

RPM: The Red Hat Package Manager

RPM: Instalación y supresión de Software

Conceptos clave

- El gestor de paquetes de Red Hat (RPM) consta de tres componentes: (1) la base de datos RPM, (2) los archivos de paquete, y (3) el comando **rpm**.
- La base de datos de RPM se encuentra en el directorio `/var/lib/rpm/`.
- Los archivos de paquete de RPM son esencialmente archivadores que contienen archivos para ser instalados e información del encabezado RPM.
- La nomenclatura del archivo de paquete convencional incluye el nombre de paquete, la versión, el número de publicación y la arquitectura.
- Los archivos de paquetes RPM se pueden instalar con **rpm -i**.
- Los paquetes RPM se pueden suprimir con **rpm -e**
- Los paquetes RPM se pueden actualizar con **rpm -U** o con **rpm -F**.
- El comando **rpm** puede hacer referencia a los archivos de paquete mediante FTP o HTTP de Erl.

RPM: The Red Hat Package Manager

Discusión

Introducción a RPM

Red Hat distribuye software mediante el *Red Hat Package Manager* (más conocido como *RPM*). Con RPM, los administradores pueden fácilmente, instalar, actualizar y administrar software.

Cuando la gente se refiere a RPM, suele referirse a todos los tres componentes en conjunto.

- La base de datos RPM
- Los archivos de paquete RPM
- El comando **rpm** ejecutable

Miraremos a cada uno de estos componentes.

La base de datos RPM

Cada sistema de Red Hat Enterprise Linux mantiene una base de datos del software instalado en la actualidad, la cual reside en el directorio `/var/lib/rpm/`. Al examinar el contenido del directorio con el comando **file**, encontramos que el directorio contiene casi exclusivamente archivos hash de base de datos binarios.

```
[root@station root]# file /var/lib/rpm/*
/var/lib/rpm/Basenames:      Berkeley DB (Hash, version 8, native byte-order)
/var/lib/rpm/Conflictname:   Berkeley DB (Hash, version 8, native byte-order)
/var/lib/rpm/Dirnames:       Berkeley DB (Btree, version 9, native byte-order)
/var/lib/rpm/Filemd5s:       Berkeley DB (Hash, version 8, native byte-order)
/var/lib/rpm/Group:          Berkeley DB (Hash, version 8, native byte-order)
/var/lib/rpm/Installtid:     Berkeley DB (Btree, version 9, native byte-order)
/var/lib/rpm/Name:           Berkeley DB (Hash, version 8, native byte-order)
/var/lib/rpm/Packages:       Berkeley DB (Hash, version 8, native byte-order)
...
```

No hay "partes que puedan recibir mantenimiento por parte del usuario" en el directorio. Aparte de saber que cada máquina de Red Hat Enterprise Linux tiene una base de datos y que puede abarcar una parte significativa del espacio del disco (en el orden de 50 MB), el administrador no necesita interferir en ella directamente.

Archivos de paquete

Red Hat distribuye software de archivos de paquete. Si este concepto es nuevo para usted, puede imaginarlos como algo similar a los archivos **tar** que contienen archivos para ser instalados en un sistema. Pero los archivos de paquete contienen más. También contienen un encabezado RPM, el cual proporciona información acerca de paquetes, tales como el nombre, el tamaño instalado, y un texto corto con la descripción de su contenido. Más importante aún, los encabezados de archivo de paquete contienen información de dependencia de qué otros paquetes, ejecutables o bibliotecas se deben instalar en el sistema para que el paquete sea útil.

Al examinar los medios de Red Hat Enterprise Linux, ya sea en un CD o en un espejo de red, los archivos de paquete generalmente residen en el directorio `RedHat/RPMS`. Después de insertar y montar un CD-ROM de Red Hat Enterprise Linux, por ejemplo, podríamos hallar lo siguiente:

```
[root@station rhel3]# ls /mnt/cdrom/RedHat/RPMS/
4Suite-0.11.1-14.i386.rpm
a2ps-4.13b-28.i386.rpm
acl-2.2.3-1.i386.rpm
alchemist-1.0.27-1.i386.rpm
amanda-2.4.4p1-0.3E.i386.rpm
...
```

Un nombre de archivo de paquete se estructura con las siguientes partes de información:

nombre-version-publicación.arquitectura.rpm

En el siguiente cuadro, describimos la nomenclatura del archivo de paquete `amanda-2.4.4p1-0.3E.i386.rpm` y el significado de cada uno de estos componentes.

Table 1. Análisis del nombre del archivo de paquete `amanda-2.4.4p1-0.3E.i386.rpm`

Componente	Ejemplo	Comentarios
Nombre de paquete	amanda	El nombre del paquete. Cuando el archivo de paquete es instalado en la máquina, el nombre será la entrada de base de datos que identifica el paquete.
Versión	2.4.4p1	La versión de código abierto del producto. Gran parte de software distribuido por Red Hat, no está escrito por empleados de Red Hat. Son los miembros de la comunidad de código abierto los que escriben y mantienen el software. Red Hat cree en la preservación de lazos con el producto de código abierto en el que los productos de Red Hat se basan y respetan el número de versión de la comunidad de código abierto de un producto determinado.
Número de publicación	0.3E	Aunque Red Hat suele iniciar con un producto mantenido por la comunidad de código abierto, Red Hat no modifica el software de código abierto. Algunas veces nuestras modificaciones son mínimas, tales como reorganizar dónde se instalan los archivos. Otras veces los cambios son más significativos. Si los ingenieros de Red Hat consideran que una cierta función se debe agregar al producto, pero no pueden convencer a la comunidad de código abierto para que la incluya, la incluirán en la versión de Red Hat. Cualquier modificación hecha por Red Hat es pública y es gratuita para cualquier miembro de la comunidad que la quiera incorporar. Las modificaciones de una determinada versión de un producto de código abierto, son rastreadas por el número de publicación.
Arquitectura	i386	La arquitectura específica para qué tipo de CPU fue creado el software. Las arquitecturas respaldadas por Red Hat Enterprise Linux incluyen <i>i386</i> , <i>ia64</i> , <i>s390</i> , y otros paquetes. Algunos archivos de paquete utilizan la etiqueta de arquitectura <i>noarch</i> , la cual indica que el paquete no contiene archivos binarios compilados, sino archivos independientes de arquitectura tales como scripts, imágenes o archivos de datos. Además, Red Hat distribuye <i>fuentes</i> RPM, es decir los archivos de paquete que no contienen aplicaciones binarias ("listas para usar"), sino el código de fuente desde el cual se compilaban las aplicaciones. La fuente RPM se identifica con una "arquitectura" <i>src</i> . La mayoría de usuarios no necesitan fuentes RPM, pero se ofrecen para los innovadores y curiosos. Con la excepción que vimos en la lección anterior, este curso trata exclusivamente de los RPM <i>binarios</i> .

El comando rpm ejecutable

El ejecutable **rpm** es un primer plano de administrador para la base de datos RPM. El ejecutable puede realizar una variedad de funciones diferentes que se distinguen por una de las siguientes opciones de línea de comando.

Table 1. Utiliza el comando rpm

Invocación	Uso
<code>rpm -i, rpm -U, rpm -F</code>	Instala o actualiza software desde archivos de paquete.
<code>rpm -e</code>	Suprime paquetes.
<code>rpm -q</code>	Solicita la base de datos RPM.
<code>rpm -V</code>	Verifica un paquete instalado.
<code>rpm --checksig</code>	Verifica la integridad de un archivo de paquete RPM.

