de comando, por lo que cualquier inserción posterior de texto deberá ir precedida por algún comando que pase a modo de entrada.

Cómo eliminar un carácter

Para eliminar un carácter, sitúe el cursor sobre el carácter a borrar y escriba `x`.

El comando `x` también elimina el espacio que ocupaba el carácter -cuando en la mitad de una palabra se elimina una letra, las letras restantes se cerrarán, sin dejar un hueco. También puede eliminar los espacios en blanco de una línea con el comando `x`.

Para eliminar el carácter anterior al cursor (a su izquierda), escriba `X` (en mayúsculas).

Cómo eliminar una palabra o parte de una palabra

Para eliminar una palabra, sitúe el cursor al principio de la palabra y escriba `dw`. Se eliminarán la palabra y el espacio que ocupaba.

Para eliminar parte de una palabra, sitúe el cursor en la palabra a la *derecha* de la parte a guardar. escriba `dw` para eliminar el resto de la palabra.

Cómo eliminar una línea

Para eliminar una línea, sitúe el cursor en cualquier lugar de la línea y escriba `dd`. Se eliminarán la línea y el espacio que ocupaba..

Cómo eliminar parte de una línea

También puede eliminar parte de una línea.

Para eliminar todo lo que se encuentre a la *derecha* del cursor, sitúelo a la derecha de la parte de la línea que desee guardar, y escriba `D`.

Para eliminar todo lo situado a la *izquierda* del cursor, sitúelo a la derecha de la parte de la línea que quiera eliminar e introduzca `d0` (`d`-cero).

Cómo eliminar hasta el final del archivo

Para eliminar todo el texto comprendido entre la línea actual y el final del archivo, escriba `dG`. También se eliminará la línea donde estaba el cursor.

Cómo eliminar desde el principio del archivo

Para eliminar todo el texto desde el principio del archivo hasta la línea actual, escriba `d1G`. También se eliminará la línea donde estaba el cursor.

Cómo copiar y mover texto -Tirar, eliminar y poner

Muchos procesadores de textos le permiten “copiar y pegar” y “cortar y pegar” líneas de texto. El editor `vi` también incluye estas características. El modo de comando de `vi` equivalente a “copiar y pegar” es *tirar y poner*; el equivalente a “cortar y pegar” es *eliminar y poner*.

Los métodos para copiar y mover pequeños bloques de texto en `vi` lleva consigo la utilización combinada de los comandos `yank`, `delete` y `put`.

Cómo copiar líneas

Para copiar una línea se necesitan dos comandos: `yy` o `Y` (“tirar”) y `p` (“poner abajo”) o `P` (“poner arriba”). Observe que `Y` hace lo mismo que `yy`.

Para tirar una línea, sitúe el cursor en cualquier lugar de la línea y escriba `yy`. Ahora mueva el cursor a la línea de arriba donde quiere colocar la línea tirada (copiada), y escriba `p`. Una copia de la línea tirada aparecerá en la línea nueva *debajo* del cursor.

Para situar la línea tirada en una línea nueva *encima* del cursor, escriba `P`.

El comando `yy` trabaja bien con un contador: para tirar 11 líneas, por ejemplo, basta con escribir `11yy`. Once líneas, contando hacia abajo desde el cursor, se tirarán, y `vi` indica esto con un mensaje en la parte inferior de la pantalla: `11 lines yanked`.

Puede utilizar también los comandos `P` o `p` inmediatamente después de cualquiera de los comandos de eliminación explicados anteriormente. Esto pone el texto que eliminó por encima o por debajo del cursor, respectivamente.



Atención– Utilice únicamente los comandos de movimiento del cursor entre las operaciones de tirar o eliminar y poner. Si elimina o tira cualquier otro texto antes de poner el texto nuevo en su lugar, las líneas que tiró o eliminó se habrán perdido.

Cómo mover líneas

Mover líneas requiere también dos comandos: `dd` (“borrar”) y `p` o bien `P`.

Para mover una línea, sitúe el cursor en cualquier lugar de la línea y escriba `dd`. Por ejemplo, para eliminar 5 líneas, escriba `5dd`.

A continuación, mueva el cursor a la línea por encima donde quiere reinsertar la línea eliminada y escriba `p`. Esto inserta el texto en una línea nueva debajo del cursor.

Por otro lado, puede poner la línea eliminada por encima del cursor escribiendo `P`.

Uso de memoria intermedia con nombre

Para insertar repetidamente un grupo de líneas en varios lugares del documento, puede tirar (o eliminar) las líneas en una memoria intermedia con nombre. Puede especificar memoria intermedia con nombre haciendo preceder un comando de dobles comillas (") y un nombre para la memoria intermedia. Por ejemplo, para tirar cuatro líneas sobre la memoria intermedia de nombre *a*, escriba `"a4yy`. Puede utilizar varias memorias intermedias. Por ejemplo, podría eliminar texto a partir de una posición y agregarlo a otras varias. Para eliminar 12 líneas de la memoria intermedia de nombre *b*, escriba `"b12dd`.

Para insertar el texto, escriba el comando `p` o `P` con "*n*", donde *n* es la memoria intermedia con nombre. Por ejemplo, para insertar las líneas guardadas en la memoria intermedia *b*, introduzca `"bP`.

Puede sobrescribir memorias intermedias con nombre con líneas nuevas. Las memorias intermedias se guardan hasta que salga de `vi`.

Cuando use memorias intermedias con nombre, puede borrar y tirar otro texto con la seguridad de no afectar a las líneas que ya ha guardado en las memorias intermedias con nombre -a no ser, por supuesto, que sobrescriba intencionalmente esta memoria intermedia con nombre.

Uso de un contador para repetir comandos

Muchos comandos `vi` pueden ir precedidos por un factor de repetición (llamado *contador*) -un número que precede el comando y le indica cuantas veces debe repetir la operación.

La mayoría de los comandos de las secciones anteriores llevan contadores. Por ejemplo, `3dd` repite el comando para eliminar una línea tres veces, por tanto elimina tres líneas. `2dw` elimina dos palabras, y `4x` elimina cuatro caracteres o espacios. También puede utilizar contadores junto con comandos para mover el cursor, como `3w` y `2Ctrl-F`. Esto se hará evidente a medida que aprenda los comandos de `vi`. En la sección, “Resumen de los comandos básicos de `vi`” al final de este capítulo, cada comando que puede llevar un contador está indicado por “[count]” antes del nombre del comando.

Si escribe un punto (.) se repetirá el comando previo de cambio de texto. Por ejemplo, si únicamente ha eliminado una línea con `dd`, puede mover el cursor a otra línea y eliminarla escribiendo sólo un punto.

Uso de los comandos `ex`

Los comandos de `ex` son más precisos y convenientes que `tirar`, `borrar`, y `poner` cuando esté trabajando con bloques grandes de texto. En vez de contar líneas en la pantalla y después buscar un punto de inserción, dará a `vi` un rango de líneas a mover o copiar y después especificar la línea antes del punto de inserción (por supuesto, con un comando de eliminación no es necesario un punto de inserción).

Cómo activar y desactivar la numeración de las líneas

Para *activar* la numeración de las líneas, escriba `:set nu` y presione Retorno.

Los números de las líneas aparecerán en el margen izquierdo. Observe que estos números no aparecerán si imprime el archivo. Sólo son visibles en la pantalla.

```
1 El deseo de escribir con mayor rapidez,  
2 y la necesidad de encerrar en poco  
3 muchas noticias, fueron causa de  
4 abreviar ciertos vocablos que  
5 pudieran adivinarse fácilmente.  
6  
7  
8  
~  
~  
~  
~  
~  
~  
~  
:set nu
```

Para *desactivar* la numeración de las líneas, escriba `:set nonu` y presione Retorno.

Cómo copiar líneas

El formato básico del comando copiar en *ex* es:

```
:line#,line# co line#
```

Los primeros dos números (separados por una coma) especifican el rango de líneas a copiar. El tercer número es la línea *antes* del punto de inserción.

Por ejemplo, para copiar las líneas de la 1 a la 5 de *texto* y situar la copia después de la línea 12, debe escribir lo siguiente:

```
:1,5 co 12
```

y presionar Retorno.

Para especificar rangos de líneas puede utilizar estas abreviaturas:

- Un punto (.) para indicar “desde la línea actual”.
- Signo dólar (\$) para indicar “hasta el final del archivo”.

Por tanto, para copiar el rango “desde la línea actual hasta la línea 5” e insertar este texto a partir de la línea 12, debe escribir:

```
:.,5 co 12
```

Para copiar el rango “desde la línea 6 hasta el final del archivo” e insertar este bloque después de la línea 2, debe escribir:

```
:6,$ co 2
```

Cómo mover líneas

El formato básico del comando de mover en `ex` es parecido al comando de copiar explicado anteriormente:

```
:line#,line# m line#
```

Los rangos de líneas y los puntos de inserción se especifican de igual forma, incluyendo el uso de las abreviaturas `.` y `$`. La diferencia en el funcionamiento es simplemente que “move” elimina un bloque de una posición y lo coloca en otra.

Por ejemplo, para mover las líneas de la 1 hasta la 5 a la línea siguiente a la 12, debe escribir:

```
:1,5 m 12
```

y presione Retorno.

Cómo eliminar líneas

Para eliminar un rango de líneas, use este formato del comand

```
:line#,line# d
```

Por ejemplo, para eliminar las líneas de la 1 a la 5, debe introducir:

```
:1,5 d
```

Cómo buscar y reemplazar con `vi`

`vi` ofrece distintas formas de situarse en un archivo localizando una cadena de caracteres. También dispone de una poderosa función de reemplazo general.

Cómo buscar una cadena de caracteres

Una *cadena de caracteres* consiste simplemente en una serie de caracteres puestos en fila. Puede contener letras, números, signos de puntuación, caracteres en pantalla, espacios en blanco, tabulaciones, o retornos de carro. Una cadena puede ser una palabra gramatical o cualquier parte de una palabra.

Para encontrar una cadena de caracteres, escriba `/` seguido de la cadena que quiera buscar, y a continuación presione Retorno. `vi` sitúa el cursor en la siguiente aparición de la cadena. Por ejemplo, para encontrar la cadena “meta”, escriba `/meta` y a continuación Retorno.

Introduzca `n` para ir a la *siguiente* aparición de la cadena; escriba `N` para ir a la aparición *previa*.

Para buscar hacia atrás en el archivo, puede utilizar `?` en vez de `/`. En este caso, el sentido de la dirección en `n` y `N` está cambiado.

La operación de buscar normalmente distingue entre mayúsculas y minúsculas: si busca “china” no encontrará “China”. Si quiere que `vi` no distinga entre mayúsculas y minúsculas al buscar, escriba `:set ic`. Para volver al modo por omisión, que tiene en cuenta las mayúsculas y minúsculas, escriba `:set noic`.

Si `vi` encuentra la cadena solicitada, el cursor se parará en su primera aparición. Si no encuentra la cadena, `vi` mostrará el mensaje `Muestra no encontrada` en la última línea de la pantalla.

Ciertos caracteres especiales (`/` `&` `!` `.` `^` `*` `$` `\` `?`) tienen un significado especial en el proceso de búsqueda y deben ir precedidos de una barra inversa cuando se utilizan en una búsqueda. Para “escapar” un carácter especial, hágalo preceder de una barra inversa (`\`). Por ejemplo, para buscar la cadena “cómo?” escriba `/cómo\?` y presione Retorno.

Estos caracteres especiales pueden utilizarse como comandos de la función de búsqueda, por lo que si quiere buscar una cadena que incluye uno o más de estos caracteres, debe indicarlo precediendo el carácter con una barra inversa. Para escapar de la propia barra inversa, escriba `\\`.

Búsqueda más exacta

Puede hacer la búsqueda más precisas marcando la cadena con indicadores para las siguientes características:

- Principio de línea
- Final de línea
- Principio de palabra
- Final de palabra
- Carácter comodín

Para que coincida con el principio de línea, comience la cadena de caracteres con el símbolo (`^`). Por ejemplo, para encontrar la siguiente línea que empiece con “Buscar”, escriba:

```
/^Buscar
```

Para que coincida con el final de línea, termine la cadena de caracteres con un signo dólar (`$`). Por ejemplo, para encontrar la siguiente línea que termine con “buscar.”, escriba:

```
/buscar\.$
```

(Observe que el punto va precedido por una barra inversa).

Para que coincida con el principio de palabra, escriba \`<` al principio de la cadena de caracteres, para que coincida con el final de palabra, escriba \`>` al final de la cadena de caracteres. Así, para que concuerde exactamente una palabra y no una cadena, combine las marcas de principio de palabra y de final de palabra en el patrón de búsqueda. Por ejemplo, para encontrar la siguiente aparición de la palabra -en oposición a la cadena -“buscar”, escriba:

```
/\<buscar\>
```

Para indicar un carácter cualquiera, escriba un punto (`.`) en la cadena en la posición donde coincide. Por ejemplo, para encontrar la siguiente aparición de “informe” o “informa”, escriba:

```
/inform.
```

Debido a que con esto se busca una cadena y no una palabra, este modelo de búsqueda podría encontrar construcciones como “información” e “informal”.

Para buscar caracteres alternativos en una cadena, escriba las alternativas entre corchetes. El modelo de búsqueda `/[md]cadena` encontrará las cadenas que empiecen por m o d. Por otra parte, `/[d-m]cadena` encontrará las cadenas que empiecen por cualquier letra entre la d y la m.

Para coincidir con cero o más apariciones del último carácter, escriba un símbolo asterisco (`*`) en la cadena. Puede combinar para más efectividad los corchetes junto con el asterisco para buscar alternativas muy concretas. Por ejemplo, para buscar todas las cadenas que terminen con cualquier letra desde la a hasta la z y que empiecen en “inform” y además encontrar las apariciones de “inform”, escriba:

```
/inform[a-z]*
```

Cómo reemplazar una cadena de caracteres

El procedimiento para reemplazar una cadena de texto está basado en los procedimientos de búsqueda explicados anteriormente. Todos los caracteres especiales para las comparaciones en las búsquedas pueden utilizarse en la operación de buscar y reemplazar.

El formato básico del comando es:

```
:g/search-string/s//replace-string/g
```

seguido de la tecla Retorno.

Por tanto, para reemplazar cualquier aparición de la cadena de caracteres “informe” con “documento”, debe escribir:

```
:g/informe/s//documento/g
```

y presionar Retorno.

Puede modificar este comando para detener la búsqueda y hacer que `vi` le pregunte cada vez si desea reemplazar o no. El comando siguiente utiliza `gc` (agregando `c` de “consultar”) para hacer que `vi` se detenga en cada aparición de “informe” y le pregunte si desea sustituir la cadena. Responda `y` para sí o `n` para no.

```
:g/informe/s//documento/gc
```

Nota – Puede cancelar una búsqueda y reemplazamiento “consultada” presionando `Ctrl-C`.

Cómo ir a una línea específica

Para ir a la última línea de un archivo abierto, escriba `G`. para volver a la primera línea del archivo, escriba `1G`.

Puede ir a cualquier otra línea escribiendo su número seguido de `G`.

Por ejemplo, suponga que sale del archivo `texto` mientras editaba la línea 51. Puede acceder a esa línea abriendo el archivo y escribiendo `51G`.

Cómo insertar un archivo en otro

Con `vi` es muy cómodo el “leer” (insertar) un archivo en el archivo que está editando. El formato general del comando es:

```
:line# r nombre_de_archivo
```

Si no especifica un número de línea, `vi` inserta el archivo en la posición actual del cursor.

Por ejemplo, si quiere insertar el archivo `cervantes` en la línea 84 del archivo `texto`, debe escribir:

```
:84 r cervantes
```

O puede situar el cursor en la línea 84 y escribir:

```
:r cervantes
```

Cómo editar archivos múltiples

`vi` le permite editar archivos múltiples. Por ejemplo, para editar el archivo `cervantes` mientras edita `texto`:

- 1. Primero, guarde su trabajo actual en `texto`. Escriba `:w` y presione Retorno.**
- 2. Para editar `cervantes`, escriba `:n cervantes` y presione Retorno.**
- 3. Haga cambios de edición en `cervantes` y guarde su trabajo.**
- 4. Cuando haya terminado de trabajar con `cervantes` y haya guardado su trabajo, tiene tres opciones:**
 - Salir de `vi`. Escriba `:q` y presione Retorno.
 - Volver a `texto`. Escriba `:n #` y presione Retorno.
 - Cambie de uno a otro archivo con el comando `:n #`.

Cómo editar una serie de archivos

Para editar una serie de archivos, liste los nombres de los archivos a continuación del comando `vi` cuando arranque `vi` en el indicador de comando:

```
$ vi texto cervantes
```

Los archivos aparecerán en el orden en el que se listaron. Primero aparecerá `texto`. Cuando haya terminado de editar `texto`, escriba `:n` para ir al siguiente archivo, `cervantes`. Para ir al siguiente archivo sin guardar los cambios hechos en el archivo actual, escriba `:n!` en vez de `:n`.

Si tiene una serie de archivos con nombres relacionados (por ejemplo, `test1`, `test2`, `test3`), puede utilizar caracteres comodines para especificar un grupo de archivos:

```
$ vi test*
```

Los archivos irán apareciendo para la edición en orden alfabético.

Cómo copiar líneas entre archivos

Para copiar líneas de un archivo a otro, haga lo siguiente:

1. Edite el primer archivo.
2. Guarde las líneas deseadas en memorias intermedias con nombre, utilizando el comando `yank`. Por ejemplo, para guardar 10 líneas en la memoria intermedia `a`, escriba `a10Y`.
3. Sin salir de `vi`, edite el siguiente archivo (`cervantes` en este ejemplo):

```
:n cervantes
```

4. Agregue las líneas del primer archivo con el comando `put`. Por ejemplo, para poner los contenidos de la memoria intermedia `a` debajo de la posición actual del cursor, escriba `ap`.

Recuerde que los contenidos de todas las memorias intermedias con nombre se pierden cuando sale de `vi`. No utilice el comando de salir (`:q`) hasta que haya terminado cualquier operación que necesite las memorias intermedias con nombre y su contenido.

Cómo establecer parámetros `vi`

`vi` tiene muchas variables que afectan a su comportamiento y a su aspecto. Puede ver una lista de estas variables (junto con su configuración actual) mientras procesa `vi`. Para ello escriba:

```
:set all
```

seguido de Retorno.

Recuperación tras una caída del sistema

Si el sistema cae, el contenido de su memoria intermedia está en peligro. Sin embargo a menudo es posible recuperar la mayoría de su trabajo al reiniciar de nuevo `vi` con el formato del comando:

```
vi -r nombre_de_archivo
```

donde *nombre_de_archivo* es el archivo que estaba editando cuando ocurrió la caída. Normalmente el sistema le enviará un aviso por correo después de que haya acabado la reinicialización, avisándole de que hay un archivo de recuperación.

Resumen de los comandos básicos de `vi`

En la tabla siguiente puede consultar de forma cómoda los comandos básicos de `vi`.

Tabla 6-1 Comandos básicos de `vi`

Comando	Significado
<i>Empezar vi</i>	
<code>vi nombre_de_archivo</code>	Abrir o crear el archivo
<code>vi</code>	Abrir un archivo nuevo para nombrarlo más tarde
<code>vi -r nombre_de_archivo</code>	Recuperar un archivo de una caída del sistema
<code>view nombre_de_archivo</code>	Abrir archivo sólo para leer
<i>Comandos del cursor</i>	
<code>h</code>	Moverse un carácter hacia la izquierda
<code>j</code>	Moverse una línea hacia abajo
<code>k</code>	Moverse una línea hacia arriba
<code>l</code>	Moverse un carácter a la derecha
<code>w</code>	Moverse una palabra a la derecha
<code>W</code>	Moverse una palabra a la derecha (pasados los signos de puntuación)
<code>b</code>	Moverse una palabra a la izquierda
<code>B</code>	Moverse una palabra a la izquierda (pasados los signos de puntuación)
<code>e</code>	Moverse al final de la palabra actual
<code>Return</code>	Moverse una línea hacia abajo
<code>Back Space</code>	Moverse un carácter a la izquierda
<code>Space Bar</code>	Moverse un carácter a la derecha
<code>H</code>	Moverse a la parte de arriba de la pantalla
<code>M</code>	Moverse al centro de la pantalla
<code>L</code>	Moverse a la parte inferior de la pantalla
<code>Ctrl-F</code>	Paginar una pantalla hacia adelante

Tabla 6-1 Comandos básicos de vi (Continuación)

Comando	Significado
Ctrl-D	Desplazarse media pantalla hacia adelante
Ctrl-B	Paginar una pantalla hacia atrás
Ctrl-U	Desplazarse media pantalla hacia atrás
Insertar caracteres y líneas	
a	Insertar caracteres a la derecha del cursor
A	Insertar caracteres al final de la línea
i	Insertar caracteres a la izquierda del cursor
I	Insertar caracteres al principio de línea
o	Insertar una línea por debajo el cursor
O	Insertar una línea por encima del cursor
Cambiar texto	
cw	Cambiar una palabra (o parte de una palabra) a la derecha del cursor
c	Cambiar una línea
C	Cambiar desde el cursor hasta el final de la línea
s	Sustituir cadena por carácter(es) desde el cursor hacia adelante
r	Reemplazar el carácter marcado por cursor por otro carácter
r Return	Partir una línea
J	Unir la línea actual con la línea inferior
xp	Transponer el carácter del cursor con el carácter a la derecha
~	Cambiar el tipo de letra (mayúscula o minúscula)
u	Deshacer el comando anterior
U	Deshacer todos los cambios en la línea actual
:u	Deshacer el comando anterior sobre la línea última
Eliminar texto	

Tabla 6-1 Comandos básicos de vi (Continuación)

Comando	Significado
x	Eliminar el carácter del cursor
X	Eliminar el carácter a la izquierda del cursor
dw	Eliminar la palabra (o la parte de la palabra a la derecha del cursor)
dd	Eliminar la línea que contiene al cursor
D	Eliminar la parte de la línea a la derecha del cursor
dG	Eliminar hasta el final de línea
d1G	Eliminar desde el principio del archivo hasta el cursor
:5,10 d	Eliminar las líneas de la 5 a la 10
Copiar y mover texto	
yy	Tirar o copiar línea
Y	Tirar o copiar línea
p	Poner la línea tirada o eliminada por debajo de la línea actual
P	Poner la línea tirada o eliminada por encima de la línea actual
:1,2 co 3	Copiar las líneas de la 1 a la 2 y ponerlas después de la línea 3
:4,5 m 6	Mover las líneas de la 4 a la 5 y ponerlas después de la línea 6
Ajustar la numeración de las líneas	
:set nu	Mostrar los números de las líneas
:set nonu	Esconder los números de las líneas
	Establecer la distinción entre mayúsculas y minúsculas
:set ic	En la búsqueda se ignora la distinción entre mayúsculas y minúsculas
:set noic	En la búsqueda se distingue entre mayúsculas y minúsculas
Encontrar una línea	

Tabla 6-1 Comandos básicos de vi (Continuación)

Comando	Significado
G	Ir a la última línea del archivo
1G	Ir a la primera línea del archivo
21G	Ir a la línea 21
Buscar y reemplazar	
/string	Búsqueda de <i>cadena de caracteres</i>
?string	Búsqueda hacia atrás de <i>cadena de caracteres</i>
n	Encontrar la siguiente aparición de <i>string</i> en la dirección de búsqueda
N	Encontrar la aparición previa de <i>la cadena de caracteres</i> en la dirección de búsqueda
:g/search/s//replace/g	Buscar y reemplazar
Limpiar la pantalla	
Ctrl-L	Limpiar (actualizar) la pantalla
Insertar un archivo en otro archivo	
:r nombre_de_archivo	Insertar (leer) el archivo a continuación del cursor
:34 r nombre_de_archivo	Insertar el archivo después de la línea 34
Guardar y salir	
:w	Guardar los cambios (escribir el contenido de la memoria intermedia)
:w nombre_de_archivo	Escribir el contenido de la memoria intermedia a un archivo con nombre
:wq	Guardar los cambios y salir de vi
ZZ	Guardar los cambios y salir de vi
:q!	Salir sin guardar los cambios

Uso del correo electrónico



SunOS dispone de un programa llamado `mailx` encargado de enviar y recibir correo electrónico (*email*). `mailx` ofrece facilidades para leer, escribir, enviar, recibir, guardar y eliminar mensajes. El programa `mailx` no se basa en el sistema de ventanas, por lo que puede ser ejecutado en cualquier terminal. Aunque es posible que prefiera utilizar un correo electrónico basado en el sistema de ventanas, el programa `mailx` le puede resultar muy útil cuando tenga que escribir una nota rápidamente. En esta sección también se explica como establecer sus propios alias de correo electrónico.

