

Prácticas

1. Esquema de partición

Esta práctica tiene como objetivo determinar el mejor esquema de partición posible, que es lo que presenta la mayor dificultad para un principiante. La práctica está pensada para todas las distribuciones. Dispone en su PC de un disco de 160 GB, de los cuales 40 están ocupados por otro sistema. Su máquina dispone de 2 GB de memoria RAM. Le quedan 120 GB de espacio en disco. ¿Cómo puede repartirlos sabiendo que quiere separar sus datos y el sistema?

1. ¿Qué tamaño debe tener la partición de intercambio SWAP?

Se utiliza la partición de intercambio cuando Linux no dispone de bastante espacio en memoria RAM para trabajar con todos sus datos. Se mueven de forma temporal los datos menos urgentes a la memoria virtual (esta partición de intercambio) para liberar memoria para otros datos.

Al disponer su sistema de 2 GB de memoria, sólo puede prever 2 GB de swap. Quedan sólo 118 GB de espacio en disco para el resto.

2. ¿Que espacio reservar a la raíz de sistema /?

Incluso si tuviera que instalar todos los productos presentes en el DVD de instalación, el total no alcanzaría 10 GB. Pero debe tener en cuenta dos cosas: siempre puede añadir software procedente de otras fuentes (nuevos repositorios, instalación manual, etc.) y la actividad sobre el sistema puede tener como consecuencia que los directorios /var y /tmp crezcan muchísimo. Si dispone de bastante espacio, ¿por qué no reservar 20 o 30 GB a la raíz? Cojamos por ejemplo 20 GB.

Quedan 98 GB.

3. ¿Qué espacio reservar al /home?

La partición que contiene /home es la que contiene sus datos o la de otros usuarios: la que ocupa más espacio, entre las fotos, la música, las películas, los documentos de trabajo, etc. Reserve los 98 GB restantes. El disco está totalmente particionado.

4. ¿Es útil crear una partición extendida?

Tiene que crear tres particiones, además de la que ya existe. O sea, un total de cuatro particiones. Cuatro es el máximo de particiones primarias. Pero piense que puede necesitar reducir, suprimir o volver a crear particiones. En este caso, ya se habría alcanzado el límite. Sea prudente y establezca una partición extendida donde ubicar particiones lógicas.

5. ¿Cuál es el esquema final del disco?

Partición primaria 1: el SO ya presente, 40 GB.

Partición extendida:

- ~ Partición lógica 1: /, 20 GB.
- ~ Partición lógica 2: /home, 98 GB.
- ~ Partición lógica 3: swap, 2 GB.

Los 160 GB están particionados.

2. Gestión de los RPM y YUM

El objetivo de esta práctica es trabajar con la base de datos de paquetes RPM ya instalados en su equipo e instalar otros nuevos. El equipo o la máquina virtual debe disponer de una distribución basada en RPM (y que soporte YUM: Red Hat, Fedora).

1. Determine el número de paquetes RPM actualmente instalados en su estación de trabajo.

Respuesta:

Ejecute el comando siguiente:

```
$ rpm -qa | wc -l
```

En la máquina del autor devuelve: 1514

2. Verifique que el paquete coreutils está en su sistema, luego determine de qué sirve con la ayuda de su manifiesto. ¿Puede conseguir obtener sólo la descripción y nada más? Lea la página del manual correspondiente para ampliar sus conocimientos.

Primero, consulte la base RPM sobre este paquete para obtener la información:

```
$ rpm -qi coreutils
```

En caso de error, seguramente no se instalará el paquete. Si no, lea el contenido del campo Descripción.

Segundo, lea la sección del manual de rpm dedicada al formato de salida. El parámetro `-q` acepta un formato de salida que puede formatear con `--queryformat`. El formato se especifica así: `%{CAMPO}` con el campo en mayúsculas:

```
$ rpm -q --queryformat="%{DESCRIPTION}\n" coreutils
```

3. Intente suprimir el paquete coreutils. ¿Puede proporcionar la lista de las dependencias que le impiden hacerlo?

Intente como root:

```
# rpm -e coreutils
```

Ha obtenido la lista de todos los paquetes que impiden su desinstalación: ¡varias decenas! Observe la existencia del parámetro `-R`, que muestra de qué depende el propio paquete, y el `--provides`, que facilita el nombre de los elementos proporcionados por el paquete.

```
$ rpm -q--provides coreutils
coreutils = 8.30-6.el8
coreutils(x86-64) = 8.30-6.el8
coreutils-full = 8.30-6.el8
fileutils = 8.30-6.el8
...
```

4. Descargue el paquete RPM de JRE (Java Runtime Environment) de Java ubicado en <http://www.java.com/en/download/manual.jsp> (tome la versión correspondiente a su arquitectura). Instale este paquete indicando que quiere ver su nombre y una barra de progreso.

