Réseaux Avancés

Myriam BEGEL

Cachan Réseau à Normale Sup'

Mardi 1er décembre 2015

Plan

Encapsuler

"L'encapsulation est un procédé consistant à inclure les données d'un protocole dans un autre protocole."

Généralement, on encapsule dans un protocole de couche supérieure.

Présentation de scapy

scapy permet de forger des paquets en Python "à la main"

Générons un ping

```
monPing= IP()/ICMP()
##Pour definir l'IP source et celle du destinataire :
monPing.src= 'adresse_IP'
monPing.dst= 'adresse_IP'
##Pour voir le paquet genere :
monPing.show()
##Pour l'envoyer :
send(monPing)
##Pour recevoir une reponse :
rep,non_rep = srp1(monPing)
```

Plan

Protocole IP insuffisant : n'indique pas le programme concerné.

On introduit : TCP et UDP, 2 protocoles de couche supérieure qui vont indiquer aux paquets le port associé au paquet.

Ports

Pour identifier l'application : numéro de port de 16 bits.
En vrai, pas de ports, juste des numéros.

En dessous de 1024, ils sont réservés, pour voir la liste
`/etc/services`
Au dessus, ils sont libres.

Adresse de socket = paire (IP,port)
Socket : représente l'extrémité d'une communication à la couche application.

User Data Protocol

Envoi simplement les données d'un couple (IP,port) vers un autre couple (IP,port).
Encapsulé dans un paquet IP

Structure sur 32 bits

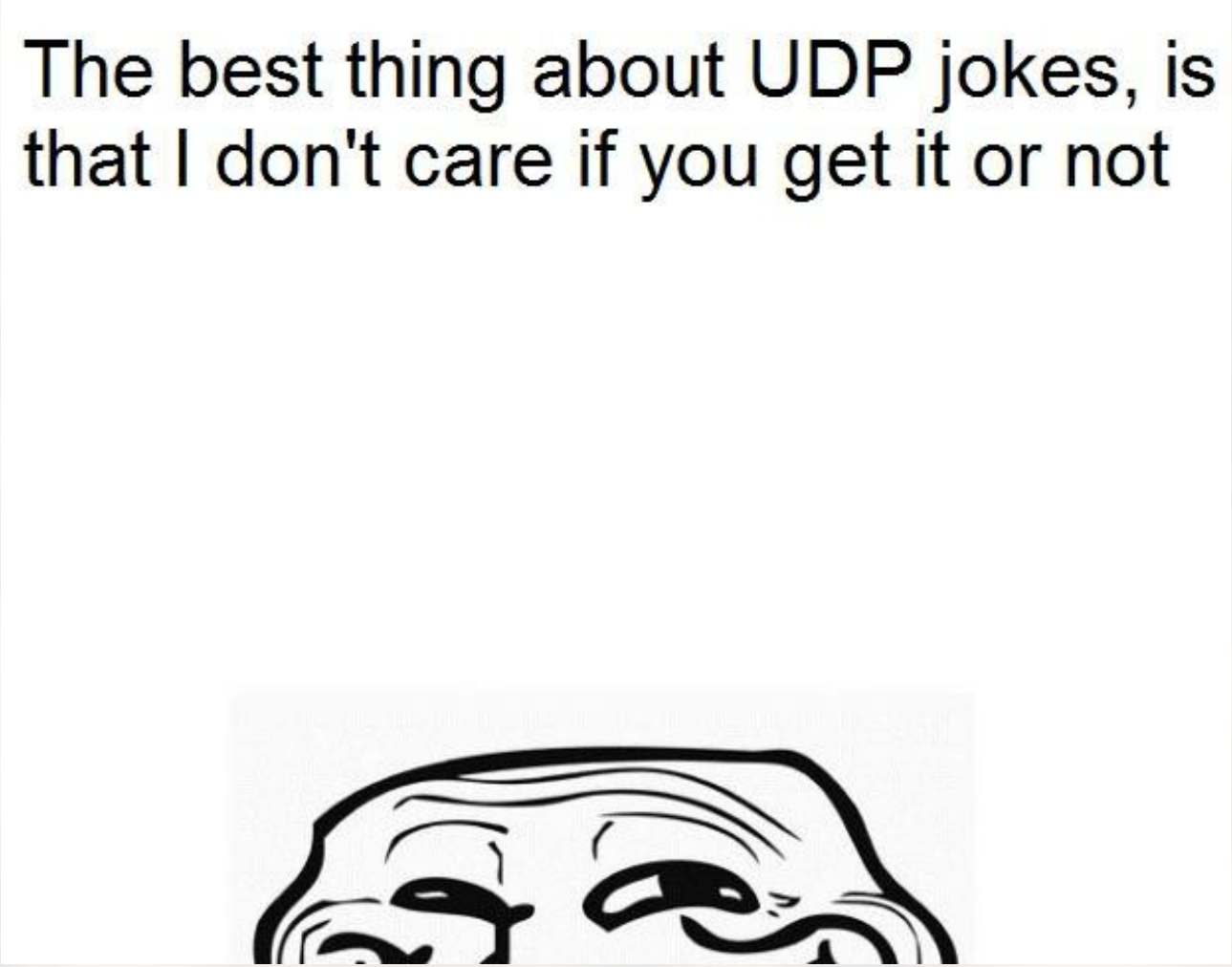
Source Port Number	Destination Port Number
Length(UDP Header + Data)	UDP Checksum
Application Data (Message)	

User Data Protocol

Avantage : rapidité par petites quantités
Inconvénient : ne garantit pas l'ordre ni l'exactitude des données reçues
Utilisation :

- Streaming
- Jeux en lignes
- DNS
- ...