La parte restante de esta lección trata de la instalación y supresión de archivos de paquetes. Las siguientes dos lecciones se centran en realizar solicitudes RPM y otros temas misceláneos relacionados con RPM.

Instalación de paquetes RPM: `rpm -i packagefilename`

El software se instala mediante el comando **rpm -i**. Por ejemplo, un administrador desea instalar un software de publicación en una máquina de Red Hat Enterprise Linux, y escoge explorar TeTeX. Desde el interior de un directorio RPMS, primero lista todos los paquetes con `tetex` en sus nombres.

```
[root@station RPMS]# ls *tetex*
tetex-1.0.7-66.i386.rpm      tetex-fonts-1.0.7-66.i386.rpm
tetex-afm-1.0.7-66.i386.rpm tetex-latex-1.0.7-66.i386.rpm
tetex-dvips-1.0.7-66.i386.rpm tetex-xdvi-1.0.7-66.i386.rpm
```

Luego, intenta instalar el archivo de paquete `tetex-dvips-1.0.7-66.i386.rpm` con **rpm -i**. Cuando **rpm** intenta instalar un paquete, hace varios chequeos incluyendo lo siguiente:

- ¿Hay suficiente espacio libre en el sistema de archivo?
- ¿El paquete RPM depende de otro paquete? Si es así, ya está instalado el otro paquete?
- ¿La instalación del paquete sobrescribirá cualquier archivo existente en el sistema de archivos?

Si alguna de estas condiciones no son satisfactorias, **rpm** rehusará instalar el paquete.

```
[root@station RPMS]# rpm -i tetex-dvips-1.0.7-66.i386.rpm
warning: tetex-dvips-1.0.7-66.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
error: Failed dependencies:
    tetex = 1.0.7 is needed by tetex-dvips-1.0.7-66
```

Desde la primera línea de advertencia, encuentra que RPM no pudo verificar la integridad del archivo del paquete, porque ninguna "clave" estaba disponible. Ampliaremos esta información más adelante. Por ahora, hagamos caso omiso de estas advertencias.

Este paquete tiene dependencias. Para que el software en el paquete *tetex-dvips* sea útil, tiene que estar instalado el paquete *tetex*. Nuestro administrador intenta resolver la dependencia instalando el archivo de paquete `tetex-1.0.7-66.i386.rpm`. Esta vez, incluye las opciones de línea de comando **-h** y **-v** que representan la salida con nombres y una serie de almohadillas (#).

```
[root@station RPMS]# rpm -ihv tetex-1.0.7-66.i386.rpm
warning: tetex-1.0.7-66.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
Preparing...                               ##### [100%]
 1:tetex                                   ##### [100%]
```

Esta vez, el paquete se instaló correctamente. Con la flecha de dirección hacia arriba rápidamente recupera la línea anterior, intenta instalar de nuevo el archivo de paquete `tetex-dvips-1.0.7-66.i386.rpm`.

```
[root@station RPMS]# rpm -i tetex-dvips-1.0.7-66.i386.rpm
warning: tetex-dvips-1.0.7-66.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
```

Cuando se resuelve la dependencia, se instala el paquete *tetex-dvips*. Debido a que las opciones **-h** y **-v** no se especificaron, el paquete se instaló silenciosamente (a excepción de la advertencia NOKEY, que ya habíamos decidido ignorar).

Una vez se ha instalado el paquete, los archivos contenidos en ese paquete se instalan en el sistema y el archivo de paquete ya no es necesario. (Así como la caja en la que venía un televisor nuevo ya no se necesita una vez se ha instalado el televisor).

El siguiente cuadro especifica las opciones de línea de comando que algunas veces se utilizan con el comando **rpm -i**.

Table 1. La opciones de línea de comando con rpm -i

Opción	Efecto
-h, --hash	Imprime signos de almohadilla (#) durante la instalación.
-v, --verbose	Imprime salida "verbosa". Una -v imprime el nombre del paquete. Múltiples -v proporcionan salida más detallada.
--nodeps	Se instala incluso si faltan prerequisites.
--replace-files	Se instala incluso si se sobrescriben los archivos
--force	Se instala incluso si el paquete ya está instalado
--test	No realiza ninguna acción, sólo imprime la salida.
--noscripts	No realiza ningún script asociado con la instalación de RPM.

Hay más opciones disponibles. Consulte la página de manual **rpm(8)** para obtener la lista completa.

Supresión de paquetes: rpm -e nombredepaquete

Una vez se ha instalado el paquete, el archivo de paquete es irrelevante y el paquete es ahora una entrada de base de datos en la máquina local. Por lo tanto, el paquete ya no se conoce con el nombre del archivo del paquete, (tal como `tetex-dvips-1.0.7-66.i386.rpm`), sino con el nombre del paquete (tal como *tetex-dvips*).

Los paquetes se suprimen con `rpm -e nombre de paquete`, donde `-e` significa "borrar".

Nuestro administrador intenta ahora suprimir los paquetes que instaló en la sección anterior.

```
[root@station RPMS]# rpm -e tetex-1.0.7-66.i386.rpm
error: package tetex-1.0.7-66.i386.rpm is not installed
```

Al darse cuenta de que ha caído en una trampa, intenta de nuevo, esta vez recordando referirse al paquete por nombre de paquete en lugar de nombre de archivo del paquete.

```
[root@station root]# rpm -e tetex
error: Failed dependencies:
    tetex = 1.0.7 is needed by (installed) tetex-dvips-1.0.7-66
```

De nuevo, el administrador encuentra problemas de dependencia. Debido a que el paquete *tetex-dvips* depende de *tetex*, *tetex* no se puede suprimir mientras *tetex-dvips* esté instalado. Al invertir el orden, los paquetes se pueden suprimir.

```
[root@station root]# rpm -e tetex-dvips
[root@station root]# rpm -e tetex
```

El siguiente cuadro especifica las opciones de línea de comando que algunas veces se utilizan con el comando **rpm -e**.

Table 1. Opciones de línea de comando utilizadas con rpm -e

Opción	Efecto
--nodeps	Suprime el paquete, incluso si los paquetes dependientes están instalados.
--test	No realiza ninguna acción, sólo imprime la salida.

SI desea obtener la lista completa de opciones, consulte la página de manual **rpm(8)**.

Referencia a los archivos de paquete por URL

Al utilizar el ejecutable **rpm**, los archivos de paquetes se pueden especificar mediante una URL de HTTP o FTP. El comando automáticamente descargará el archivo del paquete especificado y tratará el archivo como si éste estuviera especificado localmente.

Cuando se especifica un archivo con una URL de protocolo HTTP, como a continuación el tecleo puede ser tedioso, puesto que todas las versiones y números de publicación se deben especificar y el tabulador no los ayuda a completar.

```
[root@station root]# rpm -ihv http://rha-server/pub/RedHat/RPMS/xpdf-2.02-8.i386.rpm
Retrieving http://rha-server/pub/RedHat/RPMS/xpdf-2.02-8.i386.rpm
warning: /var/tmp/rpm-xfer.cG6hPI: V3 DSA signature: NOKEY, key ID db42a60e
Preparing...
1:xpdf
```

Sin embargo, cuando se utiliza una URL de protocolo FTP, se puede utilizar el comodín de archivo para que coincida con los archivos, y el comodín apuntará *en el servidor FTP*. (Oficialmente, el comodín debería ir entre comillas, pero **bash** le permite a los usuarios un poco de descuido.)

```
[root@station root]# rpm -ihv ftp://rha-server/pub/RedHat/RPMS/xpdf*
Retrieving ftp://rha-server/pub/RedHat/RPMS/xpdf-2.02-8.i386.rpm
warning: /var/tmp/rpm-xfer.g9K53X: V3 DSA signature: NOKEY, key ID db42a60e
Preparing... ##### [100%]
 1:xpdf ##### [100%]
```

Uso de rpm para actualizar RPM

Se podría decir que una de las ventajas del uso de RPM para administrar software es la facilidad para manejar actualizaciones.

