



**Réseau sous GNU/Linux**

**« *Connectez vos passions* »**

Jeudi 27 Janvier – Thomas Petazzoni



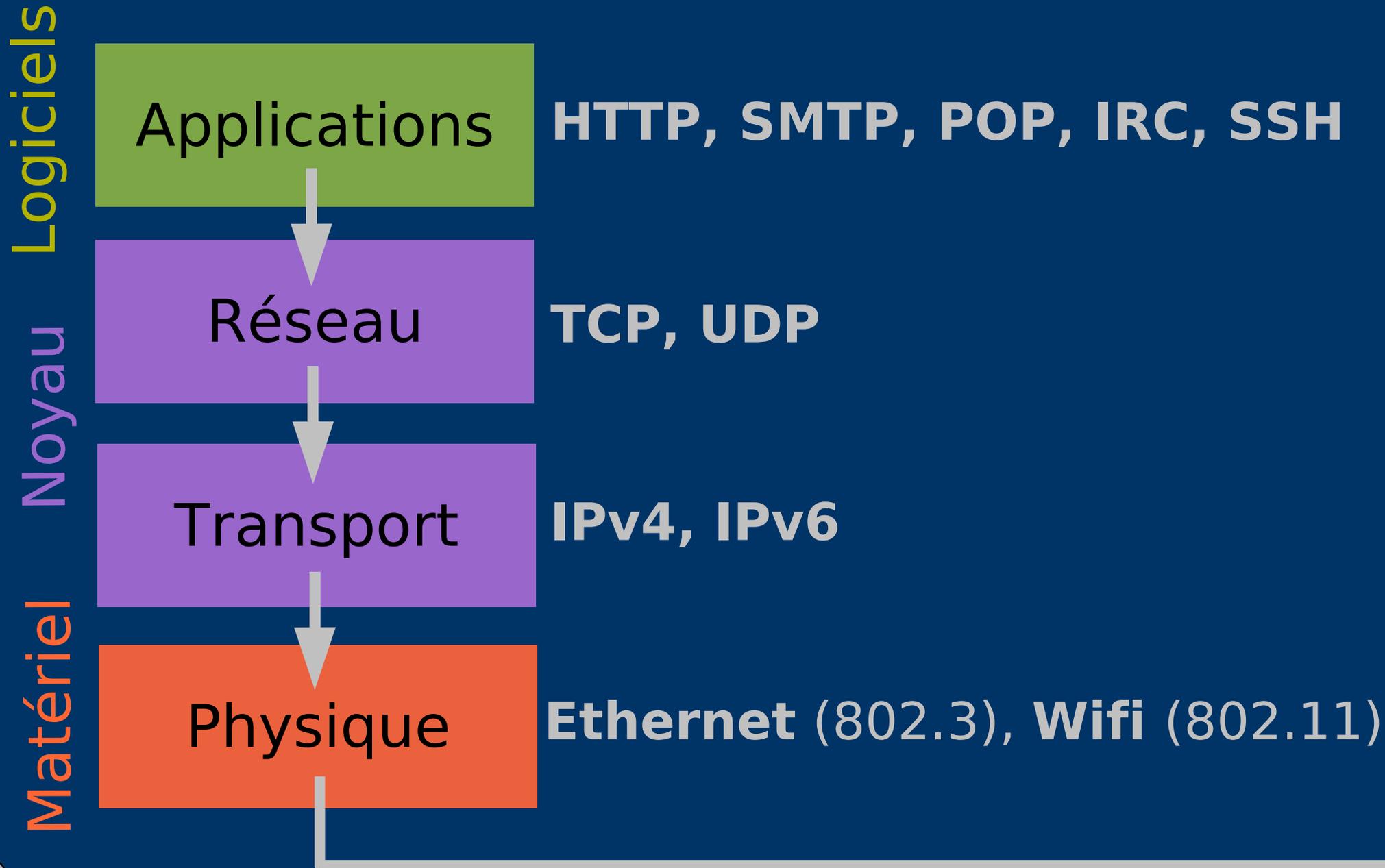
# Unité de transmission

En réseau, l'unité de transmission est le **paquet**.