UDP


The best thing about UDP jokes, is that I don't care if you get it or not

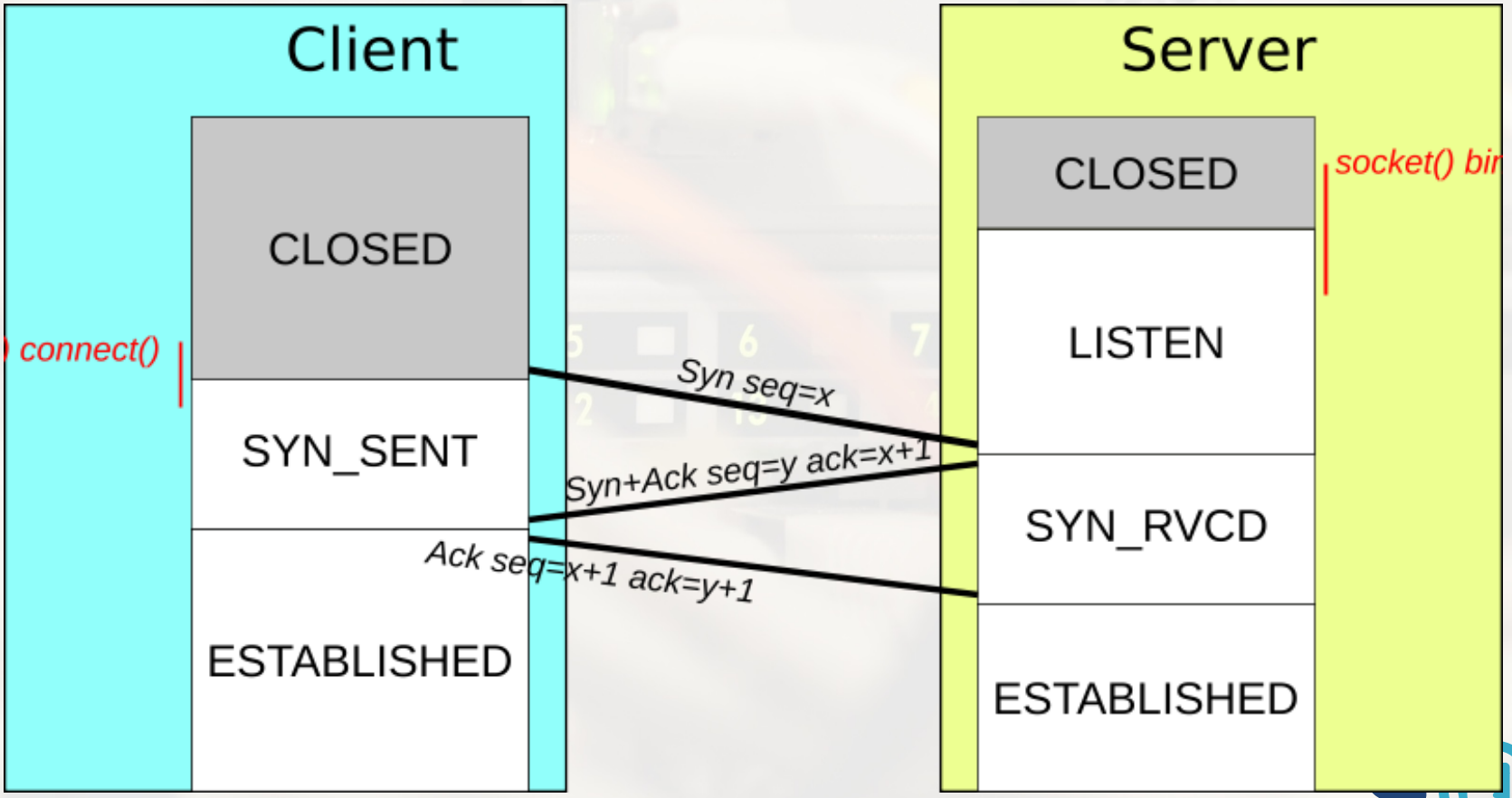
Transmission Control Protocol

Protocole qui assure l'intégrité des données mais plus complexe.