Nota - Si se encuentra en el entorno OpenWindows y aparece el icono de la Herramienta de Correo en su pantalla, abandone la Herramienta de Correo antes de realizar los ejemplos de este capítulo. De no ser así, Vd. tendría dos procesos de correo electrónico activos, y se podrían producir avisos y mensajes de error. Puede enviar mensajes de correo electrónico con seguridad en la ventana de la Herramienta de comandos o en la Herramienta del shell, pero si lee su correo electrónico y guarda o salva mensajes, puede afectar a su “bandeja de entrada”, confundiendo a la Herramienta de Correo.

Datos básicos de `mailx`

En esta sección aprenderá lo necesario para ir trabajando con el programa `mailx`. En las secciones posteriores se explicarán características y funciones que le ayudarán a dominar el programa.

La única dirección del programa `mailx` la forman el nombre de entrada del usuario al que se envía un mensaje y el nombre del sistema. Si el usuario al que se dirige se encuentra en el mismo sistema que el remitente, sólo se solicita el nombre de entrada. Cada usuario tiene un *buzón* en el que recibe correo electrónico. Dicho buzón normalmente está situado en el directorio `/var/mail/nombre_del_usuario`, donde *nombre_del_usuario* es su nombre de entrada.

El programa `mailx` le avisa cuando recibe correo electrónico y coloca su correspondencia en el buzón. Una vez que haya leído su correo electrónico, `mailx` automáticamente sitúa las cartas en un archivo de almacenamiento llamado `mbox`, que también está situado en su directorio de usuario.

Cómo empezar mailx

Empiece `mailx` escribiendo el siguiente comando tras el indicador y presionando después la tecla Retorno:

```
$ mailx
```

Si no tiene ningún correo electrónico en espera, su terminal mostrará el mensaje:

```
Ningún mensaje para nombre_del_usuario
$
```

donde *nombre_del_usuario* es su nombre de entrada.

Cómo enviarse Vd. mismo una carta de prueba

Para comprobar a primera vista como trabaja `mailx`, puede empezar por enviarse una carta de prueba a Vd. mismo. Escriba de nuevo el comando `mailx`, tras el indicador, pero esta vez incluya su dirección (su nombre de entrada y el de su sistema). Por ejemplo, si su nombre de entrada era `rosa` y su nombre de

sistema `texas`, su dirección sería `rosa@texas`, (el símbolo `@` significa “en”). Es posible que sólo pueda utilizar su nombre de entrada en una red local (si tiene dudas consulte con su administrador de sistemas).

```
$ mailx rosa@texas
```

El programa responderá con una línea `Subject` :

```
$ mailx rosa@texas
Subject:
```

Si lo desea, escriba en una palabra o dos el tema de la carta que se va a enviar Vd. mismo y presione Retorno. Ahora escriba el texto de la carta, utilice líneas breves y presione Retorno al final de ellas (observe que sólo puede hacer correcciones a medida que retrocede y vuelve a escribir las líneas *antes* de presionar Retorno).

Su carta de prueba se parecerá a lo siguiente (los espacios entre líneas se consiguen presionando dos veces Retorno):

```
$ mailx rosa@texas
Subject: a nuestros lectores

Hola:

Esperamos que la lectura de este manual
sea de vuestro interés. Para nosotros
ha sido un placer el prepararlo.

Saludos,

Montse
```

Para enviar su carta de prueba, presione Retorno para completar la última línea y después `Ctrl-D`. Después de que ha enviado su carta, el sistema muestra de nuevo el indicador de comando.

Cómo leer su carta de prueba

Para leer su carta de prueba, escriba de nuevo el comando `mailx`. Su pantalla aparecerá probablemente de esta manera:

```
$ mailx
Mail version 4.0 Thu Jan 16 12:59:09 PST 1992  Type ? for help.
"/var/mail/rosa": 2 messages 1 new
  U  2 hal@uncertain   Vie Feb 14 12:01   14/318 proyecto de es
>N  1 rosa@texas      Lun Feb 17 08:12   21/453 a una persona
&
```

La primera línea identifica la versión de `mail` que se está ejecutando; la segunda indica su buzón, generalmente situado en `/var/mail/nombre_del_usuario`, en donde se deposita el correo electrónico recibido. La tercera línea es el encabezamiento de la carta que se envía a Vd. mismo. La “N” al comienzo de la línea indica que es una carta “nueva”. Una “U” (unread) significa que la carta era nueva, pero que no fue leída antes de abandonar el programa `mailx`, (la información de esta pantalla se explicará con más detalle en la sección “Cómo leer cartas” de este capítulo).

A cada carta que es recibida se le asigna un número: La carta que Montse se escribió a sí misma se muestra como carta nº 1.

Para leer una carta, escriba su número tras el indicador de `mailx`, el símbolo (&), de la siguiente forma:

```
$ mailx
Mail version 4.0 Thu Jan 16 12:59:09 PST 1992  Type ? for help.
"/var/mail/rosa": 1 message 1 new
>N 1 rosa@texas  Vie Jul 14 12:01 21/453 a una persona
& 1

To: pepe@pipas
From: pepe@pipas
Subject: a Pepote

Hola Pepote:

Hace tiempo que no nos vemos,
llámame y quedamos para tomar unas
copas.

Saludos,

Pepe

&
```

Cómo salir de mailx

Cuando haya terminado de usar `mailx`, puede salir del programa utilizando uno de estos comandos: `q` (quit) o `x` (exit).

Si escribe `q` tras el indicador `mailx` y después presiona Retorno,

```
& q
```

verá un mensaje parecido al siguiente:

```
1 mensaje guardado en directorio_personal/mbox.
```

donde *directorio_personal* es el nombre de la ruta de acceso a su directorio de usuario.

Cuando use `q` para salir `mailx` después de leer mensajes, `mailx` traslada las cartas de su buzón y las guarda en el archivo `mbox` de su directorio de usuario. `mailx` también guarda cualquier cambio o eliminación que haya efectuado.

Si escribe `x` tras el indicador de `mailx` y luego presiona Retorno,

```
& x
```

el programa `mailx` *no* guarda ningún cambio o eliminación, ni traslada ninguna carta que haya leído al archivo `mbox`.

Cómo leer cartas

Si tiene correo electrónico, `mailx` le avisa cada vez que entra en el sistema con el mensaje

Tiene correo

o

Tiene correo nuevo

Para leer sus cartas, escriba `mailx` tras un indicador de comando y presione Retorno. Si no tiene ningún correo electrónico en espera, verá el mensaje:

No hay correo para *nombre_del_usuario*

Si ocurre lo contrario, verá una lista parecida a la siguiente:

```
$ mailx
Mail version 4.0 Thu Jan 16 12:59:09 PST 1992  Type ? for help.
"/var/mail/rosa": 4 messages 1 new 2 unread
  1 ripoll@cata Vie Feb 14 12:01 21/453 a una persona
  U 2 barna@mar Vie Feb 14 18:31 19/353 pensando en
  U 3 sol@miró Sab Feb 15 10:22 24/557 una rosa es
>N 4 costa@brava Dom Feb 16 23:59 14/280 la playa de

&
```

El programa `mailx` muestra información relativa a él mismo (número de versión y fecha) e instrucciones para obtener ayuda (Escriba `? si necesita ayuda`).

En la siguiente línea, `mailx` especifica la posición de su buzón, el número de cartas recibidas, y su estado.

A continuación, `mailx` muestra una lista numerada de cartas de su buzón. De izquierda a derecha, las columnas de cada línea especifican:

- *Estado*: indica si una carta es nueva (N), no leída (U), o leída (sin símbolo). Un signo “>” al comienzo de la línea indica la carta actual. Las cartas eliminadas aparecen marcadas con un asterisco (*).
- *Número*: indica el orden en el que la carta fue recibida.
- *Remitente*: indica el nombre del usuario (y normalmente el nombre del sistema) de donde proviene la carta.
- *Tiempo*: indica la fecha y la hora en que se envió la carta.
- *Tamaño*: indica el número de líneas/número de caracteres en la carta.
- *Subject*: indica el asunto designado por el remitente.

Si tiene un gran número de cartas en su buzón, la lista que aparezca podría no mostrar todo su correo electrónico. Si ocurre esto, escriba:

- `z` – Para mostrar la siguiente pantalla de encabezamientos de correo electrónico.
- `h-` – Para mostrar la anterior pantalla de encabezamientos de correo electrónico.
- `h` – Para volver a mostrar cada vez la lista de encabezamientos de correo electrónico.

Para ver la carta actual en la lista del buzón (que está marcada con >), presione Retorno. Si presiona Retorno de nuevo, aparecerá la siguiente carta. Para leer cualquier carta de la lista, escriba su número y presione Retorno.

Cómo eliminar (y rehacer) cartas

Al terminar de leer una carta, puede decidir si desea eliminarla en vez de tenerla guardada en su archivo `mbox` (es la opción predeterminada al salir del programa `mailx`).

Para eliminar la última carta que leyó, escriba simplemente `d` tras el indicador de `mailx`. Para eliminar una carta determinada de su buzón, utilice el siguiente formato de comando:

`d número`

Por ejemplo, para eliminar la segunda carta, introduzca este comando desde `mailx`:

```
& d 2
```

También puede eliminar varias cartas a la vez. Para eliminar las cartas 1 y 3, introduzca el comando:

```
& d 1 3
```

Para eliminar cartas consecutivas, por ejemplo de la 1 a la 3, utilice el comando:

```
& d 1-3
```

Antes de salir de `mail`, puede *rehacer* las cartas que eliminó de su buzón. Utilice el siguiente formato de comando:

`u número`

seguido de Retorno. Por ejemplo, para rehacer la segunda carta introduzca el comando:

```
& u 2
```

Si quiere deshacer su última eliminación, escriba simplemente `u` tras el indicador de `mailx` inmediatamente después de la operación de eliminación. Por ejemplo, si su último comando de eliminación fue `d 2-5`, al escribir `u` recuperará los mensajes 2, 3, 4, y 5.

Observe que todas las eliminaciones se hacen permanentes al salir de `mailx` mediante el comando `q`; es decir, las cartas eliminadas ya no pueden recuperarse. De cualquier modo, puede salir de `mailx` con el comando `x`, dejando intacto su buzón -como se mencionó anteriormente, si sale con `x` las cartas leídas se marcarán con una U, las cartas eliminadas se recuperarán, etc.

Cómo imprimir cartas

Puede imprimir una copia en papel de la carta enviándola por el canal de comunicación a la impresora. Para conseguir esto, utilice el formato de comando:

```
| número lp
```

tras el indicador de `mailx` (el símbolo `|` se denomina *canal de comunicación*.)

Por ejemplo, para imprimir una copia de la carta 2, escriba:

```
& |2 lp
```

y presione Retorno. Si no se especifica un número de carta, `mailx` envía la carta actual por el canal de comunicación a la impresora. Si desea más información acerca del uso del canal de comunicación, consulte “Cómo redireccionar y enviar por el canal de comunicación la salida del comando” en el Capítulo 2, “Comandos básicos de SunOS”.

Cómo enviar cartas

Para enviar correo electrónico con el programa `mailx`, necesita conocer el nombre o nombres de entrada del receptor o receptores de su carta. Si un receptor está en un sistema diferente, necesitará conocer el nombre del sistema del usuario. Para obtener esta información, puede utilizar los comandos `who`, `finger`, o `rusers`.

Si escribe el comando `who` se listan todos los usuarios que están conectados al servidor en el que está Vd. La lista contiene los nombres de entrada de los usuarios, sus tipos de terminal, y la fecha y hora en que entraron en el sistema. Por ejemplo:

```
$ who
  montse      tty15      Feb 20 10:22
  charo       tty04      Feb 20 10:37
  pepote      tty07      Feb 20 11:49
  carolina    tty06      Feb 20 12:02
```

Si escribe el comando `finger` se muestra el mismo tipo de información que con `who` pero más detalladamente. La información que aparece depende de cómo haya configurado este comando su administrador del sistemas. Como ejemplo, podría ser algo parecido a lo siguiente:

```
$ finger
  Login      Name           TTY      Idle   When
  elmer      Elmer Brown   tty15    43    Thu 10:22
  susan      Susan Lake    tty04    12    Thu 10:37
  stormy     Stormy Ball   tty07    12    Thu 11:49
  hankw      Hank Wilson   tty06    22    Thu 12:02
```

El comando `rusers` ofrece información de los usuarios que están conectados a su red local. Consulte el Capítulo 9, “Uso de la red,” si desea más información acerca de la utilización del comando `rusers`.

Cuando haya averiguado la información de usuario que necesitaba, complete los pasos siguientes para enviar una carta.

1. Introduzca el comando `mailx` seguido de la dirección de un usuario:

```
$ mailx usuario@sistema
```

donde *usuario* es el nombre de entrada del receptor y *sistema* es el nombre del sistema del receptor.

- Si ya ha empezado `mailx`, puede escribir simplemente `m` tras el indicador de `mailx`, seguido del nombre de entrada del receptor y del nombre de su sistema:

```
& m usuario@sistema
```

- Para enviar la misma carta a varios receptores, separe cada dirección con un espacio en blanco o una coma, por ejemplo:

```
$ mailx hank@fretful sally@dakota tex@twister
```

o

```
$ mailx hank@fretful,sally@dakota,tex@twister
```

2. Si presiona Retorno, el programa `mailx` le solicita un asunto. Escriba un asunto para su carta y presione Retorno de nuevo.

3. Escriba el cuerpo de su carta. Cuando quiera una nueva línea, presione Retorno.

Una frase que aparezca partida en su pantalla no se considerará una nueva línea hasta que presione Retorno.

Nota – Cada línea del texto de su carta puede tener hasta 256 caracteres de longitud. Si sobrepasa esta limitación, su pantalla se congelará. Si ocurre esto, presione `Ctrl-C` para cancelar su carta.

4. Cuando haya completado su carta, presione Retorno para mover el cursor a una línea nueva. A continuación presione `Ctrl-D` para enviar su carta.

Cartas no entregables

Si especifica una dirección incorrecta de usuario al enviar una carta, el sistema responderá con el mensaje

```
usuario@sistema...Usuario desconocido
```

y la carta regresará a su buzón. La próxima vez que escriba el comando `mailx`, el encabezamiento le avisará de que tiene correo electrónico devuelto, algo parecido al ejemplo siguiente:

```
N 1 Mailer-Daemon Vie Jan 3 11:13 8/49 Returned mail: usuario
desconocido
```

Cuando una carta no puede enviarse, el archivo se copiará en un archivo de su directorio de usuario llamado `dead.letter`.

Cómo cancelar una carta no enviada

Puede cancelar una carta en cualquier momento *antes* de que sea enviada presionando dos veces `Ctrl-C`.

Cómo agregar copias al carbón y borradores al carbón

Antes de enviar una carta, puede especificar que sean enviadas “copias al carbón” a otras direcciones aparte de las direcciones principales. También puede enviar “borradores al carbón” (los receptores de su carta podrán leer las direcciones de la copia, pero no las direcciones de los borradores al carbón)

Muchas personas se envían a sí mismas una copia al carbón o un borrador al carbón para tener un registro de las cartas enviadas.

Hay tres métodos para enviar copias al carbón de una carta:

- Utilice un editor de textos para editar su archivo `.mailrc` (en su directorio de usuario) e inserte la línea siguiente:

```
set askcc
```

El programa `mailx` mostrará ahora el indicador de copia (`Cc:`) después del indicador de `subject`. Introduzca la dirección de los usuarios que desea que reciban una copia al carbón. Separe las distintas direcciones con espacios en blanco.

- Cuando haya terminado de escribir el cuerpo de su carta, antes de presionar `Ctrl-D`, presione Retorno para moverse a una línea nueva y utilice el formato de comando:

`~c address(es)`

Para usar este método de enviar copias al carbón a varios receptores, separe las direcciones con espacios en blanco. Por ejemplo:

```
~c hank@fretful george@lonesome stormy@snoozer
```

- Se crea también una línea `Cc:` con el comando `~h`, que mostrará el encabezamiento completo de la carta. `~h` le solicitará datos con las líneas `To:`, `Subject:`, `Cc:`, y `Bcc:` (borrador al carbón), una línea cada vez. Las líneas vacías pueden rellenarse; las rellenas pueden reescribirse. Al igual que otros comandos de tilde, utilice siempre el comando `~h` en una nueva línea.

Nota – `~c`, `~h`, y otros comandos de tilde se describen en “Comandos de tilde” en este capítulo.

Cómo insertar una copia de una carta o un archivo

Puede insertar una copia de cualquier carta de su buzón en la carta que está escribiendo. Asimismo, puede insertar una copia de cualquier archivo de texto.

Cómo insertar una carta

El formato de comando para insertar una carta es

`~m número`

donde *número* es el número de la carta a insertar. Por ejemplo, para enviar una carta a otro usuario que incluya una copia de la carta número 3 de la lista de su buzón, puede hacer lo siguiente:

1. **En una línea nueva escriba el comando `~m 3` y presione Retorno.**
2. `mailx` **mostrará el mensaje**, `Interpolar: 3 (continuar)`
3. **Vd. no verá el texto del mensaje 3, pero el receptor sí . Puede continuar con su carta después de (continuar), o puede enviarla tal cual.**
4. **Para ver la carta completa, incluida la inserción, introduzca el comando `~p`.**

Cómo insertar un archivo

También puede insertar una copia de un archivo de texto en su carta. Utilice el formato de comando:

```
~r nombre_de_archivo
```

según está escribiendo una carta. Por ejemplo, para insertar el archivo `outline` en la carta actual, escriba:

```
~r outline
```

Cómo responder a una carta

Responda al correo electrónico con el comando

```
r número
```

tras el indicador de `mailx` (si omite el número de carta, `mailx` responderá a la carta actual). Por ejemplo, para responder al remitente de la carta 2, utilice el comando:

```
& r 2
```

`mailx` dirigirá automáticamente su carta y rellenará una línea `Re: Subject:` que repetirá la línea `Subject: original`. Envíe su respuesta como cualquier otra carta.

`R` es una variante del comando `reply` que envía su respuesta a todos los receptores de la carta original así como a su remitente. Use este comando únicamente cuando sea absolutamente necesario, para evitar el generar “correo inútil.”

Nota – Puede insertar una carta en su respuesta tal como se vio en la sección anterior. Para insertar una copia de la carta a la cual está respondiendo, use solamente el comando `~m` sin un número de carta.

Cómo guardar y recuperar cartas

Además de enviar y recibir cartas, también puede guardarlas y recuperarlas para un uso posterior. Con `mailx` puede guardar cartas agregándolas a archivos de texto normales, también puede agregar cartas a archivos especiales denominados carpetas. Ambos métodos se explican más adelante.

`mailx` distingue entre *guardar* cartas y *copiarlas*; guardar quita una carta del buzón y la añade a un archivo o carpeta, copiar deja la carta en el buzón y añade una copia a un archivo o carpeta.

Cómo guardar y copiar cartas en archivos

Para guardar una carta en un archivo, el formato de comando tras el indicador del `mailx` es:

```
s número nombre_de_archivo
```

donde *número* es el número de la carta a guardar y *nombre_de_archivo* es el archivo donde quiere guardar dicha carta. Por ejemplo, para guardar la carta 3 en un archivo denominado `~/notas/finanzas`, tendría que escribir:

```
& s 3 ~/notas/finanzas
```

(Recuerde que en un nombre de ruta de acceso, el signo `~` representa su directorio de usuario).

También puede guardar varias cartas a la vez en un mismo archivo. Por ejemplo, para guardar las cartas 3, 5, 6, 7, y 8 en `~/notas/finanzas`, tendría que escribir:

```
& s 3 5-8 ~/notas/finanzas
```

Si no existe el archivo que especificó, `mailx` lo creará. Si el archivo si existe, `mailx` añadirá la carta que esté guardando al final del archivo.

