

**Aix Marseille Université - Campus Luminy**

**UFR des Sciences**

**Rapport de TP**

**Master Informatique**

*Module Réseaux.*

**TP n°0 :**

***Découverte de la Salle et des Outils de TP***

**Réalisé par :**

ZEMMOURI Yasmine.

## Physique et Liaison de Données :

- 1- La salle de travail contient 24 stations (3 rangées de 2x4 stations).
- 2- La station comporte une carte réseau nommée « *ens160* ».
- 3- Une fiche RJ45 présente 8 connecteurs électriques. Ces connecteurs sont utilisés pour transmettre des signaux dans le cadre des réseaux Ethernet, par exemple pour les connexions LAN.
- 4- L'autre extrémité d'une fiche RJ45 est soit un appareil réseau, ou une prise RJ45.
- 5- La salle contient un ou plusieurs Switch d'après la sortie de l'exécution de la commande ***ifconfig***.
- 6- Il existe une différence entre les trois prises concernant leurs **emplacements** : la prise de carte réseau est un matériel présent sur un ordinateur permettant une connexion physique à un réseau. La prise de commutateur est spéciale aux commutateurs réseaux, et la prise murale est montée sur un mur. Mais, **leur utilité est la même** : permettre une connexion physique (par câble) à un réseau.
- 7- **Commande : *ifconfig -a* :**

```
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  
mtu 1500  
  
inet 10.192.3.58 netmask 255.255.255.0 broadcast  
10.192.3.255  
  
ether 00:50:56:b9:24:46 txqueuelen 1000 (Ethernet)  
RX packets 114296 bytes 73868876 (73.8 MB)  
RX errors 0 dropped 2139 overruns 0 frame 0  
TX packets 36652 bytes 55407537 (55.4 MB)  
TX errors 0 dropped 0 overruns 0 carrier 0  
collisions 0
```

## Mise en Œuvre :

- 1- **Commande : *quota -s* :**

```
z23024671@V-PJ-47-048:~$ quota -s  
Quotas disque pour user z23024671 (uid 23024671) :  
  
Système de fichiers  space  quota  limite  
sursisfichiers  quota  limite  sursis  
  
nfs-salsa-isi-stj.univ-  
amu.fr:/HomedirAMU/etud/z/z23024671
```

2- Commande : **df -h /tmp** :

```
z23024671@V-PJ-47-048:~$ df -h /tmp
Sys. de fichiers Taille Utilisé Dispo
Uti% Monté sur
/dev/sdc          246G   153G   83G
```

3- Commande :

**cd /tmp/**

**curl -o debianTP.box**

**<https://pageperso.lislab.fr/emmanuel.godard/boxes/debianTP.box>**

**curl -o debianTP.box.md5 <https://pageperso.lislab.fr/emmanuel.godard/boxes/debianTP.box>**

**md5sum -c debianTP.box.md5**

```
z23024671@V-PJ-47-011:~$ cd /tmp/
z23024671@V-PJ-47-011:/tmp$ curl -o debianTP.box https://pageperso.lis-
lab.fr/emmanuel.godard/boxes/debianTP.box
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100  789M  100  789M    0     0  24.4M      0  0:00:32  0:00:32 --:--:-- 26.3M
z23024671@V-PJ-47-011:/tmp$ curl -o debianTP.box.md5 https://pageperso.lis-
lab.fr/emmanuel.godard/boxes/debianTP.box.md5
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100    47  100    47    0     0    577      0  --:--:--  --:--:--  --:--:--   580
z23024671@V-PJ-47-011:/tmp$ md5sum -c debianTP.box.md5
debianTP.box: Réussi
```

**Petite recherche** : La commande `md5sum` est utilisée pour calculer et afficher la somme de contrôle MD5 d'un fichier. Cette somme de contrôle est une valeur numérique unique qui représente le contenu du fichier et est souvent utilisée pour vérifier l'intégrité des fichiers en les comparant à des sommes de contrôle MD5 préalablement connues. Si une somme de contrôle ne correspond pas au contenu du fichier correspondant, la commande affiche un message d'erreur indiquant que la vérification a échoué pour ce fichier. Si toutes les sommes de contrôle correspondent, la commande affiche un message indiquant que la vérification est réussie pour tous les fichiers.

