Réseau - Modèle OSI

Difficulté : Débutant

Notions : réseau, modèle OSI

I. Introduction

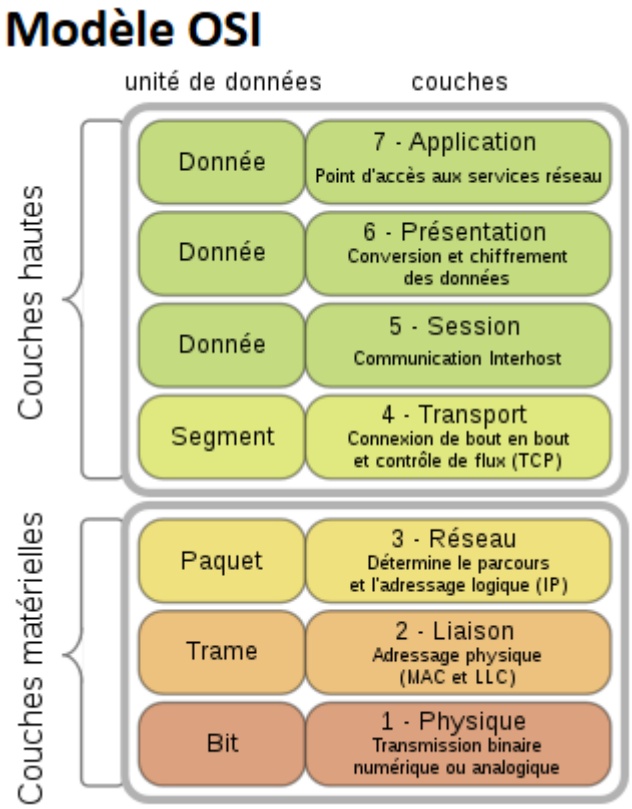
Le modèle OSI (**O**pen **S**ystems **I**nterconnection) est un modèle conceptuel en couche développé par l'**I**nternational **S**tandardisation **O**rganisation (ISO).

Il permet de conceptualiser les communications réseaux entre les ordinateurs en divisant le processus de communication en **7 couches distinctes**.

Chaque couche à un rôle spécifique et interagis avec les couches directement situées en dessous et au dessus d'elle.

Cette approche, en plus de visualiser le fonctionnement des communication réseau en permet un diagnostic efficace.

Voici la représentation des différentes couches.



Lors des communications entre machines, celles-ci seront ***encapsulées*** les unes dans les autres donnant alors ce que l'on appellera une '***trame***' réseau.

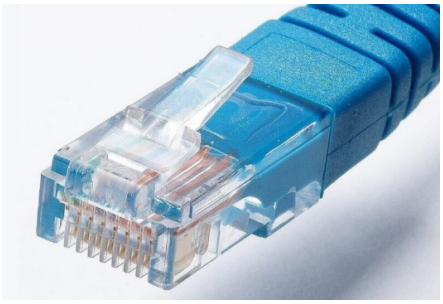
II. Les couches du modèle OSI

2.1 Couche 1 : Physique

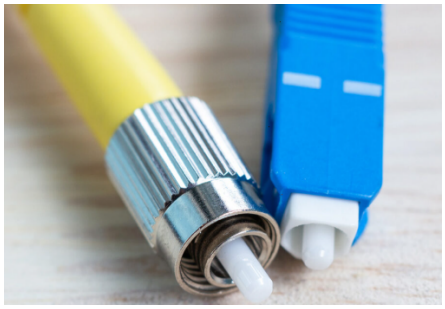
Il s'agit de la couche matérielle. Celle permettant de porter le signal réseau.

Elle est constituée des câbles réseaux, fibres optiques, ondes WIFI et des contacts électroniques des cartes réseaux.

Il s'agit de ce que l'on appelle le '***medium de propagation du signal***'



câble RJ45



connecteurs fibre

antenne wifi

Outils de diagnostics

- vérification du câble ou fibre
- état des voyants sur l'interface
- état de la carte : 'link up / link down'