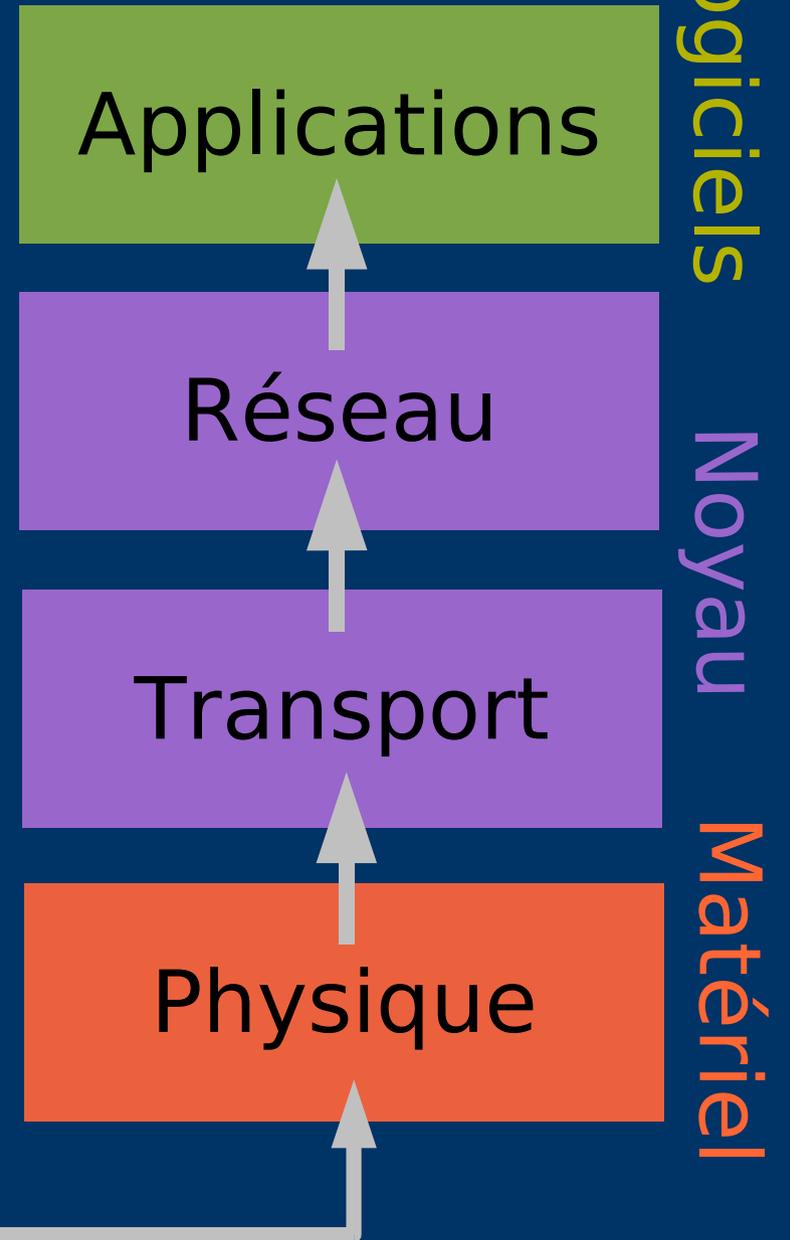
```
0x0000:  00e0 4cec 59d7 00e0 4cec 323b 0800 4500  ..L.Y...L.2;..E.
0x0010:  0050 dbef 4000 4006 7463 c0a8 0102 5138  .P..@.@.tc...Q8
0x0020:  d77d 8014 1a0b 45ac 077e f705 4c60 8018  .}....E..~..L`..
0x0030:  3e96 eaa2 0000 0101 080a 0020 1854 08d2  >.....T..
0x0040:  5830 5052 4956 4d53 4720 236b 6f73 6465  X0PRIVMSG.#kosde
0x0050:  7620 3a62 6f6e 6a6f 7572 2021 0d0a      v.:bonjour.!..
```