Instale el paquete con los parámetros `-i`, `-v` y `-h`:

```
# rpm -ivh jre-8u421-linux-x64.rpm
```

5. Si el paquete ya estuviera instalado, ¿cómo habría podido actualizarlo? Sabiendo que ya está instalado ahora, intente actualizar este paquete sin condiciones. ¿En qué caso puede ser necesario? Finalmente, suprimalo.

Puede actualizar el paquete con los parámetros `-U` o `-F`. Observe que hubiera podido instalar el paquete directamente con `-U`:

```
# rpm -Uvh jre-8u421-linux-x64.rpm
```

Pero si ya está instalada la misma versión de este paquete, eso no funcionará. Podría necesitar hacerlo si se han suprimido algunos de los archivos de este paquete: su supresión, incluso completa, no suprime el rpm de la base local. Indique la opción `--force`.

```
# rpm -Uvh --force jre-8u421-linux-x64.rpm
```

6. Instale el repositorio EPEL usando YUM y actualice el caché.

Hay un package especial que se llama `epel-release` que se encuentra presente en los repositorios de Fedora y CentOS. Instálelo con YUM:

```
# yum install epel-release
```

Actualice el caché de los repositorios:

```
# yum makecache
```

7. Haga un listado del contenido del repositorio EPEL. ¿En qué repositorio se encuentra el package `htop`? Instálelo sin que se le pida confirmación.

No se puede hacer un listado directamente del contenido de un repositorio. Hay que utilizar cuatro parámetros en contexto de yum:

- ✓ El primero desactivará el conjunto de los repositorios
- ✓ El segundo activará el repositorio EPEL
- ✓ El tercero ejecuta la acción de listado
- ✓ El último solicita solamente los que están disponibles

```
# yum --disablerepo="*" --enablerepo="epel" list -available
```

Para conocer el origen del package `htop`, se le pide a yum que muestre la información sobre este:

```
# yum info htop
```

La línea Repositorio le dará la respuesta: se trata del repositorio epel.

`htop` se instala así:

```
# yum install -y htop
```

Ya que `htop` parece que es mejor que `top`, ¿por qué no desinstalar este último? ¿De qué package proviene el comando `top`? ¿Qué verá si desea desinstalar `top` (un consejo, diga "no")? ¿Por qué?

Ubique el comando `top`:

```
# which top
/bin/top
```

Compruebe a qué package pertenece:

```
# rpm -qf /bin/top  
procps-ng-3.3.15-1.el8.x86_64
```

Ejecute yum para suprimir este package:

```
# yum remove procps-ng
```

En el servidor del autor, yum indica que va a suprimir 226 packages de dependencias. Si observa el contenido del package, verá que propone muchos comandos usados por muchos packages: `uptime`, `ps`, `watch`, `sysctl`, `pidof`, `pkill`, etc. Se trata, por lo tanto, de un package muy importante del sistema.

3. Gestión de DPKG y APT

El objetivo de esta práctica es trabajar con la base de datos de los paquetes dpkg instalados en su máquina, instalar otros nuevos y utilizar APT. Su equipo o máquina virtual debe disponer de una distribución de tipo Debian o Ubuntu.

1. Conteste a las preguntas 1 a 4 de la práctica anterior, pero con los comandos y paquetes dpkg equivalentes: coreutils está presente con el mismo nombre. Sin embargo no cuenta con el paquete Java, vamos a utilizar el paquete de Sky Enterprise (un cliente Lync/Skype Enterprise) que obtendrá aquí: <https://tel.red/linux.php>:

a - Hay que filtrar la lista de los paquetes instalados. Por defecto, dpkg facilita la lista de todos los paquetes conocidos, entre los cuales se hallan, los instalados, que empiezan por «ii»:

```
$ dpkg -l | grep ^ii | wc -l
```

En la máquina de prueba del autor, hay 1706 paquetes instalados.

b - La opción -l de dpkg acepta un filtro como parámetro:

```
$ dpkg -l "*coreutils*"
```