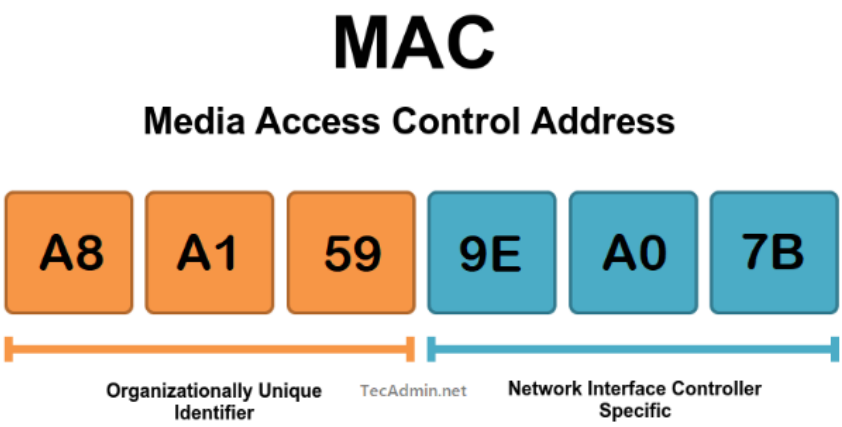
2.2 Couche 2 : Liaison

Il s'agit de la couche permettant aux machines de communiquer entre elles au plus bas niveau et de se transmettre le signal d'un nœud à un autre. En d'autre terme, d'établir une '***liaison***' entre elles.

On y retrouve notamment les adresses MAC (**M**edia **A**ccess **C**ontrol).

Une adresse MAC est un identifiant physique unique permettant d'identifier l'interface réseau au plus bas niveau.

Les plages d'adresses MAC sont distribuées aux constructeurs de matériel réseau et chaque interface matérielle est dotée de sa propre adresse MAC unique au monde.



Les protocoles utilisés à ce niveau sont :

- Ethernet
- PPP
- ...

C'est à ce niveau que va s'opérer le contrôle d'erreur (CRC) et la commutation par les switches. La gestion des VLAN, etc...

