

## Cours 43 : FTP & TFTP

Dans ce cours nous verrons le fonctionnement des protocoles FTP (File Transfer Protocol) & TFTP (Trivial File Transfer Protocol). Comme leurs noms le laisse penser ces protocoles sont utilisés pour transférer des fichiers à travers un réseau.

Dans verrons tout d'abord l'intérêt d'utiliser FTP/TFTP, ainsi que leurs fonctionnements et leurs différences. Nous verrons ensuite comment les fichiers du système sont utilisés et stockés dans un IOS Cisco. En dernier temps nous verrons comment utiliser les protocoles FTP/TFTP sur l'IOS Cisco.

FTP (File Transfer Protocol) et TFTP (Trivial File Transfer Protocol) est un protocole de l'industrie Standard utilisé pour transférer des fichiers à travers le réseau.

Ils utilisent tout deux un modèle de client/serveur. Les clients peuvent utiliser FTP ou TFTP pour copier des fichiers depuis un serveur. Les clients peuvent aussi utiliser FTP ou TFTP pour copier des fichier vers un serveur.

En tant qu'ingénieur réseau, l'utilisation la plus commune de FTP/TFTP est dans la procédure de mettre à jour l'OS d'un appareil du réseau.

On peut utiliser FT/TFTP pour télécharger une nouvelle version d'un IOS depuis un serveur, et puis redémarrer l'appareil avec la nouvelle image IOS.

Voici un schéma réseau afin de mieux comprendre :



Le serveur Cisco, récupère l'image IOS depuis Cisco, puis il place l'image IOS sur un serveur joignable par l'appareil pour être mis à jour, sur le réseau se sera le serveur FTP. Puis le routeur R1 utilise FTP/TFTP pour copier le fichier dans la mémoire flash de l'appareil.

Le Protocole TFTP (Trivial File Transfer Protocol) a été standardisé en 1981.

Il est appelé « trivial » car il est simple et a des fonctions basiques comparé à FTP.

Il permet seulement à un client de copier un fichier vers ou depuis un serveur.

Ce protocole a été publié après FTP, mais il n'est pas le remplaçant de FTP. C'est un autre outil à utiliser lorsque la simplicité d'utilisation est plus importante que les fonctionnalités.

TFTP n'utilise pas d'authentification (nom d'utilisateur/mot de passe), donc les serveur répondent à toutes les requêtes TFTP. Il n'y a pas de cryptage des données, les données sont transmises en texte clair. C'est la meilleure solution dans un environnement contrôlé pour transférer de petit fichier rapidement. Les serveurs TFTP écoutent sur le port UDP 69.

UDP est une connexion et ne fournit pas de fiabilité avec une retransmission.

TFTP a des fonctions similaires incluses au protocole.

Afin de démontrer la fiabilité de TFTP nous utiliserons le réseau suivant :



Toutes les données de messages TFTP sont confirmés, donc si le client transfère un fichier vers un serveur, le serveur enverra des messages de ACK pour acknowledgment.

Un chronomètre est utilisé et si un message attendu n'est pas reçu à temps, à la fin du temps d'attente de l'appareil celui ci renverra son message précédent.

Par exemple dans le cas du réseau présenté si le client TFTP (à droite) veut télécharger un document depuis le serveur TFTP (à gauche), le client envoie tout d'abord une requête de lecture, le serveur répond avec un message contenant la donnée et le fichier. Le client répond alors avec un message de confirmation (ACK) mais pour une raison quelconque le message n'atteint pas le serveur, puisque le client a envoyé un ACK, il attend pour le message de données suivant mais celui ci n'arrive pas car le message ACK non plus n'a pas atteint le serveur. Dans ce cas le client va renvoyer un message de confirmation une seconde fois au serveur TFTP pour lui confirmer avoir reçu le message.

Ce processus continue jusqu'à que le client ait reçu le fichier en entier.

TFTP utilise une communication dite de « lock-step ». Le client et le serveur envoient alternativement un message et attend pour une réponse (les retransmission sont envoyés si besoin).

