

INSTALACIÓN Y ACTUALIZACIÓN DE PAQUETES DE SOFTWARE

RH124

Capítulo 13

Descripción general	
Meta	Descargar, instalar, actualizar y administrar paquetes de software de Red Hat y repositorios de paquetes YUM.
Objetivos	<ul style="list-style-type: none"> • Registrar sistemas con su cuenta de Red Hat y autorizar las actualizaciones de software para los productos instalados. • Explicar el significado de un paquete RPM y el modo en que los paquetes RPM se utilizan para administrar software en un sistema con Red Hat Enterprise Linux. • Buscar, instalar y actualizar paquetes de software usando el comando yum. • Habilitar y deshabilitar el uso de repositorios YUM de terceros o de Red Hat. • Examinar los archivos de paquetes de software descargados e instalarlos.
Secciones	<ul style="list-style-type: none"> • Asignación de suscripciones a sistemas para actualizaciones de software (y práctica) • Paquetes de software RPM y YUM (y práctica) • Administración de actualizaciones de software con yum (y práctica) • Habilitación de repositorios de software yum (y práctica) • Cómo examinar archivos de paquetes RPM (y práctica)
Trabajo de laboratorio	<ul style="list-style-type: none"> • Instalación y actualización de paquetes de software

Paquetes de software RPM y yum

RH124

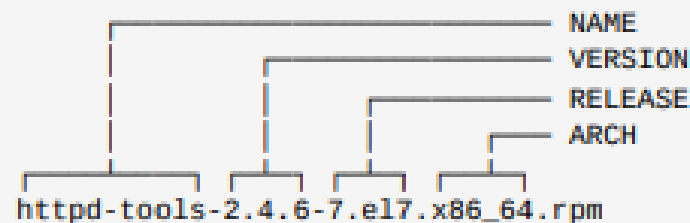
Objetivos

Tras finalizar esta sección, los estudiantes deberían poder explicar qué es un paquete RPM y cómo se utilizan los paquetes de este tipo para administrar software en un sistema con Red Hat Enterprise Linux.

Paquetes de software y RPM

Hace varios años, Red Hat desarrolló **RPM Package Manager**, que ofrece una manera estándar de colocar software en paquetes para su distribución. Administrar software como *paquetes RPM* es mucho más sencillo que trabajar con un software que simplemente se extrajo de un fichero en un sistema de archivos. Esta solución permite que los administradores controlen qué archivos se instalaron con el paquete de software y cuáles deben quitarse si el software se desinstala, y que verifiquen que los paquetes compatibles estén presentes cuando se instale. La información sobre los paquetes instalados se almacena en una base de datos de RPM local en cada sistema. Todo el software proporcionado por Red Hat para Red Hat Enterprise Linux se ofrece como un paquete RPM.

Los archivos de paquetes RPM reciben su nombre de acuerdo con una combinación del paquete **name-version-release.architecture**:



- NAME es una o más palabras que describen los contenidos (httpd-tools).
- VERSION es el número de versión del software original (2.4.6).
- RELEASE es el número de lanzamiento del paquete que se basa en la versión y que es definido por el empaquetador, que es posible que no sea el desarrollador del software original (7.el7).
- ARCH es la arquitectura de procesador en la que se compiló el paquete para su ejecución. "noarch" indica que el contenido de este paquete no es específico de la arquitectura (x86_64).

Cuando se instalan paquetes de repositorios, solo se necesita el nombre de paquete. Se instalará el paquete con la versión superior. Si hay múltiples archivos con la misma versión, se instalará el paquete con el número de lanzamiento superior.

Cada paquete RPM es un fichero especial con tres componentes:

- Los archivos instalados con el paquete.
- Información sobre el paquete (metadatos), como el nombre, la versión, el lanzamiento y la arquitectura; un resumen y una descripción del paquete; determinación de si se necesita instalar otros paquetes; licencias; un registro de cambio del paquete; y otros detalles.
- Scripts que pueden ejecutarse cuando el paquete se instala, actualiza o quita, o que se activan cuando otros paquetes se instalan, actualizan o quitan.

Los paquetes RPM pueden llevar la firma digital de la organización que los colocó en paquete. Todos los paquetes de un origen particular llevan normalmente la firma de la misma clave privada GPG. Si el paquete se modifica o daña, la firma dejará de tener validez. De esta manera, el sistema podrá verificar la integridad de los paquetes antes de instalarlos. Todos los paquetes RPM lanzados por Red Hat llevan firma digital.

Actualizaciones y parches

Cuando Red Hat revisa el código fuente ascendente de un paquete de software, se genera un paquete RPM completo. Si un paquete se añade a un sistema, se necesita solo la última versión del paquete y no cada versión del paquete desde el primer lanzamiento. En el caso de sistemas que necesitan actualizaciones, se quita la versión anterior del paquete y se instala la versión nueva. Los archivos de configuración generalmente se conservan durante una actualización, pero el comportamiento exacto de un paquete en particular se define cuando se crea la nueva versión del paquete.

