

# FTP

## Job 01

Pour ce sujet, il vous faudra installer une VM Debian, sans interface graphique pour simuler un environnement serveur. Si vous souhaitez faciliter l'utilisation de votre serveur, vous pouvez très bien lui configurer SSH.

Pour se connecter à la machine virtuelle depuis la machine hôte:

```
$ ssh root@[ IP donnée ]
```

```
machine_virtuelle $ vm-template modconsole
```

```
machine_virtuelle $ vm-start ftp
```

```
machine_virtuelle $ nano /etc/ssh/sshd_config
```

### GNU nano 5.4 /etc/ssh/sshd\_config

...

# Authentication:

#LoginGraceTime 2m

PermitRootLogin yes

...

[[^X + O + Entrée]]

```
machine_virtuelle $ service ssh restart
```

```
machine_virtuelle $ service sshd restart
```

```
machine_virtuelle $ reboot
```

## Job 02

Pour pouvoir commencer notre FTP il va vous falloir télécharger un outil qui va vous permettre de le mettre en place. Installez ProFTPD sur votre système.

```
$ apt update && apt upgrade
```

```
$ apt install proftpd
```

## Job 03

Ajoutez deux utilisateurs qui se nommeront "Merry" et "Pippin" et qui pourront chacun utiliser votre FTP avec leur mot de passe respectif : "kalimac" et "secondbreakfast"

## [ProFTPD: Logins and Authentication](#)

### **Virtual Users**

One question that often arises is "How do I create a proftpd user?" proftpd uses your system's `/etc/passwd` file by default, and so proftpd users are the same as your system users. "Virtual" users, sometimes described as FTP-only user accounts, are users that can login to proftpd, but who are separate from the normal system users, and who do not have entries in `/etc/passwd`. proftpd does not care how or where user information is defined. The daemon is designed with an abstraction layer on top of user information sources, and that abstraction is responsible for supplying that data that is required for every user: a name, a UID, a GID, a home directory, and a shell. A user's shell will not be used except in `RequireValidShell` checks, but it must still be present. The code that is responsible for supplying this data, reading it from whatever storage format is supported, lies in proftpd's various configurable authentication modules.

### [ProFTPD create user : How to add a new user?](#)

```
$ useradd Merry && passwd kalimac && usermod -md /var/www/ Merry
```

```
$ useradd Pippin && passwd secondbreakfast && usermod -md /var/www/ Pippin
```

```
$ nano /etc/proftpd/proftpd.conf
```

#### **GNU nano 5.4 /etc/proftpd/proftpd.conf**

```
...
```

```
# Use this to jail all users in their homes
DefaultRoot ~
```

```
# Users require a valid shell listed in /etc/shells to login.
# Use this directive to release that constraint.
RequireValidShell off
```

```
...
```

```
[[^X + O + Entrée]]
```

## Job 04

Il faut aussi pouvoir accéder à votre FTP sans avoir un utilisateur renseigné, ou de façon anonyme.

Adaptez votre configuration de telle sorte que la connexion anonyme soit possible.