Actualización de paquetes: rpm -U packagefilename

A continuación, el administrador descarga desde Red Hat Network un RPM de errata para el paquete *gaim*. Primero, observa su versión actual del paquete y la compara con el archivo de paquete descargado.

```
[root@server1 root]# rpm -q gaim
gaim-0.66-3.0.2
[root@server1 root]# ls gaim-0.75-3.2.0.i386.rpm
gaim-0.75-3.2.0.i386.rpm
```

Ingenuamente intenta instalar la versión más reciente de *gaim*.

```
[root@server1 root]# rpm -i gaim-0.75-3.2.0.i386.rpm
file /usr/bin/gaim from install of gaim-0.75-3.2.0 conflicts with file f
rom package gaim-0.66-3.0.2
file /usr/bin/gaim-remote from install of gaim-0.75-3.2.0 conflicts with
file from package gaim-0.66-3.0.2
...
```

Aunque, en teoría, RPM permite la instalación simultánea de múltiples versiones de un RPM, las dos versiones del RPM no deben contener los mismos archivos. Algunos paquetes RPM están diseñados de esa manera (el paquete de *kernel* es una excepción). Por lo tanto, intentar actualizar un paquete con **rpm -i** suele fallar.

El comando apropiado es **rpm -U**, donde "-U" significa "actualizar". En la actualización todas las versiones anteriores se suprimen, y la nueva versión se instala completamente. (En RPM no se puede "parchear". Un paquete de actualización no necesita paquetes anteriores.)

```
[root@server1 root]# rpm -Uvh gaim-0.75-3.2.0.i386.rpm
Preparing... ##### [100%]
 1:gaim ##### [100%]
[root@server1 root]# rpm -q gaim
gaim-0.75-3.2.0
```

Con ese sólo comando, el paquete *gaim* se ha actualizado completamente. La actualización del paquete RPM se puede calificar con las mismas opciones de línea de comando como instalación de paquete.

Refrescando los paquetes: rpm -F packagefilename

Como una ligera variación en la actualización, el sistema se puede "refrescar" con el comando **rpm -F nombre de archivo de paquete**. La única diferencia entre actualizar (con **rpm -U**) y refrescar (con **rpm -F**) está en que no hay una versión del paquete especificado instalado en el

sistema. En la actualización, el paquete especificado ya está instalado de todas maneras. En el refrescamiento, el archivo de paquete especificado se ignora.

El refrescamiento está diseñado para trabajar con colecciones de errata. Por ejemplo, un administrador podría descargar y coleccionar errata publicada en Red Hat en el directorio del servidor que pudo ser accedido con la URL `ftp://server/pub/errata`. Luego, ocasionalmente, el administrador podría refrescar otra máquina en la red local con un comando como el siguiente:

```
[root@station root]# rpm -Fvh ftp://server/pub/errata/*.rpm
```

Solamente, los paquetes en el directorio de errata pertinentes a la máquina local se actualizarían.

Actualización de RPM y archivos de configuración

Al realizar una actualización RPM, **rpm** reemplaza archivos de la versión más antigua con archivos de una nueva versión. A excepción de los archivos de configuración administrados de modo especial.

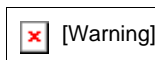
El destino de un archivo de configuración es complicado y depende del estado del archivo de configuración en el paquete original RPM, del paquete RPM más nuevo y de cómo existe en el sistema de archivos. Dependiendo de estos factores, una de las siguientes acciones se puede presentar.

- Si el archivo no ha sido editado, **rpm** supone que la configuración "por defecto" es apropiada y silenciosamente reemplaza el antiguo archivo con el nuevo.
- Si el archivo ha sido editado, pero los archivos de configuración no se diferencian en los dos paquetes RPM, **rpm** supone que la sintaxis no ha cambiado, y deja en su sitio el archivo de configuración editado. Una copia del nuevo archivo de configuración es instalada con la extensión `.rpmnew`.
- Si el archivo ha sido editado y los archivos de configuración difieren en los dos paquetes RPM, **rpm** supone que la sintaxis del archivo de configuración ha cambiado entre las dos versiones. El archivo de configuración se aparta con la extensión `.rpm_save` y el nuevo archivo de configuración predeterminado es instalado. *El administrador es responsable de migrar manualmente la configuración antigua de nuevo a su sitio.*

Para las últimas acciones, **rpm** anunciará el destino del archivo de configuración en la salida estándar. Al realizar una "actualización masiva", es una buena idea guardar la salida estándar para observarla más adelante. En ningún caso, **rpm** suprimirá completamente un archivo de configuración modificado desde el sistema de archivos.

RPM: The Red Hat Package Manager

Ejercicios en línea



Lab Exercise

Objetivo: Instalar, actualizar, y suprimir archivos de paquetes RPM

Tiempo estimado: 20 minutos.

Este laboratorio se referirá a la URL `http://rha-server/pub` como su fuente de archivos de paquetes RPM. El instructor le proporcionará una ubicación alterna.

Specification

1. Bajar las dos versiones del paquete *xosview-rha* de `http://rha-server/pub/rha/rha130/RPMS/`.
2. Instale el paquete *xosview-rha-1.8.0-1.i386.rpm*.
3. Una vez instalado, edite el archivo de configuración `/etc/xosview-rha.conf` cambiando el nivel de depuración de 0 a 1.
4. Utilice **rpm**, y el archivo de paquete *xosview-rha-1.8.0-2.i386.rpm* para actualizar su versión instalada del paquete *xosview-rha*. (Deje en su sitio los archivos de configuración cambiados.)
5. Desde `http://rha-server/pub/es4/RedHat/RPMS`, descargue los tres paquetes que comiencen por *x3270*. Utilice **rpm** para instalar cada uno de los paquetes.
6. Utilice el comando **rpm** para suprimir *únicamente* el paquete *x3270*. (Observe que **rpm** se lamentará de las dependencias rotas. Utilice la opción de línea de comando para suprimir el paquete de todas maneras.)
7. En su archivo `~/.bash_profile`, inicialice la variable de entorno `RPMS` para apuntar a `ftp://rha-server/pub/es4/RedHat/RPMS/`. Observe que el `"/` final puede ser importante.) Una vez inicializado, podrá instalar paquetes RPM mediante comandos similares al siguiente:

```
[root@station root]# rpm -ihv $RPMS/caching-nameserver*
```

8. Utilice su variable inicializada para instalar el paquete *telnet-server*.
9. Refresque su máquina de instalación con todos los paquetes del directorio `ftp://rha-server/pub/rha/rha130/errata/`. (Recuerde que *refrescar* no es lo mismo que *actualizar*.)