En la mayoría de los casos, solo una versión o un lanzamiento de un paquete puede instalarse a la vez. Generalmente, el proceso de instalación de RPM no permitirá que los archivos se sobrescriban. Si un paquete se crea de modo que no haya nombres de archivos en conflicto, pueden instalarse múltiples versiones. Este es el caso del paquete **kernel**. Como un kernel nuevo solo puede evaluarse mediante el inicio en ese kernel, el paquete está específicamente diseñado para que puedan instalarse múltiples versiones simultáneamente. Si el kernel nuevo no arranca, el kernel anterior sigue estando disponible.

Administrador de paquetes yum

Una vez instalado el sistema, los paquetes y las actualizaciones de software adicionales normalmente se instalan desde un *repositorio de paquetes* de red, la mayoría de las veces a través del servicio de administración de suscripciones de Red que se abordó en la sección anterior. El comando **rpm** puede utilizarse para instalar, actualizar, quitar y consultar paquetes RPM. Sin embargo, no resuelve dependencias de manera automática y todos los paquetes deben incluirse en una lista. Las herramientas como **PackageKit** y **yum** son aplicaciones front-end para **rpm** y se pueden utilizar para instalar paquetes individuales o *colecciones de paquetes* (a veces denominadas *grupos de paquetes*).

El comando **yum** permite buscar numerosos repositorios de paquetes y sus dependencias para que puedan instalarse de manera conjunta con la finalidad de atenuar los problemas de dependencia. El archivo de configuración principal para **yum** es **/etc/yum.conf** con archivos de configuración de repositorios adicionales ubicados en el directorio **/etc/yum.repos.d**. Los archivos de configuración de repositorio incluyen, como mínimo, una identificación de repositorio (en corchetes), un nombre y la ubicación de la URL del repositorio de paquetes. La URL puede apuntar a un directorio (archivo) local o recurso compartido de red remoto (http, ftp, etc.). Si la URL se pega en un navegador, los contenidos deben incluir los paquetes RPM, posiblemente en uno o más subdirectorios, y un directorio **repodata** con información sobre los paquetes disponibles.

El comando **yum** se utiliza para incluir en una lista repositorios, paquetes y grupos de paquetes:

```
[root@serverX ~]# yum repolist
Loaded plugins: langpacks
repo id                repo name                status
!rhel_dvd              Remote classroom copy of dvd    4,529
repolist: 4,529
[root@serverX ~]# yum list yum*
Loaded plugins: langpacks
Installed Packages
yum.noarch              3.4.3-118.2.el7         @anaconda/7.0
yum-langpacks.noarch    0.4.2-3.el7            @rhel_dvd
yum-metadata-parser.x86_64 1.1.4-10.el7           @anaconda/7.0
yum-rhn-plugin.noarch    2.0.1-4.el7            @rhel_dvd
yum-utils.noarch        1.1.31-24.el7          @rhel_dvd
Available Packages
yum-plugin-aliases.noarch 1.1.31-24.el7          rhel_dvd
yum-plugin-changelog.noarch 1.1.31-24.el7          rhel_dvd
yum-plugin-tmprepo.noarch 1.1.31-24.el7          rhel_dvd
yum-plugin-verify.noarch 1.1.31-24.el7          rhel_dvd
yum-plugin-versionlock.noarch 1.1.31-24.el7          rhel_dvd
[root@serverX ~]# yum list installed
Loaded plugins: langpacks
Installed Packages
GConf2.x86_64           3.2.6-8.el7            @rhel_dvd
ModemManager.x86_64     1.1.0-6.git20130913.el7 @rhel_dvd
ModemManager-glib.x86_64 1.1.0-6.git20130913.el7 @rhel_dvd
...
[root@serverX ~]# yum grouplist
...
Installed groups:
  Base
  Desktop Debugging and Performance Tools
  Dial-up Networking Support
  Fonts
  Input Methods
  ...
```




Referencias

Páginas del manual `yum(8)`, `yum.conf(5)`, `rpm(8)`, `rpm2cpio(8)` y `rpmkeys(8)`

Práctica: Paquetes de software RPM

Administración de actualizaciones de software con yum

RH124

Objetivos

Tras finalizar esta sección, los estudiantes deberían poder encontrar, instalar y actualizar paquetes de software mediante el uso del comando **yum**.

Trabajar con yum

yum es una herramienta eficaz de la línea de comando que puede usarse para administrar (instalar, actualizar, eliminar y consultar) los paquetes de software de modo más flexible. Los paquetes oficiales de Red Hat se descargan normalmente de la red de distribución de contenido de Red Hat. Si se registra un sistema en el servicio de administración de suscripciones, se configura automáticamente el acceso a los repositorios de software basado en las suscripciones que se adjuntan.

