IP : Internet Protocol 09.12.2023

Avec des IP on visite tous les "PI" ☺

Auteur : Pascal Fougeray

source : https://www.journaldugeek.com/2010/05/19/encore-de-lhumour-de-geek/

1 Introduction

Le terme adresse IP signifie adresse de protocole Internet.

Une adresse de protocole Internet est un **"nom" numérique unique** attribué à tout appareil électronique connecté à un réseau informatique.

Unique : si elle est **publique**

Il en existe 2 types :

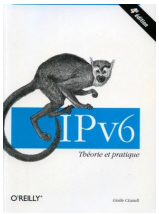
1. IPv4 l'ancêtre qui résiste ☺
2. IPv6 le futur qui doit remplacer IPv4 depuis plus de 13 ans mais qui se bat contre des vieux démons que sont les fabricants de matériel.

   IPv6 est tellement lourd que d'un point de vue logiciel et humain tout le monde (ou presque) s'y perd.

   Je ne pense (Ah oui, je rappelle que seul Pascal pense ☺)) pas que je vous ferai un CM/TD et un TP sur Ipv6.

   Mais sachez qu'il existe...

   Si certains veulent apprendre IPv6, je vous conseille ce livre :



N'essayez pas de rencontrer l'auteur : ***Gisele Cizault*** , c'est le nom d'**emprunt** d'un collectif d'universitaires et d'ingénieurs, qui participent au travers du G6 à la mise au point d'IPv6 sur le plan international. La dernière version date de 2011 et la première de 1998...

2 IPv4

Il est sur la **couche 3 du modèle OSI** : La couche réseau que nous aurons l'occasion de voir plus en détails lors du cours sur le **routage**.

Internet Protocol ou ***Darpa Internet Program***
Il est défini par la RFC 791 en septembre 1981... ***https://tools.ietf.org/html/rfc791***
Si vous voulez lire une RFC c'est celle-ci ! Elle ne fait que 45 pages en Anglais !

Son concepteur **Vinton Gray Cerf** 

2.1 Les adresses IP

Chaque machine sur Internet, Ordinateurs et Routeurs, possède une adresse IP sur chacune de ses interfaces ou presque.

J'écris presque parce qu'il est possible d'avoir des interfaces sans IP mais chut, vous faites comme si je ne vous avais rien dit pour l'instant...

En réalité il n'est pas nécessaire d'en mettre sur toutes les interfaces mais dans ce cas là on n'est pas sur la couche 3 des modèles OSI et TCP/IP, mais la couche 2.

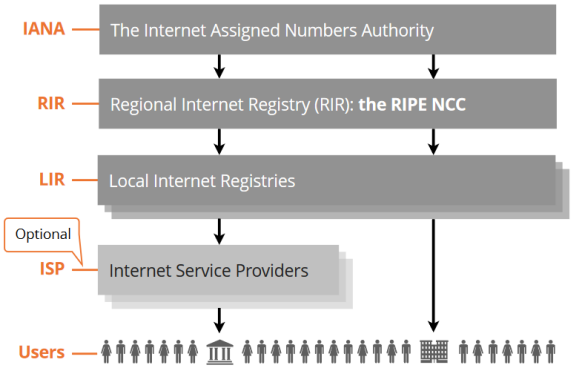
Ces adresses IP sont uniques et sont attribuées par /8 (c'est quoi ce truc ? ? ?) par l'ICANN aux RIR.

Les RIR les distribuent aux LIR en /22.

Les LIR peuvent être des ISP (ou FAI) ou bien des organisations tel RENATER qui fournit l'accès à Internet à l'Université de Caen et aux autres Université.

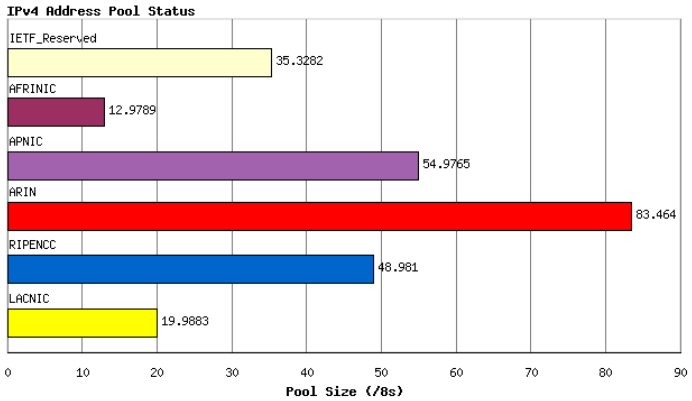
https://www.renater.fr/

Au service de la communauté Education-Recherche, RENATER offre un réseau hautement fiable et sécurisé, facilitant la collaboration et la convergence de projets scientifiques et académiques. (Wahou la pub ☺)



Source slide 7/150 de https://www.ripe.net/support/training/material/lir-training-course/lir-slides.pdf
Ne cherchez pas à en acheter un /8 ou /22, depuis 2011 il n'y en a plus... à vendre ...
Et oui depuis 2011, IPv4 est à sec et pourtant ça fonctionne encore ...