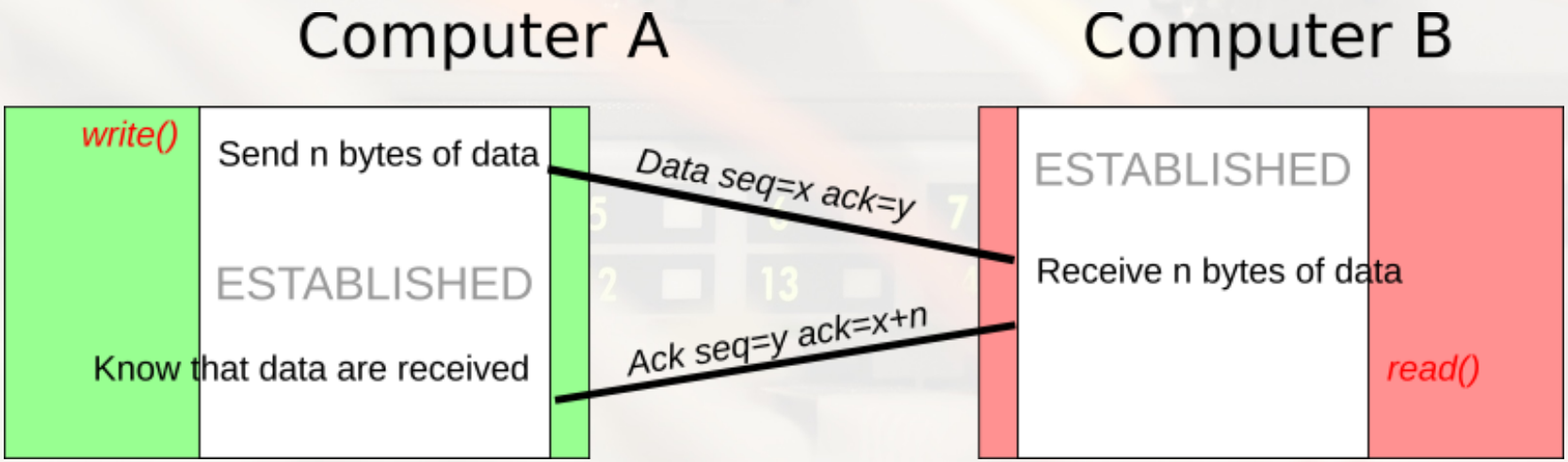
Structure :

16 bits							32 bits
Source Port							Destination Port
Sequence Number							
Acknowledgment Number (ACK)							
Offset Reserved	U	A	P	R	S	F	Window
Checksum							Urgent Pointer
Options and Padding							