La búsqueda de software con yum

- **yum help** muestra la información de uso.
- **yum list** muestra los paquetes instalados y aquellos disponibles.

```
[root@serverX ~]# yum list 'http*'
Loaded plugins: langpacks
Available Packages
httpcomponents-client.noarch          4.2.5-4.el7          rhel_dvd
httpcomponents-core.noarch           4.2.4-6.el7          rhel_dvd
httpd.x86_64                          2.4.6-17.el7         rhel_dvd
httpd-devel.x86_64                   2.4.6-17.el7         rhel_dvd
httpd-manual.noarch                   2.4.6-17.el7         rhel_dvd
httpd-tools.x86_64                   2.4.6-17.el7         rhel_dvd
```

- **yum search KEYWORD** enumera paquetes por palabras clave que se encuentran en los campos de nombre y resumen solamente.

Para buscar paquetes que contienen "servidor web" en los campos nombre, resumen y descripción, utilice **search all**:

```
[root@serverX ~]# yum search all 'web server'
Loaded plugins: langpacks
===== Matched: web server =====
freeradius.x86_64 : High-performance and highly configurable free RADIUS server
hsqldb.noarch : HyperSQL Database Engine
httpd.x86_64 : Apache HTTP Server
libcurl.i686 : A library for getting files from web servers
libcurl.x86_64 : A library for getting files from web servers
mod_revocator.x86_64 : CRL retrieval module for the Apache HTTP server
mod_security.x86_64 : Security module for the Apache HTTP Server
python-paste.noarch : Tools for using a Web Server Gateway Interface stack
```

- **yum info *PACKAGENAME*** brinda información detallada sobre un paquete, que incluye el espacio en disco necesario para la instalación.

Para obtener información sobre el servidor HTTP Apache:

```
[root@serverX ~]# yum info httpd
Loaded plugins: langpacks
Available Packages
Name       : httpd
Arch       : x86_64
Version    : 2.4.6
Release    : 17.el7
Size       : 1.1 M
Repo       : rhel_dvd
Summary    : Apache HTTP Server
URL        : http://httpd.apache.org/
License    : ASL 2.0
Description: The Apache HTTP Server is a powerful, efficient, and extensible
           : web server.
```

yum provides *PATHNAME* muestra paquetes que coinciden con el nombre de ruta especificado (que a menudo, incluye caracteres comodines).

Para encontrar paquetes que proporcionan el directorio **/var/www/html**, utilice lo siguiente:

```
[root@serverX ~]# yum provides /var/www/html
Loaded plugins: langpacks
httpd-2.4.6-17.el7.x86_64 : Apache HTTP Server
Repo                     : rhel_dvd
Matched from:
Filename                 : /var/www/html

1:php-pear-1.9.4-21.el7.noarch : PHP Extension and Application Repository
                               : framework
Repo                       : rhel_dvd
Matched from:
Filename                   : /var/www/html
```

Instalación y eliminación de software con yum

- **yum install *PACKAGENAME*** obtiene e instala un paquete de software junto con cualquier tipo de dependencia.

```
[root@serverX ~]# yum install httpd
Loaded plugins: langpacks
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.6-17.el7 will be installed
--> Processing Dependency: httpd-tools = 2.4.6-17.el7 for package:
    httpd-2.4.6-17.el7.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.6-17.el7.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package:
    httpd-2.4.6-17.el7.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package:
    httpd-2.4.6-17.el7.x86_64
--> Running transaction check
---> Package apr.x86_64 0:1.4.8-3.el7 will be installed
---> Package apr-util.x86_64 0:1.5.2-6.el7 will be installed
---> Package httpd-tools.x86_64 0:2.4.6-17.el7 will be installed
---> Package mailcap.noarch 0:2.1.41-2.el7 will be installed
--> Finished Dependency Resolution
```

Dependencies Resolved

Package	Arch	Version	Repository	Size
Installing:				
httpd	x86_64	2.4.6-17.el7	rhel_dvd	1.1 M
Installing for dependencies:				
apr	x86_64	1.4.8-3.el7	rhel_dvd	100 k
apr-util	x86_64	1.5.2-6.el7	rhel_dvd	90 k
httpd-tools	x86_64	2.4.6-17.el7	rhel_dvd	76 k
mailcap	noarch	2.1.41-2.el7	rhel_dvd	31 k

Transaction Summary

Install 1 Package (+4 Dependent packages)

Total download size: 1.4 M

Installed size: 4.3 M

Is this ok [y/d/N]:

- **yum update *PACKAGENAME*** obtiene e instala una nueva versión del paquete de software, incluidas las dependencias. Generalmente, el proceso intenta preservar los archivos de configuración, pero en algunos casos, se les cambiará el nombre si el empaquetador considera que el anterior no funcionará después de la actualización. Si no se especifica el **PACKAGENAME**, instalará todas las actualizaciones relevantes.

```
[root@serverX ~]# yum update
```

