réseau

- Création lagrenouille
- Objet : du tuto Le réseau
- Niveau requis :
  débutant, avisé
- Commentaires : *Contexte d'utilisation du sujet du tuto.* Fix Me!
- Débutant, à savoir : Utiliser GNU/Linux en ligne de commande, tout commence là !.
- Suivi :
  à-placer

Présentation

Dés lors que nous possédons un ordinateur, une des premières choses que nous voulons,c'est d'aller sur internet. Pour cela nous avons besoin de nous connecter à un routeur(pour la plupart d'entre nous ce sera une box), puis nous aurons besoin d'une imprimante, d'un téléviseur pour certains et autres appareils, vous mettez donc en marche un réseau.. Vous mettez en relations plusieurs systèmes informatique, soit à l'aide d'un câble, soit avec du wifi.. etc. Comme nous sommes quelques milliards dans ce cas il a bien fallu établir des normes de communications. Le modèle OSI est donc une norme qui préconise comment les ordinateurs devraient communiquer entre eux. Et pour faire fonctionner toutes ces communications entre elles, ce sera dans le respect de la communication par couches.

Les couches OSI : (Open Systems Interconnection)

On associe une adresse Ethernet au réseau d'un ordinateur, et non à l'ordinateur lui même. Remplacer l'interface (en l'occurrence la carte réseau) de cet ordinateur modifie son adresse physique sur le réseau. Le modèle OSI :

Il décrit sept couches portant les noms de couche : *Il est d'usage de diviser ces sept couches en deux : - les couches basses, qui se limitent à gérer des fonctionnalités de base. - les couches hautes, qui contiennent les protocoles plus élaborés. * Les couches basses, aussi appelées couches matérielles, s'occupent de tout ce qui a trait au bas-niveau, au matériel. Elles permettent d'envoyer un paquet de données sur un réseau et garantir que celui-ci arrive à destination. Elle est généralement prise en charge par le matériel et le système d'exploitation, mais pas du tout par les logiciels réseaux. Les couches basses sont donc des couches assez bas-niveau, peu abstraites. Les couches basses sont au nombre de trois.

Pour résumer, ces trois couches s'occupent respectivement des liaisons point à point (entre deux ordinateurs/équipements réseaux), des réseaux locaux, et des réseaux Internet.

LES COUCHES BASSES

`1) la couche physique :`

s'occupe de la transmission physique des bits entre deux équipements réseaux. Elle s'occupe de la transmission des bits, leur encodage, la synchronisation entre deux cartes réseau, etc. Elle définit les

standards des câbles réseaux, des fils de cuivre, du WIFI, de la fibre optique, ou de tout autre support électronique de transmission.

Le rôle principal de la couche 1 est de fournir le support de transmission de la communication. Eh oui, pour pouvoir communiquer il va bien falloir avoir un support. Vous en connaissez déjà un si vous êtes connectés à Internet : un câble RJ45 si vous êtes connectés directement à votre box, l'air libre si vous utilisez le wifi. La couche 1 aura donc pour but d'acheminer des signaux électriques, des 0 et des 1 en gros.

Avec la fibre optique, nous transportons des 0 et des 1, non plus avec de l'électricité mais avec de la lumière ! Le nom scientifique de la fibre est communément le 1000BF Aujourd'hui, dans la plupart des réseaux, nous utilisons 2 paires, soit 4 fils, une paire pour envoyer les données, et une paire pour les recevoir

`2) la couche liaison :`

s'occupe de la transmission d'un flux de bits entre deux ordinateurs, par l'intermédiaire d'une liaison point à point ou d'un bus (note3). Pour simplifier, elle s'occupe de la gestion du réseau local. Elle prend notamment en charge les protocoles MAC, ARP, et quelques autres.

Le rôle donné à la couche 2 est de connecter des machines sur un réseau local, et la détection des erreurs de transmission. Plus exactement, l'objectif est de permettre à des machines connectées ensemble de communiquer.

Nous allons donc dans ce chapitre voir ce qu'il faut mettre en œuvre pour établir une communication entre deux ou plusieurs machines Le bus de données permet à différents blocs logiques d'échanger des informations, il relie le processeur, la mémoire centrale et les contrôleurs de périphériques.