Resultados

A title

Question 1

1. El paquete instalado *xosview-rha-1.8.0-2* con un archivo de configuración modificado y almacenado automáticamente del paquete *xosview-rha-1.8.0-1* instalado anteriormente.
2. Los paquetes instalados *x3270-text* y *x3270-x11*, pero el paquete *x3270* no *debe* ser instalado.
3. La variable de entorno `RPMS` inicializada automáticamente, la cual hace referencia a una fuente de archivos de paquetes RPM de Red Hat Enterprise Linux.
4. El paquete instalado *telnet-server*.

grade

RPM: The Red Hat Package Manager

Solicitudes RPM y verificación

Conceptos clave

- Los prerequisites del paquete RPM se pueden solicitar con **rpm -q --requires**.
- Los prerequisites RPM que un paquete RPM puede llenar se pueden solicitar con **rpm -q --provides**.
- Los paquetes RPM pueden contener scripts que son ejecutados en la instalación o supresión del paquete. Estos scripts se pueden observar con **rpm -q --scripts**.
- Se puede generar una lista ordenada de paquetes y la hora de su instalación con **rpm -qa --last**.
- Los paquetes RPM instalados se pueden verificar con **rpm -V**, el cual reporta los archivos que han sido modificados desde su instalación.

RPM: The Red Hat Package Manager

Discusión

Solicitudes RPM: `rpm -q`

En el curso RHA030, el tema de solicitudes de base de datos de RPM y de archivos de paquete RPM fue descrito detalladamente. A manera de repaso, cada solicitud consta de dos preguntas: ¿qué paquetes quiere solicitar? y ¿qué información quiere ver?

Los cuadros a continuación proporcionan un rápido repaso de las varias opciones de línea de comando utilizadas para especificar estos dos aspectos de la solicitud, como se describió en RHA030.

Table 1. Opciones de solicitud RPM para especificar paquetes

Opción	Specification
<code>-a</code>	todos los paquetes instalados
<code>package-name</code>	el paquete <i>package-name</i>
<code>-f nombre de archivo</code>	el paquete que posee el archivo <i>nombre de archivo</i>
<code>-p paquete-archivo-nombre</code>	solicitud directa del archivo del paquete <i>paquete-archivo-nombre</i> . Esta opción es fundamentalmente diferente, como todas las otras opciones solicita la base de datos de los paquetes instalados.

Table 2. Opciones de solicitud RPM para especificar información

Opción	Specification
(por defecto)	paquete nombre y versión
<code>-i</code>	encabezado de información de paquete
<code>-l</code>	lista de archivos de propiedad del paquete
<code>--queryformat str</code>	lista información especificada en cadena de formato <i>str</i>

En esta lección, extendemos nuestra discusión con algunos aspectos avanzados del diseño del paquete y solicitudes RPM, centrándonos en los prerequisites y los scripts RPM.

Solicitud de prerequisites RPM: `--requires` y `--provides`

Los paquetes RPM pueden ser examinados directamente con las opciones de línea de comando `--requires` y `--provides`.

Al examinar directamente los requisitos del paquete *samba*, vemos que los requisitos pueden entrar en varias categorías:

```
[root@station root]# rpm -q --requires samba
/bin/mktemp
/bin/sh
/etc/pam.d/system-auth
...
/sbin/chkconfig
fileutils
initscripts >= 5.54-1
libacl.so.1
...
libc.so.6
...
libresolv.so.2
libssl.so.4
logrotate >= 3.4
pam >= 0.64
...
samba-common = 3.0.2
sed
```

Table 1. Tipos de requisitos de paquete RPM

Tipo	Ejemplo	Comentarios
Archivos específicos	/bin/sh, /etc/pam.d/system-auth	La lista de los archivos específicos suele generarse automáticamente, como se puede deducir de los nombres de archivo que a menudo se repiten.
Bibliotecas dinámicas	libacl.so.1, libc.so.6	El comando <i>samba</i> sirve para examinar qué bibliotecas dinámicas deben existir en el sistema para ejecutar un comando específico (ensaye por ejemplo, <code>ldd /bin/ls</code> .) Al crear un paquete RPM, la lista de todas las bibliotecas se incluye como prerequisite.
Otros paquetes	sed, samba-common = 3.0.2, pam >= 0.64	Al empaquetar un RPM, un desarrollador debe explícitamente listar todas las dependencias de paquete RPM. Los paquetes pueden estar estrechamente vinculados a otro paquete, requiriendo una versión específica (como es el caso para <i>samba-common</i>) o vinculados a otro paquete de una manera más libre, requiriendo sólo alguna versión mayor que la versión listada (como es el caso para el paquete <i>pam</i>).
Requisitos funcionales abstractos	webclient	Algunos paquetes RPM requieren que un paquete brinde alguna funcionalidad abstracta en la instalación, como por ejemplo, el cliente de red. Otros paquetes, tales como <i>mozilla</i> o <i>elinks</i> , deberían especificar si ofrecen esa funcionalidad. Este tipo de requisito es listado por muy pocos paquetes en Red Hat Enterprise Linux y no es utilizado aquí por el paquete <i>samba</i> .

Igualmente, la opción de línea de comando **--provides** sirve para listar lo que el paquete explícitamente proporciona.

```
[root@server1 root]# rpm -q --provides mozilla
libaccessibility.so
libaccessibility.so(EXPORTED)
libappcomps.so
...
libxremoteservice.so
libxremoteservice.so(EXPORTED)
webclient
mozilla = 37:1.4-3.0.18
```

Solicitud de scripts RPM: --scripts

Al empaquetar un paquete RPM, los desarrolladores pueden especificar que los scripts de shell deben ejecutarse como un efecto secundario de la instalación o supresión de un paquete. Los scripts RPM entran dentro de una de las siguientes cuatro categorías.

- Pre-install Scripts
- Post-install Scripts
- Pre-uninstall Scripts
- Post-uninstall Scripts

Los scripts (o "scriptlets", como algunas veces son llamados) se pueden examinar con la opción de solicitud **--scripts**. Considere los siguientes scripts asociados con el paquete del servidor de red *httpd*.

```
[root@server1 root]# rpm -q httpd --scripts
preinstall scriptlet (using /bin/sh): ❶
# Add the "apache" user
/usr/sbin/useradd -c "Apache" -u 48 \
    -s /sbin/nologin -r -d /var/www apache 2> /dev/null || :
postinstall scriptlet (using /bin/sh): ❷
# Register the httpd service
/sbin/chkconfig --add httpd
preuninstall scriptlet (using /bin/sh): ❸
if [ $2 = 0 ]; then
    /sbin/service httpd stop > /dev/null 2>&1
    /sbin/chkconfig --del httpd
fi
```

- ❶ En el pre-install script, el usuario *apache* es agregado al sistema. El demonio **httpd** se ejecuta como el usuario *apache*, y muchos de los archivos instalados por el paquete *httpd* tienen el usuario propietario establecido correctamente.
- ❷ En el post-install script, los enlaces específicos del nivel de ejecución se inicializan para el script de servicio *httpd*.
- ❸ En el pre-uninstall script, justo antes de suprimir el paquete, cualquier demonio **httpd** en ejecución se detiene y los enlaces simbólicos específicos del nivel de ejecución para el script de servicio *httpd* se suprimen.