Voici une distribution très "équitable" des @ IP dans le monde



source : **http://www.potaroo.net/tools/ipv4/**
Les comptes : 35,3282+12,9789+54,9765+83,464+48,981+19,9883= 256 (de 0 à 255) /8 ok, tout est alloué...
Voyons comment cela fonctionne !

2.2 Fonctionnement des adresses IP

On fera cela plus en détails en TD, mais un peu de cours ça ne peut pas faire de mal.
Je ne vais que parler d'IPv4 car pour Ipv6, je vous invite à aller à un cours de plusieurs heures...
Voyons comment sont organiser ces @IP
À l'origine, les adresses IP étaient divisées en 5 classes.
Si une entreprise voulait se connecter à Internet, elle devait choisir une adresse IP dans la classe appropriée.
Pour chaque classe, différents nombres d'octets ont été utilisés pour **identifier les réseaux**. Les octets restants déterminent **le nombre d'hôtes** dans un réseau.

Classe	Classe A	Classe B	**Classe C**	Classe D	Classe E
Plage IP	0.0.0.0 - 127.255.255.255	128.0.0.0 - 191.255.255.255	192.0.0.0 - 223.255.255.255	224.0.0.0 - 239.255.255.255	240.0.0.0 - 255.255.255.255
	/8	/16	/24	Multicast	/4 réservé ...
Masque	255.0.0.0	255.255.0.0	255.255.255.0	non défini	non défini
1er bits	0	10	110	1110	1111
Nb hôtes/Rx	16 777 214	65534	256	4096	
Nb Rx Possibles	128	16384	8192		
Total	2 Milliards	1 Milliard	2 Millions		

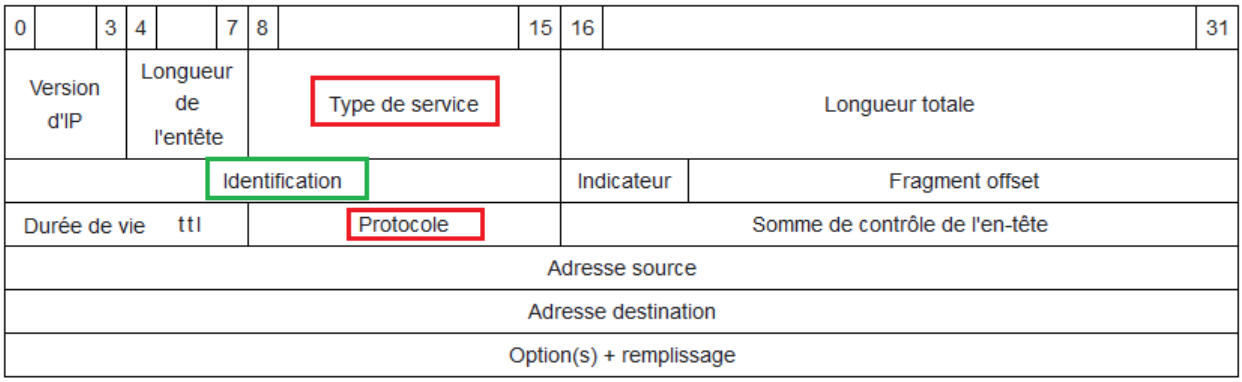
Dès **1993**... oui oui c'est vieux, on s'est rendu compte que l'on ne pouvait pas continuer comme cela et le CIDR (***classless interdomain routing***) est né.
Le CIDR aide à augmenter le nombre d'adresses disponibles sans en ajouter, la magie du découpage !
En gros on peut faire des /22 /12 /25 /31 /32 /xx comme on veut !

À savoir faire et à connaître par cœur ! ! ! ! ! ! ! ! ! ! ! !

1. Le calcul du nombre hôtes par réseau est très simple !
   À connaître pour le CT ! ! !
   — Il suffit de faire $2^{32-/x} - 2$
   exemple un /22 fait $2^{10} - 2$=1022 hôtes possibles car
   — la valeur la plus **basse** est l'@ de **Rx**
   — la valeur la plus **haute** est l'@ de **broadcast** !
   — Exemple : 192.168.128.0/22 donne
   — 192.168.128.0 @de **réseau**
   — 192.168.128.1 à 192.168.131.254 les @ des **hôtes**
   — 192.168.131.255 @ de **broadcast**
2. Calcul du Masque, il suffit de mettre les /x premiers bits à 1 et les autres à 0
   Dans un /10 cela donne 255.255.252.0
   Table des /x

/8	255.0.0.0		
/16	255.255.0.0		
/24	255.255.255.0		
/22	255.255.252.0	**Université** et **LIR**	
/30	255.255.255.252	liaison point à point	
/31	255.255.255.254	liaison point à point	
/32	255.255.255.255	Loopback	

2.3 L'entête IPv4



Définition des différents champs

Je ne vais pas tous les définir pour des raisons d'utilités et on ne peut pas tout savoir et voir !

Version (4 bits) : 4 pour Ipv4 et 6 pour Ipv6

ToS ou QoS : Il permet de distinguer différentes qualités de service (QoS) différenciant la manière dont les paquets sont traités. C'est la priorité !

Identification sur 16 bits donc 65536 paquets différents possibles ce qui fait beaucoup. C'est un N° permettant d'identifier les fragments (**fragmentation**) d'un même paquet. Surtout utilisé en TCP.