Al guardar un archivo lo elimina de su buzón; `mailx` muestra un asterisco (*) al lado del encabezamiento de cualquier carta que haya sido guardada.

Para dejar la carta en su buzón al agregarla a un archivo, utilice el comando `copy`, como se muestra a continuación:

```
& c 3 ~/notas/finanzas
```

Cómo guardar y copiar cartas en carpetas

Puede evitar el escribir nombres de rutas de acceso completos en archivos si guarda o copia las cartas en carpetas de correo electrónico. Las carpetas son archivos especiales que se almacenan en un directorio de carpeta.

Las ventajas de guardar o copiar cartas en carpetas es que sus cartas se guardarán juntas automáticamente en el mismo directorio, donde estarán fácilmente accesibles sin tener que escribir nombres de ruta de acceso largos.

Cómo establecer el directorio de carpeta

Para usar carpetas, primero debe establecer un directorio de carpeta. Es un proceso de dos pasos:

1. Primero, cree el directorio con el comando `mkdir`.

Por ejemplo, si quiere que su directorio de carpeta se denomine `Cartas`, primero crearía el directorio:

```
$ mkdir Cartas
```

2. Segundo, use un editor de texto para editar el archivo `.mailrc` de su directorio de usuario (que contiene las opciones de `mailx`) para establecer la ruta de acceso del directorio de carpeta.

Aquí necesitará editar la variable `establecer carpeta` para incluir el nombre de ruta de acceso completo de su directorio `carpeta` recientemente creado. Por ejemplo:

```
establecer carpeta=/home/austin/rose/Cartas
```

o, utilizando el método abreviado `~` del shell `C` que especifica su directorio de usuario.

```
establecer carpeta=~ /Cartas
```

Ahora su directorio de carpeta está preparado para recibir las cartas guardadas en carpetas (los cambios efectuados en el archivo `.mailrc` tendrán efecto la próxima vez que empiece `mailx`).

Cómo designar carpetas

Puede utilizar los mismos comandos para guardar o copiar cartas en carpetas y en archivos, excepto que el nombre de la carpeta va precedido por un signo (+) en vez de por un nombre de ruta de acceso. El signo + le indica al `mailx` que la carpeta se guardará en el directorio de carpeta (`cartas`).

Por ejemplo, para guardar la carta 3 en un carpeta denominada `proyectos`, escriba:

```
& s 3 +proyectos
```

`mailx` interpretará el significado del comando como “guardar la carta 3 en `~/Cartas/proyectos`” (si la carpeta no existía previamente, `mailx` la creará.)

Copie la carta en un carpeta escribiendo:

```
& c 3 +proyectos
```

Cómo enviar directamente una carta a un archivo o carpeta

Puede enviar copias de sus cartas directamente a uno de sus archivos o carpetas. Para enviar una copia a un carpeta, escriba simplemente el nombre de la carpeta en el campo `Cc:` o bien en el campo `Bcc:`. Enviar una copia a un archivo es parecido, pero debe incluir el nombre de ruta de acceso completo..

Cómo leer cartas guardadas en archivos y carpetas

Para leer cartas guardadas en un archivo, utilice el formato de comando:

```
mailx -f nombre_de_archivo
```

En el ejemplo anterior, podría leer el archivo `~/memos/finanzas` escribiendo:

```
$ mailx -f ~/memos/finanzas
```

Puede leer cartas guardadas en un carpeta con un comando similar -únicamente utilice el signo `+` en vez del nombre de ruta de acceso. Por ejemplo, para leer las cartas en el carpeta `proyectos`, tendría que escribir:

```
$ mailx -f +proyectos
```

Este comando empieza `mailx` en el archivo o carpeta designado. Únicamente se mostrarán los encabezamientos de las cartas en el archivo o carpeta. Seleccione una carta para leerla escribiendo un número tras el indicador del `mailx` y presione Retorno.

También puede trabajar con carpetas de correo electrónico con el programa `mailx`. Para ver una lista de sus carpetas, escriba lo siguiente tras el indicador de `mailx`:

```
& folders
```

Para conmutar desde su buzón a un carpeta, utilice el formato de comando:

```
& folder +nombre_de_carpeta
```

Para volver a su buzón, escriba lo siguiente tras el indicador de correo electrónico:

```
& %
```


Para volver a la carpeta anterior, escriba:etae

```
& #
```

Uso de `vi` en `mailx`

Puede usar el editor de textos `vi` para redactar cartas mientras ejecute `mailx`. Esto le permitirá corregir errores y agregar o eliminar texto antes de enviar sus cartas. Si no está familiarizado con la utilización de `vi`, consulte el Capítulo 6, “Uso del editor `vi`,” para conocerlo.

En el programa `mailx`, puede utilizar los comandos estándar de `vi` para insertar, eliminar, y cambiar texto.

Para escribir una carta con `vi`:

1. **Introduzca el comando `mailx` con una dirección tras el indicador de `mailx` (&) o en el indicador de comando (\$).**
2. **Introduzca el asunto en la línea `Subject:`. Presione Retorno.**
3. **Empiece `vi` mediante el comando `~v` en una nueva línea. Aparecerá la pantalla de `vi`, presentando un archivo vacío en su directorio `/tmp`.**
4. **Use los comandos `vi` para introducir y editar el cuerpo de su carta.**
5. **Cuando haya terminado, salga de `vi` con el comando `:wq` o `ZZ`.**

Después de salir de `vi`, `mailx` mostrará el mensaje `(continuar): y Vd. podrá ampliar la carta (pero ya fuera de vi) o enviarla presionando Ctrl-D.`

Alias de correo electrónico

Un *alias de correo electrónico* es una selección de nombres de usuario agrupados bajo un único nombre.

Utilice alias de correo electrónico si quiere enviar cartas al mismo grupo de personas una y otra vez. Por ejemplo, si desea enviar correo electrónico de vez en cuando a `hank@fretful`, `george@lonesome`, y `sally@dakota`, podría crear un alias de correo electrónico denominado `amigos`. Entonces, cada vez que envíe correo electrónico a `amigos`, las tres personas lo recibirán.

Existen dos lugares donde puede establecer sus alias de correo electrónico:

- Su archivo `.mailrc`
- El archivo `/etc/aliases`

Los alias de correo electrónico establecidos en `.mailrc` se comportan de manera diferente a los establecidos en `/etc/aliases`. Estas diferencias se resumen en la Tabla 7-1 al final de esta sección.

Cómo establecer alias de correo electrónico en `.mailrc`

Observe lo siguiente acerca del establecimiento de alias de correo electrónico en `.mailrc`:

- Los alias de correo electrónico en `.mailrc` son *privados*; es decir, sólo Vd. puede utilizarlos. Por ejemplo, si establece un alias de correo electrónico denominado `amigos` en `.mailrc` y otro usuario intenta enviar correo electrónico a `amigos`, recibiría el mensaje de error `usuario desconocido`.
- Cuando se envía el correo electrónico, los alias de `.mailrc` se expanden automáticamente para mostrar cada uno de los alias de correo electrónico. Por ejemplo, si envía correo electrónico a `amigos`, su correo electrónico llegará como si hubiera escrito los nombres de cada uno de sus receptores. Los receptores no notarán que Vd. utilizó un alias de correo electrónico para enviar su correo electrónico.

`.mailrc` está situado en su directorio de usuario. Este archivo contiene varias configuraciones que controlan el comportamiento de `mailx` y la Herramienta de Correo.

Para agregar un alias de correo electrónico en `.mailrc`, escriba:

```
$ vi ~/.mailrc
```

Nota – Puede usar cualquier editor de texto para editar el archivo `.mailrc`. El ejemplo anterior muestra el método para utilizar el editor `vi` para editar el archivo. Si no está familiarizado con `vi`, consulte el Capítulo 6, “Uso del editor `vi`”, para conocerlo.

Cada alias de correo electrónico está contenido en una línea del archivo; es decir, puede aparecer “dividido” en varias líneas, pero no puede contener retornos de carro. Cada alias de correo electrónico debe contener lo siguiente, separado por espacios en blanco:

- La palabra “alias”
- El nombre del alias de correo electrónico (debe ser una palabra)
- Los receptores (nombres de entrada y de sistema) en el alias de correo electrónico, separados por espacios en blanco

El ejemplo siguiente muestra dos alias de correo electrónico. El primero (`amigos`) contiene tres personas. El segundo (`softball`) contiene ocho. Observe en `softball` cómo los nombres aparecen divididos en la pantalla. Esto es correcto, ya que no se utilizan retornos de carro.

```
alias amigos hank@fretful george@lonesome sally@dakota
alias softball earl@woofer tex@twister elmer@farmhouse
jane@freeway hank@fretful jj@walker sally@dakota steve@hardway
```

Para enviar correo electrónico a las personas de un alias en `.mailrc`, simplemente dirígalo al nombre del alias de correo electrónico. *No* incluya su nombre de sistema. Por ejemplo, si envía lo siguiente:

```
$ mail amigos
Subject: Vamos a comer

Hey compis. ¿Qué os parece si nos vamos a comer a Atienza cordero
asado?
```

los receptores verán lo siguiente (observe la línea expandida To:):

```
To: hank@fretful george@lonesome sally@dakota
Subject: Vamos a comer

Hey compis. ¿Qué os parece si nos vamos a comer a Atienza cordero
asado?
```

Cómo establecer alias de correo electrónico en /etc/aliases

Observe lo siguiente acerca del establecimiento de correo electrónico en /etc/aliases:

Los alias de correo electrónico en /etc/aliases son *públicos*. Esto significa que si establece un alias de correo electrónico denominado `softball`, cualquiera puede enviar correo electrónico a `softball@su_nombre_de_sistema` y hacer uso del alias de correo electrónico.

- Cuando se envía el correo electrónico, los alias de correo electrónico de /etc/aliases *no* se amplían. Por ejemplo, si envía correo electrónico a `softball@nombre_de_sistema`, esto es lo que se leerá cuando se reciba el correo electrónico. Los receptores conocerán en que alias de correo electrónico se encuentran, pero no necesariamente quien más comparte el alias.

El formato de los alias de correo electrónico creados en /etc/aliases es algo diferente al de los creados en .mailrc. Cada alias de /etc/aliases debe utilizar el formato siguiente:

- El nombre del alias de correo electrónico, seguido de dos puntos (:)
- Los receptores (nombres de entrada y de sistema), separados por comas. Observe que el alias de correo electrónico *no* tiene por qué estar en una única línea.

Para modificar su archivo /etc/aliases, primero debe convertirse en raíz. Si la raíz está protegida por una contraseña, necesitará conocerla.

Escriba lo siguiente para convertirse en el usuario raíz del sistema:

```
$ su
Contraseña:
#
```

Observe que el indicador de comando cambia cuando Vd. se convierte en raíz.

El ejemplo siguiente muestra como el alias `softball@texas` se agrega al archivo predeterminado `/etc/aliases`.

```
# vi /etc/aliases
##
#Aliases can have any mix of upper and lower case on the left-
#hand side,
#but the right-hand side should be proper case (usually lower)
#
#   >>>>>>>>The program "newaliases" will need to be run after
#   >> NOTE >>this file is updated for any changes to
#   >>>>>>>>show through to sendmail.
#
#@(##)aliases 1.10 89/01/20 SMI
##
# Following alias is required by the mail protocol, RFC 822
# Set it to the address of a HUMAN who deals with this system's
mail problems.
Postmaster: root

# Alias for mailer daemon; returned messages from our MAILER-
DAEMON
# should be routed to our local Postmaster.
MAILER-DAEMON: postmaster

# Aliases to handle mail to programs or files, eg news or vacation
# decode: "|/usr/bin/uudecode"
nobody: /dev/null

# Sample aliases:
# Alias for distribution list, members specified here:
#staff:wnj,mosher,sam,ecc,mckusick,sklower,olson,rwh@ernie

# Alias for distribution list, members specified elsewhere:
#keyboards: :include:/usr/jfarrell/keyboards.list

# Alias for a person, so they can receive mail by several names:
#epa:eric

#####
# Local aliases below #
```

```
#####
softball@texas: earl@woofer tex@twister elmer@farmhouse
jane@freeway hank@fretful jj@walker sally@dakota steve@hardway
:wq          (to quit vi and save the /etc/aliases file)
# exit      (to exit root)
$
```

Puede usar cualquier editor de texto para editar el archivo `/etc/aliases`. El ejemplo anterior muestra el método de utilización del editor `vi` para editar el archivo. Si todavía no está familiarizado con `vi`, consulte el Capítulo 6, “Uso del editor `vi`”, para conocerlo.

Observe que el signo (`#`) que ve en el archivo `/etc/aliases` se utiliza para convertir el texto en *comentario* y mostrar alias de muestra. El signo (`#`) evita que el sistema procese esta información como alias reales.

No sitúe los signos (`#`) delante de los alias que agregó al archivo, a no ser que desee intencionadamente desactivar un alias.

Para enviar correo electrónico a las personas del alias `/etc/aliases`, dirija dicho correo electrónico usando el nombre del alias y el nombre del sistema donde Vd. se encuentra. Por ejemplo, si envía lo siguiente:

```
$ mail softball@texas
Subject: Fiestas en mi pueblo

Nos vamos a ver en el arco Arrebatacapas para acercarnos a la
fiesta. Nos lo vamos a pasar de maravilla comiendo pipas y quicos.
```

los receptores verán lo siguiente:

```
To: softball@texas
Subject: Fiestas en mi pueblo

Nos vamos a ver en el arco Arrebatacapas para acercarnos a la
fiesta. Nos lo vamos a pasar de maravilla comiendo pipas y quicos.
```

Observe que la línea `To:` *no* se amplía.

Siempre que envíe correo electrónico utilizando un alias de este tipo, asegúrese de incluir el nombre del sistema donde esté situado dicho alias. Si establece un alias de correo electrónico denominado `riders` en el sistema `freeway`, debería enviar su correo electrónico a `riders@freeway`.

La Tabla 7-1 ofrece un resumen comparativo entre los alias de correo electrónico creados en `.mailrc` y los creados en `/etc/aliases`.

Tabla 7-1 Comparación de los alias de correo electrónico en `.mailrc` y en `/etc/aliases`

	.mailrc	/etc/aliases
¿Debe ser raíz para modificar?	no	sí
Enviar mensajes a:	alias	<i>alias@nombre_ de_sistema</i>
¿La lista de receptores la ven los receptores?	sí	no
¿Nombres separados por comas?	no	sí
¿Los nombres deben estar todos en una línea?	sí	no
¿Los demás pueden utilizar el alias de correo electrónico?	no	sí

Si desea una información más detallada acerca de los alias de correo electrónico, escriba `man aliases` o `man addresses` tras el indicador del sistema.

Comandos de tilde

En el transcurso de la redacción de una carta, puede utilizar comandos de tilde para llevar a cabo distintas funciones. Los comandos de tilde se componen normalmente del carácter (`~`) seguido de un único carácter. La tabla siguiente describe algunos de los caracteres de tilde más útiles. Algunos de ellos ya se han explicado en este capítulo.

Nota – Si quiere incluir un carácter de tilde en una carta, escriba dos tildes seguidas. Aparecerá únicamente una tilde.

Tabla 7-2 Comandos de tilde (`mailx`)

Comando	Función
<code>~!command</code>	Escapa o se introduce en un comando de shell
<code>~.</code>	Simula el presionar Ctrl-D para indicar fin de archivo
<code>~?</code>	Lista un resumen de los comandos de tilde
<code>~b username</code>	Agrega nombre(s) de usuario a la lista de borradores al carbón (Bcc:)
<code>~c username</code>	Agrega nombre(s) de usuario a la lista de copias al carbón (Cc:)
<code>~d</code>	Lee el contenido del archivo <code>dead.letter</code> sobre la carta actual.
<code>~f number</code>	Transmite la carta especificada. Válido únicamente cuando se envía un mensaje mientras se lee correo electrónico.
<code>~h</code>	Solicita las líneas de encabezamiento: Subject, To, Cc, y Bcc.
<code>~m number</code>	Inserta texto desde la carta especificada a la carta actual. Válido sólo cuando se envía un mensaje mientras se lee correo electrónico.
<code>~p</code>	Imprime el mensaje que está siendo introducido en la pantalla.
<code>~q</code>	Simula el presionar dos veces Ctrl-C. Si el cuerpo del mensaje actual no está vacío, el contenido se guarda en <code>dead.letter</code> .
<code>~r nombre_de_archivo</code>	Lee el texto desde el archivo especificado.
<code>~s string</code>	Cambia la línea de subject a <i>cadena de caracteres</i> .
<code>~t name</code>	Agrega a la lista To. el nombre o nombres especificados
<code>~w nombre_de_archivo</code>	Escribe la carta actual sin encabezamiento en el archivo especificado.
<code>~x</code>	Salida de <code>mailx</code> . Es parecido a <code>~q</code> excepto que el mensaje no se guarda en el archivo <code>dead.letter</code> .

Cómo obtener ayuda: otros comandos de `mailx`

`mailx` dispone de dos comandos de ayuda que muestran listas de comandos y funciones. Si está en modo comando, puede escribir `?` tras el indicador de `mailx` (`&`) para ver una lista de comandos utilizados en ese modo. Asimismo,

en modo entrada (por ejemplo, cuando escriba una carta), puede introducir el comando equivalente, `~?` para ver una lista de los comandos de tilde (también denominados “escapes de tilde”).

las páginas de manual contienen información detallada acerca de `mailx` en un formato más técnico. Para ver esta entrada, introduzca el comando:

```
$ man mailx
```

o consulte *man Pages(1): User Commands*.

Uso de las impresoras



El servicio de impresión LP (abreviatura de *line printer subsystem*) es el subsistema de SunOS que proporciona las herramientas de impresión. Éste dispone de una amplia variedad de funciones, muchas de las cuales se salen del ámbito de este manual. En este capítulo sólo se explicarán los procedimientos necesarios para que pueda realizar las siguientes tareas básicas de impresión utilizando el servicio de impresión LP:

- Encargar una solicitud de impresión (enviar un archivo a la impresora)
- Determinar el estado de una impresora
- Cancelar una solicitud de impresión

Si desea una descripción completa del servicio de impresión LP, consulte *User Accounts, Printers, and Mail Administration*.

Cómo encargar solicitudes de impresión

Para imprimir un archivo desde el indicador de comando, utilice el comando `lp` para enviar una solicitud a la impresora para que imprima dicho archivo. Cuando se realiza una petición, el servicio de impresión LP la sitúa en la cola de espera de impresión, muestra el número de solicitud de identificación y después vuelve a mostrar el indicador shell.

Cómo encargar solicitudes de impresión a la impresora predeterminada

Cuando se establece el servicio de impresión Lp con una impresora predeterminada, puede encargar solicitudes de impresión sin introducir el nombre de la impresora de la forma siguiente:

```
$ lp nombre_de_archivo
```

donde *nombre_de_archivo* es el nombre del archivo que desea imprimir.

El archivo especificado se sitúa en la cola de espera de impresión de la impresora predeterminada y se muestra la *solicitud de identificación*.

Por ejemplo, para imprimir el archivo `/etc/passwd` escriba:

```
$ lp /etc/passwd
solicitud id es piña-8 (1 archivo)
$
```

Si desea información acerca de la especificación de una impresora predeterminada, consulte *User Accounts, Printers, and Mail Administration*.

Cómo encargar solicitudes de impresión utilizando un nombre de impresora

Aunque su sistema no haya designado una impresora predeterminada, puede encargar solicitudes de impresión a cualquier impresora que esté configurada para su sistema. Para encargar una solicitud de impresión a una impresora determinada escriba lo siguiente:

```
$ lp -d nombre_de_impresora nombre_de_archivo
```

donde *nombre_de_impresora* es el nombre de la impresora especificada, y *nombre_de_archivo* es el nombre del archivo que desea imprimir.

El archivo especificado se sitúa en la cola de impresión de la impresora de destino y se muestra la petición identificadora.

Por ejemplo, para imprimir el archivo `/etc/passwd` en la impresora `acorn` introduzca:

```
$ lp -d acorn /etc/passwd
solicitud identificadora es acorn-9 (1 archivo)
$
```

Si encarga una impresión a una impresora que no esté configurada en su sistema, aparecerá un mensaje informativo, como se ve en el siguiente ejemplo:

```
$ lp -d thorn /etc/passwd
UX:lp: ERROR: El destino "thorn" es desconocido para
           el servicio de impresión LP.
$
```

Si desea información acerca de la configuración de impresoras, consulte *User Accounts, Printers, and Mail Administration*. Si necesita información acerca de cómo encontrar qué impresoras están disponibles en su sistema, consulte “Cómo determinar el estado de la impresora” en este capítulo.

Cómo solicitar un aviso cuando la impresión esté completada

Cuando encargue imprimir un archivo muy amplio, puede solicitarle al servicio de impresión LP que le informe de que la impresión ha sido completada. Existen dos modos de solicitarle información al servicio de impresión LP:

- Enviar un mensaje de email
- Escribir un mensaje en su ventana de consola

Para solicitar información de email, use la opción `-m` cuando encargue la solicitud de impresión:

```
$ lp -m nombre_de_archivo
```

Para solicitar que se escriba un mensaje en su ventana de consola, use la opción `-w` cuando encargue la solicitud de impresión:

```
$ lp -w nombre_de_archivo
```

donde *nombre_de_archivo* es el nombre del archivo que se está imprimiendo.

Cómo imprimir copias múltiples

Puede imprimir más de una copia de un archivo. Cuando solicita más de una copia, el archivo se imprime el número de veces que Vd. especifique utilizando la opción `-n` para el comando `lp`. La solicitud de impresión se considera como un trabajo de impresión y sólo se imprime la primera página.

Para solicitar copias múltiples introduzca lo siguiente:

```
$ lp -n número nombre_de_archivo
```

donde *número* es el número de copias deseadas y *nombre_de_archivo* es el nombre del archivo que está imprimiendo.

Por ejemplo, para imprimir cuatro copias del archivo `/etc/passwd`:

```
$ lp -n4 /etc/passwd
solicitud id es piña-9 (1 archivo)
$
```

Cuadro resumen de las opciones lp

Puede configurar su solicitud de impresión utilizando las siguientes opciones del comando `lp`: especificar formatos, juego de caracteres, filtros, títulos, carátulas, etc. La Tabla 8-1 resume las opciones del comando `lp` más utilizadas. Puede usar dichas opciones individualmente o combinarlas en cualquier orden en la línea de comandos. Cuando combine opciones, deje un espacio entre cada opción y repita el subrayado (-).