## Première VM : Mise en place d'une « box » :

1- Commande : ***vagrant box add /tmp/debianTP.box --name "m1reseaux"***

```
z23024671@V-PJ-47-011:/tmp$ vagrant box add /tmp/debianTP.box --name "m1reseaux"
==> box: Box file was not detected as metadata. Adding it directly...
==> box: Adding box 'm1reseaux' (v0) for provider:
      box: Unpacking necessary files from: file:///tmp/debianTP.box
==> box: Successfully added box 'm1reseaux' (v0) for 'virtualbox'!
```

2- Commande : ***VBoxManage setproperty machinefolder "/tmp"***  
***vagrant box list***

```
z23024671@V-PJ-47-011:/tmp$ VBoxManage setproperty machinefolder "/tmp"
z23024671@V-PJ-47-011:/tmp$ vagrant box list
m1reseaux (virtualbox, 0)
```

## Configuration de la VM :

En suivant les étapes de l'énoncé :

```
z23024671@V-PJ-47-011:/tmp$ mkdir -p ~/reseaux/tp0/VM1
z23024671@V-PJ-47-011:/tmp$ mkdir ~/reseaux/tp0/partage
z23024671@V-PJ-47-011:/tmp$ cd ~/reseaux/tp0/VM1
z23024671@V-PJ-47-011:~/reseaux/tp0/VM1$ vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Preparing master VM for linked clones...
      default: This is a one time operation. Once the master VM is prepared,
      default: it will be used as a base for linked clones, making the
      default: creation
      default: of new VMs take milliseconds on a modern system.
==> default: Importing base box 'm1reseaux'...
==> default: Cloning VM...
==> default: Matching MAC address for NAT networking...
==> default: Setting the name of the VM: VM1_default_1695767619966_44481
```

## Compte-rendu :

1. La VM contient trois interfaces réseaux :
  1. Loopback « lo ».
  2. Ethernet0 « eth0 ».
  3. Ethernet1 « eth1 ».
2. Commande : **ip addr** :

```
m1reseaux@VM1:~$ 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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
```

```
    link/ether 08:00:27:8d:c0:4d brd ff:ff:ff:ff:ff:ff
```

```
    altname enp0s3
```

```
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
```

```
        valid_lft 86251sec preferred_lft 86251sec
```

```
    inet6 fe80::a00:27ff:fe8d:c04d/64 scope link
```

```
        valid_lft forever preferred_lft forever
```

```
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
```

```
    link/ether 08:00:27:60:09:ac brd ff:ff:ff:ff:ff:ff
```

```
    altname enp0s8
```

```
    inet6 fe80::ba42:75f4:ca07:1886/64 scope link noprefixroute
```

```
        valid_lft forever preferred_lft forever
```

3. Connection à Internet possible avec la commande : **wget** [https://pageperso.lis-lab.fr/emmanuel.godard/enseignement/tp-m1-reseaux/00\\_decouverte/](https://pageperso.lis-lab.fr/emmanuel.godard/enseignement/tp-m1-reseaux/00_decouverte/)

```
m1reseaux@VM1:~$ wget https://pageperso.lis-lab.fr/emmanuel.godard/enseignement/tp-m1-
reseaux/00_decouverte/
--2023-09-27 08:30:03-- https://pageperso.lis-lab.fr/emmanuel.godard/enseignement/tp-m1-
reseaux/00_decouverte/
Resolving pageperso.lis-lab.fr (pageperso.lis-lab.fr)... 139.124.22.27
Connecting to pageperso.lis-lab.fr (pageperso.lis-lab.fr)|139.124.22.27|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 22243 (22K) [text/html]
Saving to: 'index.html'
```

4. La bonne manière de stopper la VM est d'utiliser la *commande* **vagrant halt** :

```
z23024671@V-PJ-47-004:~/reseaux/tp0/VM1$ vagrant halt
==> default: Attempting graceful shutdown of VM...
```

5. En doublant la ligne, nous avons ajouté une interface réseau de type LAN. Nous pouvons le voir en exécutant la commande :

```
m1reseau@VM1:~$ 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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
```

```
    link/ether 08:00:27:8d:c0:4d brd ff:ff:ff:ff:ff:ff
```

```
    altname enp0s3
```

```
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
```

```
        valid_lft 86319sec preferred_lft 86319sec
```

```
    inet6 fe80::a00:27ff:fe8d:c04d/64 scope link
```

```
        valid_lft forever preferred_lft forever
```

```
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
```

```
    link/ether 08:00:27:60:09:ac brd ff:ff:ff:ff:ff:ff
```

```
    altname enp0s8
```

```
    inet6 fe80::ba42:75f4:ca07:1886/64 scope link noprefixroute
```

```
        valid_lft forever preferred_lft forever
```

```
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
```

```
    link/ether 08:00:27:18:c9:32 brd ff:ff:ff:ff:ff:ff
```

```
    altname enp0s9
```

```
    inet6 fe80::ed17:1639:179c:562/64 scope link noprefixroute
```

```
        valid_lft forever preferred_lft forever
```

## Communication avec les VMs :

**Copier/Coller** est possible depuis l'hôte vers la VM, et vice-versa. La preuve en est ce compte rendu fait à partir de l'exécution de commandes, en copiant/collant les sorties obtenues. Ce qui est de même pour l'option **Dossier Partagé**.