On a donc créé un identifiant particulier à la couche 2 qui permettrait de distinguer les machines entre elles, il s'agit de l'adresse MAC ! l'adresse MAC est en liaison avec le matériel, et notamment la carte réseau. Chaque carte a sa propre adresse MAC, unique au monde. L'adresse MAC est donc l'adresse d'une carte réseau.l'adresse MAC s'écrit en hexadécimal, codée sur 6 octets, soit 48 bits ….

En clair, Une adresse MAC est un identifiant physique inscrit en usine dans une mémoire. Elle est constituée de 6 octets souvent donnée sous forme hexadécimale (par exemple 5E:FF:56:A2:AF:15). Elle se compose de : 3 octets de l'identifiant constructeur et 3 octets du numéro de série. Un octet représente 8 bits.

Ce qui nous donne pour un octet, qui représente 8 bits : 1 octet = 2 puissance 8, = 256 valeurs ! (voir notes 1)

Un bit est une valeur binaire.etc etc

En réseau, on traduit langage de communication commun par "protocole", compréhensible par tous les systèmes d'exploitation. Le protocole va ainsi définir quelles informations vont être envoyées, et surtout dans quel ordre.

Couche - l'adresse de l'émetteur ;

- l'adresse du destinataire ;

- le message proprement dit.

La trame est le message envoyé sur le réseau.

Format d'une trame Ethernet :

adresse MAC du destinataire

adresse MAC de l'émetteur (aussi appelée adresse MAC source). protocole de la couche 3

message

CRC (Le CRC est une valeur mathématique qui est représentative des données envoyées.En gros cela veut dire que c'est un nombre qui sera différent pour chaque message.

une machine A envoie un message à une machine B.

Lors de l'envoi, A calcule le CRC (une valeur X) et le met à la fin de la trame

B reçoit le message et fait le même calcul que A avec la trame reçue (une valeur Y

B compare la valeur qu'elle a calculée (Y) avec la valeur que A avait calculée et mise à la fin de la trame (X).

Si elles sont égales, bingo ! La trame envoyée par A est bien identique à la trame reçue par B. (si non , erreur)

La taille maximale est de 1518 octets, pour une trame Ethernet.

Avec les 18 octets d'en-tête à la taille maximale, nous tombons sur un chiffre rond de 1500 octets de données pour les données à envoyer !

on sait maintenant que le rôle principal de la couche 2 est de connecter les machines sur un réseau local ; ===== Dans la couche 2, nous avons le commutateur, ou switch, sur lequel sont présentes plusieurs prises RJ45 femelles permettant de brancher dessus des machines à l'aide de câbles, mais, aussi des imprimantes, Nas, etc

Vous entendrez parfois parler de pont ou bridge en anglais (un switch avec seulement deux ports.)

Pour envoyer la trame vers la bonne machine, le switch se sert de l'adresse MAC , destination contenue dans l'en-tête de la trame.

le switch contient une table qui fait l'association entre un port du switch (une prise RJ45 femelle) et une adresse MAC. Cette table est appelée la table CAM. port 1 mac xx, port 2 mac kkk, port 3 mac GGG, etc etc .

Dans chaque switch se trouve une base de données appelée "table MAC" pour Medium-Access-Control ou "table CAM" pour Content-Addressable-Memory.

prises RJ45. câble droit ou câble croisé, si l'on utilise les switchs actuelles, ceci n'a plus d'importance avec une box,et un switch plusieurs ports, nous sommes en étoiles

Cette table fait le lien entre les ports physiques du switch (E0, E1, E2) et les adresses MAC sources qui arrivent sur ces ports. Forcément, lorsqu'on démarre un switch, ce dernier ne peut pas savoir quel PC est connecté sur tel ou tel port, la table est donc logiquement vide.

Cette table se construit en associant numéro de port et adresse MAC, le switch sait comment associer ces données, il remplit sa table CAM au fur et à mesure des connexions aux ordinateurs .

La commande « arp » permet de visualiser ou modifier la table du cache arp de l'interface. Cette table peut être statique et (ou) dynamique. Elle donne la correspondance entre une adresse IP et une adresse MAC (Ethernet).