Como un kernel nuevo solo puede evaluarse mediante el inicio en ese kernel, el paquete está específicamente diseñado para que puedan instalarse múltiples versiones simultáneamente. Si el kernel nuevo no arranca, el kernel anterior sigue estando disponible. El uso de **yum update kernel** producirá la *instalación* del kernel nuevo. Los archivos de configuración contienen una lista de paquetes que "siempre deben instalarse" aunque el administrador solicite una actualización.



nota

Utilice **yum list kernel** para detallar todos los núcleos instalados y disponibles. Para ver el kernel en funcionamiento actualmente, utilice el comando **uname**. La opción **-r** mostrará solamente la versión y el lanzamiento del kernel, y la opción **-a** mostrará el lanzamiento e información adicional del kernel.

```
[root@serverX ~]# yum list kernel
Loaded plugins: langpacks
Installed Packages
kernel.x86_64                3.10.0-123.0.1.el7          @anaconda/7.0

kernel.x86_64                3.10.0-84.el7               @rhel-7-server-htb-
rpms
[root@serverX ~]# uname -r
3.10.0-123.el7.x86_64
[root@serverX ~]# uname -a
Linux demo.example.com 3.10.0-123.el7.x86_64 #1 SMP Tue Nov 26 16:51:22 EST
2013 x86_64 x86_64 x86_64 GNU/Linux
```

- **yum remove *PACKAGENAME*** elimina un paquete de software instalado junto con cualquier paquete compatible.

```
[root@serverX ~]# yum remove httpd
```



Advertencia

`yum remove` quitará los paquetes detallados y *cualquier paquete que requiera los paquetes que se van a quitar* (y los paquetes que requieran esos paquetes, y así sucesivamente). Esto puede dar lugar a una eliminación inesperada de paquetes, por lo que debe verificar detenidamente la lista de paquetes que se quitarán.

Instalación y eliminación de grupos de software con yum

- **yum** también representa el concepto de grupos, que son *colecciones* de software relacionados e instalados en forma conjunta con un fin en particular. En Red Hat Enterprise Linux 7, hay dos tipos de grupos. Los grupos regulares son colecciones de paquetes. Los *grupos de entorno* son colecciones de otros grupos que incluyen sus propios paquetes. Puede que los paquetes o grupos provistos por un grupo sean *obligatorios* (se deben instalar si el grupo está instalado), *predeterminados* (generalmente se instalan si el grupo está instalado) u *opcionales* (no se instalan donde se encuentra el grupo, a menos que se solicite en forma específica).

Al igual que **yum list**, el comando **yum group list** (o **yum grouplist**) mostrará los nombres de grupos instalados o disponibles. Algunos grupos se instalan normalmente a través de grupos de entorno y se ocultan de manera predeterminada. Los grupos ocultos también pueden enumerarse con el comando **yum group list hidden**. Si se añade la opción **ids**, también se mostrará la ID del grupo. Los grupos pueden instalarse, actualizarse, removerse o consultarse, por nombre o ID.

```
[root@serverX ~]# yum group list
```

```
Loaded plugins: langpacks
```

```
Available environment groups:
```

```
Minimal install
```

```
Infrastructure Server
```

```
File and Print Server
```

```
Web Server
```

```
Virtualization Host
```

```
Server with GUI
```

```
Installed groups:
```

```
Base
```

```
Desktop Debugging and Performance Tools
```

```
Dial-up Networking Support
```

```
Fonts
```

```
Input Methods
```

```
Internet Browser
```

```
PostgreSQL Database server
```

```
Printing client
```

```
X Window System
```

```
Available Groups:
```

```
Additional Development
```

```
Backup Client
```

```
Backup Server
```

```
...
```

- La información sobre un grupo se muestra con **yum group info** (o con **yum groupinfo**). Incluye una lista de ID de grupos o nombres de paquetes obligatorios, predeterminados u opcionales. Los ID de grupos o los nombres de paquetes pueden tener un marcador al inicio.

Marcador	Significado
=	El paquete está instalado o fue instalado como parte del grupo.
+	El paquete no está instalado, se instalará si el grupo está instalado o actualizado.
-	El paquete no está instalado, no se instalará si el grupo está instalado o actualizado.
<i>Sin marcador</i>	El paquete está instalado, pero no se instaló a través del grupo.

```
[root@serverX ~]# yum group info "Identity Management Server"
Loaded plugins: langpacks

Group: Identity Management Server
Group-Id: identity-management-server
Description: Centralized management of users, servers and authentication policies.
Default Packages:
  +389-ds-base
  +ipa-admintools
  +ipa-server
  +pki-ca
Optional Packages:
  +ipa-server-trust-ad
  +nuxwdog
  +slapi-nis
```

- El comando `yum group install` (o `yum groupinstall`) instalará un grupo que instalará sus paquetes obligatorios y predeterminados, y los paquetes de los que depende.

```
[root@serverX ~]# yum group install "Infiniband Support"
```

```
...
```

```
Transaction Summary
```

```
=====
```

```
Install 17 Packages (+7 Dependent packages)
```

```
Total download size: 9.0 M
```

```
Installed size: 33 M
```

```
Is this ok [y/d/N]:
```

```
...
```



Importante

En comparación con Red Hat Enterprise Linux 6 y con versiones anteriores, el comportamiento de los grupos **yum** ha cambiado en Red Hat Enterprise Linux 7. En RHEL 7, los grupos se tratan como *objetos* y son rastreados por el sistema. Si un grupo instalado se actualiza y el repositorio **yum** ha añadido paquetes nuevos obligatorios o predeterminados al grupo, dichos paquetes nuevos se instalarán en la actualización.