TTL Durée de vie ou ***Time To Live*** sur 8 bits. Initialisé par l'émetteur, ce champ est décrémenté d'une unité généralement à **chaque saut de routeur**. Quand il arrive à 0, le paquet est abandonné et un message **ICMP** (voir plus loin ICMP) est envoyé à l'émetteur pour information.

RemarTTLque : Il y a bien lontemps un routeur mettait en moyenne 1s pour traiter un paquet, maintenant c'est en ms... et le TTL ne représente plus un temps mais un nombre de sauts !

2.4 Les @ IP spécifiques

Certaines adresses sont réservées à un usage particulier !

Nous avons vu que toutes les adresses IP n'étaient pas disponibles, celles qui sont **réservées** et les adresses **Multicast**

Dans les adresses ***unicast*** comprises entre 0.0.0.0 et 223.255.255.255 et bien toutes sont disponibles mais pas pour faire la même chose.

Je ne vais pas tout détailler mais ce que j'écris ici est important et doit être su !

Bloc	Usage	Total	
0.0.0.0/8	Ce réseau	16 777 216	
10.0.0.0/8	**Adresses privées**	16 777 216	La Fac de Caen
127.0.0.0/8	adresse de bouclage (**localhost**)	16 777 216	
169.254.0.0/16	adresses locales auto configurées (**APIPA**)	65536	
172.16.0.0/12	**Adresses privées**	1048576	
192.168.0.0/16	**Adresses privées**	65536	

Toutes ces adresses ne doivent pas être **routable** sur Internet ! ! !

2.5 Qu'est-ce que le réseau 169.254.0.0/16 ?

C'est l'**APIPA *(Automatic Private Internet Protocol Addressing*)** ou **IPv4LL**.

Un processus qui permet à une interface de s'attribuer automatiquement une adresse IP, lorsque **le serveur DHCP est hors service ou injoignable** .

APIPA utilise la plage d'adresses IP 169.254.0.0/16, c'est-à-dire la plage dont les adresses vont de 169.254.0.0 à 169.254.255.255.

Cette plage est réservée à cet usage auprès de l'IANA.

Et dire que l'on manque d'@IPv4 ... Bref

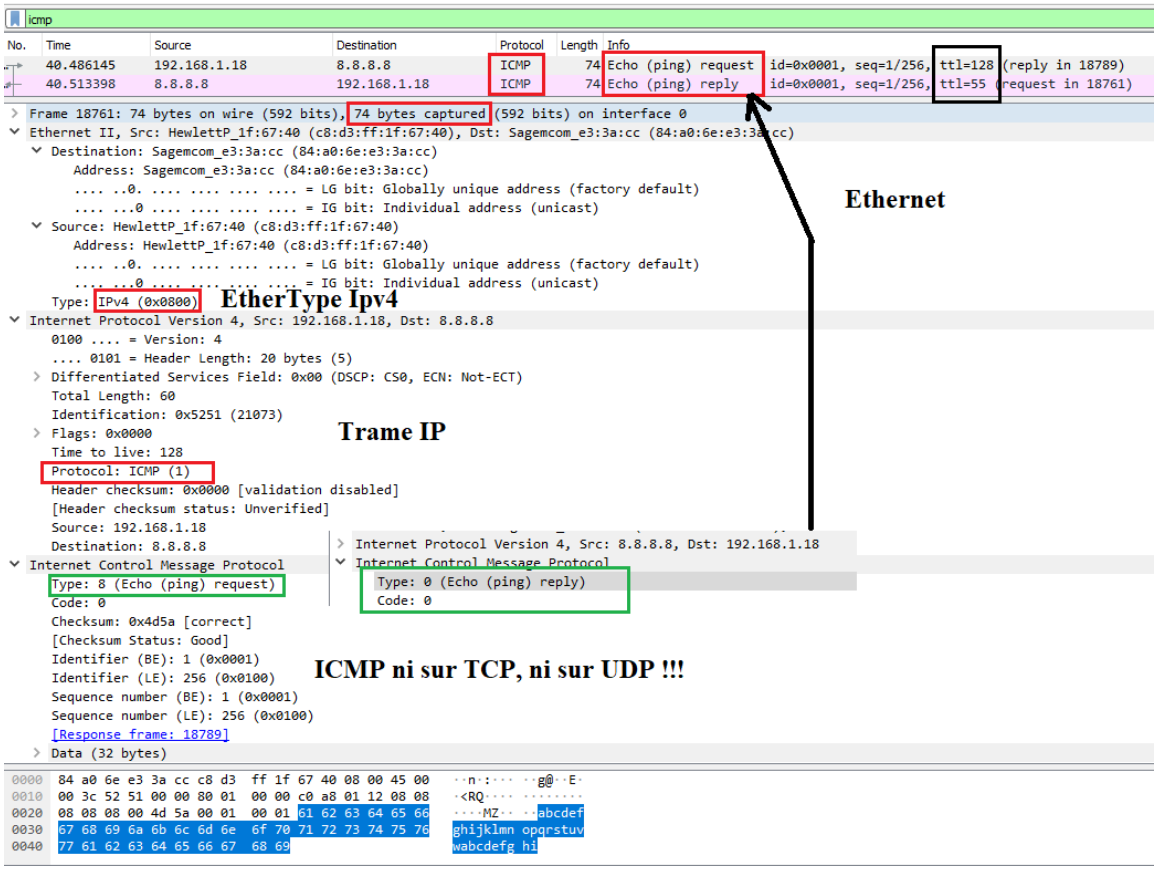
3 ICMP

Internet Control Message Protocol est l'un des protocoles fondamentaux d'IP.
Il est utilisé pour véhiculer des messages de **contrôle** et d'**erreur** pour cette suite de protocoles.
Exemples :