Avec net-tools

```
arp -a
jeanne.home (192.168.1.10) at d4:3b:04:fa:99:cf [ether] on enp3s0
? (192.168.1.18) at 00:25:d3:fc:c6:40 [ether] on enp3s0
funambule.org (192.168.1.15) at 74:d0:2b:13:6b:57 [ether] on enp3s0
 (192.168.1.11) at 00:16:d3:b3:8f:4a [ether] on enp3s0
livebox.home (192.168.1.1) at 08:3e:5d:9c:8a:ee [ether] on enp3s0
npi8440f0.home (192.168.1.22) at <incomplete> on enp3s0
```

Avec iproute2

```
ip neighbor
192.168.1.13 dev enp2s0 lladdr 70:85:c2:48:20:ef STALE
192.168.1.17 dev enp2s0 lladdr 00:18:de:cb:23:02 STALE
192.168.122.183 dev virbr0 lladdr 52:54:00:7e:fa:0f STALE
192.168.1.18 dev enp2s0 lladdr 00:25:d3:fc:c6:40 STALE
192.168.1.12 dev enp2s0 lladdr 08:60:6e:7e:4e:46 DELAY
192.168.1.16 dev enp2s0 lladdr 00:2b:67:b2:5a:15 STALE
192.168.1.1 dev enp2s0 lladdr 08:3e:5d:9c:8a:ee REACHABLE
192.168.1.11 dev enp2s0 lladdr 00:16:d3:b3:8f:4a STALE
192.168.1.24 dev enp2s0 lladdr 80:3f:5d:10:84:61 STALE
fe80::a3e:5dff:fe9c:8aee dev enp2s0 lladdr 08:3e:5d:9c:8a:ee router STALE
2a01:cb19:83c4:d500:a3e:5dff:fe9c:8aee dev enp2s0 lladdr 08:3e:5d:9c:8a:ee
router STALE
```

Avec net-tools

```
ifconfig -a
enp3s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.12  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::a60:6eff:fe7e:4e46  prefixlen 64  scopeid 0x20<link>
        ether 08:60:6e:7e:4e:46  txqueuelen 1000  (Ethernet)
        RX packets 1166988  bytes 1166071942 (1.0 GiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 697054  bytes 81478580 (77.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Boucle locale)
        RX packets 42  bytes 2468 (2.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
```

```
        TX packets 42  bytes 2468 (2.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Avec iproute2

```
ip a
ou
ip -br a
ou
 ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp3s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
UP group default qlen 1000
    link/ether 08:60:6e:7e:4e:46 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.12/24 brd 192.168.1.255 scope global dynamic
noprefixroute enp3s0
       valid_lft 83908sec preferred_lft 83908sec
    inet6 2a01:cb19:83c4:d500:3592:fd01:d8c8:739e/64 scope global temporary
dynamic
       valid_lft 1753sec preferred_lft 553sec
    inet6 2a01:cb19:83c4:d500:a60:6eff:fe7e:4e46/64 scope global dynamic
mngtmpaddr noprefixroute
       valid_lft 1753sec preferred_lft 553sec
    inet6 fe80::a60:6eff:fe7e:4e46/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

ifconfig -a m'affiche mon adresse mac (ma carte réseau)

inet6 fe80::a60:6eff:fe7e:4e46/64 scope link noprefixroute

loop c'est aussi lo, la boucle local 127.0.0.1

Chaque pile TCP/IP répond sur l'adresse 127.0.0.1.

dans /etc/network/interfaces, on a auto lo

iface lo inet loopback

de toutes les adresses IPv4 comprises entre 127.0.0.1 et 127.255.255.255, la plus utilisée est 127.0.0.1).

l'interface lo est l'interface de loopback qui a pour adresse 127.0.0.1. (c'est une interface virtuelle qui permet à la machine de se connecter à elle même sans passer par le réseau, ce qui est nécessaire pour de nombreux programmes.

On parle en français d'interface de boucle de retour, ou adresse de bouclage.)

L'interface réseau virtuelle utilisée dans cette situation se nomme l'interface de loopback (abrégée par lo sous Unix) ou boucle locale

l'adresse IPv4 127.0.0.1 constitue l'adresse loopback.c'est l'interface réseau réservée utilisée par le système local pour permettre les communications entre processus

L'hôte utilise cette adresse pour s'envoyer des paquets à lui-même.

Tout système du réseau TCP/IP doit utiliser l'adresse IP 127.0.0.1 pour le loopback IPv4 sur l'hôte local. on le voit bien dans la commande "ip addr" au dessus.