RHEL 6 y las versiones anteriores consideran la instalación de un grupo si todos sus paquetes obligatorios han sido instalados; o, en caso de que no tenga ningún paquete obligatorio, si ningún paquete predeterminado u opcional en el grupo se instaló. En *RHEL 7*, se considera la instalación de un grupo solo si **yum group install** se utilizó para su instalación. Como comando nuevo en RHEL 7, **yum group mark install GROUPNAME** puede utilizarse para marcar un grupo como instalado, y los paquetes faltantes y sus dependencias se instalarán en la próxima actualización.

Finalmente, RHEL 6 y las versiones anteriores no tenían la forma de dos palabras de los comandos **yum group**. Es decir que, en RHEL 6, el comando **yum grouplist** existía, pero el comando equivalente en RHEL 7 **yum group list** no.

Visualización del historial de transacciones

- Todas las transacciones de instalación y eliminación se registran en `/var/log/yum.log`.

```
[root@serverX ~]# tail -5 /var/log/yum.log
Feb 16 14:10:41 Installed: libnes-1.1.3-5.el7.x86_64
Feb 16 14:10:42 Installed: libmthca-1.0.6-10.el7.x86_64
Feb 16 14:10:43 Installed: libmlx4-1.0.5-7.el7.x86_64
Feb 16 14:10:43 Installed: libibcm-1.0.5-8.el7.x86_64
Feb 16 14:10:45 Installed: rdma-7.0_3.13_rc8-3.el7.noarch
```

- Un resumen de las transacciones de instalación y eliminación puede visualizarse con **yum history**.

```
[root@serverX ~]# yum history
Loaded plugins: langpacks
ID      | Login user          | Date and time    | Action(s)          | Altered
-----|-----|-----|-----|-----
  6 | Student User <student> | 2014-02-16 14:09 | Install            | 25
  5 | Student User <student> | 2014-02-16 14:01 | Install            | 1
  4 | System <unset>         | 2014-02-08 22:33 | Install            | 1112 EE
  3 | System <unset>         | 2013-12-16 13:13 | Erase              | 4
  2 | System <unset>         | 2013-12-16 13:13 | Erase              | 1
  1 | System <unset>         | 2013-12-16 13:08 | Install            | 266
history list
```


- Una transacción puede anularse con las opciones **history** **undo**:

```
[root@serverX ~]# yum history undo 6
Loaded plugins: langpacks
Undoing transaction 6, from Sun Feb 16 14:09:51 2014
  Install      dapl-2.0.39-2.el7.x86_64      @rhel-7-server-htb-rpms
Dep-Install    graphviz-2.30.1-18.el7.x86_64      @rhel-7-server-htb-rpms
Dep-Install    graphviz-tcl-2.30.1-18.el7.x86_64  @rhel-7-server-htb-rpms
  Install      ibacm-1.0.8-4.el7.x86_64      @rhel-7-server-htb-rpms
  Install      ibutils-1.5.7-9.el7.x86_64     @rhel-7-server-htb-rpms
Dep-Install    ibutils-libs-1.5.7-9.el7.x86_64     @rhel-7-server-htb-rpms
...
```

Resumen de los comandos yum

Los paquetes pueden ubicarse, instalarse, actualizarse y eliminarse por nombre o por grupos de paquetes.

Tarea:	Comando:
Enumerar paquetes instalados y disponibles por nombre	<code>yum list [NAME-PATTERN]</code>
Enumerar grupos instalados y disponibles	<code>yum grouplist</code>
Buscar un paquete por palabra clave	<code>yum search KEYWORD</code>
Mostrar detalles de un paquete	<code>yum info PACKAGENAME</code>
Instalar un paquete	<code>yum install PACKAGENAME</code>
Instalar un grupo de paquetes	<code>yum groupinstall "GROUPNAME"</code>
Actualizar todos los paquetes	<code>yum update</code>
Eliminar un paquete	<code>yum remove PACKAGENAME</code>
Mostrar historial de transacciones	<code>yum history</code>

Referencias

Páginas del manual: `yum(1)`, `yum.conf(5)`

Puede encontrar información adicional sobre `yum` disponible en la *Guía del administrador del sistema Red Hat Enterprise Linux* para Red Hat Enterprise Linux 7, que se puede encontrar en

| <https://access.redhat.com/documentation/>

Práctica: Instalación y actualización de software con yum

Habilitación de repositorios de software

yum

RH124



Objetivos

Tras finalizar esta sección, los estudiantes deberían poder habilitar e inhabilitar el uso del repositorio yum de Red Hat o de terceros.