## Vers le Routage :

### Nouvelles VMs :

- 1- On peut accéder à internet via la commande : **wget** <https://www.google.fr>

#### Machine Bleue :

```
m1reseaux@Bleue:~$ wget https://www.google.fr
--2023-09-27 08:45:17-- https://www.google.fr/
Resolving www.google.fr (www.google.fr)... 142.250.178.131, 2a00:1450:4007:819::2003
Connecting to www.google.fr (www.google.fr)|142.250.178.131|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html'

index.html          [ <=>          ] 23.58K  --.-KB/s    in 0.009s

2023-09-27 08:45:17 (2.64 MB/s) - 'index.html' saved [24145]
```

#### Machine Verte :

```
m1reseaux@Verte:~$ wget https://www.google.fr
--2023-09-27 08:44:37-- https://www.google.fr/
Resolving www.google.fr (www.google.fr)... 142.250.178.131,
2a00:1450:4007:819::2003
Connecting to www.google.fr (www.google.fr)|142.250.178.131|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html'

index.html          [ <=>          ] 23.55K  --.-KB/s    in 0.01s

2023-09-27 08:44:37 (1.64 MB/s) - 'index.html' saved [24114]
```



2- Commande : **ip addr** :

**Machine Bleue :**

```
m1reseaux@Bleue:~$ 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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000

    link/ether 08:00:27:8d:c0:4d brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
        valid_lft 86020sec preferred_lft 86020sec
    inet6 fe80::a00:27ff:fe8d:c04d/64 scope link
        valid_lft forever preferred_lft forever

3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000

    link/ether 08:00:27:be:5b:8f brd ff:ff:ff:ff:ff:ff
    altname enp0s8
    inet6 fe80::4d92:38b9:d0b1:227d/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

### Machine Verte :

```
m1reseaux@Verte:~$ 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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000

    link/ether 08:00:27:8d:c0:4d brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
        valid_lft 86105sec preferred_lft 86105sec
    inet6 fe80::a00:27ff:fe8d:c04d/64 scope link
        valid_lft forever preferred_lft forever

3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
group default qlen 1000

    link/ether 08:00:27:41:fd:f0 brd ff:ff:ff:ff:ff:ff
    altname enp0s8
    inet6 fe80::915b:5c44:d3e7:aebf/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

- 3- La mise à jour des adresses des machines Bleue et Verte se fait via l'interface et non en mettant à jour le Vagrantfile.
- 4- J'ai déjà configuré un réseau explicitement précédemment en utilisant CiscoPacketTracer.
- 5- Ping :
  - a. On ne peut pas voir le ping depuis le système hôte :

```
z23024671@V-PJ-47-011:~$ ping 192.168.1.1

PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.

^C

--- 192.168.1.1 ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4085s
```

- b. On peut voir les autres stations depuis les VMs :

**Machine Bleue :**

```
m1reseaux@Bleue:~$ ping 192.168.1.2
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
 64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=1.52 ms
 64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.808 ms
 64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.756 ms
 64 bytes from 192.168.1.2: icmp_seq=4 ttl=64 time=0.766 ms
 64 bytes from 192.168.1.2: icmp_seq=5 ttl=64 time=0.840 ms
 64 bytes from 192.168.1.2: icmp_seq=6 ttl=64 time=0.735 ms
 64 bytes from 192.168.1.2: icmp_seq=7 ttl=64 time=0.719 ms
 64 bytes from 192.168.1.2: icmp_seq=8 ttl=64 time=0.895 ms
^C
--- 192.168.1.2 ping statistics ---
 8 packets transmitted, 8 received, 0% packet loss, time 7211ms
 rtt min/avg/max/mdev = 0.719/0.879/1.520/0.247 ms
```

**Machine Verte :**

```
m1reseaux@Verte:~$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
 64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.52 ms
 64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.756 ms
 64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.766 ms
 64 bytes from 192.168.1.1: icmp_seq=7 ttl=64 time=0.719 ms
 64 bytes from 192.168.1.1: icmp_seq=8 ttl=64 time=0.895 ms
^C
--- 192.168.1.1 ping statistics ---
 8 packets transmitted, 8 received, 0% packet loss, time 7211ms
 rtt min/avg/max/mdev = 0.719/0.879/1.520/0.247 ms
```

6- Les paramètres à connaître pour configurer un réseau sont :

- a. Adresse IP.
- b. Masque de sous réseau.
- c. Passerelle (Gateway).
- d. Serveur DNS.
- e. Nom de la station.
- f. Mode de configuration.
- g. Proxy.

Mais les indispensables entre eux sont : l'adresse IP, le masque et la Gateway.

### **Observation du Trafic Réseau :**

*Protocole ARP : Problème avec le démarrage de wireshark et non reconnaissance des adresses IP des machines.*