Listado de paquetes RPM por Marca de tiempo: --last

Como un tipo de solicitud especial, **--last** se puede utilizar para ordenar paquetes RPM cuando se instalan en el sistema. El comando sirve para buscar cambios recientes que pueden presentarse en un sistema.

```
[root@server1 root]# rpm -qa --last | head -5
gaim-0.75-3.2.0           Mon 15 Mar 2004 10:04:38 PM EDT
up2date-gnome-4.2.5-1     Mon 15 Mar 2004 09:54:31 PM EDT
up2date-4.2.5-1           Mon 15 Mar 2004 09:54:30 PM EDT
gpg-pubkey-db42a60e-37ea5438 Mon 15 Mar 2004 09:48:43 PM EDT
xpdf-2.02-8               Mon 15 Mar 2004 09:46:01 PM EDT
```

Verificación de paquete: rpm -V nombredepaquete

En secciones anteriores hemos presentado variaciones en solicitudes RPM. En esta sección, presentamos un nuevo tema: la verificación del paquete RPM.

Naturalmente, la base de datos RPM del sistema almacena cada archivo asociado con un paquete instalado. Además del nombre del archivo, se almacenan muchos de los atributos, como por ejemplo, el usuario del archivo y los grupos propietarios, el modo de archivos, (permisos), la longitud de archivos y una impresión MD5 del contenido del archivo.

El comando **-V** se puede utilizar para "verificar" el paquete en cualquier momento después de la instalación. Cada archivo propiedad de un paquete será comparado con los atributos almacenados en la base de datos RPM, introducidos en la instalación del paquete y cualquier desviación será reportada.

En el siguiente ejemplo, un administrador confirma que todos los archivos pertenecientes al paquete de *bash* están intactos, mientras que para dos de los otros archivos pertenecientes al paquete *pam*, el tamaño de los archivos, la impresión MD5 y el tiempo de modificación han cambiado desde la instalación del paquete (es decir, los archivos han sido modificados de alguna manera).

```
[root@server1 root]# rpm -V bash
[root@server1 root]# rpm -V pam
S.5....T. c /etc/pam.d/system-auth
S.5....T. c /etc/security/limits.conf
```

El siguiente cuadro resume las banderas más comunes reportadas por la verificación de paquete **rpm**.

Table 1. Banderas RPM de verificación

Bandera	Atributo asociado
S	Tamaño
M	Modo (permisos)
5	MD5 sum
L	Estatus de enlace simbólico
U	Usuario propietario
G	Grupo propietario
T	Tiempo de modificación
C	Contexto SELinux

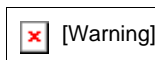
Para obtener un listado completo consulte la sección VERIFY de la página de manual **rpm(8)**.

La verificación de RPM suele utilizarse en dos casos. El primero es depuración. El paquete funcionaba cuando lo instalé, pero no funciona ahora. ¿Qué ha cambiado? El segundo caso es seguridad. Periódicamente, todo el empaquetado se puede verificar con el comando **rpm -Va**. Si

los archivos `/usr/bin/passwd` o `/usr/sbin/sshd` se han modificado, puede sospechar que hay algún problema.

RPM: The Red Hat Package Manager

Ejercicios en línea



Lab Exercise

Objetivo: Realizar solicitudes y verificaciones RPM.

Tiempo estimado: 10 minutos.

Specification

1. En el archivo `~/xscreensaver.files` liste cada *archivo* (no biblioteca o paquete) que sea prerequisite para el paquete *xscreensaver*. Los archivos deben ser listados como una referencia absoluta, uno por línea.
2. En el archivo `~/bind.scripts`, liste las palabras *pre*, *post*, *preun*, o *postun*, una por línea, como correcta para cada tipo de script incluido en el paquete *bind*. Observe que el paquete no puede ser instalado y se necesitaría obtener desde <http://rha-server/pub/es4/RedHat/RPMS>. Si es así, conteste la pregunta sin instalar el paquete.

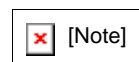
Por ejemplo, si el paquete contiene los scripts `post-install` y `post-uninstall`, el archivo `~/bind.scripts` debe contener las dos palabras *post* y *postun*.

3. En el archivo `~/recent_rpms`, liste sus 5 paquetes del sistema añadidos más recientemente, un paquete por línea. (Liste los archivos con el nombre y la versión del paquete con la solicitud **rpm** apropiada).
4. En el archivo `~/rhn-applet_modified_files`, escriba los nombres de los archivos pertenecientes al paquete *rhn-applet*, el cual **rpm** reporta como modificado desde que el paquete fue instalado. Escriba los nombres de archivo como una referencia absoluta, un archivo por línea.

Resultados

A title

Question 1



Note

El script de evaluación debe realizar solicitudes en la base de datos RPM, que pueden tomar un tiempo en realizarse. Ayude a mejorar el rendimiento del script ejecutando el siguiente comando y espere a que termine, inmediatamente antes de la calificación.

```
[root@station root]# rpm -qa > /dev/null
```

Recuerde que todos los archivos siguientes existen en el directorio raíz de su cuenta de la Academia no en `/root`.

1. El archivo `~/xscreensaver.files`, el cual lista todos los prerequisites del *archivo* para el paquete *xscreensaver*, mediante referencias absolutas.

2. El archivo `~/bind.scripts`, que contiene algunas de las palabras *pre*, *post*, *preun* o *postun*, como apropiadas para los tipos de scripts asociados con el paquete *bind* de RPM.
3. El archivo `~/recent_rpms` que contiene los nombres (incluyendo las versiones) de los 5 paquetes agregados más recientemente en su sistema.
4. El archivo `~/rhn-applet_modified_files`, el cual lista todos los archivos pertenecientes a *rhn-applet* reportados por **rpm** como modificados después de que el paquete fue instalado.

grade

RPM: The Red Hat Package Manager

Utilidades de RPM misceláneas

Conceptos clave

- Todos los archivos de paquete de Red Hat Enterprise Linux están criptográficamente firmados por Red Hat, Inc.
- La clave pública GPG utilizada para confirmar la integridad del paquete se distribuye desde el directorio raíz de cada CD-ROM de Red Hat Enterprise Linux, y en el archivo `/usr/share/rhn/RPM-GPG-KEY`.
- Las claves públicas GPG pueden ser importadas a la base de datos RPM mediante **`rpm --import`**.
- La integridad del archivo de paquete RPM se puede confirmar con **`rpm --checksig`**.
- Los archivos del paquete RPM se pueden convertir en flujo **`cpio`** con **`rpm2cpio`**.
- La aplicación gráfica **`system-config-packages`** permite a los administradores instalar paquetes RPM desde la distribución de Red Hat Linux.
- Los sistemas de Red Hat Enterprise Linux se pueden registrar con Red Hat Network mediante el comando **`up2date`**.
- Los sistemas de Red Hat Enterprise Linux registrados con Red Hat Network, pueden utilizar el comando **`up2date`** para descargar e instalar paquetes RPM.
- Los archivos de paquete de fuente RPM contienen el código fuente con el cual se construyen los paquetes RPM binarios. Ellos pueden ser instalados mediante **`rpm -i`**.

RPM: The Red Hat Package Manager

Discusión

Firmas de archivos de paquete RPM

En nuestra primera lección, tras instalar el archivo de paquete, **rpm** emitió varias advertencias concernientes a la firma, las cuales decidimos pasar por alto. Ahora centramos nuestra atención en las advertencias de firma. Para refrescar nuestra memoria, aquí presentamos la advertencia que apareció cuando fue instalado el paquete RPM *tetex-dvips*.

```
[root@station RPMS]# rpm -i tetex-dvips-1.0.7-66.i386.rpm
warning: tetex-dvips-1.0.7-66.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
```

Red Hat firma criptográficamente cada archivo de paquete RPM con una clave privada GPG (Guardia de privacidad GNU). Al verificar de modo criptográfico la firma, los usuarios se aseguran que el archivo de paquete RPM, descargado desde Internet, no ha sido modificado desde que Red Hat firmó el paquete. El comando **rpm** automáticamente intenta verificar la firma tras la instalación. No obstante, la clave pública GPG complementaria se debe obtener para realizar la verificación de la firma. La advertencia se presenta ya que el paquete no puede ser verificado porque la clave pública necesaria (ID db42a60e) no está disponible.

