

# ¿Como limpiar nuestro Debian?



A todos nos pasa y nos ha pasado (sobre todo al inicio) de instalar y desinstalar aplicaciones, probar programas y configuraciones...y por supuesto el sistema sufre un poco a causa de estas “experiencias”.

En esta pequeña y básica guía veremos algunas soluciones bastante cómodas (seguramente mucho mejor que formatear y empezar desde cero) para tratar de volver a la “normalidad” nuestra amada Debian. Y como prevenir es mejor que curar.....allá vamos!

Todos estaremos de acuerdo en que trabajar en un sistema limpio y que nos permita limpiarlo es mas cómodo que uno sucio y desordenado,y sobretodo si se trata de interventos a nivel de sistema!.

## Checkinstall

*Checkinsall* es una utilidad que permite de crear paquetes Debian a partir de los fuentes.

**CUIDADO, porque no son paquetes completos, sino que simplemente ayudan a seguir el rastro de files instalados por el programa, facilitando la actualización y el borrado del mismo.** Comenzamos con la instalación:

```
# aptitude install checkinstall
```

Su uso es bastante fácil e intuitivo,la configuración de las fuentes ( *./configure -alguna-opción*) y la compilación (*make*) se deben de llevar a cabo, mientras que el comando:

```
# make install
```

**debe ser sustituido por :**

```
# checkinstall
```

Durante la ejecución vendrán formuladas algunas preguntas, como por ejemplo el tipo de paquete a crear (siendo los posibles: deb, rpm, tgz).

Tenemos a disposición otras opciones :

**-y, -default**

Acepta los valores por defecto

**-pkgname=name**

Impone el nombre de un paquete

**-pkgversion=versión**

Impone la versión

**-A, -arch, -pkgarch=arch**

Impone la arquitectura

**-pkgrelease=release**

Impone la release versión

**-pkggroup=group**

Impone el grupo al cual el paquete debe pertenecer (doc, juegos, net...)

**-pakdir=directory**

Donde salvar el paquete

**-maintainer=email\_addr**

El maintainer del paquete

**-dpkgflags=flags**

Eventuales flags para dpkg

**-bk**

Realiza un Backup de los archivos que serán pisados.

Una vez terminado el proceso, el paquete vendrá automáticamente instalado.

**Nota: debemos tener en cuenta que serán señalados eventuales dependencias, sustituciones e/o conflictos, esto debido a que el objetivo de checkinstall es aquel de poder rastrear los files generados y no el de crear un .deb oficial.**

## Debfoster

El objetivo de este programa es de mostrar los paquetes que NO han sido instalados como dependencias, y una vez individuado/s, sera mostrada una lista indicando los paquetes “retenidos” por este/os.

```
#aptitude install debfoster
```

Salida de ejemplo en consola:

```
libxml-libxml-perl is keeping the following 3 packages installed:  
libxml-libxml-common-perl libxml-namespacesupport-perl libxml-sax-  
perl  
Keep libxml-libxml-perl? [Ynpsiuqx?], [H]elp:
```

Ahora podemos elegir cosa hacer:

**Y**

mantiene el paquete instalado

**n**

Borra el paquete señalado

**p**

Borra el paquete señalado y todos los paquetes retenidos

**s**

Salta la pregunta

**i**

Visualiza informcion sobre el paquete

**?**

Igual a i

**u**

Vuelve a formular la pregunta anterior

**q**

Cancela la operación sin realizar modificaciones

**x**

Realiza la cancelación de los paquetes

**h**

Ayuda

Cuando realizamos la limpieza con este programa,seria importante tener algunas consideraciones:

**-La información mostrada no toma en cuenta eventuales programas compilados desde las fuentes.**

**-Leer con atención las librerías mostradas como “retenidas”,puede suceder que se muestre una librería de desarrollo NO necesaria para la ejecución de otras aplicaciones,pero indispensable para la compilación (libncurses5-dev por ejemplo).**

## Deborphan

Este paquete genera una lista de los paquetes huérfanos en el sistema. Por paquete huérfano entendemos aquellas librerías que ya no son necesarias, osea,que ningún paquete instalado la indica como dependencia.**Pero...atención con los programas compilados desde las fuentes (con make install o checkinstall) ya que no serán controladas sus dependencias, por lo que podríamos causar algún problema de funcionamiento.**

```
#aptitude install deborphan
```

Una opción interesante es `-libdev`, que nos genera una lista con las librerías de desarrollo (que terminan con `-dev`) no necesarias.