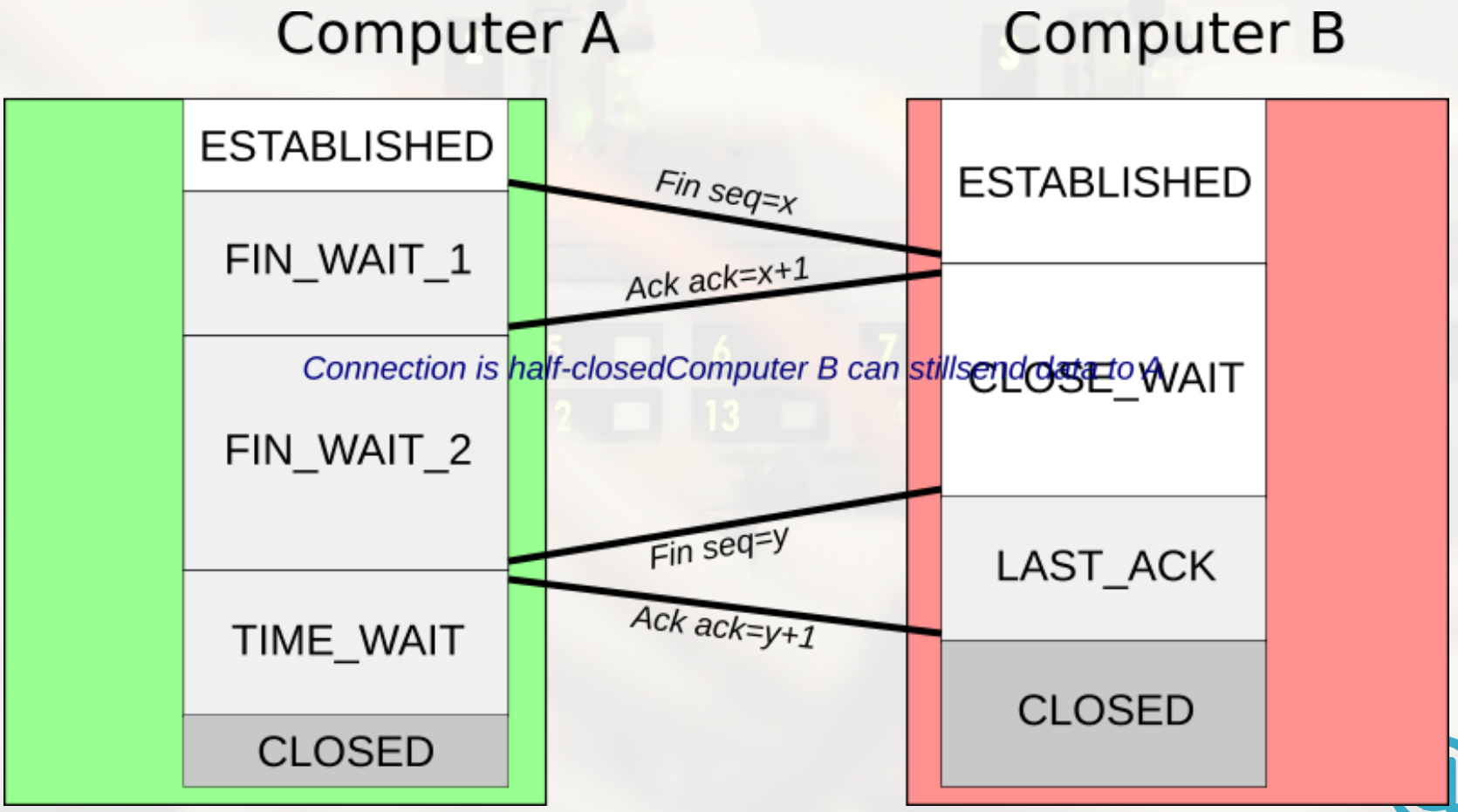
Ouverture de la connexion



Envoi des données



Arrêt de la connexion



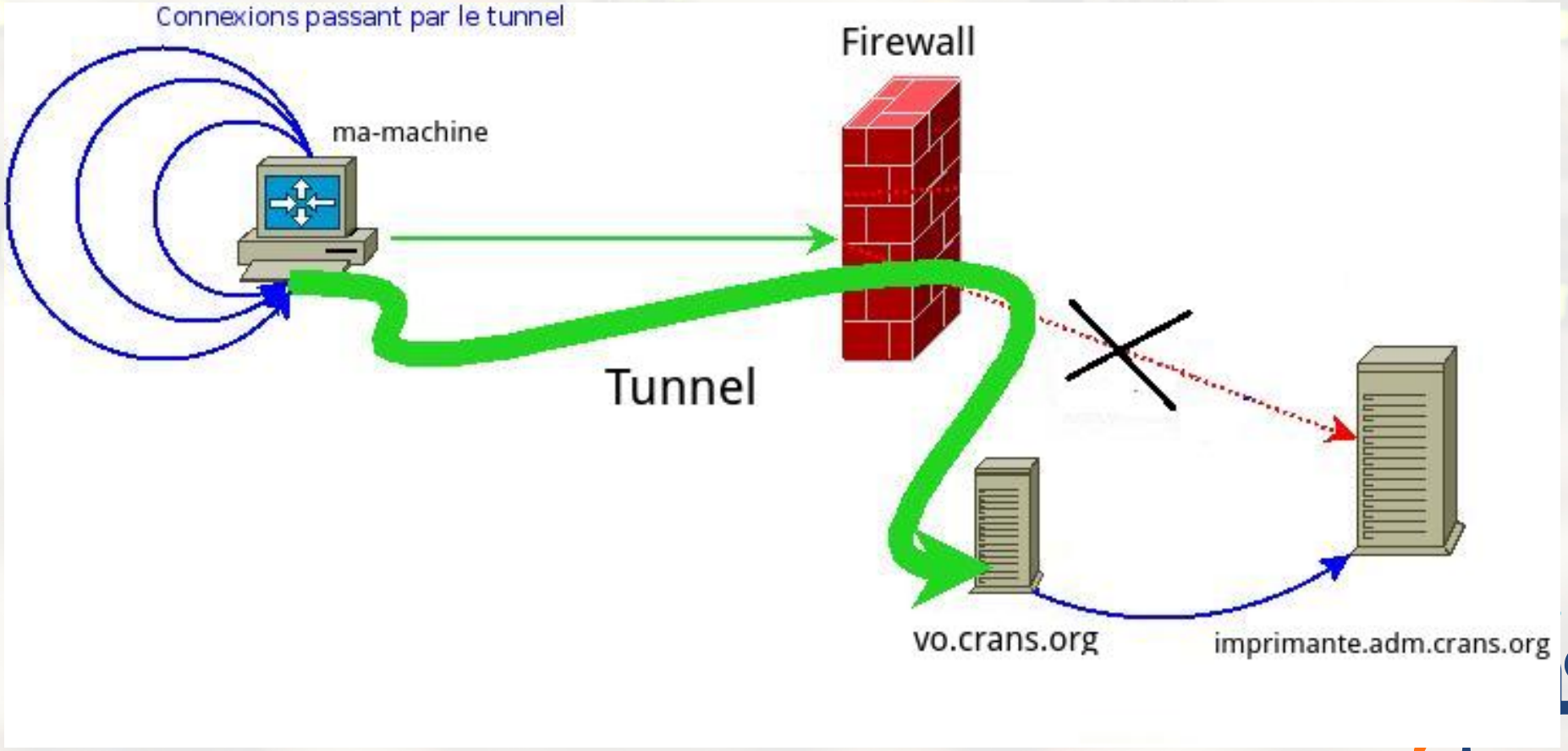
Utilisation de scapy

Ecrivons un paquet TCP :

```
monPaquet= IP()/TCP()
##Pour definir l'IP source et celle du destinataire :
monPaquet.src= 'adresse_IP'
monPaquet.dst= 'adresse_IP'
##Pour voir le paquet generer :
monPaquet.show()
##On peut modifier de meme les valeurs de seq et ack
##Pour recevoir une reponse :
rep , non_rep = srp1(monPaquet)
```

Plan

"Un tunnel, est une encapsulation de données d'un protocole réseau dans un autre, situé dans la même couche, ou dans une couche de niveau supérieur."