# Modèle en couche



# Modèle en couche



# Encapsulation

Sur la source, descente dans les couches : **encapsulation**

Sur la destination : remontée dans les couches : **désencapsulation**

Ethernet  
IP  
TCP  
HTTP

```
0000  00 e0 4c ec 59 d7 00 e0  4c ec 32 3b 08 00 45 10  ..L.Y... L.2;..E.
0010  00 3b 84 4b 40 00 40 06  33 0e c0 a8 01 02 c0 a8  .;.K@.@. 3.....
0020  01 01 80 90 00 50 38 84  c1 c6 32 05 e1 35 80 18  .....P8. ..2..5..
0030  05 b4 83 81 00 00 01 01  08 0a 00 3b 4f 9e 00 3b  ..... ..;0..;
0040  e0 12 47 45 54 20 2f 0d  0a  ..GET /. .
```



- Adresse **unique** pour chaque interface Ethernet (carte réseau)
- Fixée par le constructeur du matériel
- Difficilement modifiable
- Utilisée au niveau physique (Ethernet)
- 6 octets
- MAC = Medium Access Control

```
# ifconfig eth0  
eth0      Lien encap:Ethernet  HWaddr 00:E0:4C:EC:32:3B
```



- Utilisées pour structurer un réseau de manière logique
- Une ou plusieurs par interface réseau
- Sur 4 octets
- Utilisées au niveau du protocole IP (couche «Transport»)
- IP = Internet Protocol

```
# ifconfig eth0
eth0      Lien encap:Ethernet  HWaddr 00:E0:4C:EC:32:3B
          inet adr:192.168.1.2  Bcast:192.168.1.255  Masque:255.255.255.0
```

192 = 0xC0, 168 = 0xA8

192.168.1.2 = 0xC0A80102



Trouver la MAC à partir d'une IP : protocole  
**ARP**

ARP = Address Resolution Protocol

Message

« Qui possède l'IP w.x.y.z ? »

envoyé à tout le réseau. L'intéressé répond

« C'est moi et je suis aa.bb.cc.dd.ee.ff ! »



# Masque de sous-réseau

Associé à chaque IP, un **masque de sous-réseau** :

- forme : 255.255.255.0
- détermine la « portée » du réseau accessible directement par l'interface
- fonctionne par **ET** logique
- Ex:  $(192.168.1.2 \ \& \ 255.255.255.0) = 192.168.1.0$
- $255.255.255.0 \ \Leftrightarrow \ /24$
- $255.255.0.0 \ \Leftrightarrow \ /16$
- $255.0.0.0 \ \Leftrightarrow \ /8$



# Adresses IP spéciales

- 127.0.0.1 : Adresse IP «locale»
- 192.168.1.255 : Adresse de diffusion ou «broadcast»
- 192.168.1.0 : Adresse «réseau»

```
# ifconfig lo
lo          Lien encap:Boucle locale
           inet adr:127.0.0.1  Masque:255.0.0.0
           UP LOOPBACK RUNNING

# ifconfig eth0
eth0       Lien encap:Ethernet  HWaddr 00:E0:4C:EC:32:3B
           inet adr:192.168.1.2  Bcast:192.168.1.255  Masque:255.255.255.0
```



# Adresses IP et Internet

- Adresses IP sur 4 octets :  $2^{32}$  adresses disponibles
- Adresses uniques sur tout Internet
- Sauf classes réservées pour réseaux locaux
  - 10.0.0.1 à 10.255.255.254
  - 172.16.0.1 à 172.31.255.254
  - 192.168.0.1 à 192.168.255.254
- Nombre limité d'adresses IPs : utilisation de passerelles