Para ver los paquetes huérfanos basta lanzar el comando

```
#deborphan
```

o

```
#deborphan --libdevel
```

Es posible hacer que apt-get lea la lista de paquetes generada por deborphan:

```
# apt-get --purge remove `deborphan`
```

```
# apt-get --purge remove `deborphan --libdev`
```

La opción `-purge` como ya sabemos elimina los archivos de configuración del paquete. Para los poco amantes de la consola podemos instalar *gtkorphan*, el cual es una muy fácil e intuitiva interfaz gráfica para *deborphan*.

```
#aptitude install gtkorphan
```

## ***Archivos de configuración.***

Eliminando los archivos de configuración liberamos espacio en nuestro disco (tarde o temprano muy valioso) y mantenemos limpio el directorio */etc*.

Con el siguiente comando podremos borrar los archivos de configuración que han sido dejados por paquetes desinstalados sin la opción `-purge`.

```
# dpkg --purge `COLUMNS=300 dpkg -l | egrep "^rc" | cut -d' ' -f3`
```

Quizás te marees con esta cantidad de letras, ¡tranquilo...allá vamos!

```
dpkg --purge paquetes
```

Borra los archivos de configuración de todos los paquetes indicados. Estos paquetes los obtenemos tramite el comando:

```
`COLUMNS=300 dpkg -l | egrep "^rc" | cut -d' ' -f3`
```

que a su vez genera la lista de los paquetes que no han sido borrados completamente (osea aquellos que cuando “se fueron”, olvidaron de llevarse la configuración).

`COLUMNS=300` → modifica la longitud de la cadena, evitando cortar los nombres de los paquetes (`COLUMNS` es una variable de ambiente).

`dpkg -l` → lista los paquetes disponibles.

`egrep “^rc”` → lista las cadenas de nombres que comienzan con “rc”, que es el estado que indica que el paquete ha sido borrado.

`cut -d' ' -f3` → devuelve el nombre del paquete, usa como limitador (-d) de campo un espacio ( ' ') y toma el tercer campo (-f3.field en ingles).

# Localepurge

Este es un paquete que debería ser instalado en el sistema desde el inicio, o al menos eso pienso. El objetivo de este paquete es eliminar los idiomas que instala un paquete y no nos sirven. Normalmente nos sirven dos idiomas:

```
*en (idioma "de fabrica")
*es_ES (nuestro idioma)
```

Entre otras debemos elegir las variaciones que son utilizadas en nuestro sistema (es\_ES@euro y en\_US o la que sea).

No es aconsejable borrar el ingles, ya que es el idioma por defecto del sistema, y para evitar problemas , no molesta.

Ssí que, si aun no lo hemos hecho:

```
#aptitude install localepurge
```

Durante la configuración sera mostrada una lista con los idiomas, de estas, deberemos seleccionar las que nos interesa.

Normalmente no es necesario, pero por si las moscas, ejecutamos:

```
#localepurge
```

De ahora en adelante cada vez que instalemos un paquete, los idiomas inútiles serán borrados, ganando así, espacio en disco.

## Conclusion:

Estas son las técnicas de base para tener en orden y limpia nuestro Debian, los cambios se hacen visibles sobretodo en lo que respecta a espacio en disco.

No olvidemos la utilización de:

```
apt-get o aptitude clean
apt-get o aptitude autoclean
```

se encarga de eliminar automáticamente los paquetes .deb descargados de los repositorios que ya son inservibles o que ya tienen una versión mas nueva instalada en el sistema.

Deberemos limpiar toda cache que se nos cruce en el camino.

Una aplicación que personalmente recomiendo, siempre y cuando sea usada con cuidado:

### [KleanSweep](#)

Con esta aplicación mantendremos limpia nuestra */home* de una enorme cantidad de cosas inútiles que se van acumulando.

Si no lo deseamos instalar, siempre podemos recurrir a:

```
$ ls -a
```

lo cual nos permitirá ver los archivos y directorios de configuración, que son aquellos ocultos, los que tienen un `.` (punto) al inicio.

Una vez individuado el directorio a borrar:

```
$ rm -r nombredeldirectorio
```

donde la opción `-r` indica la recursividad de la operación, permitiendo el borrado del directorio.

**Para complementar esta “guía” :**

<http://www.esdebian.org/articulos/23921/limpiando-nuestro-debian>

Fuente:

[www.codigounix.com.ar](http://www.codigounix.com.ar)

[www.esdebian.org](http://www.esdebian.org)