La commande ping vérifie si une machine distante répond en lui envoyant des paquets On peut aussi utiliser le nom de la machine, si celle-ci est renseignée dans votre fichier Hosts ou sur un serveur DNS

```
 ping localhost
PING localhost(localhost (::1)) 56 data bytes
64 bytes from localhost (::1): icmp_seq=1 ttl=64 time=0.037 ms
64 bytes from localhost (::1): icmp_seq=2 ttl=64 time=0.060 ms
64 bytes from localhost (::1): icmp_seq=3 ttl=64 time=0.048 ms
64 bytes from localhost (::1): icmp_seq=4 ttl=64 time=0.049 ms
```

La plage d'adresses IP 127.0.0.0 - 127.255.255.255 est réservée au bouclage.

L'adresse IP de bouclage est entièrement gérée par et au sein du système d'exploitation. Ces adresses permettent aux processus serveur et client d'un même système de communiquer entre eux. Lorsqu'un processus crée un paquet avec l'adresse de destination comme adresse de bouclage, le système d'exploitation le reboucle sur lui-même sans aucune interférence de la carte réseau.

Les données envoyées en boucle sont transmises par le système d'exploitation à une interface réseau virtuelle au sein du système d'exploitation. Cette adresse est principalement utilisée à des fins de test comme l'architecture client-serveur sur une seule machine.

Par exemple, si une machine hôte peut envoyer une requête ping à 127.0.0.2 ou à toute adresse IP de la plage de bouclage, cela signifie que la pile logicielle TCP / IP sur la machine est correctement chargée et fonctionne.

broadcast L'adresse de broadcast permet d'envoyer les données et les informations à tous les appareils d'un réseau. Les différents éléments du réseau se chargent alors de la réception et du traitement des données. Le but de l'adresse IP de broadcast est de connecter ensemble tous les appareils d'un réseau.

L'émetteur établit une connexion en broadcast dans laquelle il envoie son adresse afin de permettre aux destinataires de le contacter. Le broadcast fonctionne donc de façon similaire à une liste de diffusion dans laquelle les destinataires ne sont pas visibles et l'émetteur n'a pas à connaître les adresses des participants au réseau. Les participants révèlent uniquement leur adresse lorsqu'ils entrent en contact avec l'émetteur.

Quand une machine vient de démarrer, elle n'a pas de configuration réseau (même pas de configuration par défaut), et pourtant, elle doit arriver à émettre un message sur le réseau pour qu'on lui donne une vraie configuration. La technique utilisée est le broadcast : pour trouver et dialoguer avec un serveur DHCP (note1), la machine va simplement émettre un paquet spécial, dit de broadcast, sur l'adresse IP 255.255.255.255 et sur le réseau local. Ce paquet particulier va être reçu

par toutes les machines connectées au réseau (particularité du broadcast).

Lorsque le serveur DHCP reçoit ce paquet, il répond par un autre paquet de broadcast contenant toutes les informations requises pour la configuration. Si le client accepte la configuration, il renvoi un paquet pour informer le serveur qu'il garde les paramètres, sinon, il fait une nouvelle demande. Les choses se passent de la même façon si le client a déjà une adresse IP (négociation et validation de la configuration), sauf que le dialogue ne s'établit plus avec du broadcast.

3) la couche réseau :

elle s'occupe de tout ce qui a trait à internet : l'identification des différents réseaux à interconnecter, la spécification des transferts de données entre réseaux, leur synchronisation, etc.

C'est notamment cette couche qui s'occupe du routage, à savoir la découverte d'un chemin de transmission entre récepteur et émetteur, chemin qui passe par une série de machines ou de routeurs qui transmettent l'information de proche en proche. Le protocole principal de cette couche est le protocole IP. Le rôle de la couche 3 est donc d'interconnecter les réseaux.

IP (Internet Protocol) L'adresse IP est en fait l'adresse du réseau et la machine. Plus exactement, une partie de l'adresse représentera l'adresse du réseau, et l'autre partie l'adresse de la machine.

sous debian c'est le logiciel Iproute2 qui va gérer votre réseau : /sbin/ip

lrwxrwxrwx 1 root root 7 janv. 10 2019 /sbin/ip → /bin/ip Une adresse IP est codée sur 32 bits (soit 4 octets, car vous vous rappelez bien qu'un octet vaut 8 bits). Afin de simplifier la lecture et l'écriture d'adresses IP pour les humains, nous avons choisi d'écrire les adresses avec la notation en décimal pointée.