Intérêts d'un tunnel :

- Contourner un pare-feu
- Chiffrer ses données pour les isoler du reste du réseau

proxy SOCKS

Mise en place d'un tunnel :

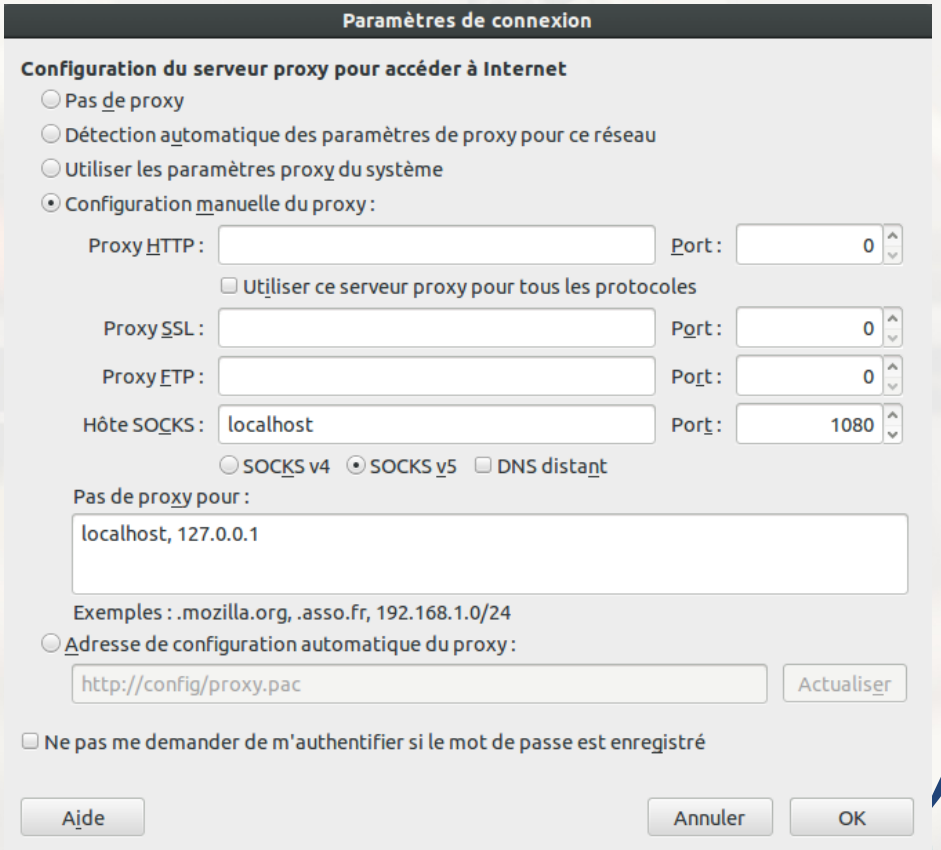
```
ssh -D port user@adress
```

par exemple

```
ssh -D 1080
begel@zamok.crans.org
```

Configurez ensuite le proxy dans votre navigateur.

Testez votre IP sur monip.fr !



Plan

Virtual Local Area Network

Normalement, chaque réseau a ses propres switch ce qui peut être coûteux et contraignant.