### [Anonymous](#)

```
$ nano /etc/proftpd/proftpd.conf
```

## GNU nano 5.4 /etc/proftpd/proftpd.conf

```
...

<Anonymous ~ftp>
  User ftp
  Group nogroup
  # We want clients to be able to login with "anonymous" as well as "ftp"
  UserAlias anonymous ftp
  # Cosmetic changes, all files belongs to ftp user
  DirFakeUser on ftp
  DirFakeGroup on ftp

  RequireValidShell off

  # Limit the maximum number of anonymous logins
  MaxClients 10

  # We want 'welcome.msg' displayed at login, and '.message' displayed
  # in each newly chdired directory.
  DisplayLogin welcome.msg
  DisplayChdir .message

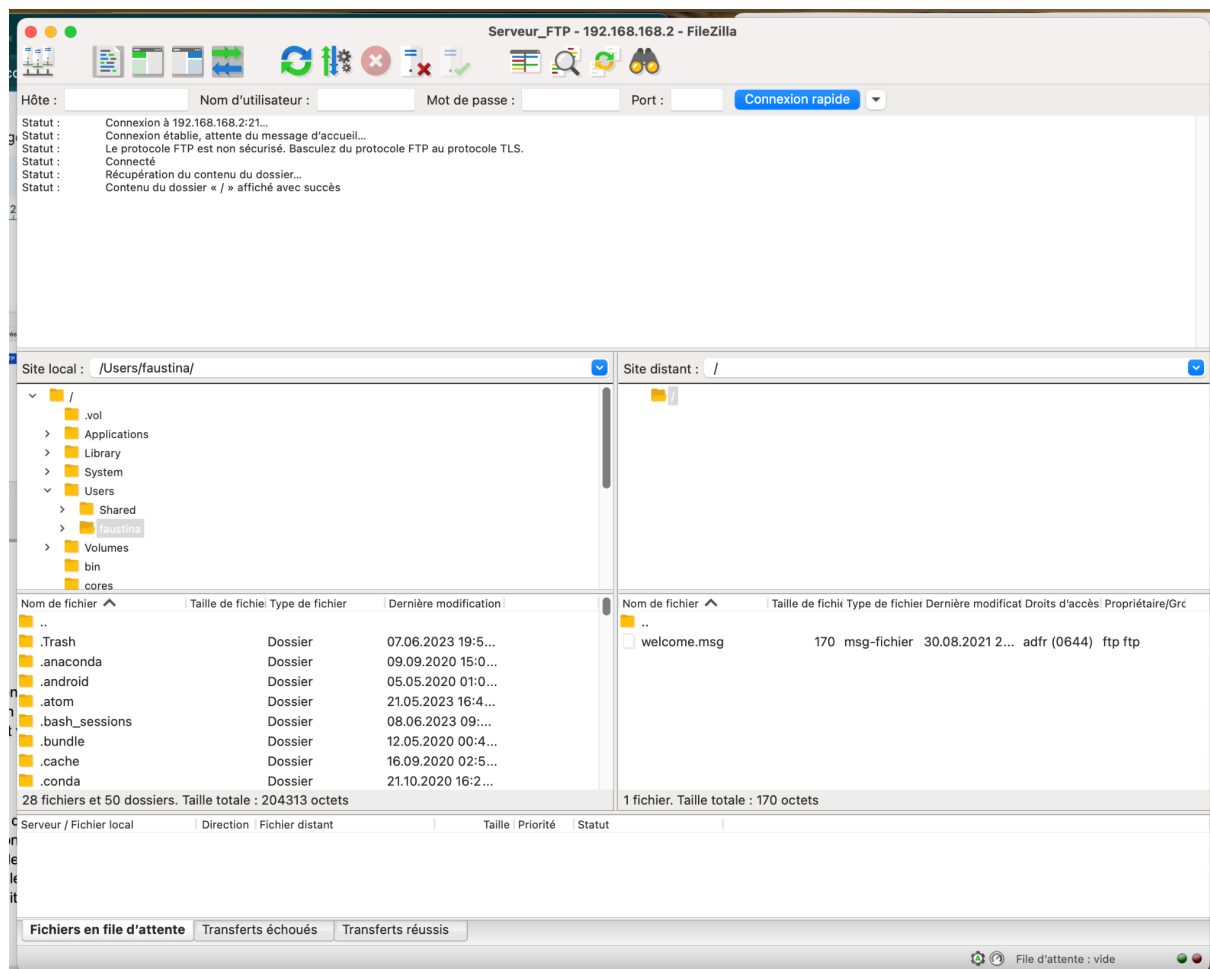
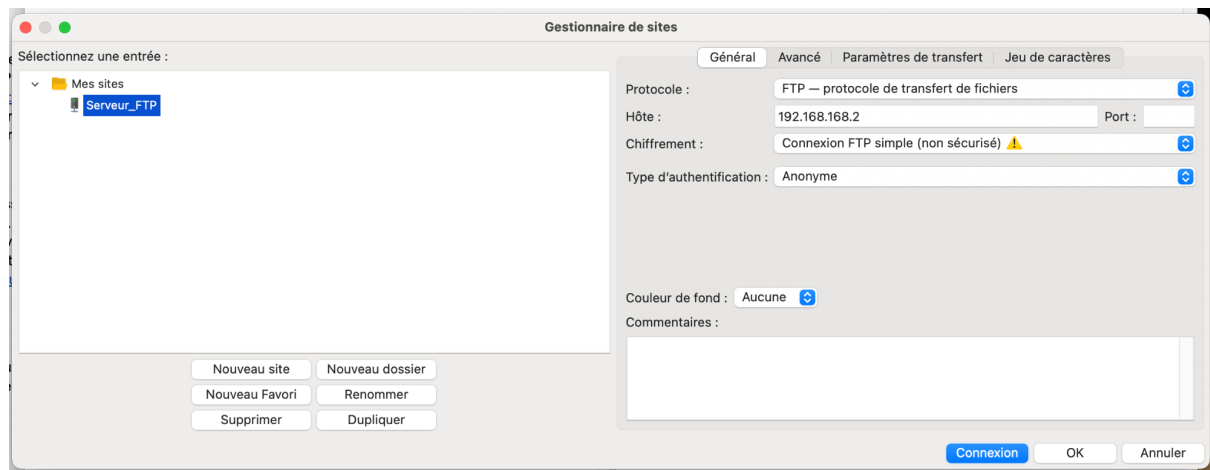
  # Limit WRITE everywhere in the anonymous chroot
  <Directory *>
    <Limit WRITE>
      DenyAll
    </Limit>
  </Directory>
</Anonymous>

...
```