Cette dernière sépare les 4 octets sous forme de 4 chiffres décimaux allant de 0 à 255. Cela donne par exemple : 192.168.0.1 On en déduit au passage que la plus petite adresse IP est : 0.0.0.0 (quand tous les bits de l'adresse sont à 0) alors que la + grande vaut : 255.255.255.255 (quand tous les bits sont à 1).

Pour des raisons d'optimisation des ressources réseau, les adresses IP sont délivrées pour une durée limitée. C'est ce qu'on appelle un bail (lease en anglais). Un client qui voit son bail arriver à terme peut demander au serveur un renouvellement du bail. De même, lorsque le serveur verra un bail arrivé à terme, il émettra un paquet pour demander au client s'il veut prolonger son bail. Si le serveur ne reçoit pas de réponse valide, il rend disponible l'adresse IP. C'est toute la subtilité du DHCP : on peut optimiser l'attribution des adresses IP en jouant sur la durée des baux. Le problème est là : si toutes les adresses sont allouées et si aucune n'est libérée au bout d'un certain temps, plus aucune requête ne pourra être satisfaite. Un sous-réseau est un espace d'adresses IP qui est divisé en espaces d'adresses plus petits. Le sous-réseau devient ainsi une partie d'un réseau dans lequel toutes les adresses IP utilisent la même adresse réseau. Si tous les sous-réseaux sont connectés à un routeur Le masque de sous-réseau (subnet mask) la création de sous-réseaux permet de segmenter votre réseau et de réduire les collisions possibles entre des données différentes.

Nous allons en fait ajouter une information supplémentaire à l'adresse IP, le masque de sous-réseau.adresse IP et masque, seront inséparables.

C'est le masque qui va indiquer quelle est la partie réseau de l'adresse, et quelle est la partie machine. Dans une adresse IP, c'est la partie gauche qui correspond à l'identité du réseau, la partie réseau de l'adresse est 192.168

la suite étant la partie machine (0.1 ou 1.1 ou 1.12 comment c'est déterminé, sous quels critères

Nombre de machines

Le masque de sous réseau par défaut est 255.255.255.0. Dans ce cas, on peut avoir jusqu'à 254 terminaux (clients) dans ce même réseau, donc 254 adresses IP.

Vous avez 254 adresses IP disponibles uniquement lorsque vous utilisez un masque de sous-réseau par défaut.

En y regardant d'un peu plus près, on peut calculer le nombre de machines que l'on peut identifier à l'aide de cet adressage.

Ainsi, on utilise 4 octets, soit 32 bits, soit encore 2^32 adresses (2 exposant 32 adresses) Or 2^32 = 4 294 967 296, on peut donc définir un peu plus de 4 milliards d'adresses !!! *- oups :) pas toujours évident

Il nous suffit de dire que les bits à 1 représenteront la partie réseau de l'adresse, et les bits à 0 la partie machine. Ainsi, on fera une association entre une adresse IP et un masque pour savoir dans cette adresse IP quelle est la partie réseau et quelle est la partie machine de l'adresse

Les couches hautes, aussi appelées couches logicielles, contiennent des protocoles pour simplifier la programmation logicielle. Elles requièrent généralement que deux programmes communiquent entre eux sur le réseau. Elles sont implémentées par des bibliothèques logicielles ou directement dans divers logiciels. Le système d'exploitation ne doit pas, en général, implémenter les protocoles des couches hautes. Elles sont au nombre de quatre :

Les COUCHE HAUTES :

4) `la couche transport :` permet de gérer la communication entre deux programmes, deux processus. Les deux protocoles de cette couche sont les protocoles TCP et UDP.

5) `la couche session :` comme son nom l'indique, permet de gérer les connexions et déconnexions et la synchronisation entre deux processus. 6) `la couche présentation :` se charge du codage des données à transmettre. Elle s'occupe notamment des conversions de boutisme ou d'alignement, mais aussi du chiffrement ou de la compression des données transmises.

