Servidor FTP multiusuario con VSFTPD + PAM + MySQL

Hoy instalaremos un servidor FTP convencional pero bastante rápido y seguro en un sistema operativo Ubuntu, basado en Debian GNU/Linux. El servidor requerirá nombre y contraseña para cada usuario que quiera acceder, y cada usuario podrá ir dirigido a un directorio distinto y modificarle el contenido (muy conveniente para un servidor-web compartido). Esta guía evita la creación de múltiples usuarios reales en el sistema operativo, y todos los usuarios virtuales actúan con una misma cuenta del sistema.

El servicio FTP tradicional consiste en que los usuarios del sistema operativo tienen acceso al sistema de ficheros con sus mismas credenciales y atribuciones, y adicionalmente hay una cuenta de usuario anónimo para que públicamente se puedan descargar determinados contenidos. La característica especial, entonces, de esta instalación es que el censo de usuarios FTP se almacena en una base de datos, y pasan a ser "usuarios virtuales" en lugar de "usuarios reales" del sistema operativo; adicionalmente se inhabilita el acceso anónimo.

Esta guía se basa en el *software* servidor de FTP **vsftpd** (vsftpd.beasts.org), pero existen alternativas tan competentes como **ProFTPD** (www.proftpd.org).

Empezamos;

Con el sistema operativo ya instalado, actualizado y en funcionamiento, nos tenemos que asegurar de que tenemos el servidor FTP (vsftpd), los módulos de autenticación (pam) y el sistema de base de datos (MySQL). Si no es así, lo podemos hacer con las instrucciones de terminal/consola:

```
sudo apt-get install mysql-server
sudo apt-get install libpam-runtime libpam-modules
sudo apt-get install libpam-mysql
sudo apt-get install vsftpd
```

El MySQL Server al instalarse pedirá que le establezcamos una contraseña para el usuario administrador (root), que necesitamos cuando queremos hacer determinadas operaciones.

También podemos usar las instrucciones de instalación para saber que los paquetes están instalados. Ahora se tratará de configurarlo todo para que los distintos programas trabajen juntos.

Si tu distribución de GNU/Linux no tiene el paquete "libpam-mysql" al alcance, lo puedes descargar de la página del proyecto *pam-mysql.sourceforge.net* ("Released Files") e instalarlo manualmente con la instrucción:

```
sudo dpkg --install nombredelfichero.deb
```

- Para poder restablecer la configuración de los ficheros de texto que se modifican (en caso de que alguna cosa falle y queramos volver atrás), es recomendable hacer una copia de cada original: antes de editar "vsftpd.conf" copiarlo a "vsftpd.conf.original".
- Si no sabes como editar los ficheros de texto desde una consola o terminal, te propongo el programa "nano", que para utilizarlo como superusuario iría así (ejemplo):

  ```
  sudo nano /etc/vsftpd.conf
  ```

  (Si el fichero no existe, lo crea. Para salir pulsar [Control] + [x])

Configuración de la base de datos

1. Nos conectamos a la consola del servidor MySQL (nos pedirá la contraseña del propio usuario "root" de MySQL):

```
mysql -u root -p
```

2. Creamos una base de datos para las cuentas de usuarios virtuales:

```
CREATE DATABASE vsftpd;
```

3. Creamos una tabla para registrar las cuentas de acceso (fijaros en los nombres de los campos):

```
CREATE TABLE vsftpd.usuarios (
nrregistro int AUTO_INCREMENT NOT NULL,
nombre varchar(30) binary NOT NULL,
contrasena varchar(50) binary NOT NULL,
estado varchar(30) binary NOT NULL,
primary key(nrregistro)
);
```

4. Creamos una tabla para registrar los intentos de acceso (fijaros en los nombres de los campos):

```
CREATE TABLE vsftpd.bitacora (mensaje varchar(255),
usuario varchar(20),
proceso int,
servidor char(32),
cliente char(32),
momento timestamp
);
```

5. Damos permisos a un nuevo usuario de MySQL para que el servidor FTP acceda a través de PAM sólo a lo imprescindible:

```
GRANT SELECT ON vsftpd.usuarios TO vsftpd@localhost IDENTIFIED BY 'contraseña1';
GRANT INSERT ON vsftpd.bitacora TO vsftpd@localhost IDENTIFIED BY 'contraseña1';
FLUSH PRIVILEGES;
```

(personalizad la contraseña1)