[[^X + O + Entrée]]

## Job 05

Trouvez un client FTP qui vous permettra de vous connecter à votre FTP. Vous devez pouvoir téléverser un fichier depuis votre ordinateur et télécharger un fichier depuis votre serveur.



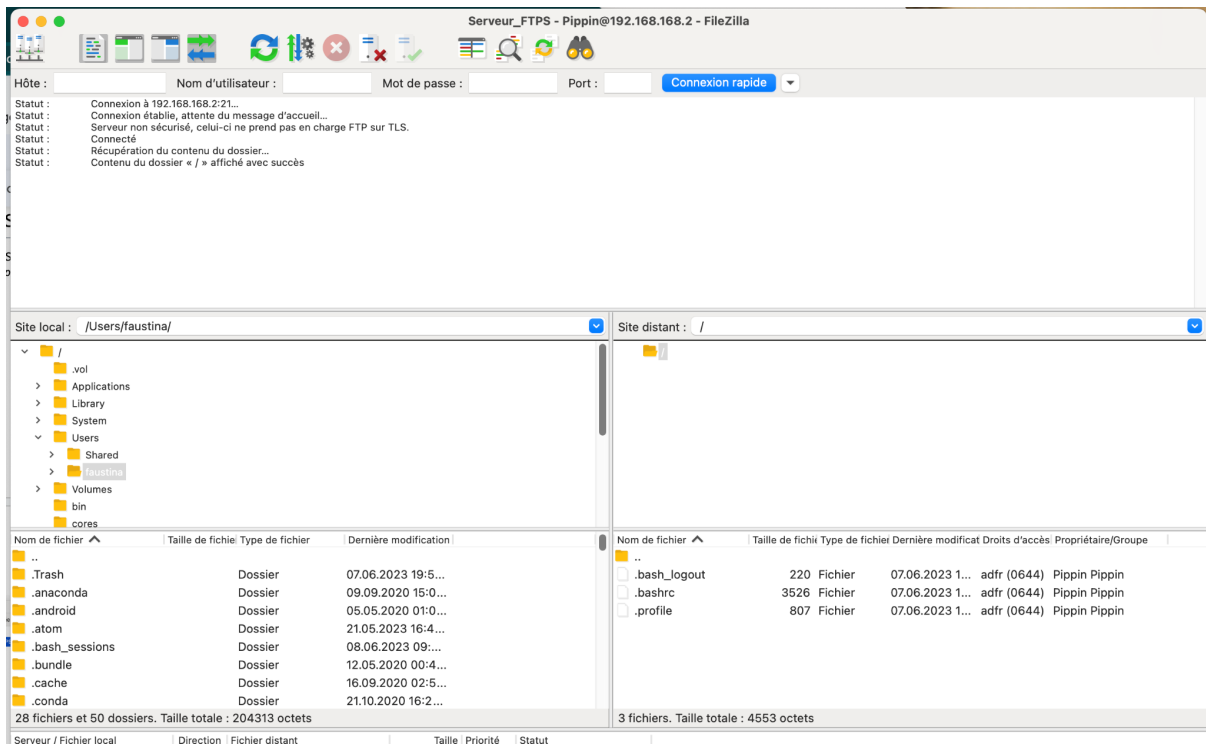
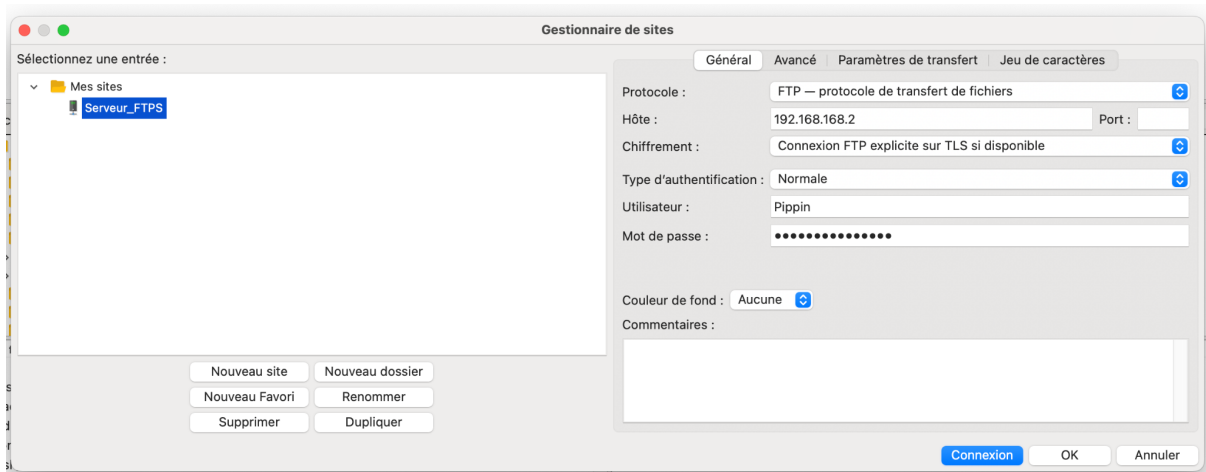
## Job 06