7) `la couche application :` prend en charge tout le reste. (pour l'instant c'est vague comme explication) Chaque couche ajoute sa propre en-tête à l'information d'origine Ce procédé s'appelle 'encapsulation

La couche 2 permet à deux machines d'acheminer les données d'un ordinateur à un autre. directement connectées de dialoguer, on dit alors que les machines sont sur un même réseau

La couche 3 permet le dialogue entre réseaux, ce que l'on appelle le routage. deux machines sur un même réseau pourront dialoguer directement, mais pour parler à une machine située sur un réseau distant, il faudra passer par des machines intermédiaires qui feront la liaison entre les réseaux (on appellera ces machines intermédiaires des routeurs).

Ces routeurs vont donc recevoir les paquets sur un réseau, et les renvoyer sur l'autre.

L'adressage MAC en couche 2 permet d'identifier les machines SUR UN MÊME RÉSEAU.

L’adressage IP en couche 3 permet d’adresser les machines SUR DES RÉSEAUX DISTINCTS.

Si vous avez compris les masques de sous réseau, vous savez que le masque permet notamment à une machine de savoir quelles sont les autres machines de son réseau.

Ainsi, quand une machine veut dialoguer avec une autre, elle va d’abord regarder si cette machine est sur son propre réseau, ou si elle va devoir passer par des routeurs intermédiaires pour lui envoyer son paquet. Ceci va être possible grâce à la table de routage.

Si le modèle osi comporte 7 couches, que nous venons d’énoncer ci dessus.

Le modèle TCP/IP, a seulement quatre couches : liaison, Internet, transport et application.

La couche liaison de TCP/IP regroupe notamment les couches physiques et liaison d’OSI. De même, la couche application de TCP/IP regroupe les couches session, application et présentation d’OSI. TCP/IP est un sigle qui recouvre deux protocoles utilisés par de nombreuses sociétés commercialisant des équipements de réseau. Ces deux protocoles IP (Internet Protocol) et TCP (Transmission Control Protocol) forment respectivement la couche réseau et la couche de transport qui ont été développées pour les besoins d’interconnexion des divers réseaux hétérogènes de la défense américaine.

L’idée de base est simple, rendre ces réseaux homogènes en leur imposant un protocole commun, le protocole IP. De cette façon, pour passer d’un sous-réseau à un autre sous réseau, il faut passer par le protocole IP qui gère le routage.

Dans les faits, ce sigle TCP/IP représente beaucoup plus que les deux protocoles développés pour interconnecter des sous réseaux entre eux ; il désigne tout un environnement qui contient, bien sûr, les protocoles TCP et IP mais aussi les applications qui ont été développées au dessus de ces deux protocoles : la messagerie électronique dénommée SMTP (Simple Mail Transport Protocol), le transfert de fichiers FTP (File Transfer Protocol), l’accès à des bases d’informations WWW (World Wide Web), etc.

la combinaison adresse IP + port est alors une adresse unique au monde, elle est appelée socket).

L’adresse IP sert donc à identifier de façon unique un ordinateur sur le réseau tandis que le numéro de port indique l’application à laquelle les données sont destinées. De cette manière, lorsque l’ordinateur reçoit des informations destinées à un port, les données sont envoyées vers l’application correspondante.

S’il s’agit d’une requête à destination de l’application, l’application est appelée application serveur.

S’il s’agit d’une réponse, on parle alors d’application cliente.

TCP/IP est un protocole, c’est à dire des règles de communication

IP signifie Internet Protocol : littéralement “le protocole d’Internet”. C’est le principal protocole utilisé sur Internet.

Internet signifie Inter networks, c’est à dire “entre réseaux”. Internet est l’ interconnexion des réseaux de la planète.

Le protocole IP permet aux ordinateurs reliés à ces réseaux de dialoguer entre eux.

UDP/IP est un protocole qui permet justement d’utiliser des numéros de ports en plus des adresses IP

(On l'appelle UDP/IP car il fonctionne au dessus d'IP).

IP s'occupe des adresses IP et UDP s'occupe des ports.

Chaque couche (UDP et IP) va ajouter ses informations.

Les informations de IP vont permettre d'acheminer le paquet à destination du bon ordinateur. Une fois arrivé à l'ordinateur en question, la couche UDP va d livrer le paquet au bon logiciel (ici : au serveur HTTP).

• Les deux logiciels se contentent d' émettre et de recevoir des données ("Hello !"). Les couches UDP et IP en dessous s'occupent de tout.

