

Semana 12:

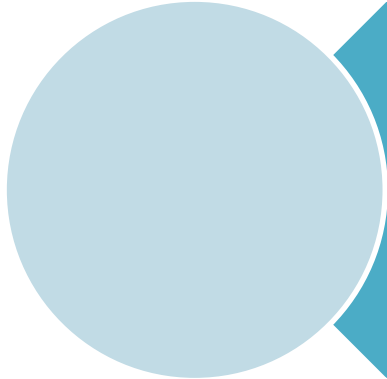
Administración de Paquetes en Linux

Introducción

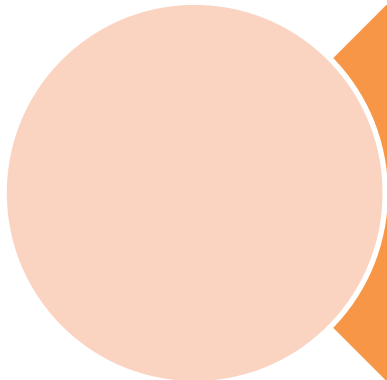
- La instalación de programas es una tarea común y necesaria en los sistemas operativos
- En Linux los programas vienen en diferentes formatos para su instalación, lo cual va a determinar la forma de instalarlos.
- En esta sesión describiremos los diferentes formatos de los paquetes de instalación y la forma de instalarlos.



Objetivos de la Sesión



Gestiona la instalación, eliminación y actualización de paquetes.



Selecciona la mejor manera de instalar un paquete en Linux

Instalación de paquetes en Linux (RPM)

- A diferencia de otros sistemas operativos, con Linux y Unix no es habitual disponer de software proporcionado con un programa de instalación interactivo (Wizard o archivo .exe).
- Con Linux es muy habitual disponer de varios productos, herramientas, actualizaciones, etc; en forma de paquetes (packages).
- Un paquete es un archivo que contiene el producto a instalar y unas reglas. Según su contenido, su tamaño puede ser muy imponente.
- El núcleo y todos sus módulos son, por ejemplo proporcionados en esta forma.

Instalación de paquetes en Linux (cont.)

A la hora de instalar paquetes se debe considerar lo siguiente.

- Gestión de las dependencias: sólo se podrá instalar el producto si los productos que él mismo utiliza están ya presentes.
- Preinstalación: se deben prever acciones antes de poder instalar el producto (cambiar permisos, crear directorios, etc.).
- Postinstalación: se deben prever acciones después de la instalación del producto (parámetros de un archivo de configuración, compilación anexa, etc.)

En Red Hat, Fedora, SuSE, Mandriva y algunas otras distribuciones el formato de paquete por defecto es el RPM (Red Hat Package Manager).

Instalación, actualización y eliminación de un RPM

- Se instala un paquete rpm con el parámetro -i. Con -h visualizará la progresión de instalación con el signo #.

```
# rpm -ih php-4.1.2-2.1.8.i586.rpm
```

La instalación no funcionará si no se resuelven las dependencias

- La actualización de un producto hacia una versión superior desde un paquete se hace con el parámetro -U. Si el paquete no está instalado, la actualización asume el papel de instalación.

```
# rpm -Uvh php-4.1.3-1.i586.rpm
```

- La supresión se efectúa con el parámetro -e. Sin embargo, observe que deberá pasar como argumento el nombre del paquete, y no el nombre del archivo de paquete.

```
# rpm -e php
```

Consultas RPM

- Se puede consultar la base de datos RPM con el parámetro `-q`. Además, podemos soportarnos con el parámetro `-a` que lista todos los paquetes instalados y el parámetro `-i` que muestra información (resumen) del paquete.

```
# rpm -qa
```

```
# rpm -qa openssh*
```

```
# rpm -qai openssh*
```

```
[root@linux ~]# rpm -qai openssh*
Name       : openssh-clients
Version    : 7.4p1
Release    : 21.el7
Architecture: x86_64
Install Date: Thu 21 May 2020 02:21:41 PM -05
Group      : Applications/Internet
Size       : 2643176
License    : BSD
Signature  : RSA/SHA256, Thu 22 Aug 2019 04:37:26 PM -05, Key ID 24c6a8a7f4a80e
b5
Source RPM : openssh-7.4p1-21.el7.src.rpm
Build Date : Thu 08 Aug 2019 08:40:49 PM -05
Build Host : x86-01.bsys.centos.org
Relocations : (not relocatable)
```