Obtención de la clave pública de Red Hat

Puesto que la seguridad de la verificación de la firma depende de la integridad de la clave pública, Red Hat distribuye la clave pública en una variedad de formas. La clave pública GPG de Red Hat se puede hallar en cualquiera de los siguientes sitios:

- El sitio Web www.redhat.com.
- Desde el directorio raíz de cualquier CD-ROM de Red Hat, en el archivo llamado `RPM-GPG-KEY`.
- En la distribución de Red Hat Enterprise Linux, como el archivo `/usr/share/rhn/RPM-GPG-KEY`.

Registro de una clave pública con RPM: `--import`

Una vez se haya obtenido la clave pública, puede registrarse con una base de datos RPM del sistema mediante `rpm --import keyfile`. A continuación, la clave pública es instalada desde el directorio raíz de un CD-ROM de Red Hat Enterprise Linux.

```
[root@station root]# rpm --import /mnt/cdrom/RPM-GPG-KEY
```

Las claves múltiples se pueden importar. Al listar todos los paquetes que comienzan por *gpg-pubkey* se puede obtener un listado de todas las claves públicas registradas en la base de datos de RPM del sistema.

```
[root@station root]# rpm -qa | grep gpg-pubkey
gpg-pubkey-db42a60e-37ea5438
```

Los detalles de una clave pública se pueden obtener al realizar una solicitud `-i`.

```
[root@station root]# rpm -q gpg-pubkey-db42a60e-37ea5438 -i
Name       : gpg-pubkey                      Relocations: (not relocateable)
Version    : db42a60e                        Vendor: (none)
Release    : 37ea5438                        Build Date: Mon 05 Jan 2004 02:00:13 PM EST
Install Date: Mon 05 Jan 2004 02:00:13 PM EST Build Host: localhost
Group      : Public Keys                     Source RPM: (none)
Size       : 0                               License: pubkey
Signature  : (none)
Summary    : gpg(Red Hat, Inc <security@redhat.com>)
Description:
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: rpm-4.2.1 (beecrypt-3.0.0)

mQGIBDfQVDgRBADBKr3Bl6P08BQ0H8sJoD6p9U7Yyl7pjtZqioviPwXP+DCWd4u8HQzcxAZ5
7m8ssAlLK1Fx93coJhDzM130+p5BG9mYSWShLabR3N1KXdXQYYcowTOMGxdwYRGr1Spw8Qyd
...
```

Comprobar manualmente una firma de paquete: --checksig

La firma digital de un paquete se puede confirmar manualmente con `rpm --checksig packagefilename`.

```
[root@station root]# rpm --checksig /mnt/cdrom/RedHat/RPMS/pxe-0.1-36.i386.rpm
/mnt/cdrom/RedHat/RPMS/pxe-0.1-36.i386.rpm: (sha1) dsa sha1 md5 gpg OK
```

Como lo indica la salida, las sumas de verificación o checksums internas y la firma GPG están "OK".

Conversión de un paquete RPM en un flujo cpio

Al introducir archivos de paquete RPM, dijimos que un paquete RPM se puede concebir como un archivo **tar**. Aunque el concepto es correcto, el protocolo está errado. En realidad, los archivos de paquete RPM están formateados con un comando de archivación llamado **cpio**. Mientras que el comando de archivación **tar** se centra en listas de argumento de nombres de archivo, el **cpio** se centra en "flujos" de nombres de archivo que fluyen desde la entrada estándar hasta la salida estándar.

Un archivo de paquete se puede convertir en flujo **cpio** mediante el comando **rpm2cpio**. Aunque se requiere conocimiento acerca de **cpio** para poder apreciar en su totalidad las implicaciones, una mirada rápida a la página de manual **cpio(1)** y a los siguientes ejemplos demostrará que **rpm2cpio** permite extraer los archivos individuales del archivo de paquete RPM, sin instalar el paquete.

A continuación, el archivo de paquete *vsftpd* cvs es convertido en un flujo **cpio**, el cual es entubado al comando **cpio**. El comando **cpio** luego lista cada archivo en el flujo **cpio**.

```
[root@station root]# rpm2cpio /mnt/cdrom/RedHat/RPMS/vsftpd-1.2.0-4.i386.rpm |
cpio --extract --list
361 blocks
./etc/logrotate.d/vsftpd.log
./etc/pam.d/vsftpd
./etc/rc.d/init.d/vsftpd
./etc/vsftpd
...
```

Reordenando un poco el comando **cpio**, los archivos individuales del paquete RPM se pueden extraer del sistema de archivo local. En el siguiente ejemplo, el archivo `./etc/vsftpdusers` es creado en el directorio local, generando los directorios necesarios a lo largo del camino.

```
[root@station root]# rpm2cpio /mnt/cdrom/RedHat/RPMS/vsftpd-1.2.0-4.i386.rpm |
cpio --extract --make-directories ./etc/vsftpd.ftpusers
361 blocks
[root@station root]# head -5 ./etc/vsftpd.ftpusers
# Users that are not allowed to login via ftp
root
bin
daemon
adm
```