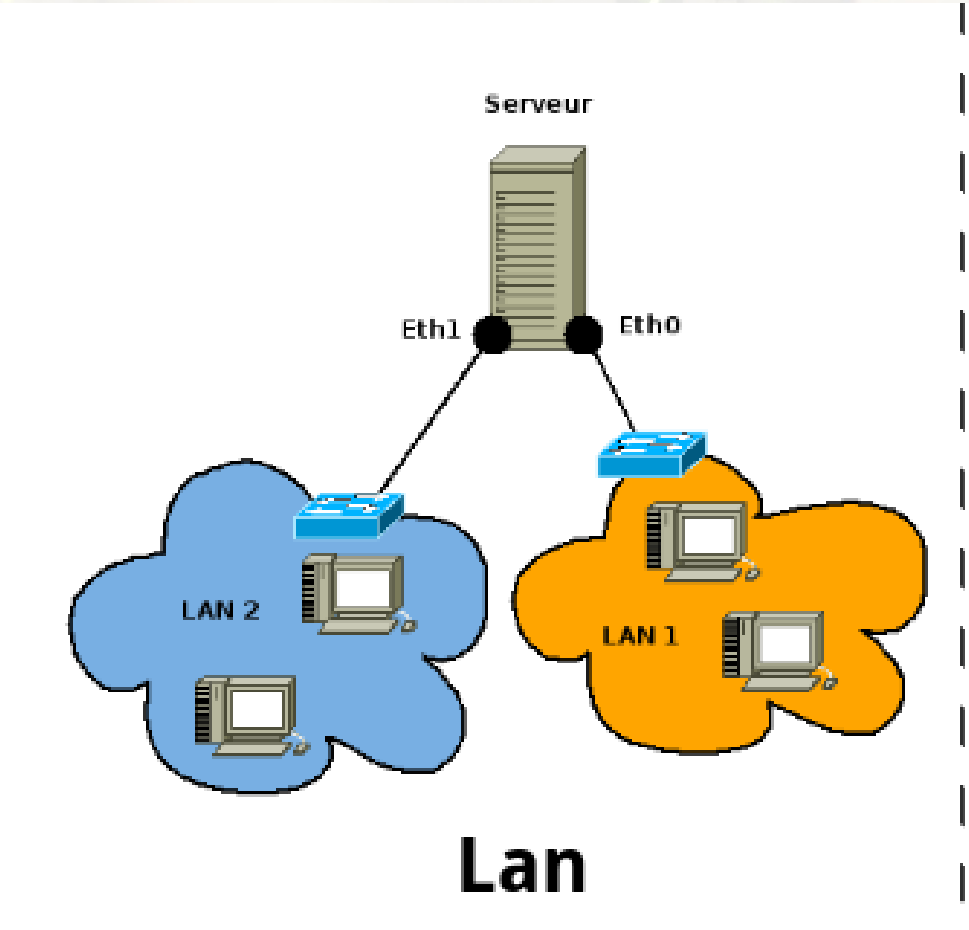
Intérêts des VLan's

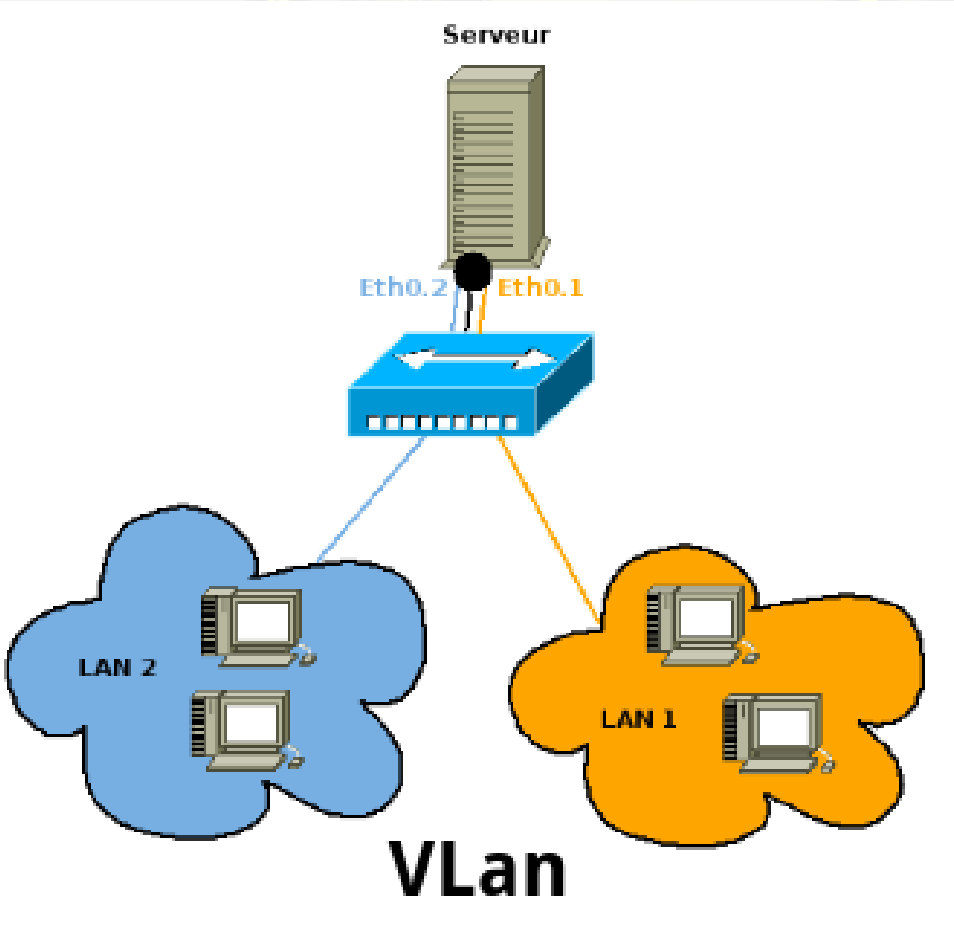
- Pouvoir séparer des flux dans un même switch
- Pouvoir changer la distribution des flux à distance

Au cr@ns, par exemple, on distingue :

- VLan Adm (administrateur)
- VLan Adhérents

VLan


Serveur
Eth1
Eth0
LAN 2
LAN 1
Lan


Serveur
Eth0.2
Eth0.1
LAN 2
LAN 1
VLan

Comment répartit-on les VLan sur les ports du switch ?

Différents niveaux :