Repositorios en CentOS

- YUM es un programa de gestión de paquetes. Recupera los paquetes dentro de los repositorios para instalarlos y gestiona las dependencias en lugar de usted.
- YUM (Yellow dog Updater Modified) se utiliza principalmente en las distribuciones Red Hat Enterprise Linux, Fedora y CentOS.
- Los repositorios se ubican o en el archivo de configuración principal (/etc/yum.conf), o en el directorio “/etc/yum.repos.d”.
- Los parámetros que tienen los archivos de los repositorios son los siguientes:

name: el nombre largo del repositorio, detallado.

baseurl: la URL del repositorio.

gpgcheck: requiere una verificación de la firma GPG del repositorio.

enabled: si no está definido o en 1, el repositorio está activo.

gpgkey: ruta de la clave pública GPG.



Parámetros del comando YUM

- Listar los repositorios disponibles para su uso.
yum repolist
- Listar un paquete
yum list openssh*
- Mostrar información de un paquete
yum info httpd
- Hace una búsqueda de todos los paquetes relacionados con el que se especifique
yum search httpd
- Instalar / desinstalar un paquete con todas sus dependencias
yum install httpd | yum remove httpd
- Ver historial de instalaciones
yum history
- Ver información de alguna instalación específica
yum history info <ID>

Parámetros del comando YUM

- Para listar las actualizaciones de todos los paquetes ejecute el siguiente comando.
yum list updates
- Para actualizar todos los paquetes del sistema ejecute lo siguiente.
yum update
- Para actualizar un paquete específico realice lo siguiente.
yum update <NombreDelPaquete>
- Si desea actualizar los paquetes y que se eliminen los paquetes considerados obsoletos ejecute el siguiente comando.
yum upgrade

Nota: Tener cuidado a la hora de usar “yum upgrade” porque dentro de los paquetes obsoletos que eliminará, puede estar uno que sea importante para el servidor.

Creando nuestro repositorio local

- Los archivos que apuntan a un repositorio siempre deben llamarse *.repo (entiéndase el símbolo * como cualquier texto).
- El archivo creado debe guardarse en “/etc/yum.repos.d” .
- Copiamos todos los archivos que contiene el disco de instalación de CentOS 7.X a un directorio cualquiera, para luego hacer referencia a ese directorio en nuestro archivo *.repo

```
[root@linux ~]# mkdir /cdrom
[root@linux ~]# mount /dev/cdrom /mnt/
mount: /dev/sr0 is write-protected, mounting read-only
[root@linux ~]# cp -r /mnt/[CGLRiErT]* /cdrom
[root@linux ~]# mkdir /cdrom/Packages
[root@linux ~]# cp /mnt/a* /cdrom/Packages/
[root@linux ~]# cp /mnt/Packages/a* /cdrom/Packages/
```

Nota: para efectos de la demostración, solo se copió los paquetes que inician ,como nombre, con la letra a; ya que se necesitaría tener 4 GB aprox de espacio libre en el disco duro del servidor linux.

Creando nuestro repositorio local

- Procedemos a crear el archivo “/etc/yum.repos.d/tecsup.repo” y tendrá como contenido lo siguiente.

```
[root@linux yum.repos.d]# pwd
/etc/yum.repos.d
[root@linux yum.repos.d]# cat tecsup.repo
[Tecsup]
name= Repositorio local Tecsup
baseurl= file:///cdrom
enabled=1
gpgcheck=0
[root@linux yum.repos.d]#
```

- Validando el acceso al repositorio.

```
[root@linux yum.repos.d]# yum repolist
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: mirror.orbyta.com
* extras: mirror.orbyta.com
* updates: mirror.orbyta.com
```

repo id	repo name	status
Tecsup	Repositorio local Tecsup	4,067
!base/7/x86_64	CentOS-7 - Base	10,070
extras/7/x86_64	CentOS-7 - Extras	397
updates/7/x86_64	CentOS-7 - Updates	754
repolist:		15,288



Instalando un repositorio remoto

- Instalaremos el repositorio más popular y utilizado en plataformas CentOS y RHEL.

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

Nota: para la instalación de paquetes debe ser necesario que el computador cuente con servicio de internet.