Ce couple (199.7.55.3:1057, 204.66.224.82:80) est appel é un socket. Un socket identifie de façon unique une communication entre deux logiciels.

TCP est capable : de faire tout ce que UDP sait faire .

- de vérifier que le destinataire est prêt à recevoir les données.

- de découper les gros paquets de données en paquets plus petits pour que IP les accepte

- de numéroter les paquets, et à la réception de vérifier qu'ils sont tous bien arrivés, de

- redemander les paquets manquants et de les réassembler avant de les donner aux logiciels.

- Des accusés de réception sont envoyés pour prévenir l'expéditeur que les données sont bien arrivées.

Internet est donc l'interconnexion de tous les réseaux de la planète.

Le protocole DNS permet de retrouver une adresse IP en fonction d'un nom d'ordinateur (un peu comme un annuaire).

• Le protocole FTP sert à transporter des fichiers d'un ordinateur à l'autre.

• Le protocole IR C permet de créer des « salons » de discussion en direct.

• Le protocole ICQ permet de savoir si quelqu'un est en ligne et de dialoguer avec lui.

• Le protocole NTP permet de mettre les ordinateurs à l'heure par internet à 500 millisecondes près.

• Les protocoles P2P permettent de partager des fichiers à grande échelle.

• Le protocole NNTP permet d'accéder à des forums de discussion sur des milliers de sujets différents.

• Le protocole SSH permet d'avoir un accès sécurisé à des ordinateurs distants.

• Le protocole SMTP permet d'envoyer des emails, et le protocole POP3 de les recevoir.

• D'autres protocoles permettent de faire du téléphone ou de la visio conférence. • etc.

Tout ces protocoles utilisent le protocole IP, le protocole d'internet (IP = « Internet Protocol »).

On dit qu'ils sont transportés par IP (c'est en effet le protocole IP qui est chargé de transporter les

paquets de données jusqu'à la destination).

Et comme internet est égalitaire, il accepte de transporter n'importe quel protocole du moment que vous utilisez le protocole IP.

Cela veut dire que vous pouvez développer votre propre protocole. Internet acceptera de transporter vos données sans problème.Vous pouvez inventer des protocoles et les utiliser pour communiquer (du moment que votre correspondant en face comprend le protocole que vous avez inventé).

La connexion physique qui relie tous les types de réseau peut être câblée (filaire) ou bien réalisée à l'aide de la technologie sans fil. Bien souvent les réseaux de communication physique constituent le fondement de plusieurs réseaux logiques, appelés VPN (Virtuel Private Network, ou réseau privé virtuel en français).

Ceux-ci utilisent un moyen de transmission physique commun, par exemple un câble de fibre optique et, lors du transfert des données, sont assignés à des réseaux virtuels logiquement différents au moyen d'un logiciel de VPN créant un tunnel (ou logiciel de tunneling). Le modèle OSI est une norme précisant comment les machines doivent communiquer entre elles.

Chaque couche ne peut communiquer qu'avec une couche adjacente

Pour comprendre cette règle, vous allez devoir comprendre comment les machines se servent du modèle OSI pour communiquer.

Vous êtes devant votre ordinateur et votre navigateur préféré. Vous entrez l'adresse d'un site dans la barre d'adresses, et le site apparaît aussitôt.

Sans le savoir, vous avez utilisé le modèle OSI ! en gros, l'application (le navigateur) de couche 7, s'est adressée aux couches réseau pour que celles-ci transmettent l'information à l'application demandée sur la machine demandée (le serveur web sur la machine google.com par exemple).

Lors d'un envoi, nous parcourons donc les couches du modèle OSI de haut en bas, de la couche 7 à la couche 1, ainsi que vous pouvez le voir sur la figure suivante.

- - application

- - présentation

- - session

- - transport

- - réseau

- - liaison de données

- - physique

Suite ici :aperçu de commandes d'administration réseau

https://debian-facile.org/utilisateurs:lagrenouille:tutos:utilisation-des-commandes-reseaux

tutos DF:

https://debian-facile.org/doc:reseau:ip-publique

https://debian-facile.org/doc:reseau:apt-p2p

https://debian-facile.org/doc:reseau:mail:mutt

https://debian-facile.org/doc:chargement_module_livecd