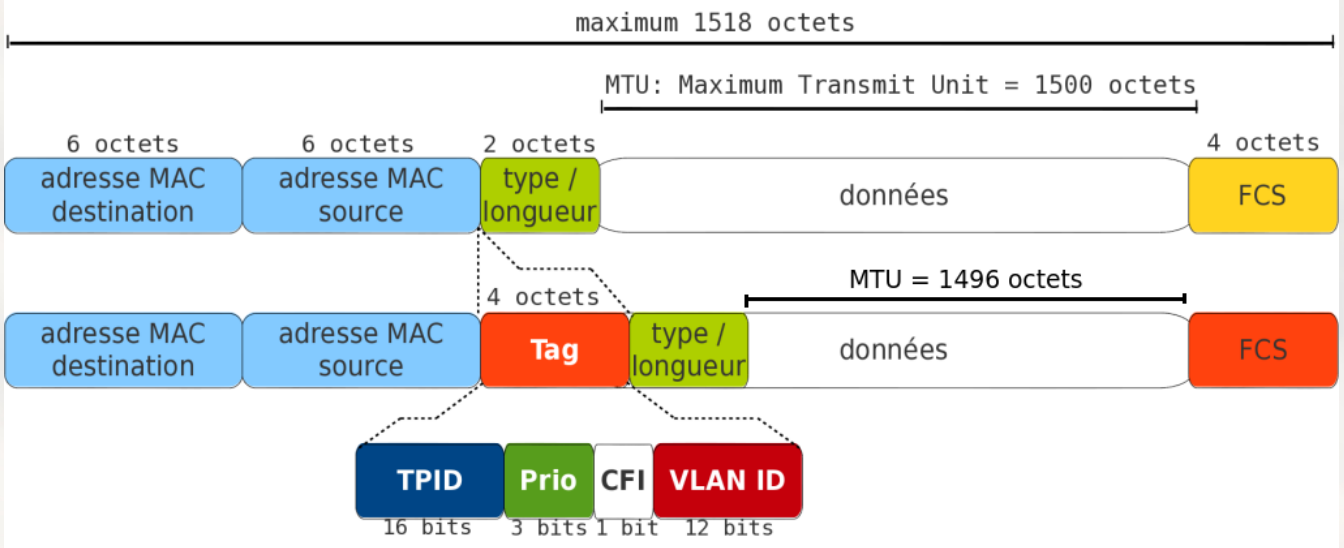
1. VLan par ports : on associe à chaque VLan les ports du switch qui lui sont rattachés
2. VLan par MAC : de même avec les adresses MAC
3. VLan par IP : de même avec les IP

Tagging

Normalement, un seul VLan peut sortir par port.
Pour remédier à cela, on peut encapsuler (norme 802.1q) le numéro du VLan dans chaque trame diffusée.
C'est le **tagging**.

Tagging

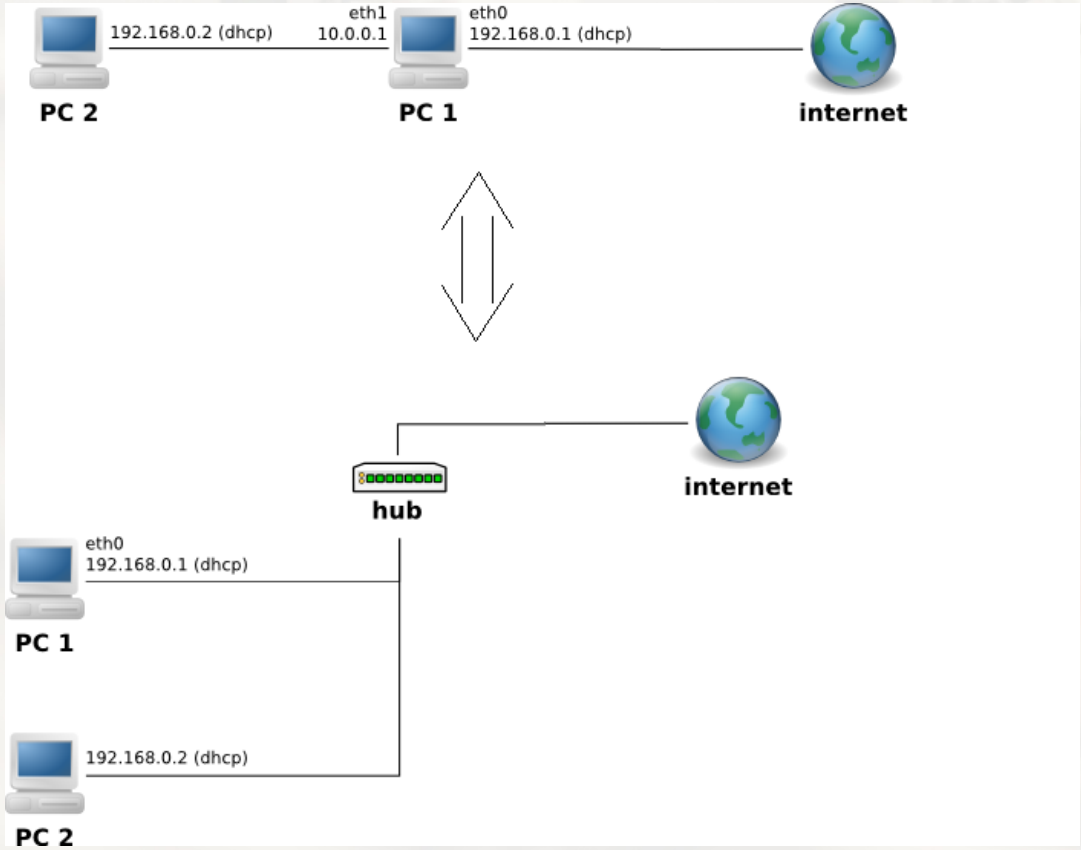
Transformation de la trame ethernet :



On remarquera bien que la MTU est diminuée de 4 octets.

Plan

Un bridge est un pont ethernet. Une machine avec 2 cartes ethernet peut servir de switch avec un bridge.



Très utile pour connecter une machine virtuelle à Internet.