Habilitación de repositorios de software de Red Hat

Si se registra un sistema en el servicio de administración de suscripciones, se configura automáticamente el acceso a los repositorios de software basado en las suscripciones que se adjuntan. Para ver todos los repositorios disponibles:

```
[root@serverX ~]# yum repolist all
Loaded plugins: langpacks
repo id                                repo name
status
rhel-7-server-debug-rpms/7Server/x86_64 Red Hat Enterprise Linux 7 Server (Debug
RPMs) disabled
rhel-7-server-rpms/7Server/x86_64      Red Hat Enterprise Linux 7 Server (RPMs)
enabled: 5,071
rhel-7-server-source-rpms/7Server/x86_64 Red Hat Enterprise Linux 7 Server (Source
RPMs) disabled
repolist: 5,071
```

Habilite o inhabilite los repositorios con **yum-config-manager**. Esta acción cambiará el parámetro **habilitado** en el archivo **/etc/yum.repos.d/redhat.repo**.

```
[root@serverX ~]# yum-config-manager --enable rhel-7-server-debug-rpms
Loaded plugins: langpacks
===== repo: rhel-7-server-debug-rpms =====
[rhel-7-server-debug-rpms]
async = True
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/7Server
baseurl = https://cdn.redhat.com/content/dist/rhel/server/7/7Server/x86_64/debug
cache = 0
cachedir = /var/cache/yum/x86_64/7Server/rhel-7-server-debug-rpms
check_config_file_age = True
cost = 1000
deltarpm_percentage =
enabled = 1
...
```

Habilitación de repositorios de software de terceros

Los repositorios de terceros son directorios de archivos de paquete de software provistos por una fuente que no es Red Hat y a la que se puede acceder mediante **yum** desde un sitio web, servidor FTP o sistema de archivos local. Los repositorios yum son utilizados por distribuidores de software diferentes a Red Hat, o se usan para pequeñas colecciones de paquetes locales. (Por ejemplo, Adobe ofrece parte de su software gratuito para Linux a través de un repositorio yum). El servidor del aula **content.example.com** aloja repositorios yum para esta clase.

Coloque un archivo en el directorio **/etc/yum.repos.d/** para habilitar el soporte para un nuevo repositorio de terceros. Los archivos de configuración de repositorio deben finalizar en **.repo**. La definición de repositorio contiene la URL del repositorio, un nombre, si se debe usar GPG para comprobar las firmas del paquete y, en ese caso, la URL que apunta a la clave GPG de confianza.

Con yum-config-manager

Si se conoce la URL para el repositorio yum, puede crearse un archivo de configuración con **yum-config-manager**.

```
[root@serverX ~]# yum-config-manager --add-repo="http://dl.fedoraproject.org/pub/epel/7/x86_64/"
Loaded plugins: langpacks
adding repo from: http://dl.fedoraproject.org/pub/epel/7/x86_64/

[dl.fedoraproject.org_pub_epel_7_x86_64_]
name=added from: http://dl.fedoraproject.org/pub/epel/7/x86_64/
baseurl=http://dl.fedoraproject.org/pub/epel/7/x86_64/
enabled=1
```

Se creó un archivo en el directorio **/etc/yum.repos.d** con el resultado que se muestra. Este archivo ahora puede modificarse para proporcionar un nombre personalizado y la ubicación de la clave GPG. Los administradores deberían descargar la llave en un archivo local en lugar de permitir que **yum** la recupere de una fuente externa.

```
[EPEL]
name=EPEL 7
baseurl=http://dl.fedoraproject.org/pub/epel/7/x86_64/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
```


Paquete de configuración de RPM para el repositorio

Algunos repositorios proporcionan este archivo de configuración y la clave pública de GPG como parte del paquete de RPM que puede descargarse e instalarse con **yum localinstall**. Un ejemplo de esto es el proyecto voluntario de Paquetes extra para Enterprise Linux (EPEL), que proporciona software no admitido por Red Hat, pero que es compatible con Red Hat Enterprise Linux.

Instalación del paquete de repositorio EPEL de Red Hat Enterprise Linux 7:

```
[root@serverX ~]# rpm --import http://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-7
[root@serverX ~]# yum install http://dl.fedoraproject.org/pub/epel/7/x86_64/e/epel-release-7-2.noarch.rpm
```