— Voir si un hôte est présent et s'il répond à l'aide de la commande **ping @IP** du destinataire
— Lorsqu'un service ou un hôte est inaccessible, la commande ***traceroute***
— Et aussi la commande ***nmap*** pour trouver les ports ouverts !

3.1 Le ping,

c'est la commande surement la plus utilisée dans le monde du réseau. Elle permet de "rassurer" l'administrateur réseau. Savoir si un Hôte est présent.
Dans ICMP c'est le type 8... mais bon on s'en moque ^^
Allez une capture Wireshark ça explique tout !



Remarque : **Ce n'est pas parce qu'une machine ne répond pas au ping qu'elle n'est pas là !**
En effet il est possible de modifier les paramètres du système donc paramètres du noyau Linux pour bloquer les réponses au ping.
Cette astuce n'a pas vraiment d'intérêt, sauf si vous voulez masquer un peu plus la présence de votre machine sur un réseau.
La commande ***nmap*** est capable d'outrepasser cela ***nmap -sP 127.0.0.1***
Méthode :
Pour **désactiver** la réponse au ping, **ouvrez** un terminal en root et **lancez** la commande suivante :
— ***echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_all***
et
— ***echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_all*** pour annuler !
Cela n'est pas permanent !
Pour effectuer une modification permanente, il faut :

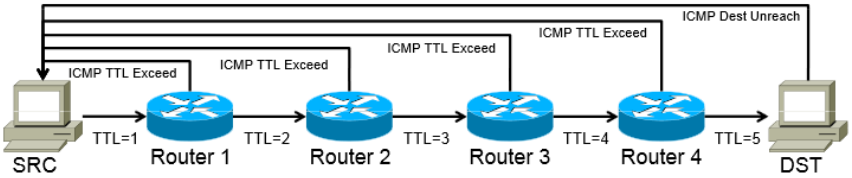
1. Modifier votre fichier ***/etc/sysctl.conf***
   — ***echo "net.ipv4.icmp_echo_ignore_all = 1" >> /etc/sysctl.conf***
2. Valider à l'aide de la commande

— ***sysctl -p /etc/sysctl.conf***

A partir de maintenant l'ordinateur ne répond plus aux pings et autres commandes utilisant le protocol ICMP

3.2 Le traceroute & tracert

L'utilitaire de diagnostic **traceroute** (Linux) ou **tracert** (Windows) **détermine l'itinéraire vers une destination** en envoyant des paquets d'écho ICMP à la destination.



Dans ces paquets, il utilise des valeurs de durée de vie (***Time-To-Live*, TTL**) IP différentes.

Un TTL correspond à un saut d'un point de vue IP, on verra cela plus en détails lors du cours sur le routage.

Étant donné que chaque routeur sur l'itinéraire doit diminuer la durée de vie d'un paquet d'au moins 1 avant de le transférer, la TTL représente effectivement le nombre de sauts.

Lorsque la TTL d'un paquet atteint zéro, le routeur renvoie un message **ICMP Temps dépassé à l'ordinateur source**.

Principe : Il envoie le premier paquet d'écho avec une TTL de 1, puis augmente la TTL de 1 à chaque transmission suivante jusqu'à ce que la destination réponde, par un **ICMP Dest Unreachable**, ou que la valeur maximale de la TTL soit atteinte.

Les messages ICMP Temps dépassé qui sont renvoyés par les routeurs intermédiaires indiquent l'itinéraire.

Notez cependant que certains routeurs rejettent silencieusement des paquets dont la durée de vie a expiré, et que ceux-ci sont **invisibles**.

Traceroute ou **tracert** impriment une liste triée des routeurs intermédiaires qui renvoient des messages ICMP Temps dépassé.

Voici un exemple fait sous windows donc **tracert**.

Vous pouvez le faire de chez vous et vous n'obtiendrez pas le même tracé sauf si vous êtes chez le même FAI.