Por ejemplo, para especificar una impresora de destino, solicitar notificación de email e imprimir seis copias de un archivo, introduciría lo siguiente:

```
$ lp -d nombre_de_impresora -m -n6 nombre_de_archivo
```

donde *nombre_de_impresora* es el nombre de la impresora deseada y *nombre_de_archivo* es el nombre del archivo que está imprimiendo.

Tabla 8-1 Resumen de las opciones `lp` más usadas

Opción	Descripción
-d	Destino. Especifica una impresora de destino según el nombre.
-m	Correo. Envía correo electrónico al solicitante cuando el archivo se ha impreso con éxito.
-n	Número. Especifica el número de copias a imprimir.
-t	Título. Especifica un título (impreso sólo en la página inicial) para una solicitud de impresión.
-o sin página inicial	Opción. Suprime la impresión de la página inicial para una solicitud individual.
-h	Encabezamiento. Coloca un encabezamiento en cada página de la solicitud de impresión.
-c	Copiar. Copia el archivo antes de imprimirlo.
-w	Escribir. Escribe un mensaje en su terminal cuando el archivo se ha impreso con éxito.

Si desea conocer la lista de opciones completa, consulte el *man Pages(1): User Commands*.

Cómo determinar el estado de la impresora

Utilice el comando `lpstat` para informarse del estado del servicio de impresión LP. Puede comprobar el estado de sus propios trabajos en la cola de espera de impresión, determinar qué impresoras están disponibles o especificar las solicitudes identificadoras de sus trabajos si desea cancelarlos.

Cómo comprobar el estado de las solicitudes de impresión

Para informarse del estado de sus propias solicitudes de impresión en espera introduzca lo siguiente:

```
$ lpstat
```

Aparecerá una lista de los archivos que ha encargado imprimir.

En el ejemplo siguiente, en el sistema `pino`, un archivo está en la cola de impresión de la impresora `piña`:

```
$ lpstat
piña-10          fred          1261   Mar 12 17:34 en pino
$
```

El comando `lpstat` muestra una línea para cada trabajo de impresión, haciendo aparecer la solicitud `id` seguida del usuario que la envió, el tamaño de la salida en bytes y la fecha y hora de dicha solicitud.

Cómo comprobar las impresoras disponibles

Para informarse de qué impresoras están configuradas en su sistema escriba lo siguiente:

```
$ lpstat -s
```

Se mostrará el estado del programador seguido del destino predeterminado y una lista de los sistemas e impresoras que estén disponibles.

En el ejemplo siguiente, en el sistema `olmo`, el programador está en funcionamiento, la impresora predeterminada es `piña` y están disponibles dos impresoras de la red: `piña` y `bellota`:

```
$ lpstat -s
programador en uso
destino predeterminado del sistema: piña
sistema de piña: pino
sistema de bellota: roble
$
```

Cómo mostrar toda la información de estado

La opción `-t` de `lpstat` le muestra una breve lista del estado del servicio de impresión LP.

Para mostrar una breve lista de toda la información de estado escriba lo siguiente:

```
$ lpstat -t
```

Aparecerá toda la información de estado disponible.

En el siguiente ejemplo no hay trabajos en la cola de espera de impresión. Cuando se envían archivos a la cola de impresión, se muestra también el estado de dichas solicitudes de impresión:

```
$ lpstat -t
programador en uso
destino predeterminado del sistema: piña
sistema de bellota: roble
piña aceptando solicitudes desde Mie Ene  2 18:20:10 PST 1991
bellota aceptando solicitudes desde Lun Mar  4 15:53:47 PST 1991
la impresora piña está inactiva. conectada desde Mie Ene  2
18:20:22 PST 1991. disponible.

la impresora bellota está inactiva. conectada desde Lun Mar  4
15:53:44 PST 1991. disponible.
$
```

Cómo mostrar el estado de las impresoras

Puede solicitar información sobre el estado de la impresión de impresoras individuales utilizando la opción `-p` de `lpstat`. Esta opción indica si la impresora está activa o inactiva, conectada o desconectada y si acepta solicitudes de impresión o no.

Para solicitar información sobre el estado de todas las impresoras de un sistema introduzca lo siguiente:

```
$ lpstat -p
```

En el ejemplo siguiente, dos impresoras están inactivas, enchufadas y disponibles. Si una de las impresoras tiene trabajos en la cola de espera de impresión, se mostrarían también dichos trabajos.

```
$ lpstat -p
impresora piña está inactiva. conectada desde Mie Ene 2 18:20:22
PST 1991. disponible.
impresora bellota está inactiva. conectada desde Lun Mar 4
15:53:44 PST 1991. disponible.
$
```

Para solicitar información sobre el estado de una impresora individual por su nombre escriba lo siguiente:

```
$ lpstat -p nombre_impresora
```

donde *nombre_impresora* es el nombre de la impresora determinada.

Mostrar las características de las impresoras

Si desea ver todas las características de una impresora, use la opción `-p` junto con la opción `-l` (largo) de `lpstat`. Este comando puede resultar especialmente útil para buscar el tipo de impresora y el tipo de contenido.

Para mostrar las características de todas las impresoras de un sistema introduzca lo siguiente:

```
$ lpstat -p -l
```

Una tabla muestra toda la información acerca de la configuración que usa el servicio de impresión LP para cada impresora.

En el siguiente ejemplo todos los campos están en blanco excepto el tipo de contenido y el tipo de impresora de la impresora piña.

```
$ lpstat -p piña -l
la impresora piña está inactiva. conectada desde Mie Ene 2
18:20:22 PST 1991. disponible.
    Tipos de contenido: PS
    Tipos de impresora: PS
    Descripción:
    Usuarios permitidos:
        (todos)
    Formatos permitidos:
        (ninguno)
    No requiere primera página
    Grupos de caracteres:
        (ninguno)
    Tono predeterminado:
    Tamaño de página predeterminado:
$
```

Tabla resumen de las opciones lpstat

Puede solicitar varios tipos de información acerca del estado de impresión utilizando el comando `lpstat`. La Tabla 8-2 resume las opciones del comando `lpstat` más utilizadas. Use dichas opciones individualmente o combínelas en cualquier orden en la línea de comando. Cuando combine opciones, deje un espacio entre cada opción y repita el guión (-).

Por ejemplo, para mostrar una larga lista de estado de una impresora individual escribiría lo siguiente:

```
$ lpstat -p nombre_de_impresora -l
```

donde *nombre_de_impresora* es el nombre de la impresora cuyo estado desea ver.

Tabla 8-2 Resumen de las opciones lpstat más usadas

Opción	Descripción
-a	Aceptación. Muestra si los destinos de impresión están aceptando solicitudes.
-c	Clase. Muestra las clases y sus miembros.
-d	Destino. Muestra el destino predeterminado.
-f	Formatos. Muestra los formatos.
-o	Salida. Muestra el estado de la salida.
-p [list] [-D] [-l]	Impresora/Descripción/Lista larga. Muestra el estado de las impresoras.
-r	Solicitud. Solicita el estado del programador.
-R	Muestra la posición del trabajo en la cola
-s	Estado. Muestra el resumen del estado
-S	Grupos. Muestra los grupos de caracteres
-u [nombre_usuario]	Usuario. Muestra las solicitudes según usuario
-v	Muestra los dispositivos

Si desea conocer la lista completa de opciones, consulte el *man Pages(1): User Commands*.

Cómo cancelar una solicitud de impresión

Use el comando `cancel` para cancelar una solicitud de impresión cuando está imprimiéndose o en la cola de impresión. Para cancelar una solicitud necesita saber la identificación de dicha solicitud. La solicitud de identificación incluye siempre el nombre de la impresora, un guión y el número de la solicitud de impresión. Cuando encarga la solicitud de impresión se muestra la solicitud id. Si no recuerda la solicitud id, escriba `lpstat` y presione Retorno. Una solicitud de impresión sólo la puede cancelar la persona que la encargó o alguien que esté conectado como `raíz` o `lp`.

Cómo cancelar una solicitud de impresión mediante el número de identificación

Para cancelar una solicitud de impresión escriba lo siguiente:

```
$ cancel requestid
```

donde *requestid* es el número de identificación de la solicitud de impresión deseada.

Aparecerá un mensaje indicándole que la petición está cancelada y empezará a imprimirse el siguiente trabajo en la cola de espera de impresión.

En el ejemplo siguiente se han cancelado dos solicitudes de impresión:

```
$ cancel piña-3 piña4
solicitud "piña-3" cancelada
solicitud "piña-4" cancelada
$
```

Cómo cancelar una solicitud de impresión mediante el nombre de la impresora

También puede cancelar únicamente el trabajo que se está imprimiendo actualmente (si lo encargó) escribiendo el nombre de la impresora en vez de la solicitud de identificación de este modo:

```
$ cancel nombre_de_impresora
```

donde *nombre_de_impresora* es el nombre de la impresora a la que envió la solicitud.

Aparecerá un mensaje indicándole que la solicitud está cancelada y comenzará a imprimirse el siguiente trabajo de la cola de espera de impresión.

En el ejemplo siguiente ha sido cancelada la solicitud de impresión actual:

```
$ cancel piña
solicitud "piña-3" cancelada
$
```

Su administrador de sistemas puede conectarse como `raíz` o `lp` y cancelar la solicitud de impresión actual utilizando el nombre de la impresora como el argumento del comando `cancel`.

Una *red* es un grupo de computadores preparados para comunicarse entre sí. Si su sistema forma parte de una red, tendrá la posibilidad de usar los recursos de otros sistemas de la red mientras permanezca conectado a ella. Puede conectar con otros computadores o ejecutar comandos remotos que afecten a otros sistemas desde su propia estación de trabajo.

En este capítulo se ofrece la siguiente información:

- Conceptos generales acerca de las redes
- Cómo entrar en sistemas remotos
- Cómo copiar archivos desde sistemas remotos
- Cómo ejecutar comandos en sistemas remotos
- Cómo solicitar información de estado en sistemas remotos

Si el computador que utiliza actualmente no está conectado a una red, la información que se va a presentar puede no afectarle. Sin embargo, puede resultarle útil al menos hojear el capítulo para sacar una idea general de las ventajas que puede ofrecer la red.

Conceptos acerca de la red

Una conexión de red entre computadores les permite transmitir información entre ellos. Las redes también se denominan frecuentemente *redes de área local* (LANs), que abarcan áreas reducidas, generalmente de no más de mil metros; *redes de área extensa* (WANs), que pueden abarcar miles de kilómetros; o *redes de área intermedia* (CANs), que tienen un tamaño intermedio.

Una red compuesta por varias redes conectadas entre sí se denomina *interconexión de redes*. Por ejemplo, su sistema puede formar parte de una red dentro de su edificio y parte de una interconexión de redes que conecte su red local con redes similares en todo el país. Como la diferencia entre red e interconexión de redes al usuario le resulta imperceptible, en este manual, el término “red” se usará para denominar tanto a una red como a una interconexión de redes.

Los sistemas conectados a una red se comunican utilizando un *protocolo de red*, o un lenguaje de red común para asegurar que dicha información se transmita a los lugares apropiados. Un *protocolo de interconexión de redes*, a veces denominado *relay*, sirve para conectar redes.

Conectar con el sistema de forma remota (rlogin)

El comando `rlogin` le permite conectar con otros sistemas UNIX de su red.

Para conectar de forma remota con otro sistema introduzca:

```
$ rlogin nombre_de_sistema
```

donde *nombre de sistema* es el nombre del sistema remoto.

Si aparece un indicador de contraseña, escriba la contraseña del sistema remoto y presione Retorno. Si su nombre de sistema está en el archivo `/etc/host-s.equív` del otro sistema, éste “se fiará” de su nombre de sistema y no le pedirá que escriba la contraseña.

```
$ rlogin solitario
Password: (escriba contraseña)
Last login: Mon Jan 6 09:37:55 from triste
Sun Microsystems, Inc.   SunOS 5.1   October 1992
( Los siguientes comandos se ejecutan en el sistema solitario.)
$ pwd
/home/jose
$ logout
Connection closed.
$
```

rlogin *sin un directorio de usuario*

En el ejemplo anterior, el usuario `jose` conectó con `solitario` en el directorio `/home/jose`, como indicó el comando `pwd`. Cuando conecta con un sistema en el que no tiene directorio de usuario, `rlogin` muestra un mensaje indicando que Vd. no tiene directorio de usuario en el sistema remoto y le conecta con el directorio raíz (`/`) del sistema:

```
$ rlogin quejica
Password:
No directory! Logging in with home=/
Last login: Fri Jan 3 10:21:59 from triste
Sun Microsystems, Inc.   SunOS 5.1   October 1992
( Los siguientes comandos se ejecutan en el sistema quejica.)
$ pwd
/
$ logout
Connection closed.
$
```

rlogin *usando el nombre de otro usuario*

Es posible que desee conectar con un sistema remoto usando el nombre de otro usuario. Por ejemplo, si está trabajando en el sistema de otro usuario (usando el nombre de usuario de éste) y desea conectar con su propio sistema utilizando su nombre, podrá hacerlo mediante la opción `-l` de `rlogin`. La sintaxis del comando es:

```
rlogin nombre_de_sistema -l nombre_de_usuario
```

Por ejemplo, el caso siguiente muestra de qué forma el usuario `jose`, en el sistema `triste`, conectaría con el sistema `solitario`, con el nombre `conde`:

```
$ rlogin solitario -l conde
Password:
Last login: Wed Jan 8 07:12:25 from triste
Sun Microsystems, Inc.      SunOS 5.1      October 1992
( Los siguientes comandos se ejecutan en solitario.)
$ pwd
/home/conde
$ logout
Connection closed.
$
```

Observe que cuando conecta con un sistema remoto con el nombre de otro usuario, Vd. es situado en el directorio de usuario de dicho usuario.

rlogin *con un sistema desconocido*

Si intenta conectar con un sistema remoto cuyo nombre desconozca su sistema, `rlogin` buscará sin éxito en las bases de datos del computador principal y mostrará a continuación la siguiente nota:

```
$ rlogin desconocido
desconocido: computador principal desconocido
$
```

Abandonar una conexión rlogin

Normalmente, Vd. termina una conexión rlogin si escribe `logout` al final de la sesión de trabajo. Si por alguna razón no puede terminar una sesión de esta forma, puede abortar la conexión escribiendo un signo `~` seguido de un punto (`~.`) al comienzo de la línea. La conexión de entrada con el sistema remoto se abortará y Vd. será devuelto a su sistema original.

Si conecta con una serie de sistemas, accediendo a cada uno mediante otro sistema distinto, y utiliza `~.` para abortar la conexión con cualquiera de los sistemas de la serie, volverá a su sistema original:

```
$ rlogin salamanca
Password:
Last login: Fri Jan 10 09:14:43 from triste
Sun Microsystems, Inc.   SunOS 5.1       October 1992
(El siguiente comando se ejecuta en salamanca.)
$ ~. (El signo ~ puede no aparecer en pantalla.)
Connection closed.
$
```

Si desea volver a una conexión intermedia rlogin, utilice dos tildes `~` seguidas de un punto (`~~.`) de este modo:

```
$ rlogin solitario
Password:
Last login: Tue Jan 7 08:12:49 from triste
Sun Microsystems, Inc.   SunOS 5.1       October 1992
(El siguiente comando se ejecuta en solitario.)
$ rlogin salamanca
Password:
Last login: Tue Jan 7 10:17:40 from solitario
Sun Microsystems, Inc.   SunOS 5.1       October 1992
(El siguiente comando se ejecuta en salamanca.)
$ ~~. (Los signos ~~ pueden no aparecer en pantalla.)
Connection closed.
$
```

Suspender una conexión `rlogin`

Cuando desee suspender una conexión `rlogin` para volver a ella más tarde, escriba el carácter (~) y después Ctrl-Z. La conexión `rlogin` se convierte en un proceso parado y a Vd. regresa al sistema con el que conectó.

Para reactivar la conexión escriba `fg`. También puede escribir el signo de tanto por ciento (%) seguido del número de proceso del proceso parado (el valor predeterminado de %, si no se incluye ningún número de proceso, es el proceso suspendido más recientemente).

```
$ rlogin solitario
Password:
Last login: Tue Jan 7 08:12:49 from triste
Sun Microsystems, Inc.      SunOS 5.1      October 1992
(El siguiente comando se ejecuta en solitario.)
~^Z ( ^Z puede no aparecer en pantalla.)
Stopped
(El siguiente comando se ejecuta en triste.)
$ pwd
/home/jose
$ %
rlogin solitario

(El siguiente comando se ejecuta en solitario.)
$ logout
Connection closed.
$
```

De igual modo que puede abortar `rlogin` con `~.`, si escribe dos signos ~ y Ctrl-Z se suspende el actual `rlogin` y Vd. se coloca en el `rlogin` intermedio.

Verificar su posición (`who am i`)

Después de conectar con varios sistemas remotos, tal vez bajo diferentes nombres de entrada, necesitará verificar el lugar exacto dónde se encuentra. Al escribir `who am i` aparece el nombre del sistema con el que está conectado actualmente así como su identidad actual.

Escriba `man rlogin` en el indicador de comando o consulte *man Pages(1): User Commands*.

Copiar archivos a distancia (r_{CP})

El comando `rCP` le permite copiar archivos de una sistema a otro. Éste utiliza los archivos `/etc/hosts.equiv` y `/etc/passwd` del sistema remoto para determinar si dispone de derechos de acceso al sistema remoto. La sintaxis del comando `rCP` es similar a la usada con `cp`.

Nota – Para copiar subdirectorios y su contenido de un sistema a otro use `rCP -r`.

Copiar de otro sistema al suyo

Para copiar de un sistema remoto a su sistema la sintaxis es:

```
rCP nombre_de_sistema:fuente destino
```

donde *nombre_de_sistema* es el nombre del sistema remoto, *fuente* es el nombre del archivo o archivos que desea copiar y *destino* es el nombre de la ruta de acceso de su sistema donde quiere situar el archivo o archivos copiados.

El siguiente ejemplo explica cómo copiar el archivo `/home/salamanca/doc/letter` del sistema remoto `salamanca` al directorio `/tmp` del sistema local `triste`:

```
$ rCP salamanca:/home/salamanca/doc/letter /tmp
$
```

Cuando use `rCP` también puede combinar varias abreviaturas y sintaxis. Por ejemplo, para copiar todos los archivos terminados en `.doc` del directorio de usuario del usuario `pepe`, conectado al sistema remoto `quejica`, en el directorio actual del sistema local `triste`, tendría que escribir lo siguiente:

```
$ rCP quejica:~pepe/*.doc .
$
```

Copiar de su sistema a otro

Para copiar de su sistema local a un sistema remoto, la sintaxis se cambia del modo siguiente:

```
rcp fuelle nombre_de_sistema:destino
```

donde *fuelle* es el archivo o archivos que desea copiar, *nombre_de_sistema* es el nombre del sistema remoto y *destino* es el nombre de la ruta de acceso del sistema remoto donde quiere situar el archivo o archivos copiados.

El siguiente ejemplo explica cómo se copiaría el archivo `austin` de su directorio `~/madrid/texas` en el directorio `~pepe/ciudades` del sistema remoto `quejica` (recuerde que `~` es su directorio de usuario y `~pepe` es el directorio de usuario del usuario `pepe`):

```
$ rcp ~/madrid/texas/austin quejica:~pepe/ciudades
$
```

Si desea más información acerca del comando `rcp` y sus opciones, consulte el *man* `Pages(1): User Commands`.

Ejecutar comandos a distancia (`rsh`)

El comando `rsh` (del *shell remoto*) le permite ejecutar un único comando en un sistema remoto sin tener que conectar anteriormente. Esto le puede ahorrar mucho tiempo cuando sólo quiera hacer una cosa en el sistema remoto.

Para ejecutar un comando en un sistema remoto escriba:

```
rsh comando del nombre_del_sistema
```

El siguiente ejemplo muestra cómo se vería el contenido del directorio `/home/solitario/guitarra` del sistema `solitario`:

```
$ rsh solitario ls /home/solitario/guitarra
collings          gibson            santacruz
fender            martin           taylor
$
```

De forma parecida a los comandos `rlogin` y `rccp`, `rsh` usa los archivos `/etc/hosts.equiv` y `/etc/passwd` del sistema remoto para determinar si Vd. tiene derecho de acceso a dicho sistema.

Si desea más información sobre el comando `rsh` y sus opciones, consulte *man Pages(1): User Commands*.

Ver información de usuario (`rusers`)

El comando `rusers` (para *usuarios remotos*) le indica quién está conectado a otros sistemas de su red. Al escribir el comando `rusers` solamente, se muestra cada máquina de la red y el usuario o usuarios que están conectados a ellas del siguiente modo:

```
$ rusers
alegre          guadalupe
triste         jose
salamanca      rosa
dormilón       jose
simpático      charo      juan      carolina
quejica        pepe
solitario      jorge
glotón         manolo
$
```

Observe que el sistema `simpático` tiene en la actualidad tres usuarios conectados.

Para mostrar información acerca de un sistema remoto determinado, escriba el comando `rusers` seguido del nombre del sistema de este modo:

```
$ rusers simpático
simpático      charo      juan      carolina
$
```

La opción `-l` del comando `rusers` ofrece una información más detallada, que incluye los nombres de los usuarios, de los sistema y los terminales, la hora en la que conectó cada usuario, cuánto tiempo ha estado inactivo cada usuario (si ha estado más de un minuto) y el nombre del sistema desde el que conectó cada usuario (si lo hubiera):

```
$ rusers -l freeway
charo      freeway:ttyd8    Feb 10 08:12      5:29
juan       freeway:console  Feb 10 09:16
carolina   freeway:ttyp0    Feb 10 11:56      36
$
```

También puede usar la opción `-l` sin facilitar ningún nombre del sistema.

Si desea más información acerca del comando `rusers` y sus opciones, consulte el *man* `Pages(1): User Commands`.

Personalizar su entorno de trabajo

10 

El sistema operativo SunOS le permite controlar y ajustar diversos aspectos de su entorno de trabajo. Esto lo puede realizar modificando las *variables de entorno* contenidas en los *archivos de inicialización* de su sistema. Cuando entra en el sistema, éste lee los archivos de inicialización y utiliza las variables para configurarse a sí mismo. Estableciendo las variables de entorno, puede “personalizar” su sistema para facilitar su trabajo y hacerlo más eficiente.