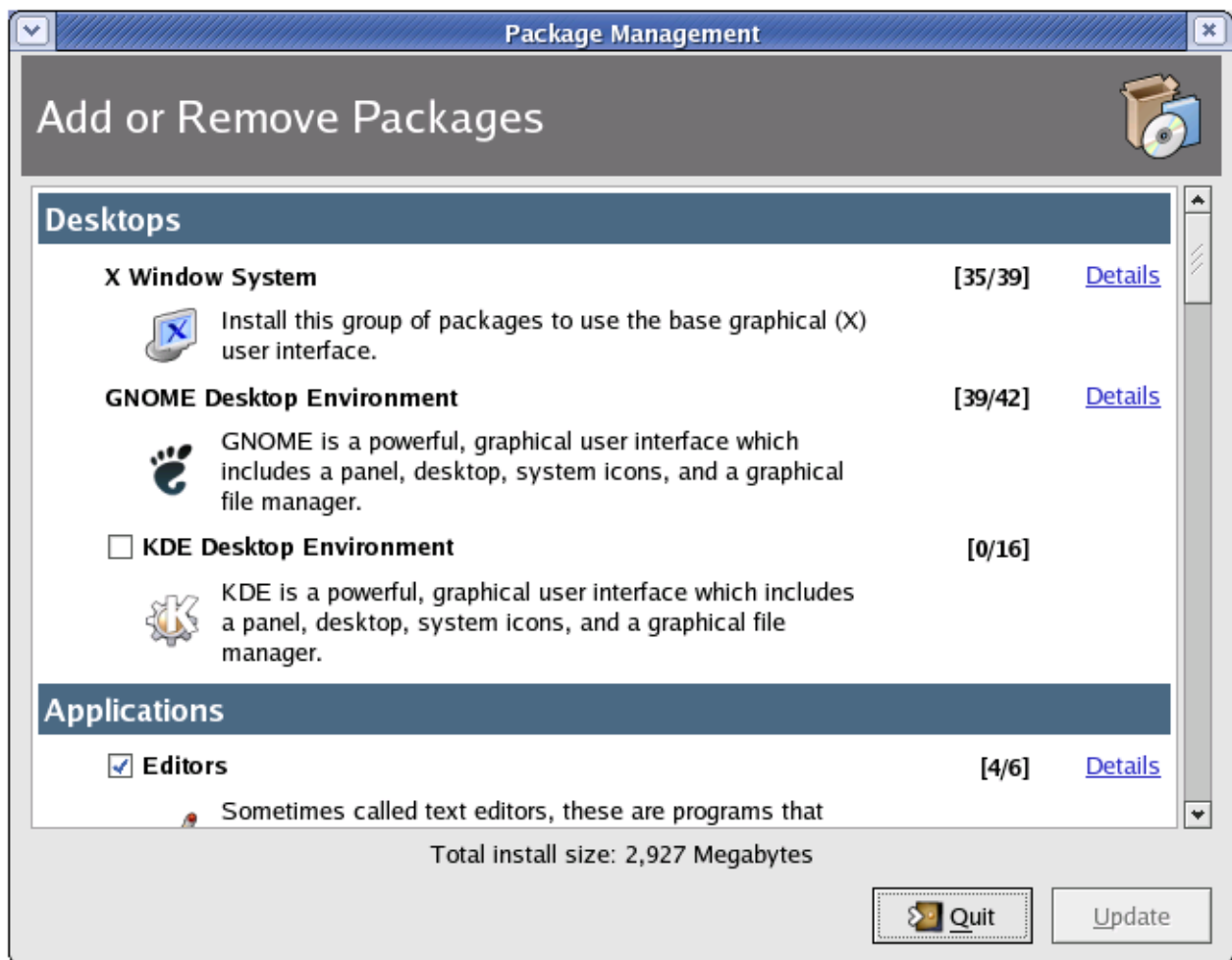
Aunque los detalles del comando **cpio** van más allá de los objetivos de este curso, los administradores aún pueden beneficiarse de la estrecha relación del archivo de paquete de un archivo de **cpio**. Trate de examinar directamente el contenido de un archivo de paquete con el comando **cat** y será recompensado con el desorden binario esperado. (Recuerde que el comando **reset** restablecerá una terminal, si es necesario.)

¿Qué sucede si un archivo de paquete se navega con **less**? El pager **less** no visualiza el contenido binario real del archivo, sino en cambio extrae silenciosamente el archivo del paquete como un archivo **cpio**, y muestra los encabezados de paquetes, el registro de cambios, y una lista detallada de archivos.

La aplicación system-config-packages

Red Hat Enterprise Linux provee una aplicación gráfica para administrar paquetes, los cuales pueden ser iniciados como **system-config-packages**.

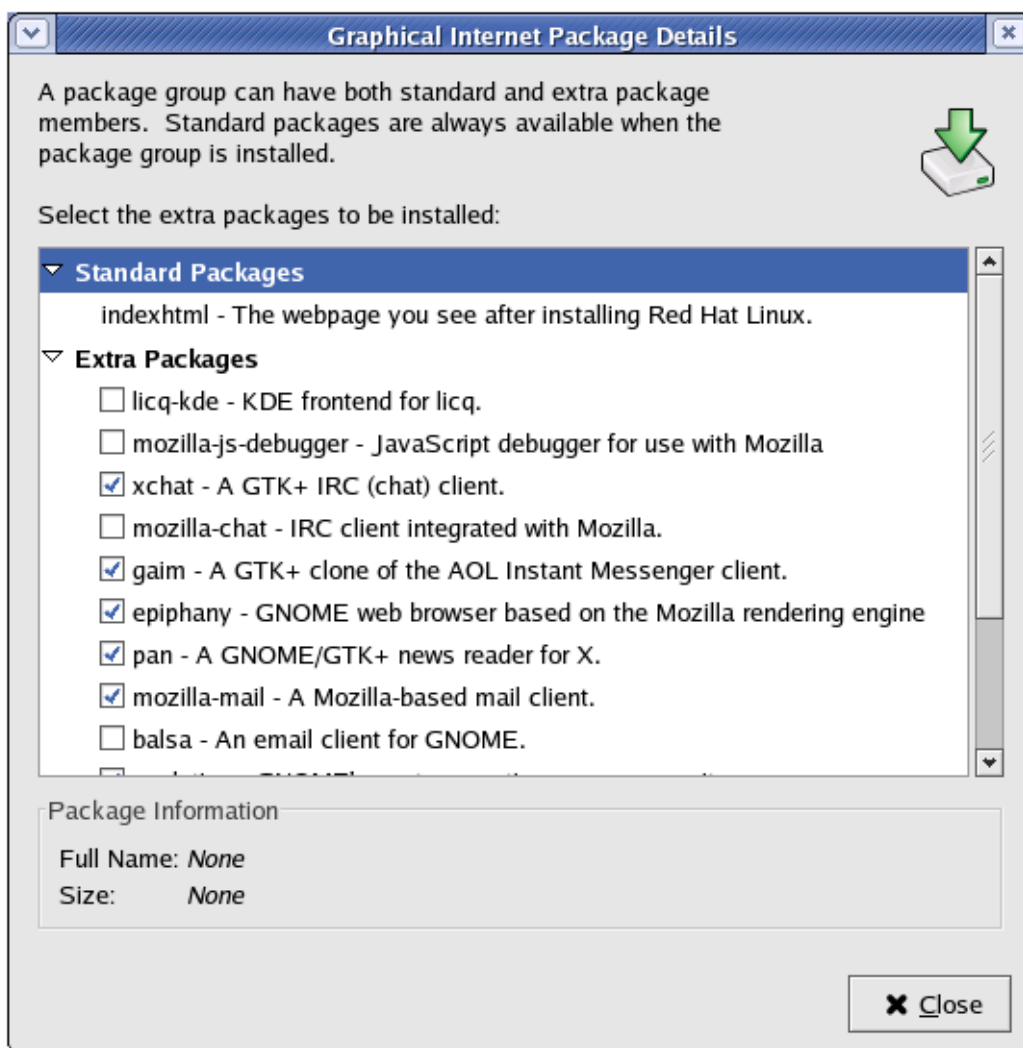
Figure 1. La aplicación system-config-packages



La aplicación presenta una lista de paquetes disponibles con Red Hat Enterprise Linux, estén o no

instalados en el sistema local. Los paquetes están organizados con los mismos grupos de componentes de software que se encuentran en el instalador. Paquetes opcionales pueden ser seleccionados dentro de los grupos de componentes.

Figure 2. La aplicación system-config-packages: selección detallada de paquete



Los nuevos paquetes se pueden instalar al seleccionar las entradas correctas. Una vez seleccionadas, la aplicación solicitará al usuario insertar el CD-ROM apropiado de la distribución de Red Hat Enterprise Linux, si es necesario. (Si el contenido de los CD-ROM de la distribución ha sido copiado en el directorio local, se puede especificar el directorio con la opción de línea de comando **--tree=ruta** en el arranque.)

Warts RPM: cierre de base de datos

Las bases de datos suelen estar diseñadas para sólo una transacción pendiente a la vez, donde el acceso a la base de datos se obtiene al adquirir una base de datos *lock*. El cierre se puede hacer al mantener un proceso a la vez. La base de datos no es una excepción.

En versiones más antiguas de RPM, el cierre de base de datos era tosco: sólo una instancia del ejecutable **rpm** podía acceder a la base de datos en cualquier momento. Si alguien estaba instalando un archivo de paquete RPM, nadie podía realizar una solicitud hasta que se completara la instalación. Aunque la integridad de la base de datos estaba asegurada, los usuarios solían frustrarse por la demora.