A menudo, los archivos de configuración enumeran varias referencias de repositorio en un solo archivo. Cada referencia de repositorio comienza con un nombre de una sola palabra entre corchetes.

```
[root@serverX ~]# cat /etc/yum.repos.d/epel.repo
[epel]
name=Extra Packages for Enterprise Linux 7 - $basearch
#baseurl=http://download.fedoraproject.org/pub/epel/7/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-7&arch=$basearch
failovermethod=priority
enabled=1
gpgcheck=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7

[epel-debuginfo]
name=Extra Packages for Enterprise Linux 7 - $basearch - Debug
#baseurl=http://download.fedoraproject.org/pub/epel/7/$basearch/debug
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-debug-7&arch=$basearch
failovermethod=priority
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
gpgcheck=1

[epel-source]
name=Extra Packages for Enterprise Linux 7 - $basearch - Source
#baseurl=http://download.fedoraproject.org/pub/epel/7/SRPMS
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-source-7&arch=$basearch
failovermethod=priority
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
gpgcheck=1
```

El parámetro **habilitado=0** puede incluirse de manera que se defina un repositorio, pero no se busque de manera predeterminada. Los repositorios pueden habilitarse o inhabilitarse en forma persistente con **yum-config-manager** o en forma provisoria con las opciones **--enablerepo=PATTERN** y **--disablerepo=PATTERN** en yum.



Advertencia

Antes de instalar los paquetes firmados, instale la clave GPG de RPM. Esta acción verificará que los paquetes pertenezcan a una clave que se haya importado. De lo contrario, **yum** le advertirá que le falta la clave. (La opción **- -nogpgcheck** puede usarse para omitir las claves GPG faltantes, pero esto podría provocar que se instalen paquetes adulterados o dudosos en el sistema y que, posiblemente, comprometan la seguridad).



Referencias

Es posible encontrar información adicional en la sección sobre la configuración de yum y repositorios yum en la *Guía del administrador del sistema Red Hat Enterprise Linux* para Red Hat Enterprise Linux 7, que se puede encontrar en

| <https://access.redhat.com/documentation/>

Páginas del manual: **yum(5)**, **yum.conf(1)**, **yum-config-manager(1)**

Práctica: Habilitar repositorios de software

Análisis de los archivos del paquete RPM

RH124

Objetivos

Tras finalizar esta sección, los estudiantes deberían poder analizar e instalar los archivos de paquete descargados.

Análisis de paquetes descargados con rpm

La utilidad `rpm` es una herramienta de nivel bajo que es útil para obtener información acerca del contenido de los archivos de paquete y los paquetes instalados. Obtiene su información desde una base de datos local o de los propios archivos del paquete.

La forma general de una consulta es:

- `rpm -q [select-options] [query-options]`
- `rpm --query [select-options] [query-options]`

Consultas RPM: opciones de selección

- `-q -a`: todos los paquetes instalados
- `-q PACKAGENAME`: **PACKAGENAME** actualmente instalado

```
[root@serverX ~]# rpm -q yum
yum-3.4.3-118.el7.noarch
```

- **-q -p PACKAGEFILE.rpm**: archivo de paquete con el nombre PACKAGEFILE.rpm

```
[root@serverX ~]# rpm -q -p http://content.example.com/rhel7.0/x86_64/dvd/Packages/  
yum-utils-1.1.31-24.el7.noarch.rpm  
yum-utils-1.1.31-24.el7.noarch.rpm
```

- **-q -f FILENAME**: qué paquete proporciona el FILENAME

```
[root@serverX ~]# rpm -q -f /etc/yum.repos.d  
yum-3.4.3-118.el7.noarch
```


Consultas de RPM: información sobre el contenido de los paquetes

- `-q`: especifica el nombre y la versión del paquete; comparar con `yum list`
- `-q -i`: información sobre el paquete; comparar con `yum info`
- `-q -l`: enumera los archivos instalados por el paquete especificado

```
[root@serverX ~]# rpm -q -l yum-rhn-plugin
/etc/yum/pluginconf.d/rhnplugin.conf
/usr/share/doc/yum-rhn-plugin-2.0.1
/usr/share/doc/yum-rhn-plugin-2.0.1/LICENSE
/usr/share/locale/aln/LC_MESSAGES/yum-rhn-plugin.mo
```

- `-q -c`: enumera sólo los archivos de configuración

```
[root@serverX ~]# rpm -q -c yum-rhn-plugin
/etc/yum/pluginconf.d/rhnplugin.conf
```

- **-q -d:** enumera sólo los archivos de documentación

```
[root@serverX ~]# rpm -q -d yum-rhn-plugin
/usr/share/doc/yum-rhn-plugin-2.0.1/LICENSE
/usr/share/man/man5/rhnplugin.conf.5.gz
/usr/share/man/man8/rhnplugin.8.gz
/usr/share/man/man8/yum-rhn-plugin.8.gz
```