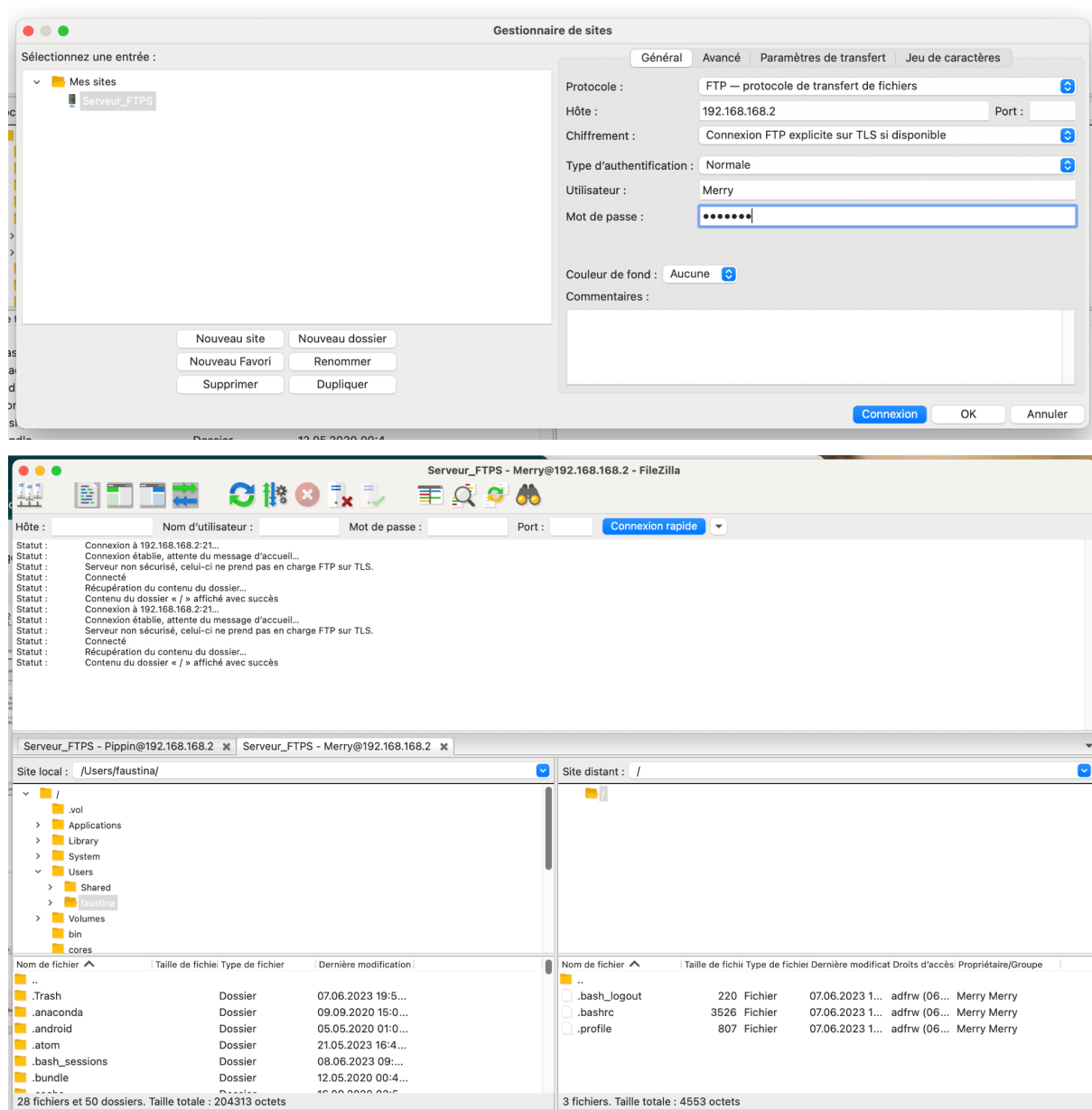
Il serait bien de sécuriser les échanges entre votre machine et votre serveur lorsque vous envoyez un fichier, pour cela, il va falloir transformer votre serveur FTP en FTPS en configurant votre serveur pour qu'il utilise TLS et SSL.

[FTP over TLS - FileZilla Wiki](#)

## TLS (FTPS) vs SSH (SFTP)

*FTPS (FTP encrypted with TLS) should not be confused with SFTP (SSH). The latter is a completely different protocol, with more information [here](#).*





## Job 07

Si on vous demande d'installer un FTP sur 140 machines, comment faire ?