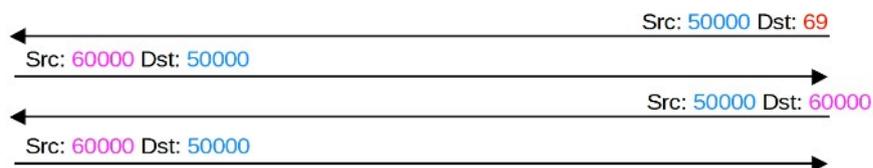
Le transfert de fichier avec TFTP ce passe en trois phases :

1. Connexion : Le client TFTP envoie une requête au serveur, et le serveur lui répond en initialisant la connexion.
2. Transfert de données : le client et le serveur échangent des messages TFTP. L'un envoie la donnée et l'autre envoie une confirmation.
3. Terminaison de connexion : Après le dernier message de donnée envoyé, une confirmation finale est envoyé pour terminer la connexion.

Lorsque le client envoie le premier message au serveur, le port de destination est UDP 69 et la source est un port aléatoire éphémère.

Ce port aléatoire est appelé « Transfer Identifier » (TID) et permet d'identifier le transfert des données. Le serveur sélectionne un TID aléatoire à utiliser comme port source lorsqu'il répond, il n'utilise plus le port 69. Lorsque le client envoie son message suivant le port de destination sera le TID du serveur et non pas 69.

Voici un schéma pour mieux comprendre cela :



Ici le port est au départ 69, un port aléatoire est utilisé, ici le port 60000 et les deux appareils continuent à échanger avec ces deux ports là.

A présent que nous avons le fonctionnement de TFTP voyons le fonctionnement de FTP. FTP est un protocole standardisé en 1971, cela est même plus ancien que TCP IP, il s'agit donc d'un très ancien protocole. FTP utilise les port TCP 20 et 21 à la place d'un seul et unique port. Des noms d'utilisateurs et mots de passes sont utilisés pour l'authentification même s'il n'y a pas de cryptage.

Pour une meilleure sécurité le protocole FTPS (FTP qui utilise SSL/TLS) peut être utilisé.

Une autre option existe avec le protocole STFT (FTP qui utilise SSH) peut aussi être utilisé.

FTP est plus complexe que TFTP et ne permet pas uniquement aux clients le transfert de fichiers, ceux ci peuvent aussi naviguer dans les répertoire, ajouter un répertoire, le supprimer, lister les fichiers etc...

Avec TFTP le client peut seulement informer le serveur de « lui donner un fichier » ou de « prendre tel fichier ».

Le client envoie des commandes FTP au serveur pour faire fonctionner ces fonctions.

Comme explique, FTP utilise deux ports : 20-21

Cela permet deux types de connexions :

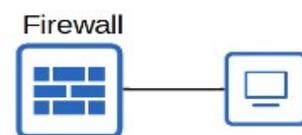
Un contrôleur de connexion FTP (TCP 21) est établi et utilisé pour envoyer des commandes FTP et les réponses.

Lorsque les fichier ou données doivent être transférés, séparément par FTP data (TCP 20) les connexions sont établis et terminés si besoin.

Tout d'abord il y a une connexion TCP initié avec le SYN, SYN-ACK, ACK de connexion TCP classique. Maintenant que le contrôle de connexion est établi, le client envoie une commande FTP au serveur, par exemple pour dire que le PC veut avoir un fichier depuis le serveur. Le serveur répond avec une confirmation (ACK). Il y a ensuite deux différents modes qui peuvent être utilisé pour établir la connexion de données FTP.

La méthode par défaut pour établir une connexion de données FTP est avec le mode actif, dans lequel c'est le serveur qui initie la connexion TCP, puis une fois la connexion établie celui procède à un échange de données FTP.

Voyons à présent comment la connexion est établit en mode passif dans ce mode c'est le client qui initie en premier la connexion, cela est souvent nécessaire lorsque le client se trouve derrière un mur de feu qui bloque les connexion entrantes depuis le serveur. Pour démontrer cela on utilisera le réseau suivant dans lequel est présent un mur de feu.



Dans ce cas la connexion est initié par le client qui effectue la requête TCP SYN.

La connexion FTP des données se fait ensuite entre le serveur/client.

Voyons quelques comparaisons des deux protocoles :

FTP : utilise TCP (20 pour les données, 21 pour le contrôle). Les clients peuvent utiliser des commandes FTP pour faire fonctionner différentes action, pas seulement pour copier des fichiers. FTP utilise une authentification par NomUtilisateur/MotDePasse. FTP est plus complexe que TFTP

TFTP : Utilise le port UDP 69, une forme de connexion basique est utilisé. Les clients peuvent uniquement copier des fichiers vers ou depuis un serveur. Il n'y a pas d'authentification requise. TFTP est plus simple que FTP.

