

Repositorio Debian

Como crear el tuyo propio



Repositorios

Configurar un Repositorio Debian propio

Seguramente te pueda ser interesante **crear tus propios paquetes Debian** y hacer que estos **estén disponibles** para la descarga de otras personas.

La ventaja de un repositorio creado de esta forma, es que puedes ir subiendo nuevos programas o actualizaciones a tus programas existentes y todo el que tenga configurado tu repositorio, con un simple **apt -y upgrade** tendrá disponibles esos nuevos programas que subas y siempre actualizados los programas con la última versión, sin necesidad de tener que buscar, descargar e instalar a mano cada uno de ellos.

Para mi la forma más sencilla ha sido utilizando una herramienta llamada **reprepro**

Te dejo los pasos necesarios para crear paso a paso tu propio repositorio:

REQUISITOS

- Paquete **reprepro**
- Servidor **Apache 2**

Atención: Todas las operaciones que se indican en esta guía se deben realizar con el usuario **root**

Instalación de reprepro

Procedemos a instalar el programa 'reprepro':

```
apt -y update
```

 Actualizamos repositorios

```
apt -y install reprepro
```

 Instalamos reprepro y sus dependencias si fuese necesario

Ahora debemos crear el directorio en donde crearemos toda la estructura necesaria de un típico repositorio Debian.

Vamos a suponer que lo queremos dejar todo en:

```
/var/www/html/repositorio
```

```
mkdir -p /var/www/html/repositorio/conf
```

Creamos todas la rutas usando **-p** en un único paso.

Por seguridad, toda la paquetería que publiquemos en nuestro repositorio tiene que estar **firmada**, para ello generaremos una clave para que en la subida de los paquetes nuevos o en sus actualizaciones estos estén

firmados:

```
cd /var/www/html/repositorio
```

 Nos cambiamos a la ruta raíz de nuestro repositorio

Y procedmos a la generación completa de la clave:

```
gpg --full-generate-key
```

Por favor seleccione tipo de clave deseado:

- (1) RSA y RSA (por defecto)
- (2) DSA y ElGamal
- (3) DSA (sólo firmar)
- (4) RSA (sólo firmar)

Su elección: 4

Escogemos la opción **4** (*4*) *RSA (sólo firmar)*

Las claves RSA pueden tener entre 1024 y 4096 bits de longitud.

¿De qué tamaño quiere la clave? (3072) 3072

Indicamos que vamos a usar una clave de **3072** bits de longitud.

Por favor, especifique el período de validez de la clave.

- 0 = la clave nunca caduca
- <n> = la clave caduca en n días
- <n>w = la clave caduca en n semanas
- <n>m = la clave caduca en n meses
- <n>y = la clave caduca en n años

¿Validez de la clave (0)? 0

La clave nunca caduca

¿Es correcto? (s/n) s

Indicamos escribiendo **0** que *la clave nunca caduca*

Respondemos con **s** para confirmar que es exactamente eso lo que queremos.

GnuPG debe construir un ID de usuario para identificar su clave.

Nombre y apellidos: Luis Gu----- Ló---

Dirección de correo electrónico: luisgulo@s*1*c*nlinux.org.es

Comentario: Firma de Paquetes Debian

Está usando el juego de caracteres 'utf-8'.

Ha seleccionado este ID de usuario:

```
"Luis Gu----- Ló--- (Firma de Paquetes Debian) "
```

¿Cambia (N)ombre, (C)omentario, (D)irección o (V)ale/(S)alir? V

Introduciremos nuestro **Nombre y apellidos**.

Luego nuestro **email**.

Aceptaremos con **V** para validar que los datos son correctos.

Nos mostrará un pequeño mensaje durante unos segundos para que generemos algo de "entropía"...

Es necesario generar muchos bytes aleatorios. Es una buena idea realizar alguna otra tarea (trabajar en otra ventana/consola, mover el ratón, usar la red y los discos) durante la generación de números primos. Esto da al generador de números aleatorios mayor oportunidad de recoger suficiente entropía.