Interprétons et voyons ce qui se passe.si j'essaie de trouver le serveur Web de la FAC de Caen.

```
C:\WINDOWS\system32>tracert www.unicaen.fr

Détermination de l'itinéraire vers rp5.unicaen.fr [193.55.120.26]
avec un maximum de 30 sauts :

  1    <1 ms    <1 ms    <1 ms  bbox.lan [192.168.1.254]
  2    23 ms    23 ms    24 ms  119-les03-th2-31-37-224-2.sfr.lns.abo.bbox.fr [31.37.[redacted]]
  3    26 ms    27 ms    26 ms  be21.cbr01-ntr.net.bbox.fr [212.194.171.28]
  4     *       25 ms    26 ms  62.34.2.56
  5    34 ms    32 ms    32 ms  renater.par.franceix.net [37.49.236.19]
  6    30 ms    30 ms    30 ms  193.51.180.42
  7     *        *        *     Délai d'attente de la demande dépassé.
  8     *        *        *     Délai d'attente de la demande dépassé.
  9    31 ms    31 ms    32 ms  syrhano-vl3201-te4-3-rouen-rtr-021.noc.renater.fr [193.51.184.129]
 10     *        *        *     Délai d'attente de la demande dépassé.
 11    44 ms    38 ms    38 ms  s2-sihes.syrhano.net [193.48.154.149]
 12     *        *        *     Délai d'attente de la demande dépassé.
 13     *        *        *     Délai d'attente de la demande dépassé.
 14     *        *        *     Délai d'attente de la demande dépassé.
 15     *        *        *     Délai d'attente de la demande dépassé.
 16    38 ms    39 ms    39 ms  rp5.unicaen.fr [193.55.120.26]

Itinéraire déterminé.
```

?

Mon-ip.com est le plus rapide et le plus simple chemin pour déterminer votre **adresse IP**. C'est l'adresse sous laquelle vous êtes connu sur Internet.

Votre adresse IP est : 31.37.248.248 @fausse [Informations complètes ici]

Son nom d'hôte associé : i16-les03-th2-31-37-248-194.sfr.lns.abo.bbox.fr

Port Utilisé : 29248

Votre IP Locale : Découvrez votre adresse IP locale en cliquant ici

Et sous Linux en étant connecté dans la salle S3-159

```
$ traceroute www.unicaen.fr
traceroute to www.unicaen.fr (10.14.128.61), 30 hops max, 60 byte packets
 1  10.38.16.2 (10.38.16.2)  0.336 ms  0.288 ms  0.491 ms
 2  i-ac5b1-ucaen-761-fwdc1-root.interco.unicaen.fr (10.1.1.36)  0.448 ms  0.439 ms  0.409 ms
 3  i-fwdc1-root-761-ac5b1-ucaen.interco.unicaen.fr (10.1.1.37)  0.355 ms  0.325 ms  0.295 ms
 4  i-ac5b1-serveur-746-fwdc1-root.interco.unicaen.fr (10.1.1.110)  0.521 ms  0.490 ms  0.460 ms
 5  ksup.unicaen.fr (10.14.128.61)  0.391 ms  0.377 ms  0.362 ms
                    fougeray@C304L-159C00:~$
```

```
$ traceroute calebasse.campus.unicaen.fr
traceroute to calebasse.campus.unicaen.fr (10.14.136.100), 30 hops max, 60 byte packets
 1  10.38.16.2 (10.38.16.2)  0.335 ms  0.523 ms  0.490 ms
 2  * * *
 3  i-fwdc1-root-761-ac5b1-ucaen.interco.unicaen.fr (10.1.1.37)  0.338 ms  0.307 ms  0.271 ms
 4  * * *
 5  calebasse.campus.unicaen.fr (10.14.136.100)  0.643 ms  0.625 ms  0.593 ms
                    fougeray@C304L-159C00:~$
```

Question : Pourquoi ce ne sont pas les mêmes adresses ?

Réponse : À **l'extérieur** de la Fac ce sont des @IP **publics** à **l'intérieur** des @IP **privées** ! ! !

4 ARP

L'***Address Resolution Protocol*** (ARP, protocole de résolution d'adresse) est un protocole utilisé pour traduire une @ de protocole de couche réseau donc IPv4 en une @ de protocole de couche de liaison donc une @ MAC. Il se situe à l'interface entre la couche réseau (couche 3 du modèle OSI) et la couche de liaison (couche 2 du modèle OSI).

Il a été défini dans la RFC 8261 : An Ethernet Address Resolution Protocol.

Le protocole ARP est nécessaire au fonctionnement d'IPv4 utilisé au-dessus d'un réseau de type Éthernet.

Remarque : En IPv6, les fonctions de ARP sont reprises par le ***Neighbor Discovery Protocol*** (**NDP**)

Un ordinateur connecté à un réseau informatique veut émettre une trame Ethernet à destination d'un autre ordinateur dont il ne connaît que l'@ IP et placé dans le **même réseau !**

S'il ne connaît pas l'@ MAC du destinataire, il va placer son émission en attente et effectuer une requête ARP en **broadcast** de niveau 2 (FF :FF :FF :FF :FF :FF).

Cette requête est de type « *Quelle est l'adresse MAC correspondant à l'adresse IP adresseIP ? Répondez à monAdresseIP* ».

Who as 8.8.8.8 ? Tell 16.64.33.51

Puisqu'il s'agit d'un ***broadcast***, tous les ordinateurs du segment vont recevoir la requête.

En observant son contenu, ils pourront déterminer quelle est l'adresse IP sur laquelle porte la recherche.

La machine qui possède cette adresse IP sera la seule à répondre en envoyant à la machine émettrice une réponse ARP du type « *je suis adresseIP, mon adresse MAC est adresseMAC* ».