- **-q --scripts:** enumera los scripts de la shell que pueden ejecutarse una vez que se instaló o eliminó el paquete

```
[root@serverX ~]# rpm -q --scripts openssh-server
preinstall scriptlet (using /bin/sh):
getent group sshd >/dev/null || groupadd -g 74 -r sshd || :
getent passwd sshd >/dev/null || \
    useradd -c "Privilege-separated SSH" -u 74 -g sshd \
    -s /sbin/nologin -r -d /var/empty/sshhd sshd 2> /dev/null || :
postinstall scriptlet (using /bin/sh):

if [ $1 -eq 1 ] ; then
    # Initial installation
    /usr/bin/systemctl preset sshd.service sshd.socket >/dev/null 2>&1 || :
fi
preuninstall scriptlet (using /bin/sh):

if [ $1 -eq 0 ] ; then
    # Package removal, not upgrade
    /usr/bin/systemctl --no-reload disable sshd.service sshd.socket > /dev/null
    2>&1 || :
    /usr/bin/systemctl stop sshd.service sshd.socket > /dev/null 2>&1 || :
fi
postuninstall scriptlet (using /bin/sh):

/usr/bin/systemctl daemon-reload >/dev/null 2>&1 || :
if [ $1 -ge 1 ] ; then
    # Package upgrade, not uninstall
    /usr/bin/systemctl try-restart sshd.service >/dev/null 2>&1 || :
fi
```

- **-q --changelog**: enumera la información de cambios para el paquete

```
[root@serverX ~]# rpm -q --changelog audit
* Thu Oct 03 2013 Steve Grubb <sgrubb@redhat.com> 2.3.2-3
resolves: #828495 - semanage port should generate an audit event

* Thu Aug 29 2013 Steve Grubb <sgrubb@redhat.com> 2.3.2-2
resolves: #991056 - ausearch ignores USER events with -ua option
...
```

```
[root@serverX ~]# wget http://classroom/pub/materials/wonderwidgets-1.0-4.x86_64.rpm
[root@serverX ~]# rpm -q -l -p wonderwidgets-1.0-4.x86_64.rpm
/etc/wonderwidgets.conf
/usr/bin/wonderwidgets
/usr/share/doc/wonderwidgets-1.0
/usr/share/doc/wonderwidgets-1.0/README.txt
```



nota

El comando **repoquery** también puede usarse para obtener información sobre paquetes y sus contenidos. La diferencia con **rpm** es que busca esa información en los repositorios yum, en lugar de hacerlo en la base de datos local de los paquetes instalados.

Uso de yum para instalar archivos de paquete locales

El comando `yum localinstall PACKAGEFILE.rpm` se puede usar para instalar los archivos de paquetes de manera directa. Automáticamente, descarga todas las dependencias que tiene el paquete desde cualquier repositorio `yum` configurado.

```
[root@serverX ~]# yum localinstall wonderwidgets-1.0-4.x86_64.rpm
```

```
[root@serverX ~]# rpm -q wonderwidgets
wonderwidgets-1.0-4.x86_64
```



nota

`rpm -ivh PACKAGEFILE.rpm` también se puede usar para instalar archivos de paquete. Sin embargo, el uso de `yum` ayuda a mantener un historial de transacciones que conserva `yum` (consulte `yum history`).



Advertencia

Tenga cuidado al instalar paquetes de terceros, no solo por el software que pueden instalar, sino también porque el RPM puede ejecutar scripts arbitrarios como usuario `root` durante el proceso de instalación.

Extracción de archivos de los paquetes RPM

Los archivos del paquete RPM pueden extraerse sin instalar el paquete con **cpio**, que es una herramienta para archivar, como **zip** o **tar**. Canalice los resultados de **rpm2cpio PACKAGEFILE.rpm** en **cpio -id** y extraerá todos los archivos almacenados en el paquete RPM. Se crearán árboles de subdirectorios, según sea necesario, con respecto al directorio de trabajo actual.

Seleccione los archivos que también puedan extraerse mediante la especificación de la ruta del archivo:

```
[root@serverX ~]# rpm2cpio wonderwidgets-1.0-4.x86_64.rpm | cpio -id "**txt"
11 blocks
[root@serverX ~]# ls -l usr/share/doc/wonderwidgets-1.0/
total 4
-rw-r--r--. 1 root root 76 Feb 13 19:27 README.txt
```

Resumen de los comandos de consulta rpm

Los paquetes instalados pueden consultarse directamente con el comando **rpm**. Agregue la opción **-p** para consultar un archivo de paquete antes de la instalación.

Tarea:	Comando:
Muestra información sobre el paquete.	<code>rpm -q -i NAME</code>
Enumera todos los archivos que están incluidos en el paquete.	<code>rpm -q -l NAME</code>
Enumera los archivos de configuración incluidos en un paquete.	<code>rpm -q -c NAME</code>
Enumera los archivos de documentación incluidos en un paquete.	<code>rpm -q -d NAME</code>
Muestra un resumen breve del motivo de lanzamiento del paquete nuevo.	<code>rpm -q --changelog NAME</code>
Muestra la secuencia de comandos de shell incluida en un paquete.	<code>rpm -q --scripts NAME</code>



Referencias

Páginas del manual **yum(8)**, **rpm(8)**, **repoquery(1)**, **rpm2cpio(8)** y **cpio(1)**

Práctica: Trabajar con los archivos de paquete del RPM