

yum, centos

Yum

Yum es la herramienta de Centos para la gestión de paquetes

Instalar únicamente actualizaciones de seguridad

En versiones anteriores había que instalar un plugin **yum-security** (para Centos 5) o yum-plugin-security (para Centos 6 y 7)

Para comprobar si lo tenemos instalado

```
rpm -qa |grep -i yum-security
```

```
yum install yum-plugin-security
```



Desde Centos 7 ya viene incluido por lo que sólo es necesario instalar el plugin

- Para ver un resumen de las actualizaciones que necesita nuestro sistema

```
yum updateinfo summary
```

- Para actualizar las listas de información disponibles

```
yum updateinfo list available
```

- Para ver las actualizaciones necesarias de una lista en concreto

```
yum updateinfo list bugfix
```

- Para ver la lista de actualizaciones de seguridad necesarias para nuestro sistema

```
yum updateinfo list security all
```

- Para ver información sobre una actualización en concreto

```
yum updateinfo ELBA-2018-0840
```

- Para actualizar los paquetes a la versión del mismo más actualizada

```
yum --security update-minimal
```

- Para actualizar los paquetes a la última versión disponible

```
yum --security update
```

Más información en <https://linux.die.net/man/8/yum-security>

Script para actualizaciones automáticas

Creamos el siguiente script para cron.

- Si queremos que se ejecute cada día → /etc/cron.daily/yumupdate.sh
- Si queremos que sea una vez en semana → /etc/cron.weekly/yumupdate.sh

```
#!/bin/bash
YUM=/usr/bin/yum
$YUM -y -R 100 -d 0 -e 0 update yum
$YUM -y -R 10 -e 0 -d 0 update
```

Donde:

- La primera línea actualiza yum
- la segunda línea actualiza el sistema
- la opción -R indica el tiempo máxima de espera antes de ejecutar el comando
- -e establece el nivel de error. 0 es que sólo nos informa de errores críticos
- -d establece el nivel de debug



no hay que olvidarse de darle los permisos de ejecución al fichero de script

Ver los paquetes instalados en el sistema

```
yum list installed
```

Buscar paquete

Por ejemplo para buscar paquetes que el nombre contenga php

```
yum list installed | grep -i php
```



el -i es para que no distinga entre mayúsculas o minúsculas

Información sobre cuando se instaló

Listado de la fecha de instalación de todos los paquetes

```
rpm -qa -last
```

Para saber cuándo se instaló un paquete en particular

```
rpm -q nombre_paquete -last
```

Deshacer cambios o Roll back

Con el comando

```
yum history
```

Obtenemos un histórico de los cambios.

```
[root@]# yum history
```

ID	Linea de comandos	Día y hora	Acción(es)	Modific
29	-y install php php-curl	2018-05-21 13:21	Install	3
28	update	2018-05-21 13:17	I, U	10
27	install yum-utils	2018-05-21 13:14	Install	4
26	install http://rpms.rem	2018-05-21 13:00	Install	1
25	install php-gd	2018-05-21 12:23	Install	11
24	install wget	2018-05-21 11:52	Install	1
23	install httpd	2018-05-21 11:46	Install	5
22	update	2018-05-21 11:43	Update	27

Podemos obtener más información sobre un determinado cambio con **yum history info ID**

```
yum history info 29
```

que daría como salida

```
ID de transacción : 29
Hora inicial      : Mon May 21 13:21:47 2018
Rpmdb inicial    : 403:f5122f693c232264926551793ecd11b87b85e057
Hora final       : 13:21:49 2018 (2 segundos)
Rpmdb final      : 406:357a12e260ab03908297ee2c9985d509ae422f03
Usuario          : root <root>
Codigo-obtenido  : Exito
Línea de comando : -y install php php-curl php-cli php-gd
Transacción realizada con:
  Instalado      rpm-4.11.3-32.el7.x86_64 @base
  Instalado      yum-3.4.3-158.el7.centos.noarch @base
  Instalado      yum-metadata-parser-1.1.4-10.el7.x86_64 @anaconda
  Instalado      yum-plugin-fastestmirror-1.1.31-45.el7.noarch @base
```

Paquetes modificados:

```
Instalación de dependencias libargon2-20161029-2.el7.x86_64 @epel
Instalar php-7.2.5-1.el7.remi.x86_64 @remi-php72
Instalar php-cli-7.2.5-1.el7.remi.x86_64 @remi-php72
```

Para deshacer el cambio ejecutaríamos **yum history undo <ID>**

```
yum history undo 29
```

Pero si hemos hecho varios cambios y queremos volver a un punto concreto ejecutaremos **yum history rollback <ID>**

```
yum history rollback 22
```

Descargar paquetes sin instalarlos

A veces es útil descargar ciertos paquetes para luego poder copiarlos a un servidor que no tiene conexión a Internet. Para eso podemos usar yum de la siguiente forma **yum install -downloadonly -downloadaddir=/directorio_destino <paquete>** Por ejemplo para descargar el servidor apache.

```
yum install --downloadonly --downloadaddir=/root/paquetes httpd
```

Una vez copiado los paquetes a la máquina destino, para instalarlos **yum -nogpgcheck localinstall <paquete>**

```
yum --nogpgcheck localinstall httpd-2.4.6-80.el7.centos.x86_64.rpm
```

FAQ

Si al hacer un yum update obtenemos un error del tipo metadata file does not match checksum se soluciona añadiendo al fichero /etc/yum.conf lo siguiente

```
http_caching=packages
```

Ejecutamos entonces un

```
yum clean metadata
```

o bien

```
yum clean all
```

y ya podemos volver a ejecutar el yum update sin problemas.

HTTP Error 404

si al ejecutar yum update nos da un error **HTTP Error 404 : Not found** podemos ejecutar para solucionarlo



```
yum clean all
```

```
yum update
```

Referencias

- <http://rm-rf.es/rollback-de-instalaciones-y-actualizaciones-con-yum-history/>
- <https://blog.carreralinux.com.ar/2018/05/usar-yum-descargar-paquetes/>
- Artículo original de <http://rm-rf.es/6-trucos-utiles-del-gestor-de-paquetes-yum/>
- <https://www.elarraydejota.com/como-realizar-unicamente-actualizaciones-de-seguridad-en-red-hat-y-derivadas/>

From:

<http://intrusos.info/> - **LCWIKI**

Permanent link:

<http://intrusos.info/doku.php?id=linux:centos:yum&rev=1620985745>

Last update: **2023/01/18 14:18**