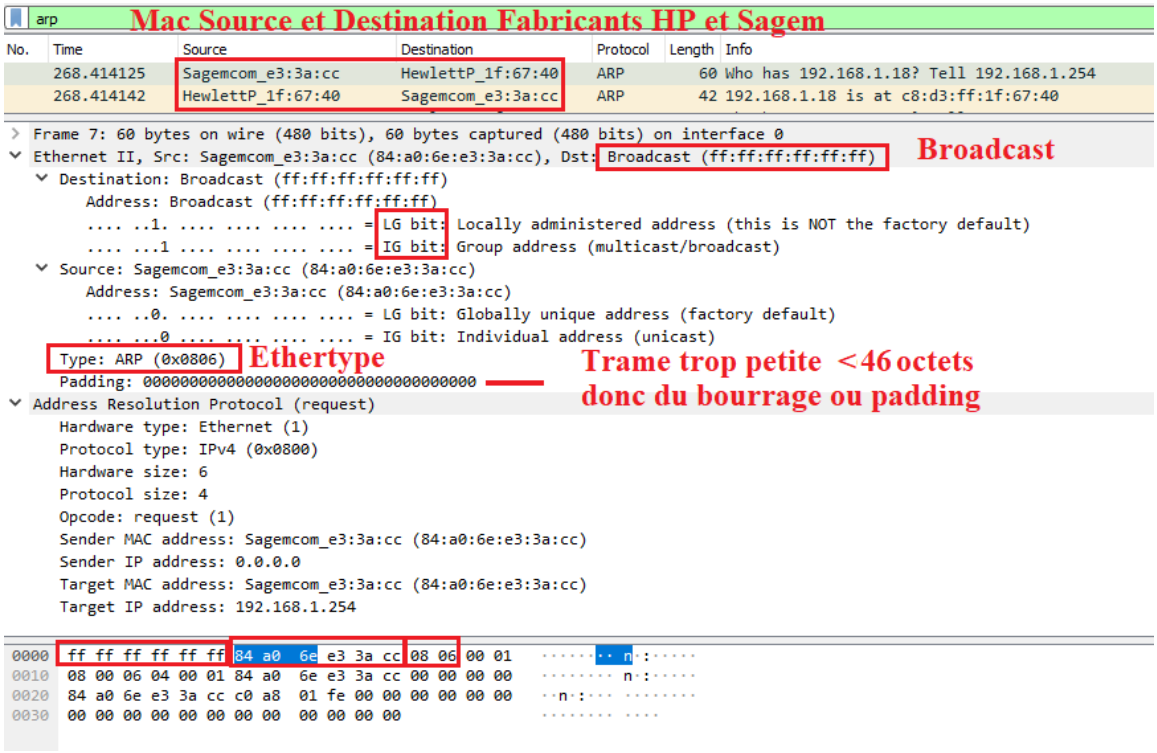
8.8.8.8 is at 45 :AB :C4 :F8 :BA

Pour émettre cette réponse au bon ordinateur, il crée une entrée dans son cache ARP à partir des données contenues dans la requête ARP qu'il vient de recevoir.

La machine à l'origine de la requête ARP reçoit la réponse, **met à jour son cache ARP** et peut donc envoyer à l'ordinateur concerné le message qu'elle avait mis en attente.

Il suffit donc d'un broadcast et d'un unicast pour créer une entrée dans le cache ARP de 2 ordinateurs.

Je ne vais pas détailler plus en détails ce protocole juste vous montrer une capture via Wireshark



ARP permet donc la correspondance entre @IP et @MAC et vice versa

Une trame ARP ne traverse pas un routeur ! ! !

5 Sous linux

On pourrait le faire sous un autre OS, Windows et MacOsX disposent aussi de ces commandes !

5.1 ARP

Pour connaître le cache ARP d'une machine et bien c'est la commande **arp** tout simplement qui renvoie :

```
etudiant@machine:~$ arp
Address HWtype HWaddress Flags Mask Iface
machine.mshome.net ether 00:15:5d:9d:84:3a C eth0
```

Elle lit dans le fichier /proc/net/arp ce que le noyau connait !

```
etudiant@machine:~$ cat /proc/net/arp
IP address HW type Flags HW address Mask Device
192.168.1.254 0x1 0x2 84:a0:6e:e3:3a:cc * enp0s3
```

ATTENTION : Une machine ne peut connaître que les @MAC des machines reliées à elle par un fil (donc la couche 1) et/ou un switch (donc la couche 2)

Une trame ARP ne traverse pas un routeur ! ! !

5.2 Ping

C'est l'acronyme de ***Packet INternet Groper***) est l'outil d'administration de réseau le plus connu.

Il est sûrement le plus simple puisqu'il permet, grâce à l'envoi de paquets, de **vérifier si une machine distante répond** et, par extension, qu'elle est accessible par le réseau.

Elle s'appuie sur le protocole ICMP.

ping - send ICMP ECHO_REQUEST to network hosts

Elle peut rendre plusieurs paramètres !

-c le nombre de paquets

-I pour choisir l'interface que l'on veut utiliser pour sortir de la machine ! ! !

Elle fait de l'***echo Request*** et de l'***echo Reply***

ICMP n'est pas ping !