Este capítulo explica cómo personalizar su sistema modificando sus archivos de inicialización y estableciendo las variables de entorno más comunes. También enseña cómo poner alias a los comandos de SunOS, cómo cambiar el indicador del sistema, cómo establecer los permisos de archivo predeterminados y cómo personalizar las fuentes de OpenWindows.

Archivos de inicialización

Los archivos de inicialización encargados de la configuración del sistema dependen del shell que el administrador del sistema haya especificado como shell predeterminado cuando instaló el sistema. El shell Bourne es el shell predeterminado de SunOS, pero también puede usar el shell C o el Korn. Cada uno de estos shell dispone de su propio archivo (o archivos) de inicialización.

Si no está seguro de que shell es el shell predeterminado (denominado *shell de entrada*):

1. Escriba `echo $SHELL`:

```
$ echo $SHELL
/bin/sh
```

2. Observe la salida del comando. Si es:

- `/bin/sh` – su shell de entrada será el shell Bourne
- `/bin/csh` – su shell de entrada será el shell C
- `/bin/ksh` – su shell de entrada será el shell Korn

Sin tener en cuenta el shell que esté usando, cuando entra por primera vez en el sistema, éste procesa generalmente el archivo perfil: `/etc/profile`. Usualmente, el administrador del sistema es el dueño de este archivo, y los usuarios lo pueden leer (pero no escribir en él).

Después de que el sistema ejecute el perfil del sistema, este procesa el *perfil del usuario*. El perfil del usuario es uno o varios archivos de inicialización que definen el entorno de trabajo. Por ejemplo, si Vd. se encuentra en el entorno OpenWindows su sistema comprobará dicho archivo o grupo de archivos cada vez que empiece una nueva Herramienta del shell o una ventana de la Herramienta de comandos.

Según que shell predeterminado esté establecido, su perfil de usuario será uno de los siguientes:

- `.profile` (para los shell Bourne y Korn)
- `.login` y `.cshrc` (para el shell C)

Su archivo u archivos de perfil de usuario están situados en su directorio de usuario y le permiten configurar su entorno de trabajo según sus preferencias.

Variables de entorno

El sistema establece el entorno de sistema mediante un conjunto de especificaciones definidas en los archivos de inicialización. Si desea modificar temporalmente su entorno para la sesión actual de trabajo puede escribir comandos directamente tras el indicador de comandos. Sin embargo, si desea modificar su entorno de trabajo de un modo más permanente, puede almacenar variables de entorno “permanentes” en los archivos `.profile`, `.login`, o `.cshrc`.

Para mostrar las variables de entorno establecidas actualmente en el sistema:

1. Escriba el comando `env` y presione Retorno:

```
$ env
HISTORY=100
HOME=/home/guadalajara/pancho
HZ=100
LOGNAME=pancho
MAIL=/var/mail/pancho
MANSECTS=\1:1m:1c:1f:1s:1b:2:\3:3c:3i:3n:3m:3k:3g:3e:3x11:3xt:3
w:3b:9:4:5:7:8
PATH=/usr/bin
SHELL=/bin/sh
TERM=sun
TZ=EST5EDT
```

Nota – También puede usar el comando `env` para identificar su shell de entrada. Éste está especificado en la variable de entorno `SHELL`. En el ejemplo anterior, está establecido como `/bin/sh` (el shell Bourne).

El perfil del usuario

Esta sección describe algunas de las variables de entorno más usadas. Muchas de dichas variables pueden estar ya incluidas en su perfil de usuario. Como ya se mencionó anteriormente, su archivo de perfil de usuario (`.profile` para los shell Bourne y Korn, y `.cshrc` para el shell C) se encuentra en su directorio de usuario.

Nota – Los archivos (“punto”) ocultos pueden listarse escribiendo `ls -la`.

A continuación se enumera una lista parcial de variables de entorno que se pueden incluir en su perfil de usuario. La sintaxis para definir variables de entorno dependerá del shell que esté usando:

- `CDPATH` – Especifica los directorios que se han de buscar cuando se escribe un único nombre de directorio sin un nombre de ruta de acceso completo.
- `HISTORY` – Establece el número de comandos disponibles para el comando `history` (sólo para el shell C).

- HOME – Define la ruta de acceso absoluta de su directorio raíz. El sistema utiliza esta información para determinar que el directorio cambie cuando escriba el comando `cd` sin argumentos.
- LANG – Especifica el lenguaje local. Los valores adecuados son: japonés, alemán, francés, sueco, español e italiano.
- LOGNAME – Define su nombre de entrada. El valor predeterminado para esta variable se establece automáticamente al nombre de entrada especificado en la base de datos `passwd` como parte del proceso de entrada. Si desea información acerca de la base de datos `passwd`, consulte *User Accounts, Printers, and Mail Administration*.
- LPDEST – Define la impresora predeterminada.
- MAIL – Especifica la ruta de acceso de su correo, normalmente situado en el directorio `/var/mail/username`, donde `username` es su nombre de entrada. Si desea más información acerca de este archivo, consulte el Capítulo 7, “Uso del correo electrónico”.
- MANSECTS – Establece las secciones de documentación en línea disponibles.
- PATH – Lista por orden los directorios que busca el sistema para encontrar un programa a ejecutar, cuando escribe un comando. Si el directorio apropiado no se encuentra en la ruta de acceso de búsqueda, deberá introducirlo o escribir el nombre completo de la ruta de acceso cuando introduzca un comando.

El valor predeterminado de esta variable se define automáticamente y se establece como se especifica en el archivo `.profile` (shell Bourne o Korn), o en el archivo `.cshrc` (shell C) como parte del proceso de entrada.

- PS1 – Define el indicador de comando. El indicador predeterminado del shell Bourne y del Korn es un signo de dolar (`$`). El indicador predeterminado del shell C es el signo de tanto por ciento (`%`). El indicador predeterminado de raíz en cada shell es el signo (`#`).
- SHELL – Define el shell usado por `vi` y por otras herramientas.
- TERMINFO – Especifica el nombre de ruta de acceso de un terminal no soportado que ha sido agregado a la base de datos `terminfo`. No es necesario que establezca esta variable para terminales predeterminados de esta base de datos. Si desea más información sobre la base de datos `terminfo`, consulte *User Accounts, Printers, and Mail Administration*.

- `TERM` – Define el tipo de terminal que está usando actualmente. Cuando ejecuta un editor, el sistema busca un archivo con el mismo nombre que la definición de dicha variable. Primero busca la ruta de acceso (si la hay) remitida por la variable `TERMINFO`, y después el directorio predeterminado `/usr/share/lib/terminfo`, para determinar las características del terminal. Si no se encuentra una definición en ninguna de las dos posiciones, el terminal se identifica como “sin inteligencia”.
- `TZ` – Define la zona horaria del reloj del sistema.

Cómo establecer la variable `PATH`

La variable de entorno `PATH` se usa para ubicar comandos dentro de la jerarquía de directorios de SunOS. Al fijar `PATH` se crea un conjunto de directorios fijos que el sistema buscará siempre que Vd. escriba el nombre de un comando.

Por ejemplo, si no tiene establecida ninguna variable `PATH` y quiere copiar un archivo, tendrá que introducir el nombre de ruta de acceso completo para el comando, `/usr/bin/cp`. Sin embargo, si ha establecido la variable `PATH` para incluir el directorio `/usr/bin`, podrá simplemente escribir `cp` y el sistema ejecutará siempre el comando. Esto es debido a que su sistema busca el comando `cp` en todos los directorios nombrados en la variable `PATH` y lo ejecuta cuando lo encuentra. Si utiliza la variable `PATH` para listar los directorios de comando de SunOS puede facilitar significativamente su trabajo.

Para los shell Bourne y Korn, la variable `PATH` se especifica en el archivo `.profile` (en su directorio personal) utilizando la siguiente sintaxis:

```
PATH=.: /usr/bin: /home/bin
```

donde *home* representa el nombre de la ruta de acceso de su directorio de *usuar*.

Para el shell de C, la variable `PATH` se especifica en el archivo `.cshrc` (en su directorio de usuario) utilizando la siguiente sintaxis:

```
set path=(. /usr/bin home/bin)
```

donde *home* es el nombre de ruta de acceso de su directorio de usuario.

Nota – En el shell de C puede utilizar el método abreviado `~` para representar el nombre de la ruta de acceso de su directorio de usuario.

Si modifica la variable `PATH`, y está ejecutando el shell de C, use el comando `source` para que los cambios surtan efecto en la ventana actual sin tener que salir del sistema:

```
example% source .cshrc
```

Si está ejecutando el shell Bourne o el Korn, para que los cambios surtan efecto en la ventana actual sin tener que abandonar el sistema, escriba lo siguiAlias (sólo el shell de C):

```
$ . .profile
```

Alias (Sólo C Shell)

Los alias son métodos abreviados que resultan muy útiles para comandos que utilice frecuentemente. Por ejemplo, la configuración predeterminada del comando eliminar (`rm`) no solicita confirmación antes de eliminar archivos. A veces esto resulta incómodo, ya que un error de escritura puede eliminar un archivo no deseado. Sin embargo, el shell de C le permite usar la variable `alias` para cambiar esto agregando al archivo `.cshrc` la línea siguiente:

```
alias rm 'rm -i'
```

Con esta línea incluida en el archivo `.cshrc`, escribir `rm` no será lo mismo que escribir `rm -i`, que es el formato interactivo del comando `rm`. Entonces, siempre se pedirá que confirme el comando antes de eliminar ningún archivo. Las comillas de `rm -i` del ejemplo anterior son necesarias para incluir el espacio en blanco entre `rm` y `-i`. Si no existieran, el shell de C no interpretaría correctamente el texto que va después del espacio.

Para hacer que los cambios efectuados en el archivo `.cshrc` surtan efecto inmediatamente en la ventana actual, use el comando `source`. El comando `source` hace que el sistema lea el archivo `.cshrc` actual y ejecute sus comandos:

```
example% source .cshrc
```

Cómo cambiar el indicador de comando

La sintaxis que utiliza para cambiar el indicador de comando dependerá de que esté utilizando el shell Bourne, Korn o el de Shell Bourne y Shell Korn y C shell.

Para los shell Bourne o Korn, Vd. redefine el indicador de comando con el comando `PS1`. Aquí tiene tres ejemplos:

```
PS1=": "
PS1="`hostname` : "
PS1="`hostname`{'id'}: "
```

- El primer ejemplo establece el indicador como el signo de dos puntos (:), seguido de un espacio.
- El segundo ejemplo crea un indicador formado por su nombre de sistema, seguido de dos puntos y un espacio.
- El tercer ejemplo establece el indicador como su nombre de sistema, seguido de su nombre de entrada entre llaves {}, dos puntos y un espacio.

Realice cualquiera de los ejemplos anteriores para cambiar el indicador de comando actual. El indicador se conservará hasta que lo cambie otra vez o salga del sistema.

Si desea realizar cambios más duraderos, agregue uno de los ejemplos anteriores (o un indicador creado por Vd. mismo) al archivo `.profile`. Si realiza esto, el indicador que especifique aparecerá cada vez que entre en el sistema o empiece un nuevo shell.

Shell de C

Para el shell de C, puede personalizar su indicador de comando con el comando `set prompt`. Aquí tiene tres ejemplos:

```
set prompt="% "
set prompt="`hostname`\\!: "
set prompt="`hostname`{'id'}: "
```

- El primer ejemplo establece el indicador como el signo de tanto por ciento seguido de un espacio.
- El segundo ejemplo crea un indicador formado por su nombre de sistema seguido del nombre de historial del comando (`hostname1`, `hostname2`, `hostname3`, etc.).
- El tercer ejemplo establece el indicador como su nombre de sistema, seguido por su nombre de entrada entre llaves, dos puntos y un espacio.

Escriba cualquiera de los ejemplos anteriores para cambiar el indicador de comando actual. El indicador se conservará hasta que lo cambie o salga del sistema.

Si desea hacer cambios más duraderos, agregue uno de los ejemplos anteriores (o un indicador creado por Vd. mismo) al archivo `.cshrc`. Si realiza esto, el indicador que especifique aparecerá cada vez que entre en el sistema o empiece un nuevo shell.

Otras variables útiles

Existen muchas otras variables que puede establecer en los archivos `.profile` o `.cshrc`. Si desea una lista completa, consulte *man Pages(1): User Commands*. A continuación se dará una breve descripción de las opciones utilizadas más frecuentemente.

Use `set noclobber` para prevenir que se sobrescriba accidentalmente en los archivos cuando se use el comando `cp` para copiar un archivo. Esta variable sólo afecta al shell de C. Escriba lo siguiente en el archivo `.cshrc`:

```
set noclobber
```


Use `set history` para establecer el número de comandos que se van a guardar en su lista de historia. El comando `history` resulta muy útil para introducir comandos que haya escrito anteriormente. El archivo `history` también puede usarse para repetir comandos anteriores. Esta variable sólo afecta al shell de C. Escriba lo siguiente en el archivo `.cshrc`:

```
set history=100
```

También puede afectar a los shells Bourne y Korn de la misma manera escribiendo lo siguiente en el archivo `.profile`:

```
HISTORY=100
```

Cómo establecer permisos de archivos predeterminados

El `umask` comando establece un permiso de archivo predeterminado para todos los archivos y directorios que cree. Por ejemplo, si le preocupa la seguridad y desea proteger a los miembros de su grupo y todos los usuarios, simplemente lea y ejecute los permisos (`-rwxr-xr-x`) en sus directorios y archivos, Vd. puede establecer el comando `umask` en los archivos `.cshrc` y `.profile` de modo que cada nuevo archivo o directorio que cree quede protegido por estos permisos.

Al igual que el comando `chmod`, `umask` utiliza un código numérico para representar permisos de archivo absolutos. Sin embargo, el método utilizado para calcular el código de `umask` es bastante diferente del utilizado con `chmod`.

Primeramente, si `umask` se establece a `000`, todos los archivos que cree tendrán los siguientes permisos (de lectura y escritura pero no de ejecución):

```
rw-rw-rw- (modo 666)
```

y todos los directorios creados tendrán los siguientes permisos (de lectura, escritura y ejecución):

```
rwxrwxrwx (modo 777)
```

Para determinar el valor de uso de `umask`, se resta el valor de los permisos que desee (utilizando el valor para el comando `chmod`) de los permisos predeterminados actuales asignados a los archivos. El resto es el valor a usar por el comando `umask.d`.

Por ejemplo, suponga que desee cambiar el modo de acceso predeterminado de los archivos que van desde 666 (`rw-rw-rw-`) hasta 644 (`rw-r--r--`). Reste 644 de 666. El resto, 022, es el valor numérico a utilizar con `umask` de este modo:

```
umask 022
```

Similar al código numérico del comando `chmod` son los tres números utilizados con `umask` de la siguiente manera:

- El primer dígito controla los permisos del usuario
- El segundo controla los permisos del grupo
- El tercer dígito controla los permisos de todos los demás

La Tabla 10-1 muestra los permisos de archivo creados por cada dígito del código numérico del comando `umask`.

Table 10-1 Permisos de `umask`

código umask	Permisos
0	<code>rwX</code>
1	<code>rw-</code>
2	<code>r-x</code>
3	<code>r--</code>
4	<code>-wX</code>
5	<code>-w-</code>
6	<code>--X</code>
7	<code>---(ninguno)</code>

Si desea una mayor información acerca del comando `umask`, consulte *man Pages(1): User Commands*.

Cómo configurar las fuentes de OpenWindows

Puede personalizar si lo desea el tamaño y el tipo de fuentes que aparecen en las aplicaciones OpenWindows. Las siguientes secciones explican cómo configurar dichas fuentes.

Cómo especificar el tipo de fuente y el tamaño del punto

La fuente predeterminada para las ventanas es Lucida Sans de 12 puntos (mediana); la fuente predeterminada de los encabezamientos de ventana es Lucida Sans Bold. Si lo prefiere, puede especificar otro tamaño y tipo de fuente para las ventanas y los encabezamientos de ventana. Puede realizar un cambio para una sola ventana o un cambio permanente para todas las aplicaciones con la ventana de propiedades del área de trabajo. Las siguientes subsecciones describen cada una de estas opciones.

Fuentes de anchura fija y de espacios proporcionados

Observe que existen dos categorías generales de fuentes: *de anchura fija* y *de espacios proporcionados*. Todos los caracteres de una fuente de anchura fija ocupan el mismo espacio. Por el contrario, los caracteres de una fuente de espacios requieren diversos espacios, dependiendo de su anchura individual. Las fuentes con espacios proporcionados resultan más agradables para la lectura. Sin embargo, algunas aplicaciones (como la Herramienta de comandos, la Herramienta del shell y `xterm`, una conocida aplicación de emulador de terminal) trabajan mejor con fuentes de anchura fija.

Cómo escoger entre fuentes fijas y proporcionadas

Observe que la fuente predeterminada mostrada por la Herramienta de comandos y la Herramienta del shell es una fuente de espacios proporcionados. Aunque esta fuente resulta muy agradable para la lectura, causa problemas en el alineamiento de los caracteres (con la tabulación y el espaciado) con cualquier fuente de espacios proporcionados en ventanas del terminal. Si el alineamiento de los caracteres en el espaciado y la tabulación le causan problemas, es mejor que escoja una fuente de anchura fija para dichas ventanas. En los ejemplos siguientes, sólo se usan fuentes de anchura fija para las ventanas del terminal. En los ejemplos de otras ventanas y encabezamientos se usan fuentes con espacios proporcionados.

Cómo especificar la fuente para una sola ventana

Esta sección explica cómo abrir una sola aplicación con un tipo de fuente y tamaño de punto modificados. Observe que no se pueden realizar cambios en ventanas ya existentes; para mostrar una nueva fuente se debe empezar una nueva aplicación. Para empezar una nueva aplicación debe escribir su nombre en una línea de comando.

El comando básico, que se mostrará a continuación, especifica el nombre de la aplicación, la opción `-fn` (nombre de fuente) y el tipo y tamaño de fuente. El signo (&) hace regresar el indicador del sistema a la ventana después de que escriba el comando, esto le permite seguir usando dicha ventana.

```
$ aplicación -fn tipo de fuente-tamaño del punto&
```

Los siguientes ejemplos explican cómo utilizar el comando para abrir una aplicación con un tamaño y tipo de fuente especificados.

- El ejemplo siguiente empieza una nueva Herramienta de comandos con la fuente de espacios proporcionales `Lucida Sans Typewriter Bold`.

No se especifica el tamaño del punto, por lo tanto se usa el valor predeterminado (12 puntos).

```
$ cmdtool -fn lucidasans-typewriter-bold &
```

- El ejemplo siguiente empieza una nueva Herramienta del shell con `Lucida Sans Typewriter Bold` y aumenta el tamaño de la fuente de 12 puntos a 14.

Observe que cuando cambia el tamaño de la fuente cambia también el tamaño de la ventana.

```
$ shelltool -fn lucidasans-typewriter-bold-14 &
```

- El ejemplo siguiente empieza con una nueva ventana del terminal `xterm` con la fuente `terminal-bold` de 16 puntos:

```
$ xterm -fn terminal-bold-16 &
```

- El ejemplo empieza un nuevo Editor de Textos con la fuente `Helvetica Bold` con 14 puntos:

```
$ textedit -fn helvetica-bold-14 &
```

Use la opción `-fn` con cualquier aplicación y cualquier tipo y tamaño de fuente que desee. La sección, “La lista de fuentes disponibles” en la página 164” de este capítulo explica como listar todas las fuentes disponibles de las aplicaciones de OpenWindows.

Cómo hacer permanentes las asignaciones de fuentes

Si está procesando continuamente aplicaciones con fuentes configuradas, tal vez desee agregar la configuración al menú del área de trabajo. Esto lo puede hacer utilizando la categoría Menú de Programas de la ventana de propiedades del área de trabajo. Así se evitará el trabajo de escribir cada vez las opciones de la línea de comando. Por ejemplo, si desea ejecutar frecuentemente el editor de texto con un tamaño de punto más grande, tendría que agregar la siguiente línea de comando al menú de programas:

```
textedit -fn lucidasans-typewriter-14
```

Puede tener más de un ejemplo de la misma aplicación en su menú de programas si desea que tengan diferentes tamaños de fuente. Esto resultará útil si está procesando una gran variedad de tamaños de punto. Por ejemplo, es posible

que desee tener la opción de ejecutar un Editor de Textos utilizando fuentes de 12, 14 ó 18 puntos. Agregaría los siguientes comandos a su menú de programas:

```
textedit -fn lucidasans-typewriter-12
textedit -fn lucidasans-typewriter-14
textedit -fn lucidasans-typewriter-18
```

Una vez que haya personalizado de este modo su menú de programas desde su ventana de propiedades del área de trabajo, puede invocar al Editor de Textos en cualquiera de dichos tamaños de punto, simplemente seleccionando el elemento adecuado en su menú de programas.

Nota – Las líneas de comando agregadas al menú de programas no deben ir seguidas del signo (&).

Cómo listar las fuentes disponibles

Es posible que desee experimentar con más fuentes de las que se han mostrado en los ejemplos anteriores y que quiera aplicarlas en otras aplicaciones de OpenWindows. Para hacer esto liste primero las fuentes existentes y selecciónelas después.

La lista de fuentes disponibles

Puede ver toda la lista de fuentes disponibles escribiendo `xlsfonts` tras el indicador de la ventana del emulador de terminal. Es más conveniente usar la Herramienta de comandos para ver la lista ya que es más probable que la lista se desplace hacia la parte superior de la pantalla, y la Herramienta de comandos dispone de una barra de desplazamiento que le permite ver toda la lista.

Nota – La lista generada desde `xlsfonts` es muy larga; existen más de 400 fuentes. Si la lista de su pantalla no contiene el número de fuentes esperado, revísela con su administrador de sistemas. Es posible que se instalase un subgrupo de fuentes.

Cada fuente tiene un nombre completo así como una versión abreviada. Por ejemplo, el nombre completo de `lucidasans-typewriter` es:

```
-b&h-lucida sans typewriter-medium-r-normal-sans-12-120-72-72-m-0-iso8859-1
```

Las fuentes que ve en el listado `xlsfonts` son los nombres enteros seguidos de los abreviados. En este capítulo use los nombres abreviados.