Nos solicitará la **clave**:


```
reprepro -VV createsymlinks
```

Configuración del servidor Apache2

Suponemos que ya tenemos nuestro servidor Apache2 funcionando (la instalación y configuración da para un artículo completo), vamos a configurar un `VirtualHost` poder tener nuestro repositorio separado de otros servicios web.

Para que el repositorio sea accesible desde internet con una URL similar a: `repositorio-tudominio`, vamos a crear la siguiente configuración:

Crea un fichero `repositorio.conf` para realizar esto:

```
cd /etc/apache2/sites-available
```

```
touch repositorio.conf
```

Editalo para que tenga un contenido similar, cambiando el literal `tu-dominio` por el valor correcto:

```
<VirtualHost *:80>
    ServerName repositorio.tu-dominio
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html/repositorio
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
RewriteEngine on
RewriteCond %{SERVER_NAME} =repositorio.tu-dominio
RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
</VirtualHost>
<IfModule mod_ssl.c>
    <VirtualHost _default_:443>
        ServerName repositorio.tu-dominio
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html/repositorio
        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
        SSLEngine on
        <FilesMatch "\.(cgi|shtml|phtml|php)$">
            SSLOptions +StdEnvVars
        </FilesMatch>
        <Directory /usr/lib/cgi-bin>
            SSLOptions +StdEnvVars
        </Directory>
    </VirtualHost>
</IfModule>
```

Ahora hay que activar este `VirtualHost` en Apache2, simplemente ejecuta:

```
a2ensite repositorio.conf
```

Ya tenemos nuestro propio Repositorio Debian perfectamente creado y configurado

Configuración de los equipos clientes que usen nuestro repositorio

Deberán crear un fichero `.list` con el nombre de tu servidor en la ruta de configuración de apt:

```
touch /etc/apt/sources.list.d/tu-dominio.list
```

El contenido de este fichero debe de ser:

```
# Repositorio tu-dominio
deb http://repositorio.tu-dominio/ buster main contrib non-free
```

Importar tu **clave pública** para confiar en la paquetería que suministras y tener instalado instalado 'apt-transport-https':

```
apt-get install -y apt-transport-https
```

Y descargarse e instalar tu clave, mediante el comando:

```
wget -q0 - https://repositorio.tu-dominio/repositorio.key | sudo apt-key add -
```

Y luego simplemente actualizar la lista de paquetes disponibles de todos los repositorios que tengan configurados:

```
apt -y update
```

Gestión de los Paquetes en nuestro repositorio

Incluir paquete Debian en el Repositorio

```
reprepro -b /var/www/html/repositorio/ includedeb fichero.deb
```

Se incluye en la Sección que tenga definida el paquete por defecto.

Si queremos forzar a que el paquete se deposite en otra sección usaremos el parámetro **-S sección**:

```
reprepro -S main -b /var/www/html/repositorio/ includedeb fichero.deb
```

Puedes usar las secciones que creaste en el fichero `distributions` en la línea `Components: main contrib non-free`

Eliminar un paquete del Repositorio

```
reprepro remove buster fichero.deb
```

Para el borrado es necesario indicar el `CodeName` o distribución `-buster-` en la que está el paquete a borrar.

Listar la paquetería de nuestro Repositorio

```
reprepro -b /var/www/html/repositorio/ list buster
```

Para listar también es necesario indicar el `CodeName` o distribución: `buster`.

NOTA: Si en algún momento tienes algún problema con `reprepro` al realizar alguna operación, puede ser debido a la petición de la clave, simplemente incluye como parámetro `--ask-passphrase` para forzar a que te pida la clave.

Esta guía es una adaptación y ampliación al documento de la wiki: <https://wiki.debian.org/DebianRepository/SetupWithReprepro>

Autor: Luis Gutiérrez López (luisgulo)

