**Réseau**

Ex : Si on travaille en TCP/IP : Transport TCP + Internet IP

Couches (Applications,Présentation et Session) sont appelées Application en TCP/IP

Couche Transport est appelée Transport en TCP/IP

Couche Réseau est appelée Internet en TCP/IP

Couches Liaison et Physique sont appelées Network Access en TCP/IP

TCP est plus long que l’UDP (du coup on préférera l’UDP pour des échanges rapides styles VOIP)

On prend un serveur FTP avec OS (Unix), le FTP va installer un Service qui ouvre un port (Port 21 par défaut) => proftpd (software) avec adresse IP 88.191.21.14

Dans proftpd => création d’un nouvel utilisateur (etu1 pwd1)

Le nouveau user veut envoyer des fichiers sur le FTP en utilisant une application client ici : FileZilla

Dans le software on entre le nom du ftp, l’IP correspondante, le user (etu1) et le password (pwd1), une fois que l’on est connecté on peut voir ce qu’il y’a de présent sur notre ordinateur et sur le serveur FTP, il ne reste plus qu’à drag & drop les fichiers que l’on veut mettre en ligne

En couche réseau on est bien souvent en IP

En couche transport on est en TCP ou UDP mais on n’a pas la main dessus (à part si l’on crée notre propre application réseau)

Multiplexing permet de faire plusieurs choses en même temps (par exemple : Game de LoL en vocal Discord + Probuilds) mais comment savoir quel data va où ? Avec les ports

Port source = ports ouverts dans le client pour envoyer la trame

Port destination = ports ouverts sur le serveur pour recevoir la trame

Sequence number = numéro du paquet qu’on a envoyé au serveur

Btw, les ports sont des trous quand on parle à quelqu’un avec la bouche (port source) les paroles (trames,paquets) sont envoyés jusqu’aux oreilles du destinataire (port de destination).

Une fois le message envoyé au serveur, une réponse est renvoyée (port source et destination sont inversées du coup)

Couche 2 (Iiaison) ne change jamais => ADRESSE MAC

Couche 3 (réseau) ne change quasi jamais => IP

Nombre de ports ouvrables => 0 à 65535

Ports 7 et 9 : Wake ON LAN c-à-d pc éteind, carte réseau allumée qui écoute, tu envoies un magic packet et tu allumes le pc à distance

Ports 20 : Data envoyé par ce port (FTP)

Ports 21 : transfert de fichiers (ftp)

Ports 22 : SSH (pour configurer quelque chose qui n’est pas pré-configuré) SFTP => Secure FTP Connexion => invite de commandes (pas de GUI)

Ports 23 : Telnet, même chose que le SSH mais sans protection

Ports 25 : SMTP, ports pour envoyer des mails uniquement

Ports 69 : SFTP

Ports 80 : HTTP

Ports 443 : HTTPS

UDP : Rapide, Petite charge (pour communiquer), Pas besoin d’un accusé (un flag), Ne renvoie pas les données perdus, données envoyées quand elles arrivent

TCP : Fiable, Données avec un accusé (un flag), renvoie les données perdues, livre les données dans un ordre séquencé

Segment TCP (32bits) : 16 premier bit (port source) 16 d’après (port destination) sur 20 bytes de largeur

Segment UDP (32bits) : 16 premier bit (port source) 16 d’après (port destination) sur 8 bytes de largeur

OS travaille avec la couche transport

PACKET TRACER DERNIERE VERSION

FIREFOX ESR AVEC FLASH 8