192.168.1.3



192.168.1.1



81.56.146.155



192.168.1.2



# Deux méthodes

# Configuration IP

- statique
- dynamique : DHCP ou BOOTP

## Sous GNU/Linux

- directement avec la commande `ifconfig` pour du statique
  - `ifconfig eth0 192.168.1.32`
- directement avec un client DHCP pour du DHCP
  - `dhclient eth0`
  - `pump -i eth0`
- Debian : dans le fichier `/etc/network/interfaces`

```
# /etc/network/interfaces

auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
        address 192.168.1.1
        netmask
255.255.255.0

auto eth1
iface eth1 inet dhcp
```



Chaque ordinateur possède une **table de routage** qui explique comment gérer les paquets en fonction de leur destination.

```
# route -n
Table de routage IP du noyau
Destination      Passerelle      Genmask          Indic  Metric  Ref    Use  Iface
192.168.1.0      0.0.0.0         255.255.255.0   U      0        0      0   eth0
0.0.0.0          192.168.1.1    0.0.0.0         UG     0        0      0   eth0
```



# *TCP et UDP, notion de port*

Deux principaux protocoles au dessus d'IP

- **TCP**, protocole «connecté»  
Transmission Control Protocol
- **UDP**, protocole «non connecté»  
User Datagram Protocol

Organisation des flux de paquets par «tuyaux»  
logiques : les ports

65 536 ports disponibles



Dans tout transfert de paquet TCP ou UDP, il y a :

- un logiciel « serveur » qui attend des paquets sur un **port** donné
- un logiciel « client » qui envoie des paquets sur ce port à travers le réseau, s'adressant ainsi au logiciel « serveur »

```
22:21:01.331989 192.168.1.2:33088 > 192.168.1.1:80: P 1:8(7) ack 1 win
1460 <nop,nop,timestamp 11418975 11457301> (DF) [tos 0x10]
0x0000  4510 003b 0830 4000 4006 af29 c0a8 0102      E..;.0@.@..)....
0x0010  c0a8 0101 8140 0050 1346 312b 0ca7 bde7      .....@.P.F1+....
0x0020  8018 05b4 76d2 0000 0101 080a 00ae 3d5f      .....v.....=_
0x0030  00ae d315 4745 5420 2f0d 0a          ....GET./..
```



# *Ports : conventions*

Existence de conventions pour les numéros de ports :

- http : 80 (tcp)
- https : 443 (tcp)
- pop : 110 (tcp)
- smtp : 25 (tcp)
- telnet : 23 (tcp)
- dns : 53 (udp)
- tftp : 69 (udp)

Voir une liste plus complète dans `/etc/services`



# Serveurs : processus et inetd

## Logiciel « serveur »

- processus classique qui attend des connexions sur un **port** précis
- logiciels en écoute listables par *netstat --ip -l*
- lancé :
  - soit de manière autonome
    - le logiciel écoute lui-même sur le port
    - exemple : apache, ssh, proftpd ...
  - soit par le super-démon **inetd**
    - **inetd** écoute sur tous les ports et transmet au logiciel si nécessaire
    - exemple : echo, time, exim, apache ...



# Résolution de noms

Retenir les adresses IP : Argh ;-(

=> Faire correspondre des noms aux adresses

Deux solutions :

- Fichier /etc/hosts

- Comment synchroniser tout ça ?

- Protocole DNS

- Système hiérarchisé de stockage de correspondances noms/adresses IP
- Interrogé automatiquement par les applications
- Configuration dans /etc/resolv.conf
- Serveur DNS : Bind
- Client DNS : host



- Installation d'un serveur genre Apache ?
- Explications sur Netfilter : NAT et suivi de connexion ?
- Les entrées de DNS : CNAME, A et MX ?
- ?

