vsftpd con usuarios virtuales y FTPS en Ubuntu 22.04

Instalar vsftpd

```
sudo apt update
sudo apt install vsftpd db-util apache2
```

Crear el directorio del usuario virtual

Ejemplo con el usuario virtual `paco`, cuya raíz FTP será `/var/www/paco`.

```
sudo mkdir -p /var/www/paco
sudo chown -R ftp:www-data /var/www/paco
sudo chmod -R 2775 /var/www/paco
```

- `www-data` será el usuario real al que se mapearán los usuarios virtuales.
- El usuario virtual no existe en el sistema.
- El usuario quedará encerrado (chroot) en este directorio.

Configurar PAM para usuarios virtuales

Crear el archivo PAM específico:

```
sudo nano /etc/pam.d/vsftpd.virtual
```

Contenido:

```
auth required pam_userdb.so db=/etc/vsftpd/virtual_users
account required pam_userdb.so db=/etc/vsftpd/virtual_users
```

3.1 Qué es PAM y cómo interviene en vsftpd

PAM (Pluggable Authentication Modules) es el sistema de autenticación de Linux.

vsftpd no valida usuarios directamente. El flujo real es:

1. vsftpd recibe usuario y contraseña
2. Llama a PAM
3. PAM responde OK o NO
4. vsftpd permite o rechaza el acceso

Este archivo indica a PAM cómo autenticar cuando vsftpd usa el servicio `vsftpd-virtual`.

3.2 Explicación de las líneas PAM

Línea de autenticación:

```
auth required pam_pwdfile.so pwdfile /etc/vsftpd/virtual_users.db
```

- `auth`: fase de autenticación
- `required`: obligatorio
- `pam_pwdfile.so`: autentica contra un archivo
- `pwdfile`: base de datos Berkeley DB con usuarios virtuales

Línea de control de cuenta:

```
account required pam_permit.so
```

- `account`: fase de validación de cuenta
- `pam_permit.so`: permite siempre

Se usa porque los usuarios virtuales no existen como cuentas reales.

Crear usuarios virtuales y base de datos PAM

```
sudo mkdir -p /etc/vsftpd
sudo nano /etc/vsftpd/virtual_users.txt
```

Contenido:

```
paco
pacoPass
```

Convertir a base de datos:

```
sudo db_load -T -t hash -f /etc/vsftpd/virtual_users.txt
/etc/vsftpd/virtual_users.db
sudo chmod 600 /etc/vsftpd/virtual_users.*
```

Configurar vsftpd

Backup:

```
sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.bak
```

Editar:

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

Contenido completo:

```
# Escucha conexiones FTP en IPv4
listen=YES

# Desactiva IPv6
listen_ipv6=NO

# Deshabilita acceso anónimo
anonymous_enable=NO
```

```
# Habilita usuarios locales (necesario para usuarios virtuales)
local_enable=YES

# Permite escritura
write_enable=YES

# Umask por defecto
local_umask=022

# Mensajes de directorio
dirmessage_enable=YES

# Hora local en logs
use_localtime=YES

# Logs de transferencia
xferlog_enable=YES

# Puerto 20 para modo activo
connect_from_port_20=YES

# Chroot por usuario
chroot_local_user=YES


# Usuarios virtuales
guest_enable=YES
guest_username=www-data
user_sub_token=$USER
local_root=/var/www/$USER
pam_service_name=vsftpd-virtual
virtual_use_local_privs=YES


# Permitir escritura en chroot
allow_writeable_chroot=YES


# FTPS / TLS
ssl_enable=YES
allow_anon_ssl=NO
force_local_data_ssl=YES
force_local_logins_ssl=YES
rsa_cert_file=/etc/ssl/private/vsftpd.crt
rsa_private_key_file=/etc/ssl/private/vsftpd.key
ssl_tlsv1=YES
ssl_sslv2=NO
ssl_sslv3=NO
require_ssl_reuse=NO


# Modo pasivo
pasv_enable=YES
pasv_min_port=30000
pasv_max_port=31000
```

Cómo vsftpd convierte un usuario virtual en www-data (funcionamiento interno)

Este punto es clave para entender por qué la configuración funciona.

Cuando un usuario virtual se autentica correctamente:

1. PAM valida usuario y contraseña.

2. vsftpd **no crea una sesión con ese usuario**.
3. vsftpd cambia internamente al usuario real definido en:

```
guest_username=www-data
```

4. A nivel del sistema:
    - UID efectivo: www-data
    - GID efectivo: www-data
5. El nombre del usuario virtual sigue existiendo solo a nivel lógico.

El parámetro:

```
virtual_use_local_privs=YES
```

indica que el usuario virtual hereda **exactamente los permisos Linux** de `www-data`.

El parámetro:

```
user_sub_token=$USER
local_root=/var/www/$USER
```

hace que vsftpd:

- Sustituya $USER por el nombre del usuario virtual (`paco`)
- Aplique el chroot a `/var/www/paco`

Resultado:

- Todos los accesos se ejecutan como `www-data`
- Cada usuario queda aislado en su propio directorio
- Los permisos se gestionan con chmod/chown estándar
- No existen usuarios reales adicionales en el sistema

Este diseño es simple, seguro y muy usado en hosting.

Crear certificado TLS

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
-keyout /etc/ssl/private/vsftpd.key \
-out /etc/ssl/private/vsftpd.crt

sudo chmod 600 /etc/ssl/private/vsftpd.*
```

Iniciar y habilitar vsftpd

```
sudo systemctl enable vsftpd
sudo systemctl restart vsftpd
sudo systemctl status vsftpd
```

Debe aparecer `active (running)`.

Firewall (si se usa UFW)

```
sudo ufw allow 20/tcp
```

```
sudo ufw allow 21/tcp
sudo ufw allow 30000:31000/tcp
sudo ufw reload
```

Probar conexión FTPS

Cliente FTPS (FileZilla, WinSCP):

- Protocolo: FTP sobre TLS explícito
- Host: IP o dominio
- Puerto: 21
- Usuario: paco
- Contraseña: definida
- Modo pasivo

Resultado esperado:

- Raíz: `/var/www/paco`
- Chroot efectivo
- Transferencias cifradas
- Escritura permitida