Outils de diagnostics

Vérification des tables ARP

par exemple :

```
C:\Users\ochabran>arp -A

Interface: 192.168.41.2 --- 0x17
  Internet Address      Physical Address      Type
  192.168.41.254        e4-3a-6e-5e-80-e1     dynamic
  192.168.41.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

2.3 Couche 3 : Réseau

La couche réseau est la plus connue et manipulée directement par les administrateurs réseaux et systèmes.

Il s'agit de la couche qui permet de faire abstraction de la couche matérielle afin de construire des topologies réseau logiques.

Celles-ci pourront de ce fait être cartographiées et des '***routes***' permettrons d'interconnecter ces topologies entres elles.

C'est à ce niveau là qu'interviens le protocole ***IP*** et que l'on retrouve nos adresses logiques uniques sur le réseau.

- Réseau - Adressage IPV4
- Réseau - Adressage IPv6

A ce niveau, l'on parle de ***trames*** ip. Ces trames ont une taille limite de 65536 Octets. Mais celle-ci ne sera que rarement atteinte, car elle va être limitée par la taille maximale que les réseaux qui la portent peuvent supporter.

Cela se négocie grâce au ***MTU*** (**M**aximum **T**ransfer **U**nit). Passé cette taille, les trames seront fragmentées.

Cette fragmentation est gérée par le routeur.

> **Attention** : Rien ne garantis à ce stade que les paquets arriveront dans le bon ordre. Cela sera géré sur la couche suivante.

Voici les en-têtes présents sur cette couche :

en-tête	Description
TTL (**T**ime **T**o **L**ive)	ce champs permet de définir une expiration sur les paquets afin que ceux-ci puissent "s'autodétruire" si leur durée de vie est expirée. Empêchant ainsi une saturation du réseau.
SRC (Source address)	Adresse IP de la source de la transmission.
DST (Destination address)	Adresse IP de la destination de la transmission.

Checksum	Somme de contrôle IP.

C'est sur cette couche que sont géré entre autre le routage, le diagnostic et les tests de connectivité (avec ICMP).

Outils de diagnostics

- ping
  - **general failure** : problème sur la pile TCP ou pas de signal
    - *vérifier câble et configuration IP*
  - **host unreachable** : le ping n'est pas parvenu à l'hôte de destination
    - vérifier l'état de la passerelle, les configuration IP et le routage
  - **Request timed out** : l'hôte de destination a bien reçu la requête mais n'a pas renvoyé de réponse.
    - pare-feu ? route retour ?
  - **TTL expired in transit** : le paquet a dépassé le nombre de saut maximum (sans doute une boucle de routage).
  - **Anwer from <host>** : OK

- tracert / traceroute : permet de voir les sauts pour vérifier la route empruntée

2.4 Couche 4 : Transport

La couche 4 sert à faire passer des communication à travers le réseau IP.

On y retrouve principalement 2 protocoles de transfert :

- TCP (**T**ransmission **C**ontrol **P**rotocol) : un protocole permettant un transfert de donnée contrôlé avec une gestion des erreurs.
- UDP (**U**ser **D**atagram **P**rotocol) : un protocole sans contrôle d'erreur, mais permettant des interactions plus rapide.

La différence majeure entre les deux tient au fait que TCP gère la transmission, là ou UDP se contente d'envoyer des paquets sans aucun contrôle de la transmission.

2.4.1 TCP

Un paquet TCP est appelé un ***Segment***.

TCP prends en charge la gestion des erreurs. A chaque trame transmise, des informations de contrôle y sont adjointes.

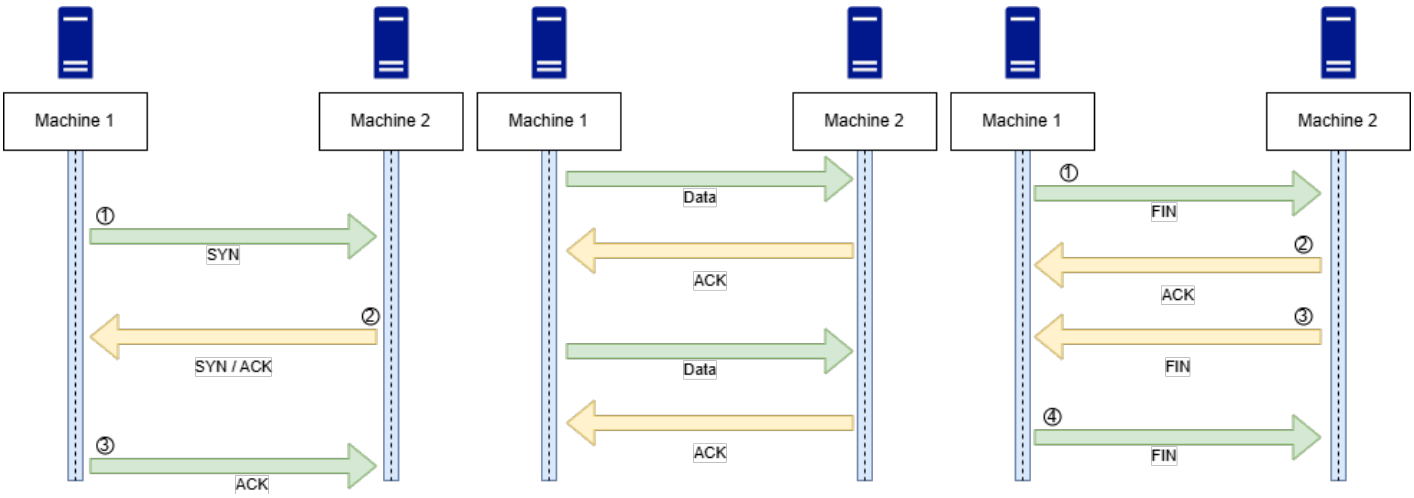
Ces informations permettent ainsi au récepteur de pouvoir s'assurer du numéro de paquet dans la transmission, du fait que le paquet soit complet et reçu sans erreur, de gérer les erreur éventuelles en demandant un nouvel envoi de paquet et enfin de savoir quand la transmission est complétée.

> *Cela le rends donc idéal pour tous les protocoles nécessitant que la donnée soit reçue de façon certaine. Par exemple pour le FTP, SFTP, SMB, HTTP, etc...*

Un échange TCP se déroule de la façon suivante :

Etape	Message	Description
1	SYN	Un message SYN est envoyé par la machine 1, dans le cadre d'un processus appelé '***handshake***'. Ce paquet est issu afin d'initier la connexion et synchroniser les deux machines.
2	SYN / ACK	Ce paquet est envoyé par la machine 2 après réception du paquet de la machine 1 permettant de valider la réception de la demande d'échange.
3	ACK	Ce message sera ensuite envoyé après chaque transmission, afin d'en valider la réception.