Es posible que encuentre dos paquetes con este nombre; por lo tanto, deberá leer los resultados o buscar una correspondencia exacta:

```
$ dpkg -l coreutils
```

Para obtener los detalles del paquete que tiene ya instalado, hay que dirigirse al manual, donde se le informa de que es posible utilizar el comando **dpkg-query** y el parámetro `-W`:

```
$ dpkg-query -W coreutils
```

Pero falta la descripción. El manual de dpkg-query proporciona una información adicional: puede modificar el formato de salida con `-f`:

```
$ dpkg-query -W -f='${Description}' coreutils
```

c - Para suprimir un paquete Debian, utilice la opción `-r`:

```
# dpkg -r coreutils
```


Obtendrá errores:

```
# dpkg -r coreutils
dpkg: error al procesar el paquete coreutils (--remove):
este es un paquete esencial - no debe ser desinstalado.
Se encontraron errores al procesar:
coreutils
```

Para conocer qué paquetes dependen de coreutils, utilice de nuevo dpkg-query:

```
# dpkg-query -W -f='${Provides}' coreutils
textutils, shellutils, fileutils
```

d - Para instalar un paquete Debian, utilice el parámetro `-i` y anote los errores. ¿Qué ocurre?:

```
# dpkg -i sky.deb
```

Como ya se ha indicado en este capítulo, no hay un método directo equivalente a rpm para la actualización de un paquete. Si ya está instalado el paquete, `-i` lo actualizará. Debe comprobar antes si efectivamente ya está instalado (para ello, vea la respuesta a la primera pregunta).

Suponiendo que no tenemos suerte, obtendremos errores de dependencias. Ejecute el siguiente comando para corregirlos:

```
# apt-get -f install
```

2. APT es un gestor de metapaquetes: gestiona las dependencias en su lugar y trabaja con repositorios, y no con paquetes individuales. Java (oficial) se encuentra en un repositorio alternativo. El objetivo es gestionar este repositorio y las instalaciones asociadas.

a - Añada la línea «deb http://ppa.launchpad.net/linuxuprising/java/ubuntu eoan main» en el archivo de los repositorios. Sustituya Xenial por su versión de Ubuntu.

b - Actualice la base de datos local de APT.

c - Instale únicamente Java 13 oficial de Oracle.

d - Actualice la totalidad de su distribución con los posibles nuevos paquetes disponibles dentro de sus repositorios.

Respuesta:

a - Debe modificar el archivo /etc/apt/sources.list o crear un archivo /etc/apt/sources.list.d/java.list y añadir la línea «deb http://ppa.launchpad.net/linuxuprising/java/ubuntu eoan main».

Puede también utilizar el commando siguiente que lo hará en su lugar y gestionará la firma GPG:

```
$ sudo add-apt-repository ppa:webupd8team/java
```

b - Actualice la base local APT con el comando siguiente:

```
# apt-get update
```

c - En principio, encuentre el nombre del paquete:

```
# apt-cache search java13 | grep oracle
oracle-java13-installer - Oracle Java™ Development Kit (JDK) 13
oracle-java13-set-default - Set Oracle JDK 8 as default Java
oracle-java13-unlimited-jce-policy - Unlimited JCE Policy for Oracle Java 13
oracle-java13-installer - Oracle Java™ Development Kit (JDK) 13
```

Luego instale Oracle-java13-installer:

```
# apt-get install oracle-java13-installer
```

Observe que, a diferencia de lo visto en la primera pregunta, APT gestiona las dependencias y va a instalar Supertuxkart, así como las dependencias asociadas.

d - Actualice su sistema con:

```
# apt-get upgrade
```

4. Las fuentes

Esta práctica propone compilar las fuentes de un servidor FTP llamado proftpd cuyos fuentes están disponibles en <https://github.com/proftpd/proftpd/releases>.

1. Descargue las fuentes en el formato tar.gz en su directorio personal, el autor utilizó el comando siguiente:

```
$ wget https://github.com/proftpd/proftpd/archive/v1.3.6b.tar.gz
```

Para descomprimir las fuentes, teclee:

```
$ tar xvf v1.3.6b.tar.gz
```

Cambie el nombre del directorio resultante de la descompresión:

```
$ mv proftpd-cvs-20170118 proftpd
```

Acceda al directorio:

```
$ cd proftpd
```

2. La compilación de este producto se efectúa ejecutando el comando `./configure`. Haga que el producto se instale en `/opt/proftpd` y luego ejecute la compilación.