5.3 IP ou ifconfig ou ipconfig (Windows)

Il existe 2 commandes pour affecter une adresse IP à une interface

1. La commande **ifconfig** qui est l'ancienne commande et que l'on ne devrait plus utiliser... mais bon je l'utilise qu'en même !

   Si on veut l'utiliser il faut être **root** et avoir installé (*apt install*) le paquet **net-tools** (Pas installé par défaut !)

   Exemple : ***ifconfig enp0s3 192.168.16.1 netmask 255.255.255.0***

   Attention : La commande ifconfig n'est pas dans le PATH par défaut d'un utilisateur.

   En effet, si on lance
   - la commande ***which ifconfig*** elle renvoie **/usr/sbin/ifconfig**
   - la commande **echo $PATH** elle renvoie
   - pour un utilisateur : usr/local/bin :/usr/bin :/bin :/usr/local/games :/usr/games <– cool des jeux ☺
   - pour root : /usr/local/sbin :/usr/local/bin :**/usr/sbin** :/usr/bin :/sbin :/bin

2. La commande **ip** avec les bonnes options, et il y en a beaucoup mais vraiment beaucoup ... mais bon je l'utilise qu'en même !

   L'utiliser avec certaines options il faut être **root** et avoir installé le paquet **iproute2** (Installé par défaut !)

   Exemple : ***ip addr add 192.168.16.1/24 dev enp0s3***

   Si vous voulez en savoir plus sur la commande **ip** et bien vous avez l'embarras du choix sur Internet et sur ecampus je vous ai mis un pdf .

   Un site sympa : http://www.linuxcertif.com/doc/keyword//bin/ip/

```
etudiant@machine:~$ ip
Usage: ip [ OPTIONS ] OBJECT { COMMAND | help }
       ip [ -force ] -batch filename
where  OBJECT := { link | address | addrlabel | route | rule | neigh | ntable |
                   tunnel | tuntap | maddress | mroute | mrule | monitor | xfrm |
                   netns | l2tp | fou | macsec | tcp_metrics | token | netconf | ila |
                   vrf | sr | nexthop | mptcp }
       OPTIONS := { -V[ersion] | -s[tatistics] | -d[etails] | -r[esolve] |
                    -h[uman-readable] | -iec | -j[son] | -p[retty] |
                    -f[amily] { inet | inet6 | mpls | bridge | link } |
                    -4 | -6 | -I | -D | -M | -B | -0 |
                    -l[oops] { maximum-addr-flush-attempts } | -br[ief] |
                    -o[neline] | -t[imestamp] | -ts[hort] | -b[atch] [filename] |
                    -rc[vbuf] [size] | -n[etns] name | -N[umeric] | -a[ll] |
                    -c[olor]}
```

Pour information : Sous Windows c'est la commande **ipconfig**
Exemple

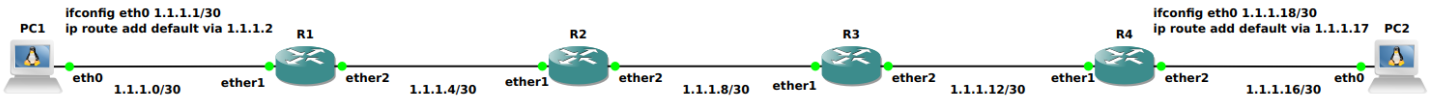
```
Carte Ethernet VirtualBox Host-Only Network :

   Suffixe DNS propre à la connexion. . . :
   Adresse IPv6 de liaison locale. . . . .: fe80::fdcf:81dd:7751:c38%10
   Adresse IPv4. . . . . . . . . . . . . .: 192.168.56.1
   Masque de sous-réseau. . . . . . . . . : 255.255.255.0
   Passerelle par défaut. . . . . . . . . :
```

5.4 Traceroute

Pour l'utiliser pas besoin d'être **root** mais il faut que le paquet **traceroute** soit installé !
Ici on fait un traceroute vers le DNS de Google...
Le jour où il ne répond plus, dites vous que ça vient de chez vous ☺

```
etudiant@machine:~$ traceroute 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  Seize64.mshome.net (172.22.48.1)  0.298 ms  0.270 ms  0.260 ms
 2  bbox.lan (192.168.1.254)  3.537 ms  3.528 ms  3.520 ms
 3  i19-les03-th2-31-37-248.248.sfr.lns.abo.bbox.fr (31.37.248.248)  17.607 ms  28.586 ms  23.897 ms
 4  * * *
 5  62.34.2.125 (62.34.2.125)  28.512 ms  38.015 ms  28.497 ms
 6  * * *
 7  72.14.204.68 (72.14.204.68)  27.058 ms  25.460 ms  25.440 ms
 8  * * *
 9  dns.google (8.8.8.8)  27.021 ms  24.924 ms  22.565 ms
```

On peut faire du **traceroute** en **UDP** ou en **TCP**. Pour ce dernier il faut ajouter l'option **-T** et être **root** ! ! !
Voir la structure suivante. Vous ne connaissez pas tout dans cette structure, mais ça va venir.
R1 à R4 sont des routeurs, des machines qui reçoivent des paquets et qui ne regarde que la couche réseau (couche 3 IP) et qui transfèrent les paquets...
PC1 veut pouvoir atteindre PC2 et vice versa, par où passe-t-il ?