Una vez que haya escogido una fuente, siga las instrucciones contenidas en “Cómo especificar el tipo de fuente y el tamaño del punto” en la página 161 ,” en este capítulo para personalizar las fuentes de las ventanas de aplicación.

Cómo cambiar a la versión 3.3 de OpenWindows, o posteriores



Puede ocurrir que esté ejecutando una versión del software del entorno de usuario que ya no sea compatible con Solaris, que ejecuta OpenWindows como su entorno de usuario predeterminado. Por ejemplo, si está ejecutando el software del entorno de usuario SunView o una versión de OpenWindows anterior a la 3.3, tendrá que modernizar el software y cambiar a la Versión 3.3, o a versiones posteriores de OpenWindows. Este apéndice le explica cómo hacerlo.

Nota – El software de SunView ya no es soportado por la Versión 3.3 de OpenWindows, o versiones posteriores. A diferencia de las anteriores versiones de OpenWindows, una vez que actualice la Versión 3.3 o una posterior, ya no tendrá la posibilidad de ejecutar también SunView.

SPARC *Cómo cambiar desde el entorno SunView*

Si cambia al entorno OpenWindows desde el entorno SunView, la siguiente información puede facilitarle el cambio:

SPARC *Los archivos .defaults y .xdefaults*

Para personalizar su entorno OpenWindows de la misma manera que su entorno SunView, puede convertir el archivo `.defaults` (utilizado por el software de SunView) en un archivo `.xdefaults` (utilizado por el software de

OpenWindows). Si tiene un archivo `.defaults` en su directorio de usuario, deberá ejecutar el programa `convert_to_Xdefaults(1)` de su directorio de usuario de la siguiente manera:

```
$ cd
$ $OPENWINHOME/bin/convert_to_Xdefaults .defaults
```

Así se creará un archivo `.Xdefaults` en su directorio de usuario que se utilizará para personalizar su entorno OpenWindows cuando empiece el software.

Cómo cambiar desde una versión del entorno OpenWindows anterior a la 3.3

Lea esta sección detenidamente si con anterioridad ejecutaba una versión del entorno OpenWindows anterior a la 3.3 y ahora desea cambiar a la Versión 3.3, o posteriores. En particular, mucha de esta información va dirigida a los usuarios que usaban la Versión 2 del entorno OpenWindows.

La variable de entorno OPENWINHOME

Si está procesando en la actualidad una versión del software de OpenWindows anterior a la Versión 3.3, es posible que haya configurado su sistema para que utilice la variable de entorno `OPENWINHOME`. Ya no es recomendable que los usuarios establezcan la variable de entorno `OPENWINHOME`, ni manualmente ni mediante un archivo de inicio.

Cuando ejecute el comando `openwin`, éste automáticamente establecerá la variable de entorno `OPENWINHOME` como `/usr/openwin`; por lo tanto, Vd. no necesitará hacerlo.

Si ha establecido la variable de entorno `OPENWINHOME` en el archivo `.profile` o en el `.cshrc` en su directorio de usuario, haga un comentario en la línea o elimínela entera antes de ejecutar la Versión 3.3 de OpenWindows, o versiones posteriores.

Para quitar o convertir en comentario la variable de entorno `OPENWINHOME` en el archivo `.profile` o en el `.cshrc`:

- 1. Abra el archivo `.profile` o el `.cshrc` usando un Editor de Textos como por ejemplo el `vi`.**

2. Escriba un signo # antes de la variable, como se muestra a continuación o elimine la línea entera.

Si está trabajando en el archivo `.profile`, siga el ejemplo a, si está trabajando en el archivo `.cshrc`, siga el ejemplo b

a. En el archivo `.profile`:

```
#OPENWINHOME=/usr/openwin
```

a. En el archivo `.cshrc`:

```
#setenv OPENWINHOME /usr/openwin
```

3. Guarde y salga del archivo.

El archivo `.xinitrc`

A continuación se enumera una serie de observaciones importantes sobre el uso de los archivos `.xinitrc` y `$OPENWINHOME/lib/Xinitrc`:

1. En el entorno de la Versión 2 de OpenWindows, el archivo de comandos `openwin` creaba de forma automática una copia de `$OPENWINHOME/lib/Xinitrc` en un archivo llamado `.xinitrc` de su directorio de usuario. Esto ya no sucede en el entorno de la Versión 3.3 de OpenWindows, lo que es importante por las siguientes razones:
 - a. El archivo de comandos de inicio `openwin` utiliza el archivo de inicio predeterminado `$OPENWINHOME/lib/Xinitrc`, a no ser que exista un archivo `.xinitrc` en su directorio de usuario que sobrescriba al archivo predeterminado.
 - b. Es importante que utilice el archivo predeterminado `$OPENWINHOME/lib/Xinitrc` que se incluye en el software de la Versión 3.3 de OpenWindows, o versiones posteriores. No obstante, si desea conservar cualquier cambio especial realizado en el archivo `.xinitrc` del software de la Versión 2, puede hacerlo siguiendo las instrucciones que se darán en esta sección
2. Si ejecuta su sistema utilizando pantallas múltiples, ya no necesitará múltiples versiones de `olwm`.

Uso del archivo de inicio correcto

Si está ejecutando actualmente una versión del software de OpenWindows anterior a la Versión 3.3, es importante determinar el estado del archivo `.xinitrc`. Este archivo `.xinitrc` puede contener opciones específicas de usuario.

Para determinar el estado del archivo `.xinitrc`, escriba los siguientes comandos:

```
$ cd
$ ls -a .xinitrc
```

Dependiendo de la salida de este comando, haga lo siguiente en cada caso:

- Si no tiene un archivo `.xinitrc` (es decir, el resultado del comando `ls -a` anterior no muestra un listado del archivo) no haga nada. Si no existe ningún archivo `.xinitrc` en su directorio de usuario, OpenWindows utilizará el archivo de inicio predeterminado del sistema.
- Si tiene un archivo `.xinitrc` (es decir, el resultado del comando `ls -a` anterior muestra un listado del archivo), pero no ha efectuado ningún cambio en el archivo o no desea guardar los cambios que ha realizado, siga el paso 1 de “Procedimientos de inicio del archivo”.
- Si tiene un archivo `.xinitrc` (es decir, el resultado del comando `ls -a` anterior muestra un listado del archivo) y ha realizado cambios en el archivo que desea conservar, siga el paso 2 de “Procedimientos de inicio del archivo”.

Procedimientos de inicio del archivo

1. Para eliminar el archivo `.xinitrc` de su directorio de usuario, escriba el comando siguiente:

```
$ rm .xinitrc
```

2. Para conservar los cambios efectuados en el archivo `.xinitrc`, siga estos pasos:

a. Mueva `.xinitrc` a `.xinitrc.save`:

```
$ mv .xinitrc .xinitrc.save
```

b. Copie `$OPENWINHOME/lib/Xinitrc` en `.xinitrc` en su directorio de usuario:

```
$ cp $OPENWINHOME/lib/Xinitrc $HOME/.xinitrc
```

c. Agregue las líneas que desee conservar de `.xinitrc.save` en `.xinitrc`.



Atención– Cuando edite el archivo `.xinitrc`, no agregue una segunda versión de `olwm`, no agregue `svenv` ni elimine la línea que contenga `$OPENWINHOME/lib/openwin-sys`.

Ventana de propiedades del área de trabajo

En anteriores versiones de OpenWindows (antes de la versión 3.3), si se efectuaba un cambio en el menú de la ventana de propiedades del área de trabajo se almacenaba en el archivo `.Xdefaults` de su directorio de usuario. En la versión 3.3 de OpenWindows, y versiones posteriores, los cambios realizados en el menú de la ventana de propiedades del área de trabajo se almacenan en el archivo `.OWdefaults`, también de su directorio de usuario. El archivo `.Xdefaults` puede todavía existir, pero tienen prioridad las personalizaciones realizadas en el archivo `.OWdefaults`.

El archivo `.Xdefaults` *sólo* debe usarse para realizar cambios de personalización adicionales que no se puedan efectuar mediante la ventana de propiedades del área de trabajo. Por ejemplo, puede editar el archivo `.Xdefaults` utilizando un editor de texto como el `vi` para realizar personalizaciones en aplicaciones que no pertenezcan a OpenWindows o agregar macros del preprocesador de C. El uso de la ventana de propiedades del área de trabajo no afecta a dichas personalizaciones.

Si ya tiene un archivo `.Xdefaults` en su directorio de usuario y no desea efectuar en él ninguna personalización, no hará falta que lo elimine. Como el archivo `.OWdefaults` tiene prioridad sobre el archivo `.Xdefaults`, éste no causa molestias.

Cómo personalizar el menú del área de trabajo

En OpenWindows 3.3, o versiones posteriores, el submenú Programas se personaliza en el menú del área de trabajo usando la ventana de propiedades del área de trabajo. Antes de la versión 3.3 de OpenWindows esta personalización se realizaba mediante el archivo `.openwin-menu` de su directorio de usuario.

Nota – Si no dispone de un archivo `.openwin-menu` en su directorio de usuario, no hará falta que realice el procedimiento siguiente. Puede personalizar el menú de área de trabajo mediante la ventana de propiedades del área de trabajo.

Si tiene un archivo `.openwin-menu`, para personalizar el menú de área de trabajo mediante la ventana de propiedades del área de trabajo tendría que realizar los pasos siguientes.

Si ve esta línea en el archivo `.openwin-menu`:

```
"Programs"MENU $OPENWINHOME/lib/openwin-menu-programs
```

elimínela y reemplácela por esta línea:

```
"Programs"INCLUDE openwin-menu-programs
```

Si el archivo `.openwin-menu` no contiene la citada línea, simplemente agregue al archivo `.openwin-menu` la línea sustituta como se indicó anteriormente.

Al agregar o sustituir dicha línea, el menú predeterminado Programas se agrega al menú del área de trabajo. Esto le permite personalizarlo mediante la ventana de propiedades del área de trabajo.

Si en el menú del área de trabajo se encontrara con elementos redundantes, simplemente elimínelos borrando las líneas repetidas del archivo `.openwin-menu`.

Cómo modificar el teclado



En este apéndice se dan instrucciones para remapear el teclado. También se incluyen opciones de remapeo para teclas especiales del teclado. A su vez, ofrece información sobre cómo activar y desactivar la tecla Componer del teclado.

Si desea información acerca del remapeo de los botones del ratón (por ejemplo, para que los usuarios zurdos puedan utilizarlo con mayor facilidad), consulte *Guía del Usuario Solaris*.

Cómo activar/desactivar la tecla Componer

x86 – La tecla Componer equivale a Ctrl-Mayús-F1 en los sistemas x86.

Si no utiliza la tecla Componer, puede desactivarla para que no se pulse de forma accidental. Primeramente, busque el código de tecla para Multi_key:

```
$ xmodmap -pk | grep Multi_key
```

Su sistema mostrará una línea parecida a esta:

```
nn 0xff20 (Multi_key)
```

La información importante es el número de dos dígitos del código de tecla al comienzo de línea, representado por *nn*. Use este número de código para construir la siguiente línea en su fichero `.xinitrc`:

```
xmodmap -e 'keycode nn = NoSymbol'
```

Para reactivar la tecla Componer, convierta en comentario la línea anterior de su archivo `.xinitrc` y reinicie el software OpenWindows.

SPARC *Remapeo del teclado para zurdos*

El archivo de comandos para remapear teclas de esta sección (proporcionado para los teclados del tipo 4 y 5) remapea la mayoría de las teclas especiales en los paneles derecho e izquierdo del teclado, (esto es, los bloques de teclado a la izquierda y derecha del área principal del teclado).

SPARC – Observe que las siguientes secciones pertenecientes a “Remapeo del teclado para zurdos” se refieren sólo a sistemas basados en SPARC.

SPARC *Uso del archivo de comandos de remapeo*

Para crear y utilizar el archivo de comandos de remapeo:

- 1. Cree un archivo llamado `lefty.data` usando cualquier editor de textos.** Puede estar situado en cualquier directorio. El paso cuatro debe ocurrir en el mismo directorio en el que crea dicho archivo.
- 2. Escriba en el archivo de comandos tal como se indica en la Tabla B-1 “El archivo de comandos `lefty.data`”.** Cualquier línea que tenga delante un signo de exclamación es una línea de comentario y no ejecuta ninguna operación.
- 3. Guarde los cambios y salga del editor.**
- 4. En el indicador, escriba: `xmodmap lefty.data`**
Debe estar en el mismo directorio que el archivo guión.

5. Haga clic en un botón del ratón en el área de trabajo para que el guión surta efecto.

Una vez que haya completado estos pasos, puede utilizar el teclado de forma que las teclas estén mapeadas para que las pueda usar una persona zurda.

Escriba lo siguiente en el archivo `lefty.data`, tal como se describe en el paso 1.

Tabla B-1 El archivo de comandos `lefty.data`

```
!  
! lefty.data  
!  
! Datos de xmodmap para establecer las teclas de función derecha e izquierda para  
! personas zurdas en el teclado Sun del tipo 4. Para usar estos datos escriba lo  
! siguiente, donde <filename> es el nombre del archivo (es decir, lefty-data):  
!  
! xmodmap <filename>  
!  
! Los comentarios siguientes corresponden a las asignaciones del código de tecla que  
! siguen inmediatamente después.  
!  
! intercambiar L2 (Repetir) con R1 (Pausa)  
! intercambiar L3 (Props) con R6 (KP_Multiplicar)  
! intercambiar L4 (Anular) con R4 (KP_Igual)  
! intercambiar L5 (Delante) con R9 (KP_9)  
! intercambiar L6 (Copiar) con R7 (KP_7)  
! intercambiar L7 (Abrir) con R12 (KP_6)  
! intercambiar L8 (Pegar) con R10 (Izquierda)  
! intercambiar L9 (Buscar) con R15 (KP_3)  
! intercambiar L10 (Cortar) con R13 (KP_1)  
!  
! cambiar R3 (Break) a L1 (Parar)
```

Tabla B-1 El archivo de comandos `lefty.data`

```
! cambiar R2 (Imprimir) a R10 (Izquierda)
! cambiar R5 (KP_Separar) a R12 (Derecha)
!
! cambiar Final de línea a Control-R
!
código de tecla 10 = R1      R1      Pause
código de tecla 28 = L2      L2      SunXK_Again
código de tecla 32 = R6      R6      KP_Multiply
código de tecla 54 = L3      L3      SunXK_Props
código de tecla 33 = R4      R4      KP_Equal
código de tecla 52 = L4      L4      SunXK_Undo
código de tecla 56 = R9      R9      KP_9      Prior
código de tecla 77 = L5      L5      SunXK_Front
código de tecla 58 = R7      R7      KP_7      Home
código de tecla 75 = L6      L6      SunXK_Copy
código de tecla 79 = Derecha R12      KP_6
código de tecla 100 = L7     L7      SunXK_Open
código de tecla 80 = Izquierda R10     KP_4
código de tecla 98 = L8      L8      SunXK_Paste
código de tecla 102 = R15    R15     KP_3      Next
código de tecla 121 = L9     L9      SunXK_Find
código de tecla 104 = R13    R13     KP_1      End
código de tecla 119 = L10    L10     SunXK_Cut
código de tecla 30 = L1      L1      SunXK_Stop
código de tecla 29 = Left    R10     KP_4
código de tecla 53 = Right   R12     KP_6

código de tecla 118 = Control_R
agregar control = Control_R
```

SPARC *Cómo deshacer el remapeo del teclado*

Existen dos maneras de conmutar las teclas a su configuración original. La primera consiste en abandonar el software de OpenWindows y empezarlo de nuevo. El segundo método, que es preferible si desea volver a cambiar las teclas periódicamente, consiste en crear un segundo guión e iniciarlo siempre que quiera volver a cambiar.

Siga estas instrucciones para crear el segundo guión:

- 1. Use cualquier editor para crear un archivo llamado `nolefty.data`.**
Éste debe estar en el mismo directorio que contenga el archivo de comandos `lefty.data`.
- 2. Escriba en el archivo de comandos tal como se muestra en la Tabla B-2 “El archivo de comandos `nolefty.data`”.**
Cualquier línea que lleve delante un signo de exclamación es una línea de comentario y no ejecuta ninguna operación.
- 3. Guarde los cambios y salga del editor.**
- 4. Tras el indicador, escriba:**

```
$ xmodmap nolefty.data
```

Para que el archivo `nolefty.data` surta efecto, debe introducir el comando anterior en el mismo directorio que el archivo de comandos.

Tabla B-2 El archivo de comandos `nolefty.data`

```
!  
! nolefty.data  
!  
! Datos de xmodmap para restaurar las teclas de función a la derecha y a la izquierda,  
! después de ser establecidas para el teclado tipo 4 de Sun para personas zurdas. Para  
! usar estos datos escriba lo siguiente, donde <filename> es el nombre de este archivo:  
!  
! xmodmap <filename>  
!
```

Tabla B-2 El archivo de comandos `nolefty.data`

!Reasignar valores predeterminados a las teclas de función de la izquierda

!

código de tecla 10 = L2	L2	SunXK_Again
código de tecla 32 = L3	L3	SunXK_Props
código de tecla 33 = L4	L4	SunXK_Undo
código de tecla 56 = L5	L5	SunXK_Front
código de tecla 58 = L6	L6	SunXK_Copy
código de tecla 79 = L7	L7	SunXK_Open
código de tecla 80 = L8	L8	SunXK_Paste
código de tecla 102 = L9	L9	SunXK_Find
código de tecla 104 = L10	L10	Sun XK_Cut

!

! Reasignar valores estándar a las teclas de función de la derecha.

!

código de tecla 28 = R1	R1	Pause
código de tecla 29 = R2	R2	Print
código de tecla 30 = R3	R3	Scroll_Lock Break
código de tecla 52 = R4	R4	KP_Equal
código de tecla 53 = R5	R5	KP_Divide
código de tecla 54 = R6	R6	KP_Multiply
código de tecla 75 = R7	R7	KP_7 Home
código de tecla 77 = R9	R9	KP_9 Prior
código de tecla 98 = Left	R10	KP_4
código de tecla 100 = Right	R12	KP_6
código de tecla 119 = R13	R13	KP_1 End
código de tecla 121 = R15	R15	KP_3 Next

!

! Reasignar la tecla de Fin de línea como tal y eliminarla del mapa de control.

Tabla B-2 El archivo de comandos `nolefty.data`

!

quitar control = Control_R

5código de tecla 118 = Fin de línea



Cómo ejecutar aplicaciones en red



Este apéndice describe una característica avanzada del entorno OpenWindows que le permitirá ejecutar aplicaciones que residan en otro sistema de su red.

Nota – La mayoría de los usuarios no necesitarán leer este apéndice. Si quiere explorar la posibilidad de ejecutar aplicaciones en red, puede hablar con su administrador de sistemas acerca de las aplicaciones especiales que podrían estar disponibles en su red.

Normalmente, en el ambiente OpenWindows todas las aplicaciones de su pantalla (como por ejemplo la Herramienta de Correo y el Calendario) son programas que se ejecutan en su sistema local. Sin embargo, si su estación de trabajo forma parte de una red, puede ejecutar aplicaciones en otro sistema y verlas en su pantalla local. Al ejecutar una aplicación de esta forma evita ciclos de cpu en su sistema local, y tiene acceso a una red completa de aplicaciones.

En este apéndice se describe el escenario más simple para ejecutar una aplicación en un sistema remoto y verla en su pantalla local. Debido a que su entorno de computación puede variar, tendrá que seguir con flexibilidad estas instrucciones. La sección “Más acerca de la seguridad”, proporciona información adicional acerca de las dificultades que conlleva el ejecutar aplicaciones en red.

Para usar el siguiente procedimiento de ejecución de una aplicación remota, es necesario que se cumplan estos requisitos:

- Vd. debe tener derechos de acceso al sistema remoto.

- Su directorio personal debe ser montable en NFS en el sistema remoto.
- La aplicación y las bibliotecas apropiadas deben estar instaladas en el sistema remoto o *computador principal*.

Si no entiende estos requisitos contacte con su administrador de sistemas.

Uso de `rlogin` para ejecutar una aplicación en red

La clave para ejecutar una aplicación en red que está en un sistema remoto consiste en asegurarse de que sus variables de entorno están establecidas correctamente:

- La variable de entorno `HOME` de su shell correspondiente en el sistema remoto debe estar establecida en su directorio de usuario.
- La variable de entorno `DISPLAY` de su shell correspondiente en el sistema remoto debe estar establecida en su pantalla local.
- Si las bibliotecas OpenWindows no han sido instaladas en los directorios estándar de la biblioteca compartida `/usr/lib` o `/usr/local`, debe establecer la variable de entorno `LD_LIBRARY_PATH` en el directorio apropiado (`$OPENWINHOME/lib`).

A continuación se cita un ejemplo de la ejecución una Herramienta de comandos en un sistema remoto utilizando `rlogin`. En este ejemplo, el directorio de usuario está montado en el sistema remoto como `/home/mydirectory`, y el software OpenWindows está situado en `/usr/openwin` en el sistema remoto. Cambie las variables, *mydirectory* y *mymachine* según su planificación. Además, reemplace *cmdtool* por el nombre de la aplicación que quiera ejecutar.

```
$ rlogin remotemachine
.
.
( Los comandos siguientes se ejecutan en el sistema remoto. ) ]
.
.
$ HOME=/home/mydirectory
$ DISPLAY=mymachine:0
$ LD_LIBRARY_PATH=/usr/openwin/lib
$ /usr/openwin/bin/cmdtool &
```


Después de introducir la última línea, aparece en su pantalla una ventana de la Herramienta de comandos. Aunque puede interactuar con esta aplicación como con cualquier otra de su pantalla, de hecho, la propia Herramienta de comandos se está ejecutando en el sistema remoto.

Aunque no se obtiene una ventaja especial si se ejecuta una Herramienta de comandos de esta forma (está disponible localmente en su sistema y no utiliza una cantidad significativa de los recursos de su computador), este ejemplo le muestra cómo utilizar una aplicación remota que esté disponible.