4	DATA	Une fois la connexion établie, les données sont transmises à travers des messages data, qui seront 'acknowledged'.
5	FIN	A la fin de la transmission, ce paquet est utilisé pour clôre proprement la session.

#	RST	Ce paquet est utilisé en lieu et place du paquet FIN si il y a eu un problème durant la transmission. Celui-ci terminera alors de façon abrupte la communication.



Ainsi, les données importantes présentes dans les trames TCP Sont :

en-tête	Description
SRC (Source address)	Adresse IP de la source de la transmission.
DST (Destination address)	Adresse IP de la destination de la transmission.
SRC port	Le port source de la transmission.
DST port	Le port de destination de la transmission.
Sequence Number	Lors de l'établissement de la connexion, le premier paquet reçois un numéro initial aléatoire. il constituera le premier numéro de la séquence à transmettre.
ACK Number	Après l'envoi d'un bloc de donnée ayant reçu un Sequence Number, le nombre du prochain bloc reçois le nombre de séquence + 1 et ainsi de suite.
Checksum	Somme de contrôle, pour vérifier l'intégrité du paquet.
Data	Là où se situe la donnée envoyée dans la trame.
Flag	Ce champ détermine comment la trame doit être traitée par les deux machines durant le processus de Handshake. (SYN / ACK / FIN / RST)

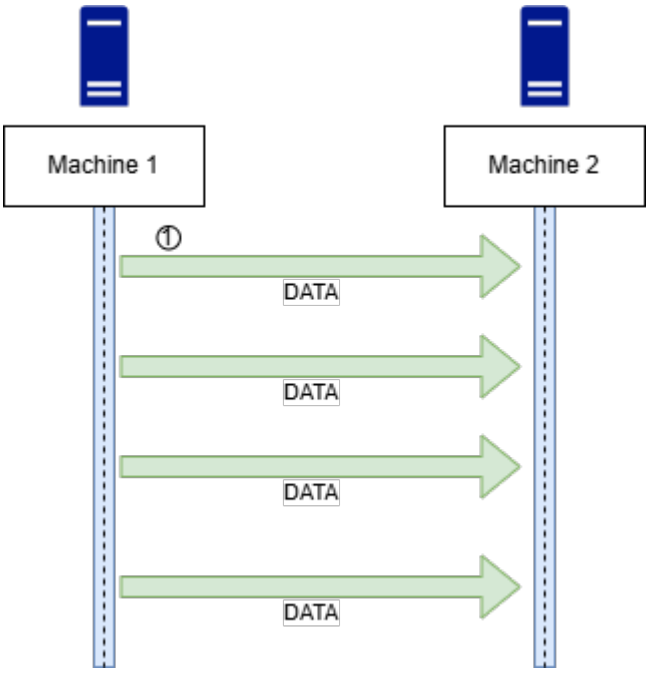
Cela permet de s'assurer que la transmission se déroule bien, de gérer les erreurs.

2.4.1 UDP

Un paquet UDP est appelé un ***Datagrame***.

Contrairement à TCP, le protocole UDP ne prends pas en charge de contrôle de transmission ou d'erreur. Ce qui rends la perte de paquet irrémédiable.

Mais cette absence de contrôle en fait également son point fort. En effet, du fait de sa simplicité, il est très utile pour gérer des flux continus.



> *Cela rends UDP parfait pour des transmissions continues et massive, comme par exemple des flux vidéos, VoIP ou autres...*

Outils de diagnostics

Outils	DIgnostics
netstat	Ports, sessions, TCP states
ss	Sockets, queues, states
tcpdump	Flags, handshake, resets
Wireshark	analyse complète
lsof	Process ↔ port mapping
telnet / nc	Disponibilité du port
curl / wget	TCP errors, resets
(PowerShell) test-netconnection	Windows TCP diagnostics
iperf	Throughput, retransmissions
ssldump	TLS handshake issues

2.5 Couche 5 : Session

La couche 5 permet le contrôle des '***sessions***' de communications entre les applications.

C'est ici que les échanges vont être commencé, suivis, terminés entre les interlocuteurs.

C'est également ici que les erreurs de communications, les mises en attentes de paquets vont être gérées.

2.6 Couche 6 : Présentation

La couche 6 permet la mise en forme préalable des données de la couche suivante, soit sa '***présentation***'.

C'est sur cette couche que ce fait par exemple le chiffrement du flux de données avant l'envoi de la trame (SSL / TLS).

C'est donc aussi sur cette couche que s'opèrera le déchiffrement.

Si des algorithmes de compression de flux sont mis en œuvre, ils se reposeront aussi sur cette couche pour la compression et décompression des flux.

2.7 Couche 7 : Application

C'est enfin sur cette couche que passe la communication des différents services et application.

C'est là que se retrouvent l'ensemble des protocoles liés aux applications comme le DNS, DHCP, NTP, HTTP, etc...

Outils de diagnostics

Côté client, tester la connexion TCP avec telnet, netcat ou autre.

```
Telnet
```

III. Conclusion

Lors d'un diagnostic sur un problème réseau, il est important d'avoir ce modèle en tête et de réfléchir au problème couche par couche en partant de la plus basse.

Cela permet de gagner du temps sur la résolution d'un problème.

A moins bien sûr que la couche qui pose problème soit clairement identifiée au départ par le retour obtenu ou le message d'erreur.

Basiquement, pour une résolution de problème sur la connectivité des VM, les questions à se poser sont :

1. La carte réseau existe et est connectée
2. Je suis situé sur le bon Vswitch (VMBR)
3. Je suis dans la bonne plage IP des deux côté
4. Les pare-feu / routeurs sont bien configurés

Si le signal est OK et que le ping passe,

1. Le service est démarré sur le serveur,
2. Le port est ouvert
3. la configuration de mon service est correcte
4. le contenu est accessible.

Revision #3
Created 2026-07-17 20:31:39 UTC by Olivier
Updated 2026-07-17 20:38:23 UTC by Olivier