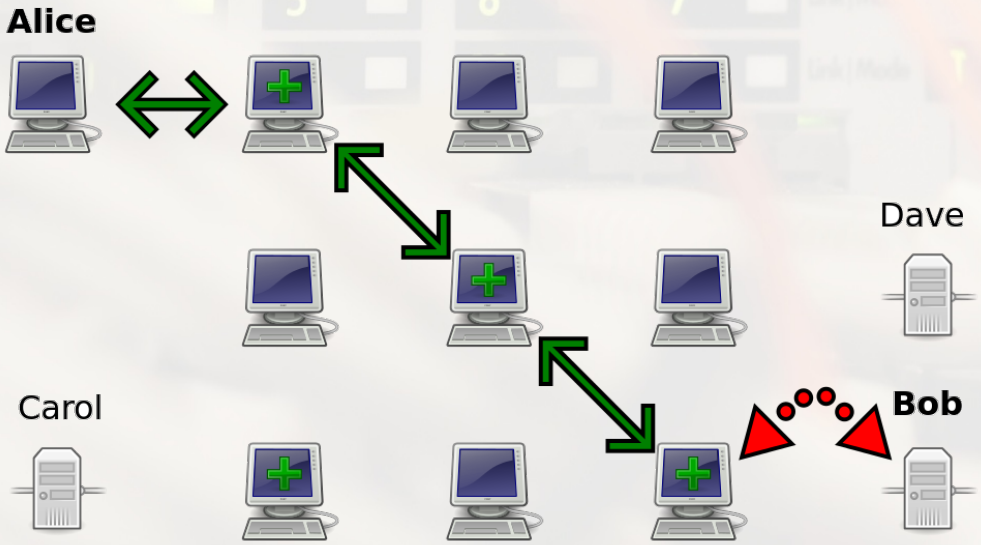
3 étapes :

1. Listening : écoute tout ce qui se passe sur le réseau
2. Learning : déduit la configuration du réseau
3. Forwarding : dispatche correctement les paquets

Plan

The Onion Router

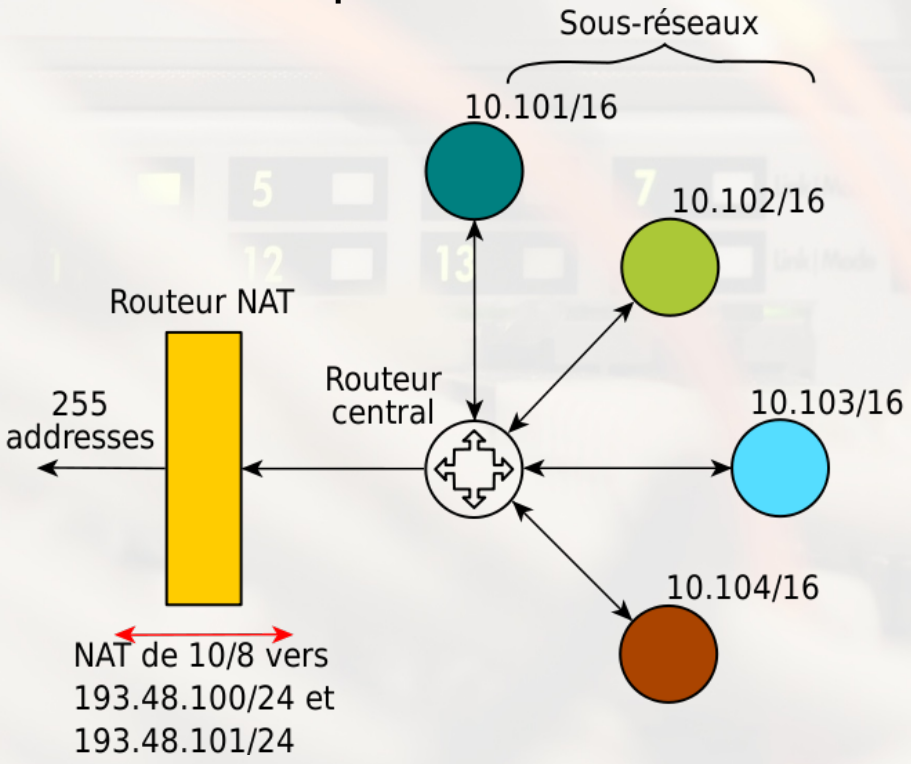
TOR est un ensemble de routeurs organisé en couches qui empêchent théoriquement l'analyse du trafic d'un utilisateur.
Pour utiliser TOR, il suffit de rediriger votre trafic par le port dédié (par défaut 9050).
https ://www.torproject.org/



Plan

Network Address Translation

Pour palier au manque d'adresse IPv4, on peut utiliser un NAT :
Un routeur va avoir des adresses IP globales et les machines à l'intérieur uniquement des IP locales.



NAT en pratique

Quand un paquet sort, le routeur lui change son adresse de socket avec son IP.
Il enregistre l'association entre la socket locale et globale.
Quand un paquet arrive, il l'ouvre, regarde à quel port il est adressé et le redistribue correctement.

NAT c'est mal

- des tas de gens n'ont pas d'IP routables
- tellement de gens qu'on peut pas développer de nouveaux protocoles
- le routeur ouvre les paquets et même les modifie

D'où l'IPv6 !

Questions ?