Más acerca de la seguridad

Esta sección describe algunos conceptos fundamentales referentes a la seguridad de la red que podría encontrar útiles a medida que ejecute aplicaciones en la red, incluyendo:

- Mecanismos de control de acceso basados en el usuario y en el computador principal
- Protocolos de autorización MIT-MAGIC-COOKIE-1 y SUN-DES-1
- Cuándo y cómo cambiar el control de acceso a un servidor
- Cómo ejecutar aplicaciones de forma remota, o localmente como un usuario diferente

Quién debe leer esta sección

La configuración de seguridad predeterminada en el software de la Versión 3.3 de OpenWindows, o versiones posteriores, no necesita cambiarse excepto si ejecuta aplicaciones en alguna de las configuraciones siguientes:

- Ejecuta una aplicación enlazada con versiones de Xlib o libcps *anteriores al software de la Versión 2 de OpenWindows* o X11R4.
- Ejecuta una aplicación que está enlazada estáticamente a las bibliotecas de la Versión 2 de OpenWindows y quiere utilizar el protocolo de autorización SUN-DES-1.
- Ejecuta una aplicación en un servidor remoto.

Mecanismos de control de acceso

Un mecanismo de control de acceso es un medio para decidir qué *clientes*, o aplicaciones, tienen acceso al servidor X11. Sólo a los clientes debidamente autorizados se les permite conectar con el servidor; al resto se les deniega el acceso, y se termina el intento de conexión, enviándoles un mensaje de error.

Hay dos tipos diferentes de mecanismos de control de acceso: *basado en el usuario* y *basado en el computador principal*. Es decir, un mecanismo garantiza el acceso a una cuenta de usuario determinada, mientras que el otro garantiza el acceso a un *computador principal* o sistema particular. A no ser que la opción `-noauth` se utilice con el comando `openwin`, estarán activos tanto el mecanismo de control basado en el usuario como el basado en el computador principal. Si desea más información, consulte “Manipulación del acceso al servidor” de este capítulo.

Acceso basado en el usuario

Un mecanismo basado en el usuario, o basado en la autorización le permite conceder acceso explícito a un usuario determinado de cualquier sistema principal. El sistema cliente de ese usuario pasará los datos de autorización al servidor. Si dichos datos coinciden con los datos de autorización del servidor, al usuario se le permitirá el acceso.

Acceso basado en el computador principal

Un mecanismo basado en el computador principal es un mecanismo de uso general. Le permite conceder el acceso a un computador principal determinado, de forma que todos los usuarios de dicho computador principal puedan conectar con el servidor. Es el método más impreciso para controlar el acceso: si ese computador principal tiene acceso al servidor, entonces todos los usuarios de dicho computador principal podrán conectar con el servidor.

El mecanismo basado en el computador principal se utiliza principalmente para compatibilidad con el software anterior. Las aplicaciones enlazadas con versiones de `Xlib` o de `libcps` anteriores al software de la Versión 2 de OpenWindows o X11R4 no reconocen al nuevo mecanismo de control de acceso basado en el usuario. Para ofrecer a estas aplicaciones la posibilidad de conectar con el servidor, un usuario deberá cambiar al mecanismo de control basado en el computador principal, o bien reenlazar con las nuevas versiones de `Xlib` y de `libcps`.

Nota – Si fuera posible, las aplicaciones clientes enlazadas con versiones anteriores de `Xlib` o de `libcups` deberían reenlazarse con las nuevas versiones de estas bibliotecas para permitir la conexión con el servidor utilizando el nuevo mecanismo de control de acceso basado en el usuario.

Protocolos de autorización

En esta versión del software OpenWindows se soportan dos protocolos de autorización: `MIT-MAGIC-COOKIE-1` y `SUN-DES-1`. Se diferencian en los datos de autorización que utilizan, y se parecen en el mecanismo de control de acceso que usan. En todo momento, el servidor implementa únicamente un protocolo. El protocolo `MIT-MAGIC-COOKIE-1` con el mecanismo de control basado en el usuario es la opción predeterminada del software de OpenWindows.

`MIT-MAGIC-COOKIE-1`

El protocolo de autorización `MIT-MAGIC-COOKIE-1` se desarrolló en el Instituto de Tecnología de Massachusetts. En el inicio del servidor, se crea un *magic cookie* para el servidor y el usuario que empezó el sistema. En cada intento de conexión, el sistema cliente del usuario envía el magic cookie al servidor formando parte del paquete de conexión. Este magic cookie se compara con el magic cookie del servidor. Se permite la conexión si los magic cookies coinciden, o se deniega si no coinciden.

SUN-DES-1

El protocolo de autorización `SUN-DES-1`, desarrollado por Sun Microsystems, está basado en Secure RPC (abreviatura de Remote Procedure Call; Llamada a Procedimiento Remoto) y requiere soporte DES (abreviatura de Data Encryption Software; Software de Cifrado de Datos). La información de autorización consiste en el netname, o nombre de red de un usuario, que es independiente del sistema. Esta información es cifrada y enviada al servidor, formando parte del paquete de conexión. El servidor descifra la información, y si el nombre de red es conocido, se permite la conexión.

Este protocolo ofrece un mayor nivel de seguridad que el protocolo MIT-MAGIC-COOKIE-1. Ningún usuario puede utilizar su nombre de red independiente del sistema para acceder al servidor, sin embargo es posible que otro usuario utilice el magic cookie para acceder al servidor.

El apartado “Cómo permitir el acceso cuando se utilice SUN-DES-1”, de este capítulo, explica cómo permitir el acceso a su servidor para otro usuario, añadiendo el el nombre de red de ellos a la lista de acceso de su servidor.

Cómo cambiar el protocolo de autorización predeterminado

El protocolo de autorización predeterminado, MIT-MAGIC-COOKIE-1, puede cambiarse a SUN-DES-1, el otro protocolo de autorización soportado, o bien cambiar para no utilizar ningún mecanismo de control de acceso basado en el usuario. Puede cambiar el valor predeterminado mediante opciones del comando `openwin`. Por ejemplo, para cambiar el valor predeterminado de MIT-MAGIC-COOKIE-1 a SUN-DES-1, empiece el software OpenWindows de esta forma:

```
$ openwin -auth sun-des
```

Si tiene que ejecutar el software OpenWindows sin mecanismo de acceso basado en el usuario, utilice la opción `-noauth` en la línea de comandos:

```
$ openwin -noauth
```



Atención– Si utiliza `-noauth` se debilita la seguridad. Es equivalente a ejecutar el software OpenWindows únicamente con el mecanismo de control de acceso basado en el computador principal; el servidor desactiva el mecanismo de control de acceso basado en el usuario. Cualquier persona que pueda ejecutar aplicaciones en el sistema local de Vd., tendrá permiso de acceso a su servidor.

Manipulación del acceso al servidor

A no ser que la opción `-noauth` se utilice con `openwin` (consulte “Cómo cambiar el protocolo de autorización predeterminado”), estarán activos tanto el mecanismo de control de acceso basado en el usuario como el basado en el computador principal. El servidor verifica primero el mecanismo basado en el usuario, y a continuación el mecanismo basado en el computador principal. La configuración de seguridad predeterminada utiliza `MIT-MAGIC-COOKIE-1` como el mecanismo basado en el usuario, y una lista vacía como el mecanismo de control basado en el computador principal. Debido a que la lista basada en el computador principal está vacía, únicamente será efectivo el mecanismo basado en el usuario. La utilización de la opción `-noauth` le ordena al servidor que desactive el mecanismo de control de acceso basado en el usuario, e inicializa la lista basada en el computador principal, agregando el computador principal local.

Existen tres programas con los que puede cambiar el mecanismo de control de acceso de un servidor: `xhost`. Estos programas acceden a dos archivos binarios creados por el protocolo de autorización. Estos archivos contienen datos de autorización específicos de la sesión. Un archivo es de uso interno del servidor, y el otro está colocado en el directorio `$HOME` del usuario:

- `.Xauthority` Archivo de autorización del cliente

Utilice los programas `xhost` y `xauth` para cambiar en el servidor la lista de acceso basada en el computador principal. Puede agregar o borrar computadores principales de la lista de acceso. Si ha empezado con la configuración predeterminada -con la lista de acceso basada en el computador principal que está vacía- y utiliza `xhost` para agregar un nombre de sistema, reducirá el nivel de seguridad. El servidor permitirá el acceso al computador principal que ha agregado, así como a cualquier usuario que especifique el protocolo de autorización predeterminado. Consulte “Acceso basado en el computador principal” si desea una explicación de porqué el mecanismo de control de acceso basado en el computador principal es considerado un nivel de seguridad inferior.

El programa `xauth` accede a los datos de autorización en el archivo `.Xauthority` del cliente. Puede extraer esta información de su archivo `.Xauthority` para que otro usuario pueda mezclar esos datos en su archivo `.Xauthority`, permitiéndole así acceso a su servidor, o bien al servidor al que esté Vd. conectado.

Consulte “Cómo permitir el acceso cuando se utilice MIT-MAGIC-COOKIE-1” si desea ejemplos de cómo utilizar `xhost` y `xauth`.

Archivo de autorización del cliente

El archivo de autorización del cliente es `.Xauthority`. Contiene entradas con este formato:

```
connection-protocol      auth-protocol      auth-data
```

Como valor predeterminado, `.Xauthority` contiene MIT-MAGIC-COOKIE-1 como el `auth-protocol`, y entradas para la pantalla local como `connection-protocol` y `auth-data`. Por ejemplo, en el computador principal `anyhost`, el archivo `.Xauthority` podría contener las entradas siguientes:

```
anyhost:0  MIT-MAGIC-COOKIE-1  82744f2c4850b03fce7ae47176e75
localhost:0  MIT-MAGIC-COOKIE-1  82744f2c4850b03fce7ae47176e75
anyhost/unix:0  MIT-MAGIC-COOKIE-1  82744f2c4850b03fce7ae47176e75
```

Cuando se inicia el cliente, se lee una entrada correspondiente al `connection-protocol` en `.Xauthority`, y el `auth-protocol` y el `auth-data` se envían al servidor formando parte del paquete de conexión. En la configuración predeterminada, `xhost` muestra listas de acceso basadas en el computador principal que están vacías e informan que la autorización está activa.

Si ha cambiado el protocolo de autorización del valor predeterminado a SUN-DES-1, las entradas en `.Xauthority` contendrán SUN-DES-1 como `auth-protocol` y el nombre de red del usuario como `auth-data`. El nombre de red tiene el formato siguiente:

```
unix.userid@NISdomainname
```

Por ejemplo, en el computador principal *anyhost*, el archivo `.xauthority` podría contener las entradas siguientes, donde `unix.15339@EBB.Eng.Sun.COM` es el nombre de red del usuario, que es independiente del sistema:

<code>anyhost:0</code>	<code>SUN-DES-1</code>	<code>"unix.15339@EBB.Eng.Sun.COM"</code>
<code>localhost:0</code>	<code>SUN-DES-1</code>	<code>"unix.15339@EBB.Eng.Sun.COM"</code>
<code>anyhost/unix:0</code>	<code>SUN-DES-1</code>	<code>"unix.15339@EBB.Eng.Sun.COM"</code>

Nota – Si no conoce su nombre de red, o nombre de red independiente del sistema, consulte a su administrador de sistemas.

Cómo permitir el acceso cuando se utilice MIT-MAGIC-COOKIE-1

Si utiliza el protocolo de autorización MIT-MAGIC-COOKIE-1, siga estos pasos para permitir a otro usuario el acceso a su servidor:

- 1. En el sistema donde se ejecute el servidor, utilice `xauth` para extraer una entrada correspondiente a `hostname:0` en un archivo.**

En este ejemplo, `hostname` es *anyhost* y el archivo es `xauth.info`:

```
$ $OPENWINHOME/bin/xauth nextract - anyhost:0 > $HOME/xauth.info
```

Observe que por la longitud de la línea, ésta se divide en una segunda línea; de todas formas, debe aparecer una única línea cuando la escriba en su pantalla.

- 2. Envíe el archivo que contenga la entrada del usuario que solicite acceso (utilice la Herramienta de Correo, `rcp` o cualquier otro método de transferencia de archivos).**

Nota – Enviar por correo electrónico el archivo que contenga su información de autorización es un método más seguro que utilizar `rcp`. Si utiliza `rcp`, *no* sitúe el archivo en un directorio al que pueda acceder fácilmente otro usuario.

- 3. El otro usuario debe introducir dicha entrada en su archivo `.Xauthority`.**
En este ejemplo, *userhost* introduce `xauth.info` en su archivo `.Xauthority`:

```
$ $OPENWINHOME/bin/xauth nmerge - < xauth.info
```

Nota – El valor *auth-data* es específico para la sesión; por tanto, será válido únicamente hasta que el servidor sea reiniciado.

Cómo permitir el acceso cuando se utilice SUN-DES-1

Si está utilizando el protocolo de autorización SUN-DES-1, siga estos pasos para permitir el acceso a su servidor a otro usuario:

- 1. En el sistema donde se ejecute el servidor, utilice `xhost` para que el servidor reconozca al nuevo usuario.**

En este ejemplo, para permitir al nuevo usuario *somebody* ejecutar en *myhost*:

```
$ xhost + somebody@
```

- 2. El nuevo usuario debe utilizar `xauth` para agregar la entrada en su archivo `.Xauthority`.**

En este ejemplo, el nombre de red independiente del sistema para el nuevo usuario es `unix.15339@EBB.Eng.Sun.COM`. Observe que este comando se debe escribir en una línea sin retorno de carro. Después del símbolo del canal de comunicación, deje un espacio en blanco y escriba el resto del comando.

```
$ echo 'add myhost:0 SUN-DES-1 "unix.15339@EBB.Eng.Sun.COM" ' |
$OPENWINHOME/bin/xauth
```

Observe que a causa de la longitud de la línea, ésta se divide en una segunda línea; de todas formas, deberá aparecer en una única línea cuando escriba en la pantalla.

Cómo procesar clientes remota o localmente como otro usuario

Los clientes X utilizan el valor de la variable de entorno `DISPLAY` para obtener el nombre del servidor al que deben conectarse.

Para ejecutar clientes remota o localmente como otro usuario, siga estos pasos:

1. En el sistema donde se ejecute el servidor, permita el acceso a otros usuarios.

Dependiendo del protocolo de autorización que utilice, siga los pasos indicados en “Cómo permitir el acceso cuando se utilice MIT-MAGIC-COOKIE-1” o bien en “Cómo permitir el acceso cuando se utilice SUN-DES-1”.

2. Establezca `DISPLAY` en el nombre del computador principal donde se ejecute el servidor.

En este ejemplo, el computador principal es *remotehost*:

```
$ DISPLAY=remotehost:0
```

3. Ejecute el programa del cliente de esta forma.

```
$ client_program&
```

El cliente se mostrará en el sistema remoto, *remotehost.t*.

SPARC - Interconexión de redes DECnet (DNI)



Este apéndice explica cómo interconectar el ambiente OpenWindows y el ambiente DECwindows™ mediante el protocolo de transporte NSP DECnet.

SPARC – Observe que todo este capítulo perteneciente a “ Interconexión de redes DECnet” se refiere sólo a sistemas basados en SPARC. La interconexión de redes DECnet también es disponible sólo con DNI 8.x.

Existen dos posibles situaciones de DNI:

- Ejecutar un cliente X11 en un sistema VAX (bajo el sistema operativo VMS®) y mostrar la ventana del cliente en un sistema OpenWindows
- Ejecutar un cliente X11 en un sistema OpenWindows y mostrar la ventana del cliente en un sistema VAX

Estas dos situaciones se describen en las próximas secciones tras una sección inicial en la que se explica cómo instalar el software de DNI para cada situación.

Cómo instalar la interconexión de redes DECnet

Para instalar la interconexión de redes DECnet siga estos pasos:

1. Active una conexión mediante DNI.

Las bibliotecas del servidor y cliente de OpenWindows utilizan una versión de la biblioteca de transporte de DNI `libdni` que se puede cargar de forma dinámica. Debe establecer la variable de entorno `DNI_X_ENABLE` en el directorio donde esté instalado `libdni.so` para que las bibliotecas del cliente y el servidor carguen `libdni`.

El ejemplo siguiente asume que se ha cargado DNI por medio de `pkgadd` en la posición predeterminada:

```
$ DNI_X_ENABLE=/opt/SUNWconn/dni/lib
```

2. Empiece el servidor de OpenWindows.

Como valor predeterminado, el servidor OpenWindows soporta la seguridad "MIT-MAGIC-COOKIE". Este mecanismo de seguridad se basa en el usuario en vez de en el computador principal, Vd. decide qué usuarios pueden conectar con el servidor en vez de qué sistemas pueden hacerlo. En el modo predeterminado, el comando `xhost` devuelve una lista vacía, e indica solamente que la seguridad está activada. Puede desactivar este modo de seguridad (y volver al modo de seguridad de las versiones anteriores del servidor OpenWindows) utilizando la opción `-noauth` con el comando `openwin`.

```
$ openwin -noauth
```

3. Solicite al propietario del sistema que procesa el software de OpenWindows la utilización del comando `xhost` a fin de dar permiso a DEC® VAX® para que tengan una conexión X11 con el servidor de OpenWindows.

Para que los clientes X11 conecten con el servidor de OpenWindows por medio del software de DNI, las direcciones de nodo DECnet deben estar mapeadas según sus nombres de nodo DECnet. Esto se realiza creando e inicializando la base de datos NCP. Esto también debe hacerse en el sistema DEC VAX.

```
$ xhost decvax:::
```

El doble signo de dos puntos especifica el transporte DECNet.

Cómo mostrar un cliente remoto en un sistema OpenWindows

Puede procesar clientes X11 de VMS utilizando el comando SunLink DNI `dnilogin` para conectar con el sistema VAX. En primer lugar, establezca la variable de entorno `DISPLAY` de su sistema local para que sea el servidor X11 del sistema remoto. Luego ejecute un cliente X11 introduciendo el nombre de dicho cliente, representado aquí por `x11_client`. Si desea más información acerca del uso del sistema operativo VMS, consulte *VMS DECwindows User's Guide, Running Applications Across the Network*.

Por ejemplo:

```
$ dnilogin decvax
.
.
.
$ define DECW$DISPLAY OW_machine::0
$ spawn/nowait run x11_client
```

Cómo mostrar un cliente remoto en un VAX

Puede ejecutar clientes X11 en un sistema OpenWindows y mostrarlos en un servidor DECwindows estableciendo la variable `display` `DISPLAY` al sistema remoto VAX.

Antes de que pueda ejecutar cualquiera de los clientes X11, debe compilar e instalar las fuentes OpenWindows en el servidor DECwindows. Dichas fuentes están disponibles en la versión MIT X11R4 o en el paquete opcional de fuentes suministrado junto al software de la Versión 3.3 de OpenWindows. Siga estos pasos para instalar las fuentes adecuadas en el servidor DECwindows:

- 1. Instale las fuentes opcionales de OpenWindows (de las fuentes de la versión MIT X11R4) en el sistema OpenWindows.**
- 2. Si desea leer las instrucciones de instalación de las fuentes, consulte la *OpenWindows Server Programmer's Guide*.**
- 3. Copie las fuentes en un directorio del sistema VAX.**

```
$ cd $OPENWINHOME/share/src/fonts/misc
$ dnicp *.bdf `decvax::[vaxdir]`
```

4. Compile las fuentes de cursor del sistema VAX.

Se obtendrán archivos como: `olcursor.decw$font;1`
`olglyph10.decw$font;1...`

```
$ font olcursor.bdf
$ font olglyph10.bdf
...
```

5. Copie las fuentes en el directorio `sysfont`:

```
$ set def sys$sysroot:[sysfont.decw.user_cursor16]
$ copy [vaxdir]olcursor.decw$font;1 *
```

Nota – Para copiar las fuentes en el directorio `sysfont`, Vd. debe estar conectado como “sistema” en el sistema DEC VAX.

- 6. También debe seguir del paso 2 al 4 para el resto de las fuentes de cursor y para las fuentes Lucida en `$OPENWINHOME/share/src/fonts/75dpi` y `$OPENWINHOME/share/src/fonts/100dpi`.**

Nota – Las fuentes Lucida deben instalarse en `sys$sysroot:[sysfont.decw.user_75dpi]` y `sys$sysroot:[sysfont.decw.user_100dpi]`.

La lista siguiente muestra el número mínimo de fuentes de trabajo a instalar necesario para ejecutar las herramientas Deskset de OpenWindows. Si esta utilizando fuentes predeterminadas para las aplicaciones, sólo debe instalar dichas fuentes. Sin embargo, puede instalar más fuentes si es necesario.

- `olcursor.bdf`
- `olglyph10.bdf`
- `olglyph12.bdf`
- `olglyph14.bdf`
- `olglyph19.bdf`
- `luBS08.bdf`
- `luBS10.bdf`
- `luBS12.bdf`
- `luBS14.bdf`
- `luRS08.bdf`
- `luRS10.bdf`
- `luFS12.bdf`
- `lutBs12.bdf`
- `lutRS10.bdf`
- `lutRS12.bdf`

7. Reinicie de nuevo el servidor DECwindows.

8. Puede comprobar que las fuentes están instaladas listando las fuentes existentes en el servidor DECwindows:

```
$ DISPLAY=decvax:0
$ xlsfonts | grep Sun      (Fuentes de cursor)
$ xlsfonts | grep Lucida
```

9. Asegúrese que ha concedido permiso al nodo de OpenWindows para que se muestre en el servidor DECwindows, utilizando el menú Seguridad del Administrador de sesión DECwindows.

10. Ejecute una aplicación X11 (por ejemplo una herramienta Deskset de OpenWindows).

```
$ DISPLAY=decvax::0
$ mailtool
```

Nota - `DNI_X_ENABLE` debe estar establecido en la posición de la biblioteca de transporte DNI `libdni`. Lea el paso 1 de “Cómo instalar la interconexión de redes DECnet,” de este capítulo.

Si se imprime un mensaje de error como el siguiente, tendrá que instalar dicha fuente en el servidor DECwindows para procesar la aplicación.

```
XView aviso: No puede cargarse fuente '-b&h-lucida-medium-r-*-*--80-*--*--*--*--*' (Paquete de fuente)
```

Este mensaje de error significa que debe instalarse la fuente `luRS10.bdf`.

Si desea más información acerca de las fuentes, consulte la *OpenWindows Server Programmer's Guide*.