El comando **rpm** ha adoptado recientemente más bases de datos de cerramiento de alto grado. Los

procesos necesitan adquirir un cierre para sólo una porción de la base de datos que están modificando, no para toda la base de datos. La estrategia de alto grado significa que los cierres no se mantienen por mucho tiempo. La desventaja es que el cerramiento de alto grado es más difícil de diseñar.

Con la conversión a cerramiento de alto grado, el mecanismo de cerramiento de la base de datos **rpm** en ocasiones se atasca, generalmente como resultado de un comando **rpm** cancelado previamente. El síntoma es un comando **rpm**, tal como **rpm -qa** que simplemente se cuelga tratando de adquirir un cerramiento inalcanzable. Cualquier otro comando **rpm** iniciado después de este punto también se cuelga. Infortunadamente, Red Hat Enterprise es propenso a este problema.

Afortunadamente, la solución es bastante simple.

1. Mate todos los procesos **rpm** que se estén ejecutando (o que se estén colgando), con un comando tal como **killall -9 rpm**.
2. Suprima todos los archivos comenzados por `__db.` desde el directorio de la base de datos RPM (`/var/lib/rpm/`).

Para realizar el último paso (y posiblemente el primero), deberá tener acceso root. Si no tiene acceso root, el usuario puede simplemente reiniciar la máquina. En el script de inicio de Red Hat Enterprise Linux `/etc/rc.d/rc.sysinit`, se encuentra la siguiente línea sutil sin comentar.

```
rm -f /var/lib/rpm/__db*
```

Como resultado, cualquier cierre "atascado" se limpiará cada vez que se reinicie el sistema.

Red Hat Network

Red Hat distribuye software principalmente a través de Red Hat Network, el cual se halla en <http://rhn.redhat.com>. Al comprar un derecho de Red Hat Network, obtendrá acceso a la aplicación de red. A través de una cuenta activa de Red Hat Network, podrá hacer lo siguiente:

- Descargar imágenes ISO de la distribución de medios de Red Hat Enterprise Linux.
- Descargar errata publicada después de una determinada publicación de Red Hat Enterprise Linux.
- Administrar una configuración de software de una máquina o grupo de máquinas remotas a través de una interfaz de red.

Una vez esté instalado Red Hat Enterprise Linux en el sistema, se puede registrar con una cuenta de Red Hat Network mediante el comando **up2date**. Luego, se puede utilizar **up2date** para descargar e instalar errata automáticamente o lo contrario, desinstalar paquetes de Red Hat Network.

Fuentes RPM

Nuestro cuaderno sobre RPM se enfocó casi todo en paquetes RPM *binarios* que son el único tipo de archivos de paquete concernientes a la mayoría de los usuarios. Esta sección proporciona una introducción rápida a las *fuentes* RPM.

Las fuentes RPM suelen hallarse en los CD-ROM de la distribución de Red Hat Enterprise Linux. De la misma manera que los RPM binarios tradicionalmente se localizan con el directorio `RedHat/RPMS/`, las fuentes RPM se hallan dentro de un directorio `SRPMS/`, el cual suele ser un hermano del directorio `RedHat/`.


```
[root@station root]# ls /mnt/cdrom/SRPMS/
ant-1.5.2-23.src.rpm
apmd-3.0.2-18.src.rpm
ash-0.3.8-16.src.rpm
aspell-de-0.1.1-17.src.rpm
aspell-es-0.2-13.src.rpm
aspell-it-0.1-16.src.rpm
...
```

Instalación de fuentes RPM

Al igual que los paquetes RPM binarios, las fuentes RPM se instalan mediante el comando **rpm -i**. No obstante, hay diferencias importantes en la instalación de paquetes de fuentes RPM.

- Para instalar una fuente RPM, se debe instalar el paquete RPM binario *rpm-build*. Este paquete provee, entre otras cosas, el directorio **/usr/src/redhat/**.
- Las fuentes RPM *no están en base de datos*. En la instalación de una fuente RPM, los archivos se agregan al sistema de archivos, pero no se hace ninguna comprobación de dependencia ni se modifica la base de datos RPM.
- Las fuentes RPM se instalan completamente en los directorios **/usr/src/redhat/SOURCES/** y **/usr/src/redhat/SPECS/**.

Qué hay en una fuente RPM

Las fuentes RPM generalmente contienen los siguientes elementos.

- Una copia de una "fuente pristina" de un proyecto de código abierto, distribuido como un archivo comprimido tar.
- Una colección de "archivos de parche" que define los cambios hechos por Red Hat a los proyectos de código abierto.
- Un "archivo spec", que proporciona una fórmula para extraer la fuente pristina, aplica los parches para modificar el código abierto, compilar los ejecutables, y luego recoger los productos recién compilados en un archivo de paquete RPM binario.

En el siguiente ejemplo, la fuente RPM para el paquete *less* está instalada.

```
[root@station root]# rpm -ihv /mnt/cdrom/SRPMS/less-378-11.src.rpm
1:less ##### [100%]
[root@station root]# ls /usr/src/redhat/SOURCES/
less-378+iso247-20030108.diff  less-378-ncursesw.patch  less.csh
less-378-ko.patch           less-378-rhl.patch       lesspipe.sh
less-378-multibyte.patch     less-378.tar.gz          less.sh
[root@station root]# ls /usr/src/redhat/SPECS/
less.spec
```

Al explorar una fuente RPM obtenemos una vista de la segunda mitad del diseño detrás de RPM la cual suele estar oculta. No sólo RPM es un sistema de distribución de software, también es un sistema para *construir* proyectos de código abierto.

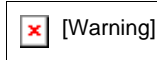
Aunque la compilación de software de código abierto va más allá de los objetivos de este curso, al ejecutar **rpmbuild -ba archivospec** podrá observar una construcción automática en acción.

```
[root@station root]# rpmbuild -ba /usr/src/redhat/SPECS/less.spec
Executing(%prep): /bin/sh -e /var/tmp/rpm-tmp.51951
+ umask 022
+ cd /usr/src/redhat/BUILD
+ LANG=C
+ export LANG
...
```

Si la construcción tiene éxito, los productos compilados deberían hallarse en el directorio `/usr/src/redhat/BUILD` y el archivo de paquete binario resultante en un subdirectorio de `/usr/src/redhat/RPMS`.

RPM: The Red Hat Package Manager

Ejercicios en línea



Lab Exercise

Objetivo: Instalar una clave pública GPG y realizar una verificación de firma en un archivo de paquete RPM.

Tiempo estimado: 10 minutos.

Specification

1. Importar la clave pública GPG en `/usr/share/rhn/RPM-GPG-KEY` a su base de datos RPM.
2. Los paquetes *xosview-rha* utilizados en un ejercicio anterior han sido firmados digitalmente por la clave personal de Red Hat Academy. Busque la clave pública GPG en <http://rha-server/pub/rha/rha130/RHA-GPG-KEY>. Descárguela e impórtela a su base de datos RPM.
3. Si los archivos del ejercicio anterior no están disponibles, descargue el archivo de paquetes *xosview-rha* hallados en <http://rha-server/pub/rha/rha130/RPMS/>. Utilice **rpm** (con la clave recién instalada) para verificar la integridad de los archivos.

Resultados

A title

Question 1

1. Una base de datos **rpm** con la RPM de Red Hat y las claves GPG de Red Hat Academy instaladas.

grade