Sur PC1 d'@ IP 1.1.1.1, la commande traceroute 1.1.1.18, @ IP de PC2 renvoie :

```
root@PC1:~# traceroute 1.1.1.18
traceroute to 1.1.1.18 (1.1.1.18), 30 hops max, 60 byte packets
 1  1.1.1.2 (1.1.1.2)  1.046 ms  1.009 ms  0.979 ms
```

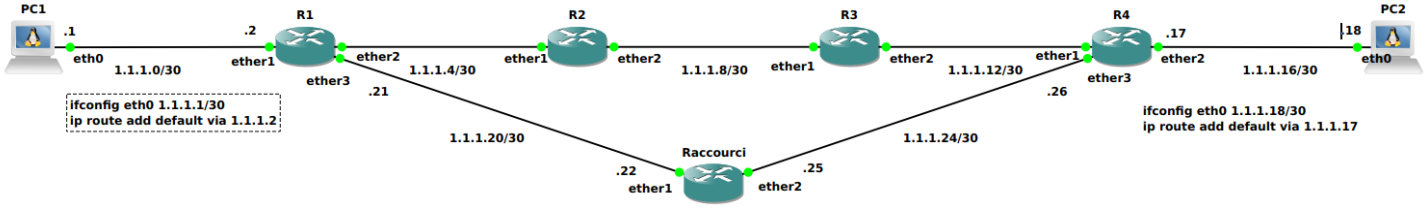
```
 2   1.1.1.6 (1.1.1.6)   3.343 ms   3.341 ms   3.305 ms
 3   1.1.1.10 (1.1.1.10)   5.608 ms   5.613 ms   5.568 ms
 4   1.1.1.14 (1.1.1.14)   6.217 ms   6.752 ms   7.022 ms
 5   1.1.1.18 (1.1.1.18)   10.183 ms   10.165 ms   10.112 ms
```

Et sur PC2 d'@ IP 1.1.1.18, la commande traceroute 1.1.1.1, @ IP de PC21renvoie :

```
root@PC2:~# traceroute 1.1.1.1
traceroute to 1.1.1.1 (1.1.1.1), 30 hops max, 60 byte packets
 1   1.1.1.17 (1.1.1.17)   0.950 ms   0.889 ms   0.848 ms
 2   1.1.1.13 (1.1.1.13)   2.154 ms   2.148 ms   2.127 ms
 3   1.1.1.9 (1.1.1.9)   4.685 ms   4.674 ms   4.643 ms
 4   1.1.1.5 (1.1.1.5)   4.901 ms   4.873 ms   4.844 ms
 5   1.1.1.1 (1.1.1.1)   5.103 ms   5.052 ms   4.993 ms
```

Que remarque t'on ?
Que dans les 2 cas nous avons 5 sauts, en effet nous avons dans cette structure 5 réseaux différents !
Les réseaux IP 1.1.1.0/30, 1.1.1.4/30, 1.1.1.8/30, 1.1.1.12/30 et 1.1.1.16/30
Que le prochain saut est l'@ IP connectée à soi !
— 1.1.1.2 est **connectée** à 1.1.1.1 @ PC1
— 1.1.1.17 est **connectée** à 1.1.1.18 @ PC2

Si on modifie la structure comme suit



Sur PC1 d'@ IP 1.1.1.1, la commande traceroute 1.1.1.18, @ IP de PC2 renvoie :

```
root@PC1:~# traceroute 1.1.1.18
traceroute to 1.1.1.18 (1.1.1.18), 30 hops max, 60 byte packets
 1   1.1.1.2 (1.1.1.2)   9.830 ms   10.240 ms   10.205 ms
 2   1.1.1.22 (1.1.1.22)   34.054 ms   34.020 ms   33.970 ms
 3   1.1.1.26 (1.1.1.26)   45.747 ms   45.698 ms   45.636 ms
 4   1.1.1.18 (1.1.1.18)   48.063 ms   48.013 ms   47.952 ms
```

Et sur PC2 d'@ IP 1.1.1.18, la commande traceroute 1.1.1.1, @ IP de PC21renvoie :

```
root@PC2:~# traceroute 1.1.1.1
traceroute to 1.1.1.1 (1.1.1.1), 30 hops max, 60 byte packets
 1   1.1.1.17 (1.1.1.17)   3.875 ms   4.166 ms   4.117 ms
 2   1.1.1.25 (1.1.1.25)   9.382 ms   9.521 ms   9.470 ms
 3   1.1.1.21 (1.1.1.21)   14.236 ms   14.204 ms   14.138 ms
 4   1.1.1.1 (1.1.1.1)   16.266 ms   16.207 ms   16.133 ms
```

On passe de 5 sauts à 4 sauts, les paquets IP passent par le routeur raccourci.
On verra pourquoi dans le CM/TD/TP

6 Conclusion

IP est à Internet ce que les 06 et 07 sont au smartphone. Sans lui pas d'Internet.
À retenir : **IP**, **ICMP** (ping et traceroute) et **ARP** pour les @MAC vues au cours précédent sur Ethernet et ici.
Savoir ce qu'est un traceroute et à quoi il sert ! ! !