- Al instalar este repositorio, automáticamente se crearán los archivos *.repo en “/etc/yum.repos.d” y nuestro servidor tendrá acceso al repositorio EPEL.

```
[root@linux ~]# ls /etc/yum.repos.d/
CentOS-Base.repo      CentOS-fasttrack.repo  CentOS-Vault.repo  tecsup.repo
CentOS-CR.repo        CentOS-Media.repo      epel.repo
CentOS-Debuginfo.repo CentOS-Sources.repo     epel-testing.repo
```

```
[root@linux ~]# yum repolist
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: mirror.orbyta.com
* epel: mirror.twinklakes.net
* extras: mirror.orbyta.com
* updates: mirror.orbyta.com
```

repo id	repo name	status
Tecsup	Repositorio local Tecsup	4,067
base/7/x86_64	CentOS-7 - Base	10,070
epel/x86_64	Extra Packages for Enterprise Linux 7 - x86_64	13,323
extras/7/x86_64	CentOS-7 - Extras	397
updates/7/x86_64	CentOS-7 - Updates	754

Copias de seguridad y trabajo con tar-gzip-bzip2

- Es un comando de fácil uso para guardar y recuperar archivos en una copia de respaldo. La copia de respaldo puede ser una unidad de cinta, aunque también puede ser un archivo normal.
- Este comando permite “empaquetar” archivos y carpetas; en un archivo tar. Por otra parte, puede además añadirle compresión para que tenga un menor tamaño.

Opciones:

cf : Crea un archivo tar

cvf: Crea un archivo tar y muestra en consola lo que empaqueta

tvf : Muestra contenido del archive tar

xf : Desempaqueta archivos desde un archivo tar

xvf : Desempaqueta archivos desde un archivo tar pero muestra en consola lo

que realiza

- Empaquetando el directorio /etc
tar cvf etc.tar /etc
- Desempaquetando el archivo tar
tar xvf etc.tar



Compresión de tamaño en archivos TAR

- Los formatos de compresión más comunes son:

gzip	.gz
bzip2	.bz2

- El standard de compresión bajo los sistema GNU/Linux es el formato gzip.
- Se puede almacenar y comprimir archivos conjuntamente; usando gzip o bzip2.
- Empaquetando un archivo con compresión gzip.

```
# tar czvf etc.tar.gz /etc
```

- Desempaquetando un archivo con compresión gzip.

```
# tar xzvf etc.tar.gz
```

- Empaquetando un archivo con compresión bzip2. Para ver el contenido de un archivo tar use “tvf”

```
# tar cjvf etc.tar.bz2 /etc | # tar tvf *.tar
```

- Desempaquetando un archivo con compresión bzip2.

```
# tar xjvf etc.tar.bz2
```

Nota: con el parámetro -C puede especificar para que desempaquete en otra ubicación.



Iniciando archivos ejecutables (.sh, .bin, .run)

- La forma de iniciar archivos ejecutable , en una plataforma, linux es anteponiendo un ./ al inicio del nombre del archivo. Además, dicho archivo debe tener habilitado el permiso de ejecución (x).

```
[root@linux webmin-1.941]# ls -l setup.sh
-rwxr-xr-x. 1 root bin 25016 Jan 14 19:10 setup.sh
[root@linux webmin-1.941]# ./setup.sh
```

- Procederemos a crear un script básico que imprima en pantalla nuestro nombre.

```
[root@linux tmp]# echo '#!/bin/bash' > nombre
[root@linux tmp]# echo "echo \"Roberto Rodríguez\"" >> nombre
[root@linux tmp]# ls -l
total 4
-rw-r--r--. 1 root root 36 Jun 14 16:06 nombre
[root@linux tmp]# chmod 744 nombre
[root@linux tmp]# ls -l
total 4
-rwxr--r--. 1 root root 36 Jun 14 16:06 nombre
[root@linux tmp]# ./nombre
Roberto Rodríguez
```


FIN DE LA UNIDAD

Bibliografía

- Adelstein, Torn (2007). Administración de Sistemas Operativos Linux. Madrid: Anaya Multimedia (005.43L/A23)
- Alegría Loainaz, Iñaki (2005). Linux Administración del Sistema y la Red. Madrid: Pearson Educación (005.43L/A37)
- Negus, Christopher (2013). Linux, Bible. Albany NY: A.De Boeck (005.43L/N36)

Próxima Sesión:

Configuración de Red en Linux y DHCP

Sistemas Operativos