Configurar impresoras y módems



Configurar un módem para su sistema

Un *módem* es un dispositivo que permite a su sistema transmitir y recibir información a través de las línea telefónicas. Puede configurar con facilidad un módem en su sistema usando la Herramienta de administración gráfica. Siga para ello estos pasos:

1. Conecte el módem.

Ejecute las siguientes tareas para conectar un módem a su sistema:

- Conecte físicamente el módem al sistema o instálelo (si tiene una tarjeta de módem).
- Configure los enchufes y/o la velocidad en baudios, puerto y otras características del módem.
- Enchufe el módem o su adaptador a una toma de corriente en caso necesario.

Consulte la documentación del suministrador del módem, y la documentación de instalación para su sistema para saber si ha de ejecutar las tareas explicadas.

2. Entre en el sistema como raíz.

A menos que sea Vd. miembro del grupo de administración de sistemas especial de UNIX (GID 14), tendrá que acceder al sistema como *raíz* para usar la Herramienta de administración. Raíz es un usuario de sistema con permisos especiales para modificar sistemas de archivos.

Use el comando `su` para acceder al sistema como raíz::

```
$ su
Password:      (entre aquí la contraseña para raíz)
#
```

Si la cuenta de entrada de raíz está protegida por una contraseña, tendrá que conocer la contraseña de raíz. Si no es el caso, presione simplemente retorno tras el indicador de contraseña. Si no conoce la contraseña de raíz contacte con su administrador de sistemas para obtener ayuda.

3. Empezar la Herramienta de administración.

Escriba:

```
# /usr/bin/admintool &
#
```

La Herramienta de administración aparecerá a los pocos segundos.

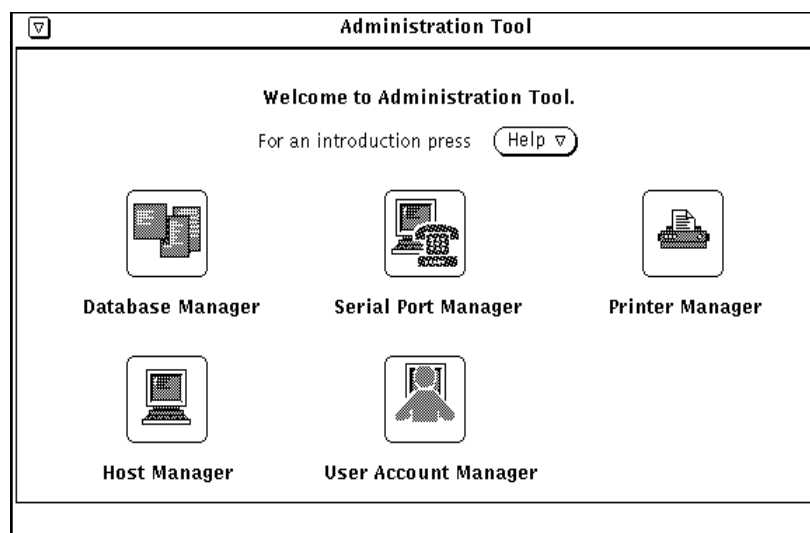


Ilustración E-1 La Herramienta de administración

4. Selecciona la Serial Port Manager.

Para empezar la Serial Port Manager, haga clic sobre el icono de Serial Port Manager en la ventana de la Herramienta de administración. Esto abre una nueva ventana con controles y opciones para administrar módems y terminales.

5. Siga las instrucciones de la documentación en línea.

La ayuda en línea del Serial Port Manager ofrece instrucciones detalladas sobre cómo usar el Serial port Manager para instalar un módem. Para ver las instrucciones de Ayuda, seleccione Ayuda desde la ventana del Serial Port Manager.

Para información adicional sobre módems ...

Si desea más información sobre cómo instalar y configurar módems, consulte el manual *Peripherals Administration* que es parte del *Solaris 2.4 System Administrator AnswerBook*.

Configurar una impresora para usar en su sistema

La forma más fácil de configurar una impresora es usar la Herramienta de administración gráfica. Las instrucciones a continuación explican cómo configurar una impresora local para uso en su sistema personal. Si desea instrucciones sobre cómo configurar las impresoras para la red, consulte el manual *Peripherals Administration* que es parte del *Solaris 2.4 System Administrator AnswerBook*.

A continuación le explicamos los pasos necesarios para configurar una impresora local:

1. Conecte la impresora.

Las siguientes tareas son, por regla general, necesarias para conectar la impresora a su sistema:

- Conecte físicamente la impresora al sistema
- Configure cualquier enchufe y/o la velocidad en baudios, puerto y otras características a la impresora.
- Enchufe la impresora a una toma de corriente.

Puede conectar la impresora al sistema y encenderla antes o después de usar la Herramienta de administración para agregarla. Consulte el manual de impresora de su distribuidor, y el manual de instalación de su sistema para obtener la información necesaria sobre los enchufes y cables necesarios. Normalmente, se conectan los cables a un puerto en serie, pero en algunos casos, y dependiendo de los requisitos de la impresora, se puede usar un puerto paralelo.

2. Entre en el sistema como raíz.

A menos que sea Vd. miembro del grupo de administración de sistemas especial de UNIX (GID 14), tendrá que acceder al sistema como *raíz* para usar el Administration Tool. Raíz es un usuario de sistema con permisos especiales para modificar sistemas de archivos.

Use el comando `su` para acceder al sistema como raíz::

```
$ su
Password:      (entre aquí la contraseña para raíz)
#
```

Si la cuenta de entrada de raíz está protegida por una contraseña, tendrá que conocer la contraseña de raíz. Si no es el caso, presione simplemente retorno tras el indicador de contraseña. Si no conoce la contraseña de raíz, contacte con su administrador de sistemas para obtener ayuda.

3. Empezar la Herramienta de administración.

Escriba:

```
# /usr/bin/admintool &
#
```

Administration aparecerá a los pocos segundos.(Vea la Ilustración E-1 en la página 201).

4. Seleccione el Printer Manager.

Para empezar el Printer Manager, haga clic sobre el icono Printer Manager en la ventana de la Herramienta de administración. Esto abre una nueva ventana que controla las opciones para administración de impresoras.

5. Siga las instrucciones de la ayuda en línea.

La ayuda en línea del Printer Manager ofrece instrucciones detalladas sobre cómo usar el Serial port Manager para instalar un módem. Las instrucciones de ayuda las encontrará en Ayuda ...de la ventana Printer Manager.



Para información adicional sobre impresoras ...

Si desea información adicional sobre la administración de impresoras, consulte el manual *Peripherals Administration* que es parte de *Solaris 2.4 System Administrator AnswerBook*.

Índice

Símbolo

. . símbolo, 36

A

administrador del sistema, 1

alias, 156

aplicación

en red

con DNI (DECnet), 193

en un sistema remoto, 182

aplicaciones en red

con DECnet, 193

apropos comando, 25

archivo

.cshrc, 152

.login, 152

.mailrc, 120

.profile, 152

/etc/aliases, 122

/etc/hosts.equiv, 147, 149

/etc/passwd, 147, 149, 151

/etc/profile, 152

buscar, ?? a 44

cambiar permisos, 47 a 50

comprobar tipo, 32

copiar, 29

crear nuevo, 29

definición de, 27 a 28

establecer permisos

predeterminados, 159 a 160

establecer valores

predeterminados, 159 a 160

guardar correo en, 115

imprimir, 130

inicialización, 151 a 152

insertar en una carta, 114

listado invisible, 46

mostrar contenido, 31 a 32

permisos, 44 a 45

seguridad, 44 a 53

y el diff comando, 39 a 41

archivo mbox

Consulte también correo

archivos de inicialización, 151 a 152

archivos de punto

.xinitrc, 169

archivos de texto, 27

archivos ejecutables, 28

archivos ocultos

.xinitrc, 169

archivos punto, 46

ayuda en línea, 23

B

`bdiff` comando, 41

borrar

 correo, 107 a ??

 vi texto con `ex` comandos, 90

botones de remapeo del ratón, 173

Bourne shell

 y archivos de inicialización, 151 a 152

buscando

 con buscar, 41 a ??

buscar

 con `grep`, 55 a 62

buzón, 102

 ajustar la ruta para, 154

Consulte también correo

C

C shell

 e historial de comando, 20 a ??

 y alias, 156

 y archivos de inicialización, 151 a 152

 y comando `history`, ?? a 21

cadena de caracteres, 90

cadena de signos, 55

cambiar directorios, 34 a 36

canal de comunicación

 comando de salida, 22

 enviar correo al comando `lp`, 109

cancelar comando, 138

carpetas, 116 a 119

caso de sensibilidad

 y comandos, 18

 y el `vi` editor, 90

`cat` comando, 32

`cd` comando, 28, 34 a 36

`CDPATH` variable, 153

`chmod` comando, 47, 50

 y `umask`, 160

cola de espera de impresión, 136

comando abrir línea, 81

comando agregar, 81

comando buscar, 41 a ??

comando `env`, 153

comando `finger`, 110

comando `history`, ?? a 21
 definir, 153

comando insertar, 81

comando `mkdir`

 y correo, 116

comando `put`, 86

comando `rusers`, 110

comando `umask`, ?? a 160

comando `who`, 109

comando `yank`, 85

comandos

 ejecutar en una máquina remota, ?? a 149

 entrar `long`, 19

 entrar `multiple`, 19

 procesándose en el fondo, 23

 redireccionar salida, ?? a 21

 repetir anterior, 20 a ??

 repetir `vi`, 87

 tilde, 125

 y caso de sensibilidad, 18

 y determinar funcionalidad, 24

 y opciones, 21

 y palabras clave en colores, 25

 y redireccionar salida, 21 a 22

 y salida del canal de
 comunicación, 22

 y sintaxis, 24

comandos de tilde, 125

comandos SunOS, xiv

comillas, 62

`command prompt`

 definir, 157

Cómo entrar, 1

compatibilidad con SunView, 167

`.defaults` archivo (SunView), 167

`.Xdefaults` archivo, 167

`convert_to_Xdefaults`

 programa, 168

concatenado, *Consulte* `cat` comando, 32

-
- conectar con el sistema
 - de forma remota, 142 a 146
 - utilizando el nombre de otro usuario, 144
 - conexiones remotas, *Consulte* `rlogin` comando
 - contraseña, 1
 - cambiar, 64
 - con fecha de caducidad, 65 a 66
 - seleccionar, 63
 - Contraseñas, procesos y almacenamiento en disco, 63
 - `convert_to_xdefaults` programa, 168
 - copia recursiva, 38
 - copiar
 - a un sistema remoto, 148
 - archivos, 29
 - correo a archivo, 116
 - correo en carpetas, 116 a 117
 - de una máquina remota, 147
 - directorios, 38
 - líneas entre `vi` archivos, 95
 - `vi` texto, 85
 - `vi` texto con `ex` comandos, 88
 - copias, 112
 - copias ciegas, 112
 - correo
 - alias
 - definición de, 119
 - en `.mailrc` archivo, ?? a 122
 - archivo mbox, 102
 - ayuda, 127
 - cancelar cartas que no se han enviado, 112
 - carpetas, 116 a 119
 - comandos de tilde, 125
 - copias, 112
 - copias ciegas, 112
 - determinar una dirección del usuario, 110
 - eliminar cartas, 107 a 109
 - empezar, 102
 - enviar, 109 a 111
 - varios receptores, 110
 - guardar, 115 a 117
 - imprimir, 109
 - insertar otra carta, 113
 - insertar un archivo, 114
 - leer, 104, 106 a 107
 - mostrar listado de cartas, 107
 - responder a, 114
 - salir, 105
 - usar `vi` con, 119
 - verificar versión, 104
 - y el buzón, 102
 - correo alias, 119
 - Consulte también* correo
 - correo electrónico, *Consulte* correo
 - `cp` comando, 29, 38
 - archivo.cshrc, 152 a 160
- ## D
- default
 - file permissions, 159 a ??
 - default ring on menus, xvi
 - `.defaults` archivo, 167
 - `df` comando, 68
 - `diff` comando, 39 a 41
 - `diff3` comando, 41
 - directorio
 - cambiar, 34 a 36
 - cambiar permisos, 47 a 50
 - comprobar utilización, 68
 - copiar, 38
 - de usuario, 34
 - `dead.letter`, 111
 - definición de, 28
 - eliminar, 39
 - establecer permisos
 - predeterminados, ?? a 160
 - establecer una carpeta, 116
 - jerarquía, 33 a 34
 - mostrar actual, 34
 - mover, 38
 - raíz, 33
 - renombrar, 38

seguridad, 44 a 53
y nombre de ruta de acceso, 33
directorio de usuario, 28
y el método abreviado tilde, 35
y rlogin comando, 143
directorio maestro, 36
directorio personal, 34
definir, 154
directorio raíz, 33
directory
setting default permissions, 159 a ??
DISPLAY variable de entorno, 191
displaying
users on your file server, 109
displays, starting OpenWindows with
more than one monitor, 14
ditroff programa, 71
du comando, 69

E

echo, 2, 64
eliminar
correo, ?? a 109
directorios, 39
vi texto, 84
email, *Consulte* correo
empezar
compatibilidad con SunView, 167
pantallas múltiples, cómo
procesar, 16
ejemplos, 15 a ??
tecla componer,
desactivar/activar, 173
entrada
casos especiales, 12
entrar en el sistema
con windows, 12
enviar por canal de comunicación
salida mediante grep, 56
enviar por el canal de comunicación
du salida mediante sort, 69
ps salida mediante grep, 67

escapes de tilde, *Consulte* comandos de tilde

ex comandos, 75, 87 a 90
expresiones regulares, 59

F

fecha comando, 18
file
searching for, 41 a ??
file comando, 32
folder de correo, 116
Consulte también folder
fuentes
cambiar valor por omisión, 161
de las aplicaciones en red, 197
listar, 164

G

grep comando
como filtro, 56
sintaxis para, 55 a 56
y cadenas de signos, 58
y expresiones regulares, 59 a 60
y metacaracteres, 60 a 61

H

Herramienta de Comandos
en un sistema remoto, 182
historial comando, 20
historial de comando, 20 a ??
HOME variable, 154

I

impresoras
puerto
paralelo versus serie, 202
imprimir
a una impresora específica, 130 a 131
cancelar solicitudes, 138 a 140
comprobar disposición de
impresora, 134

comprobar estado de, 133 a 136
 copias múltiples, 132
 correo, 109
 en la impresora predeterminada, 130
 número de solicitud id, 139
 vi archivos, 77
 y determinar tipo de impresora, 136
 y solicitar un aviso cuando la
 impresión esté
 completada, 131

Indicador de comando, 17
 indicativo de comando
 definir, 154
 interconexión de redes, 142

K

kill comando, 67
 Korn shell
 y archivos de inicialización, 151 a 152

L

LANG variable, 154
 lenguaje local, 154
 línea de estado, 73
 archivo.login, 152
 logname comando, 19
 LOGNAME variable, 154
 lp comando, 129
 tabla de opciones, 133
 LPDEST variable, 154
 lpstat comando, 133 a 138
 sintaxis para, 137
 tabla de opciones, 138
 ls comando, 29
 y listar archivos invisibles, 46
 y opción de formato largo, 45 a 46

M

mail
 alias
 en /etc/aliases, 122 a ??
 en archivo.mailrc, 120 a ??
 tabla de comparación, 125
 alias
 in /etc/aliases, ?? a 125
 and undeliverable letters, 111
 copy command, 116
 MAIL variable, 154
 archivo.mailrc, 120
 mailx programa
 Consulte también correo
 man comando, 3, 24
 man pages, *Consulte* páginas del manual
 de referencias
 MANSECTS variable, 154
 mbox archivo, 102
 memoria intermedia con nombre, 86
 metacaracter, 30
 metacaracteres, 59
 mkdir comando, 37
 módem
 (definido), 199
 modo comando, 73, 74
 modo entrada, 73
 more comando, 31
 mostrar
 contenido de archivo, 31 a 32
 estado de impresora, 134 a 136
 listado del buzón, 107
 permisos de archivo, 45
 su posición de entrada remota, 146
 usuarios remotos, 149 a 150
 utilización del directorio, 68 a 69
 utilización del disco, 68
 mover
 archivos, 30
 directorios, 38
 vi texto, 86
 vi texto con ex comandos, 89
 multiple screens, start-up, 14
 mv comando, 30

Ñ

-noauth opción de openwin, 186
-noauth opción para openwin, 13
nohup comando, 23
nombre de entrada, 1
 definir, 154
nombre de ruta de acceso, 33
 relativo, 37
nombre de ruta de acceso relativo, 37
nombre de usuario, 1
nroff programa, 71
número de solicitud id, 139
números de identificación del proceso,
 Consulte PIDs

O

opciones, 21
opciones de comandos, 21
openwin
 -dev opción, 14, 15, 16
 -noauth opción, 13
 -noauth opción, 184, 186
 opciones de dispositivo
 opción, 14
OpenWindows
 en monitores duales, 14
 entrar y salir, 12
 seguridad con MIT-MAGIC-
 COOKIE, 194
 y archivos de inicialización, 152
 y correo, 101

P

páginas del manual de referencia, 3, 154
páginas del manual de referencias, 24
pantallas
 visualización izquierda-derecha de
 OpenWindows, 15
 visualizaciones arriba/abajo de
 OpenWindows, 16
parámetros

 establecer para vi, 96
passwd comando, 65
PATH variable, ?? a 156
perfil del sistema, 152
perfil del usuario, 152
permiso
 cambiar, 47 a 50
 categorías para, 44
 establecer absoluto, 50 a ??
 establecer predeterminado, 159 a ??
 fijar absoluto, ?? a 53
 tipo de, 44
 y el metacaracter *, 50
permiso de ejecución, 44
permiso de escritura, 44
permiso de lectura, 44
permisos
 establecer predeterminado, ?? a 160
permisos absolutos, 50 a 53
 Consulte también permiso
PIDs, 66 a 68
predeterminado
 directorio, 28
 impresora, 130
 indicador de comando, 17
 permisos de archivo, ?? a 160
imprimir
 Consulte también lp comando
procesos
 kill, 67 a 68
 procesando, 66 a 67
archivo.profile, 152 a 160
programa mailx
 salir, 105
protocolos de autorización, *Consulte*
 seguridad
ps comando, 66
PS1 variable, 154, 157
puerto en serie, 202
puerto paralelo, 202
pwd comando, 28, 34

R

ratón
 botones de remapeo, 173
r`cp` comando, 147 a 148
red
 definición de, 141
 protocolo, 142
red de área extensa, 142
red de área intermedia, 142
red de área local, 142
red, grande, xiii
relay, 142
reloj del sistema, 155
renombrar
 archivos, 30
 directorio, 38
repetir comandos, 20 a 21
r`login` comando, 142 a 146
 y abandonar una conexión, 145
 y suspender una conexión, 146
r`m` comando, 31, 39
r`mdir` comando, 39
r`sh` comando, 148 a 149
r`users` comando, 149

S

salir del sistema, 3 a ??
seguridad
 archivo.Xauthority, 187 a ??
 mecanismos de control de acceso
 definición de, 184
 MIT-MAGIC-COOKIE-1 protocolo de
 autorización, 185
 -noauth opción, 184
 protocolos de autorización, 185 a 191
 xauth programa, 190
seguridad MIT-MAGIC-COOKIE, 194
SELECT, xv
sensibilidad a letras mayúsculas o
 minúsculas
 y el `vi` editor, 78

servicio de impresión LP, *Consulte*
 imprimir
sesión de trabajo estándar, 1
shell de entrada, 2 a ??, 151
SHELL variable, 154
símbolo canal de comunicación, 22
sort comando, 69
subsistema impresora en líneas, *Consulte*
 imprimir
SunLink, red, 195

T

tecla componer, desactivar/activar, 173
teclado
 deshacer el remapeo para las
 personas zurdas, 177
 reordenar para el usuario zurdo, 174
TERM variable, 155
TERMINFO variable, 154
texto en comentario, 124
tiempo de cpu, 66
touch comando, 29
trabajos en el fondo, 23
t`roff` programa, 71
TZ variable, 155

U

umask command, 159 a ??
uso de las personas zurdas
 del ratón, 173
usuario raíz, 123
usuario zurdo
 del teclado, 174
utilizar disco, 68

V

valor por omisión
 impresora, 154
valor predeterminado
 shell, 151

variable HISTORY, 153
variable PATH, 154
variables de ambiente
 en archivos de inicialización, 152 a 156
VAX, como host (computador principal) de una aplicación cliente
 X11, 193

vi
 usar con correo, 119

vi
 acabar
 completo, 75 a 96
 archivo imprimir, 77
 buscar y reemplazar, 90 a 93
 cambiar texto en, 82 a 83
 cambiar una línea, 82
 cambiar una palabra, 82
 caso de sensibilidad en una búsqueda, 90
 comando abrir línea, 81
 comando agregar, 81
 comando insertar, 81
 comando put, 86
 comando yank, 85
 copiar texto, 85
 crear un archivo, 72
 cursor, 72
 movimiento, 78
 definición de, 71
 deshacer cambios, 83
 desplazarse en, 79
 eliminar texto, 84 a 85
 eliminar un caracter, 84
 eliminar una línea, 84
 eliminar una palabra, 84
 establecer parámetros, 96
 ex comando copiar, 88
 ex comando mover, 89
 insertar otros archivos, 94
 insertar repetido, 86
 insertar texto, 81
 ir a una línea específica, 93
 línea de estado, 72 a 73
 memoria intermedia con nombre, 86
 modo comando, 73, 74
 modo entrada, 73 a 74
 mover texto, 86
 navegar en, 78 a ??
 navigate en, ?? a 80
 numeración de las líneas, 87
 paginar por archivos, 79
 partir líneas, 83
 recuperación tras caída, 96
 reemplazar un caracter, 82
 repetir comandos, 87
 resumen de los comandos básicos, 97 a ??
 sustituir caracteres, 82
 trabajar con archivos múltiples, 94 a 96
 transponer caracteres, 83
 unir líneas, 83
 uso de un contador, 87
 y ex comandos, 75, 87 a 90

vi editor, *Consulte vi*

view comando, 71

W

what is comando, 24

X

xauth programa, 190
.Xauthority file, 187 a ??
.Xdefaults
 y aplicaciones SunView, 167
.xinitrc archivo, 169

xmodmap
 comando, 173
 y reordenación del teclado, 174