6. Damos permisos a un nuevo usuario de MySQL distinto para la administración de usuarios:

```
GRANT INSERT,SELECT,DELETE ON vsftpd.usuarios TO ftpadmin@localhost IDENTIFIED BY 'contraseña2';
GRANT SELECT,DELETE ON vsftpd.bitacora TO ftpadmin@localhost IDENTIFIED BY 'contraseña2';
FLUSH PRIVILEGES;
```

(personalizad la contraseña2, que sea distinta)

7. Nos desconectamos de la consola del servidor MySQL:

```
quit
```

Configuración del sistema operativo

1. Debemos asegurarnos de que en el registro de programas de consola/terminal (fichero **/etc/shells**) hay una línea como esta:

```
/bin/false
```

Si no está, la añadimos al final. Esto permite crear las cuentas de usuario de los programas impidiendo que sean usadas para acceder a la computadora vía terminal.

2. Creamos un directorio para los perfiles de usuario FTP:

```
sudo mkdir /etc/vsftpd
```

3. Creamos dentro otro directorio para los usuarios FTP sin perfil todavía hecho:

```
sudo mkdir /etc/vsftpd/vacio
```

4. Fijamos el directorio vacío sin permisos de escritura:

```
sudo chmod 555 /etc/vsftpd/vacio
```

5. Creamos un grupo y cuenta de usuario para el servidor FTP:

```
sudo groupadd vsftpd
sudo useradd -d /etc/vsftpd -s /bin/false -g nogroup vsftpd
```

6. Creamos un grupo y cuenta de usuario para los clientes FTP:

```
sudo groupadd ftp
sudo useradd -d /etc/vsftpd/vacio -s /bin/false -g ftp ftp
```

Si ya existía (*user ftp exists*) conviene que lo modifiquemos:

```
sudo usermod -d /etc/vsftpd/vacio -s /bin/false -g ftp ftp
```

Configuración del autenticador

Pluggable Authentication Modules (PAM) es un sistema múltiple para que los programas puedan autenticar la identidad de usuarios y otros agentes del entorno informático. Lo utilizamos aquí para que el servidor VSFTPD no tenga que hacer esta tarea, y simplemente consulte a PAM si el usuario que llama se corresponde en el censo de la base de datos.

Revisamos los parámetros del perfil de VSFTPD para PAM, con tal que se hagan las consultas a MySQL. Abrimos el fichero **/etc/pam.d/vsftpd** con un editor de texto y:

1. Para que no se exija la existencia de los usuarios en el sistema operativo, desactivamos "pam_ftp.so", "common-account", "common-auth" y "pam_shells.so" añadiendo el símbolo "#" al inicio de cada línea donde aparezcan:

```
# pam_ftp.so.
#@include common-account
#@include common-auth
#auth    required        pam_shells.so
```

2. Para especificar las credenciales de acceso, tablas y campos a consultar en la base de datos MySQL, añadimos estas dos laargas líneas al final del fichero:

```
auth  required    pam_mysql.so      verbose=0                  user=vsftpd
passwd=contraseña1       host=localhost       db=vsftpd       table=usuarios
usercolumn=nombre    passwdcolumn=contrasena    statcolumn=estado    crypt=2
sqllog=true  logtable=bitacora  logmsgcolumn=mensaje  logusercolumn=usuario
logpidcolumn=proceso loghostcolumn=cliente logtimecolumn=momento


account     required    pam_mysql.so      verbose=0             user=vsftpd
passwd=contraseña1       host=localhost       db=vsftpd       table=usuarios
usercolumn=nombre    passwdcolumn=contrasena    statcolumn=estado    crypt=2
sqllog=true  logtable=bitacora  logmsgcolumn=mensaje  logusercolumn=usuario
logpidcolumn=proceso loghostcolumn=cliente logtimecolumn=momento
```

(La contraseña1 de las líneas debe ser la misma que hemos fijado para el usuario "vsftpd" de MySQL)

Configuración del servicio FTP

Revisamos los parámetros de la configuración básica de VSFTPD, editando el fichero **/etc/vsftpd.conf** y fijando los que especificamos de la siguiente manera (hay que buscarlos en el fichero, y si no están añadirlos al final):

```
anonymous_enable=NO
local_enable=YES
write_enable=YES
xferlog_enable=YES
nopriv_user=vsftpd
chroot_local_user=YES
secure_chroot_dir=/etc/vsftpd/vacio
pam_service_name=vsftpd
# En octal: 0022 xor 0777 = permisos 755
local_umask=0022
file_open_mode=0777
dirmessage_enable=NO
# Número máximo de sesiones de usuarix conectadxs simultáneamente
max_clients=50
# Nr. máximo de ss. de usuarix para un mismo origen (IP).
max_per_ip=5
# Unificación de los clientes como únicx usuarix del sistema
guest_enable=YES
# Usuarix del sistema con que actuaran lxs usuarixs FTP
guest_username=ftp
# Dónde encontrar las configuraciones específicas de cada usuarix
user_config_dir=/etc/vsftpd
# Directorio FTP predeterminado. Si no creamos un perfil, no toquen nada.
local_root=/etc/vsftpd/vacio
# Privilegios locales para escribir, para lxs usuarixs FTP remotxs.
virtual_use_local_privs=YES
# Enmascarar la información de usuarix/grupo en directorios como "ftp"
hide_ids=YES
```