Voyons rapidement la gestion des fichiers systèmes IOS.

Un fichier est une manière de contrôler comment une donnée est stocké et récupérer.

Il est possible de voir le système de fichier d'un appareil sous IOS Cisco avec la commande :  
show file systems

```
Router#show file systems
File Systems:
  Size(b)      Free(b)      Type  Flags  Prefixes
* 2142715904   1994403840   disk  rw     flash0: flash:#
  -           -           disk  rw     flash1:
  966656       962560       disk  rw     flash2:#
  -           -           disk  rw     flash3:
  -           -           opaque rw     archive:
  -           -           opaque rw     system:
  262144       256791       nvram rw     nvram:
  -           -           opaque rw     tmpsys:
  -           -           network rw     snmp:
  -           -           opaque rw     null:
  -           -           network rw     tftp:
  -           -           opaque ro     xmodem:
  -           -           opaque ro     ymodem:
  -           -           opaque wo     syslog:
  -           -           network rw     rcp:
  -           -           network rw     pram:
  -           -           network rw     ftp:
[output omitted]
```

On peut voir le type de fichiers avec des informations qui sont :

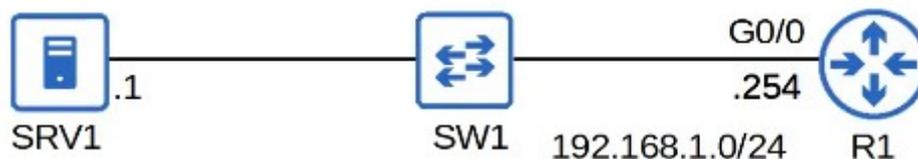
Disk : stockage de l'appareil comme une mémoire flash.

Opaque : utilisé pour des fonctions internes.

Nvram : NVRAM interne. La fichier startup-config est stocké à l'intérieur.

Network : représente un fichier externe du système, par exemple un serveur FTP/TFTP.

Voyons à présent comment utiliser TFTP et FTP pour transférer des fichiers dans la mémoire flash d'un appareil Cisco. Nous utiliserons pour démontrer cela le réseau suivant :



Il est possible d'afficher la version actuelle d'un IOS avec la commande : show version

```
R1#show version
Cisco IOS Software, C2900 Software (C2900-UNIVERSALK9-M), Version 15.1(4)M4, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Thurs 5-Jan-12 15:41 by pt_team
[output omitted]
```

On peut afficher le contenu de la mémoire flash avec la commande : show flash

```
R1#show flash
System flash directory:
File Length Name/status
 3 33591768 c2900-universalk9-mz.SPA.151-4.M4.bin
 2 28282 sigdef-category.xml
 1 227537 sigdef-default.xml
[33847587 bytes used, 221896413 available, 255744000 total]
249856K bytes of processor board System flash (Read/Write)
```



On peut voir qu'il s'agit bien de la version installé depuis le serveur TFTP.  
Si l'on souhaite supprimer l'ancienne version de l'IOS utilisé, on utilise pour cela la commande `delete` suivi du chemin du fichier.

```
R1#delete flash :c2900-universalk9-mz.SPA.151-4.M4a.bin
```

Voyons à présent comment faire lorsque l'on souhaite utiliser le protocole FTP pour transférer des fichiers.

```
R1(config)#ip ftp username cisco
R1(config)#ip ftp password cisco
R1(config)#exit
R1#copy ftp: flash:
Address or name of remote host []? 192.168.1.1
Source filename []? c2900-universalk9-mz.SPA.155-3.M4a.bin
Destination filename [c2900-universalk9-mz.SPA.155-3.M4a.bin]?

Accessing ftp://192.168.1.1/c2900-universalk9-mz.SPA.155-3.M4a.bin...
Loading c2900-universalk9-mz.SPA.155-3.M4a.bin from
192.168.1.1: !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[output omitted]
```

Pour cela on commence tout d'abord par s'authentifier au serveur ftp avec les commandes :

```
R1(config)#ip ftp username cisco
R1(config)#ip ftp password cisco
R1(config)#exit
R1#copy ftp: flash:
```

Les commandes pour copier, supprimer, les fichiers sont identiques à celle utilisés pour TFTP.  
La principale différence se situe lors de la connexion, avec FTP il faut tout d'abord se connecter au serveur FTP en utilisant le nom d'utilisateur/MotDePasse du serveur.