L'installation que vous avez doit être automatisée. Créez un script qui permet l'installation complète de telle sorte qu'une machine vierge (disposant seulement d'une Debian fonctionnelle avec internet) puisse devenir un serveur FTP fonctionnel avec les fonctionnalités listés dans les jobs du dessus.

**\$ source <(curl -s**

**[https://raw.githubusercontent.com/faustina-maria-giaquinta/ftp\\_server/main/server\\_in\\_stall.sh](https://raw.githubusercontent.com/faustina-maria-giaquinta/ftp_server/main/server_in_stall.sh))**

**GNU nano 5.4 server\_install.sh**

**# !/bin/bash**

```
apt update && apt upgrade -y  
apt install proftpd git -y
```

```
wget -P /etc/proftpd/  
https://raw.githubusercontent.com/faustina-maria-giaquinta/ftp_server/main/proftpd.conf  
&& service proftpd restart  
wget -P /etc/ssh/  
https://raw.githubusercontent.com/faustina-maria-giaquinta/ftp_server/main/sshd_config  
&& service sshd start && service ssh start
```

[[^X + Y + Entrée]]

## Job 08

Créez un script qui va vous permettre de faire le travail inverse, de telle sorte que votre machine soit libre de toute installation/configuration liée à la mise en place d'un serveur FTP.

```
$ source <(curl -s  
https://raw.githubusercontent.com/faustina-maria-giaquinta/ftp\_server/main/server\_u  
ninstall.sh)
```

GNU nano 5.4 server\_uninstall.sh

```
# !/bin/bash
```

```
systemctl stop proftpd ssh && apt autoremove proftpd git -y && apt autoclean
```

[[^X + Y + Entrée]]

## Job 09

Créez un script qui va parcourir un fichier CSV contenant les éléments nécessaires à la création d'un ou plusieurs utilisateurs FTP.

Votre script devra automatiquement créer un utilisateur avec ses informations, lui autoriser l'accès au FTP, lui donner les droits Sudo ou non (selon son rôle) et lui créer son /home en le rendant propriétaire de ce dernier.

```
$ source <(curl -s  
https://raw.githubusercontent.com/faustina-maria-giaquinta/ftp\_server/main/ftp\_users  
.sh)
```

GNU nano 5.4 ftp\_users.sh

```

# !/bin/bash
csv_url="https://raw.githubusercontent.com/faustina-maria-giaquinta/ftp_server/main/userlist.csv"
temp_csv="/tmp/temp.csv"

remove_spaces() {
    local str="$1"
    echo "${str}" | awk '{$1=$1};1'
}

create_user() {
    curl -s "$csv_url" >"$temp_csv"

    while IFS=, read -r id prenom nom passwd role || [ -n "$id" ]; do
        name=$(remove_spaces "$prenom")_${remove_spaces "$nom"}
        role=$(remove_spaces "$role")

        if [ "$role" = "Admin" ]; then
            group="root"
        else
            group="users"
        fi

        useradd "$name" -u "$id" -p "$passwd" -sG "$group" && \
        usermod -md /var/www/ "$name"

    done <"$temp_csv"

    rm "$temp_csv"
}

create_user

```

[[^X + Y + Entrée]]

## Pour aller plus loin...

Créer une tâche automatique qui va permettre une sauvegarde automatique de votre installation ainsi que les fichiers qui sont envoyés sur votre FTP.

Il devra récupérer les différents fichiers et les compresser dans une archive qui aura un nom qui permettra de dater le fichier de sauvegarde, exemple :

**backup\_jour-mois-année\_heure:minutes.tar.gz**

Le fichier de sauvegarde devra être envoyé vers une autre machine (vous pouvez tester avec une deuxième machine virtuelle).



## Rendu

Le projet est à rendre sur <https://github.com/prenom-nom/FTP>

Votre repository devra contenir tous les scripts créés dans les jobs de ce sujet

Pensez à mettre votre repos en public et à donner les droits sur le répertoire à deepthoughtlaplateforme !