Para esto, ejecute en primer lugar el comando:

```
$ ./configure --prefix=/opt/proftpd
```

Verifique y resuelva si es necesario las dependencias faltantes. Luego ejecute el comando:

```
$ make
```

Después de la compilación, instale proftpd. ¿Cómo evitamos que ocurran problemas de permisos? A menos que sea root, Ud. no podrá instalar el software. Cambie al usuario root o utilice `su` o `sudo`:

```
$ sudo make install
```

3. Si proftpd se ha compilado correctamente, no debe haber problemas para ejecutarlo. En otro caso, ¿qué puede haber pasado?

```
$ /opt/proftpd/sbin/proftpd --version  
ProFTPD Version 1.3.5
```

Proftpd depende de ciertas librerías. ¿Están instaladas correctamente en su máquina? Si no, utilice su gestor de paquetes para hacerlo, y vuelva a ejecutar la compilación.

5. Librerías compartidas

En esta práctica, va a gestionar las librerías compartidas vinculadas al programa compilado en la práctica anterior. Observe que se aconseja encarecidamente efectuar estas operaciones en un equipo temporal, virtual, al que habremos posiblemente hecho una instantánea (snapshot).

Muy importante: ¡deje una consola root abierta! La operación tendrá consecuencias desastrosas en los permisos.

1. Compruebe a qué librerías compartidas está vinculado el programa proftpd. ¿Encuentra una librería llamada lybcrypt?

El comando **ldd** permite ver todas las librerías compartidas utilizadas por un programa:

```
$ ldd /opt/proftpd/sbin/proftpd
```

Aísle la línea que busca:

```
$ ldd /opt/proftpd/sbin/proftpd | grep -i libcrypt
```

Debe devolver una única línea. En la máquina de prueba la librería está ubicada en la ruta `/lib/x86_64-linux-gnu/libcrypt.so.1`.

2. Partiendo del principio (falso y peligroso) de que ningún programa utiliza la librería compartida `libcrypt.so.1`, diríjase a su ubicación como root y mueva todos los archivos asociados en `/tmp/lib`. Ejecute luego el comando **ldconfig**. ¿Puede ejecutar `proftpd`? Explíquelo.

Cree el directorio `/tmp/lib`:

```
$ mkdir /tmp/lib
```

Diríjase a `/usr/lib` (o la ruta de la librería):

```
$ cd /lib/x86_64-linux-gnu/libcrypt.so.1
```

Mueva los archivos:

```
# mv libcrypt.so.1 libcritp-2.24.20 /tmp/lib
```

Ejecute `proftpd`. No se inicia:

```
$ /opt/proftpd/sbin/proftpd --version
/opt/proftpd/sbin/proftpd: error while loading shared libraries:
libcrypt.so.1: cannot open shared object file: No such file or directory
```

El comando **ldd** le informa de la desaparición de la librería movida:

```
$ ldd /opt/proftpd/sbin/proftpd | grep -i libcrypt
libcrypt.so.1 => not found
```

3. Edite el archivo de configuración `/etc/ld.so.conf` y añádale la ruta `/tmp/lib`. Ejecute de nuevo `ldconfig` y luego vuelva a iniciar `proftpd`. ¿Qué está pasando?

Si ya se encuentra como root, al añadir la ruta `/tmp/lib` y luego actualizar la caché del cargador dinámico con `ldconfig`, la librería vuelve a estar accesible.

Para realizar todo esto, escriba los comandos:

```
# vi /etc/ld.so.conf
# ldconfig
# ldd ./proftpd
libcrypt.so.1 => /tmp/lib/ libcrypt.so.1
```

El programa vuelve a funcionar.

Sin embargo, si usted no es root, felicidades: acaba de romper su instalación de Linux. Será difícilmente reparable, salvo si arranca de un soporte (CD, USB) para reparar los daños (punto 4) o arranca empleando una de las opciones específicas del cargador de arranque.

En efecto, esta librería gestiona una parte de los permisos y en concreto la codificación/descodificación de las contraseñas. Será imposible convertirse en root, ya que no podrá introducir la contraseña.

4. Vuelva a ponerlo todo en su estado inicial.

Mueva la librería hacia su posición de origen, suprima `/tmp/lib` de `ld.so.conf` y vuelva a ejecutar `ldconfig`. Suprima `/tmp/lib`.