Algunos de los parámetros no son imprescindibles así, pero de esta manera también funciona. El parámetro "local_umask" es para los permisos que se fijaran a los ficheros que suban los usuarios vía FTP; por defecto serían 077, pero con 755 permite que el usuario propietario de los ficheros y carpetas pueda manipular plenamente y en cambio el público de una web sólo pueda leer.

Puesta en marcha

Con esta instrucción reiniciamos el servicio de VSFTPD para que aplique la nueva configuración, o también lo iniciamos si estaba parado:

```
sudo /etc/init.d/vsftpd restart
```

Cómo crear un usuario FTP

1. El directorio en donde tenga que acceder el usuario debe estar creado, con propietario "ftp" y con permisos para operar:

```
sudo mkdir /ruta/a/su/directorio
sudo chown -R ftp:ftp /ruta/a/su/directorio
sudo chmod -R 755 /ruta/a/su/directorio
```

2. Con esta instrucción tendremos una nueva cuenta de usuario en la base de datos:

```
mysql -u ftpadmin --password=contraseña2 -e "INSERT INTO vsftpd.usuarios
(nombre, contrasena) VALUES ('elnombredeusuario',
PASSWORD('lacontraseña'));"
```

Estos "elnombredeusuario" y "elnombredeusuario" (así como la localización pública del servidor) son los datos que podremos dar a las personas que queremos que se conecten por FTP.

3. Creamos un fichero de texto para el perfil de usuario como **/etc/vsftpd/elnombredeusuario** en donde escribiremos la ruta del directorio al que accederá el usuario vía FTP:

```
local_root=/ruta/a/su/directorio
```

En este fichero podemos añadir cualquiera de los parámetros que son válidos en **/etc/vsftpd.conf** para personalizarlos para este usuario en concreto. También los podemos blindar un poco estableciendo permisos:

```
sudo chmod -R 440 /etc/vsftpd/elnombredeusuario
```

Cómo eliminar un usuario FTP

1. Eliminamos el usuario de la base de datos:

```
mysql -u ftpadmin --password=contraseña2 -e "DELETE FROM vsftpd.usuarios
WHERE nombre = 'elnombredeusuario';"
```

2. Eliminamos el perfil FTP de usuario:

```
sudo rm /etc/vsftpd/elnombredeusuario
```

3. Si queremos, podemos eliminar su directorio (se perderá todo el contenido !):

```
sudo rm -R /ruta/a/su/directorio
```

Problemas comunes, primeros auxilios

- **Mensaje "500 OOPS: cannot change directory:/etc/vsftpd/vacio" del servidor al intentar entrar como usuario desde un programa FTP-cliente:** Al crear la cuenta no has creado su perfil (/etc/vsftpd/elnombredeusuario), o bien no le has especificado correctamente la ruta del parámetro "local_root". Revisa también los permisos del directorio del que da error.
- **Al entrar como usuario desde un programa FTP-cliente se ve un directorio vacío y no se puede hacer nada en él (/etc/vsftpd/vacio):** Al crear la cuenta no has creado su perfil (/etc/vsftpd/elnombredeusuario), o bien no has especificado correctamente la ruta del parámetro "local_root", o bien el fichero de perfil tiene permisos demasiado restrictivos.
- **No consigo acceder con un mismo usuario a distintos sitios web:** Si pretendes utilizar el servidor FTP para mantener sitios virtuales de un servidor web, y te gustaría entrar con un mismo nombre de usuario para acceder diferenciando el nombre de dominio o la dirección IP, te conviene hacer uso del *software*-interfaz de red *xinetd* para hacer llamadas al VSFTPD especificando distinto fichero de configuración en la línea del comando. Más información en:

  es.wikipedia.org/wiki/Xinetd

  www.xinetd.org
- **Tengo problemas y necesito encontrar de dónde vienen:** Puedes consultar diversas bitácoras para ver qué pasa con las entradas de clientes FTP:
  1. Bitácora del servidor FTP VSFTPD: fichero de texto **/var/log/vsftpd.log**
  1. Bitácora del autenticador PAM (para todos los accesos al sistema): fichero de texto **/var/log/auth.log**
  1. Bitácoras del servidor de base de datos MySQL: ficheros de texto **/var/log/mysql.log** y **/var/log/mysql.err**
  1. Bitácora entre VSFTPD y PAM (éxitos y fracasos en los intentos de entrada normal de los usuarios FTP). Podemos ver las anotaciones en la base de datos con la instrucción:

     ```
     mysql -u ftpadmin --password=contraseña2 -e "SELECT * FROM vsftpd.bitacora;"
     ```

     Podemos vaciar (eliminar todo el histórico) esta bitácora con la instrucción:

     ```
     mysql -u ftpadmin --password=contraseña2 -e "DELETE FROM vsftpd.bitacora;"
     ```
  1. Censo de usuarios registrados. Podemos ver todos los usuarios virtuales con la instrucción:

     ```
     mysql -u ftpadmin --password=contraseña2 -e "SELECT * FROM vsftpd.usuarios;"
     ```

     Podemos consultar una sola cuenta de usuario con la instrucción:

     ```
     mysql -u ftpadmin --password=contraseña2 -e "SELECT * FROM vsftpd.usuarios WHERE nombre = 'elnombredeusuario';"
     ```
  1. Bitácora del instalador Debian de programas: fichero de texto **/var/log/installer/status**
- **Necesito controlar la cantidad de datos que sube un usuario (cuota de espacio):** Como podías imaginar, en esta guía no encontrarás respuestas ni pistas para todo, pues a día de hoy yo tampoco sé cual es el camino para aplicar esta funcionalidad desde VSFTPD con usuarios virtuales.

Más seguridad, control de velocidad, comprobaciones

Hay que tener en cuenta que en esta guía se ha dado la manera de escribir las contraseñas de forma visible (para facilitar automatizaciones con *script*), así que las instrucciones escritas quedan almacenadas tal cual en el historial de la consola. Para vaciar después este historial y también el historial del MySQL puedes utilizar las siguientes instrucciones:

```
rm $HOME/.bash_history
rm $HOME/.mysql_history
```

(tiene efecto si seguidamente cerramos la sesión de consola/terminal)

Si no quieres dejar escritas las contraseñas con las líneas de instrucciones al llamar a MySQL, puedes cambiar su parámetro `--password=contraseña` por `-p` , y pedirá teclear la contraseña de forma no visible. Un ejemplo:

```
mysql -u ftpadmin -p -e "SELECT * FROM vsftpd.usuarios;"
```

Un salto en seguridad es la autenticación del servidor, es decir, que un cliente FTP tenga la certeza de conectarse al auténtico servidor y no a un pescador de contraseñas. Esto, y la comunicación encriptada, se consigue con la capacidad SSL del servidor. Pero el objetivo de esta guía es ayudar a poner en marcha un servidor FTP de forma sencilla, pues consúltense manuales más avanzados para aprender sobre comunicaciones SSL, o barreras adaptativas como *ban2fail*.

El VSFTPD permite algo de control de ancho de banda dedicado a las conexiones FTP. Se puede establecer en *bytes* por segundo en **/etc/vsftpd.conf** con los parámetros "local_max_rate" y "anon_max_rate". Consúltese la documentación de VSFTPD para más detalles o novedades.

Esta guía publicada en catalán y castellano el 14 de febrero del 2008 por per Narcís Garcia Langa está comprobada con la Ubuntu GNU/Linux 7.10 server, contando con la versión de serie 0.6.2 de PAM-MySQL (pero puede funcionar con la mayoría de variantes de Debian). Referencias útiles:

- www.ubuntu.com
- doc.ubuntu.com/ubuntu/serverguide/C/
- www.debian.org

No hubiera sido posible completar la guía sin:

- Alejandro Ayuso: monocaffe.blogspot.com/2007/09/servidor-virtual-de-ftp-seguro-con.html
- La documentación oficial de vsftpd de Chris Evans: vsftpd.beasts.org
- La documentación de Miles Brennan: www.brennan.id.au/14-FTP_Server.html
- La documentación oficial de PAM-MySQL de Moriyoshi Koizumi:

  pam-mysql.sourceforge.net/Documentation/
- El manual de referencia de MySQL: dev.mysql.com/doc/refman/5.0/es/
- El código fuente publicado por Gunay Arslan
- o la gran cantidad de preguntas y respuestas publicadas en Internet por la inquieta comunidad informática, libre. Especial mención merece el GiLUG (www.gilug.org)