

Comment dépanner ?

Pour dépanner un poste informatique il faut passer par plusieurs étapes pour nous simplifier la tâche. Voici donc un guide de dépannage en plusieurs étapes et par la suite, des explications détaillées du fonctionnement des commandes.

I - Le dépannage

1. On vérifie le câblage :
 - On vérifie visuellement si il est bien branché, pas cassé ou autre.
 - On vérifie à l'aide d'un utilitaire comme mii-too ou ethtool.
 2. On vérifie si la carte est allumée :
 - Pour cela on va utiliser des utilitaires (comme ip link) en ligne de commande sur un terminal (linux) ou un invité de commande (sur Windows).
 3. On vérifie son adresse IP :
 - Si il n'y en a pas, on lance le service DHCPCLIENT.
 - Si il n'y a pas de service DHCPCLIENT non plus, on la configure manuellement.
 4. On vérifie sa table de routage :
 - On va regarder les routes par défaut
 5. Si tout est bon jusqu'à là, alors inspectez la configuration du routeur.
 6. Diagnostics :
 - Ping pour vérifier la communication.
 - Wireshark pour regarder, capturer les paquets.
 - Tcpdump pour capturer les paquets en ligne de commande.
-

II -Explication des commandes de dépannage

Rappelons avant tout, que toutes les commandes et/ou utilitaires présentes dans ce document disposent d'un manuel.

Pour linux il est disponible avec la commande "man" :

La commande :



Le résultat :



Pour Windows il est disponible avec la commande "?" :

La commande :



Note, il existe une variante :



De plus les démonstrations de commandes sont effectuées sur machine virtuelle, si besoin voici comment en installer une : [Guide installation de VM](#)

Vérification du câblage

Tout d'abord, on vérifie le câblage, on teste si les câbles sont bien branchés. Même si ils sont bien branchés, on peut vérifier si le câble est fonctionnel correctement grâce à des utilitaire pour vérifier le câblage il existe plusieurs commandes et sur différents supports.

Pour linux il en existe plusieurs qui ont le même but mais on va s'intéresser à une seule commande qui est mii-tool.

1. On l'installe dans le terminal avec la commande :

```
apt install net-tools
```

(Rappel : il faut être en admin pour installer un paquet, on se connecter en admin avec "su root" ou bien "sudo -i")

2. Pour l'utiliser on a juste a tapé la commande suivante :

```
mii-tools [nom de la carte réseau]
```

NB : le nom de la carte réseau est visible grâce à la commande :

```
ip a
```

Cette commande nous liste les différentes cartes réseaux de la façon suivante (dans ce cas, c'est la numéro 2 qui nous intéresse) :



3. On obtiendra le résultat suivant :



4. Ce qui nous intéresse c'est la dernière partie, si il y a marqué "link ok", cela veut dire que la connection est bonne, si non cela veut dire que l'on est pas connecté. Dans ce cas la, il faudra tester les câbles en remontant jusqu'au routeur pour trouver le câble défaillant.

Pour Windows la commande équivalente serait ipconfig :

1. Elle est installée de base sur Windows, elle est utilisable dans powershell en tapant la commande suivante :

```
ipconfig
```

2. Quand on l'utilise il y a une ligne nommée statut du média qui nous permet de savoir si il est connecté ou pas :



3. La résolution du problème se fera comme dit précédemment.

Vérification si la carte réseau est activée

Si on sait que l'on est connecté la panne peut potentiellement être due à l'état de la carte réseau, si elle est désactivée on n'aura pas l'accès aux réseaux.

Pour linux on utilise cette commande : "ip link show" :

1. On l'utilise de la façon suivante :

```
ip link show [nom de la carte réseau]
```

2. Ce qui nous donne comme résultat :



3. Si derrière "statut" il y a marqué "UP" cela veut dire qu'elle est activée, si elle est marquée "DOWN" cela veut dire qu'elle est désactivée.
4. Pour l'activer on utilise la commande suivante :

```
ip link set up dev [nom de la carte réseau]
```

5. Exemple :



Pour Windows il existe une commande équivalente qui est Get-NetAdapter :

1. Elle permet de voir l'état des carte réseau, elle est utilisée de la façon suivante :

```
Get-NetAdapter
```

2. Resultat :



3. Si une carte est désactivée on peut la réactiver avec la commande suivante :

```
netsh int set int name=[nom de la carte réseau] admin=enable
```

4. Exemple :



Vérification de son adresse ip

Maintenant que l'on sait que nos cartes réseaux sont activées et bien connectées, il faut leur attribuer une route, pour cela il existe deux grandes méthodes, la première est l'utilisation d'un serveur DHCP qui attribue automatiquement les adresses, la deuxième plus fastidieuse est de les attribuer manuellement.

Sur linux :

1. Pour activer le DHCP il suffit simplement de taper la commande suivante en administrateur :

```
dhclient
```

2. Si il est déjà activé voici ce que retourne le terminal :



3. Maintenant pour ajouter manuellement l'ip on utilise la commande suivante :

```
ip addr add [IP]/[mask] dev [nom de la carte réseau]
```

4. Exemple :



NB : pour supprimer toutes les ip d'une carte on peut utiliser la commande :

```
ip addr flush dev [nom de la carte réseau]
```

Exemple :



Pour Windows il existe des commandes équivalentes qui sont :

1. Pour savoir si le DHCP est activé il suffit simplement de taper la commande suivante en administrateur (elle permet aussi de voir l'IP de la carte, l'IP du réseau et l'IP de la passerelle):

```
netsh interface ip show address
```

2. Exemple :



3. Pour l'activer on utilise la commande suivante :

```
netsh interface ip set address [nom de la carte réseau] dhcp
```

4. Si il est déjà activé voici ce que retourne powershell :



5. Maintenant pour ajouter manuellement l'ip, on utilise la commande suivante :

```
netsh interface ip set address [nom de la carte réseau] static [IP de la machine] [Masque] [IP de la passerelle]
```

6. Exemple :



7. Résultat :



NB : pour supprimer toutes les ip d'une carte on peut utiliser la commande :

```
ipconfig /release
```

Exemple :



Vérification de sa table de routage

Maintenant que l'on a rajouté les IP on doit rajouter les routes par défaut du routeur si on a utilisé les commandes précédentes, sur Windows on a pas besoin de les rajouter, sur linux la route vers le réseau se rajoute automatiquement quand on affecte l'IP de la machine il ne nous manque donc plus qu'à vérifier ou ajouter l'IP vers la passerelle sur linux.

1. Pour vérifier les routes sur linux, il existe la commande suivante :

```
ip route
```

2. Exemple :



3. Pour ajouter la route par défaut vers la passerelle, on utilise la commande suivante :

```
ip route add default via [ip de la passerelle] dev [nom de la carte réseau]
```

4. Exemple :



NB : pour supprimer une route il existe la commande :

```
ip route delete [ip]/[masque]
```

Exemple



1. Pour vérifier les routes sur Windows, il existe la commande suivante :

```
route printe
```

2. Exemple :



3. Pour ajouter la route par défaut vers la passerelle on utilise la commande suivante :

```
route add 0.0.0.0 mask 0.0.0.0 [adresse ip de la passerelle] metric 3 if  
[numéro de la carte réseau(visible sur la comande precedante)]
```

4. Exmeple :



NB : pour supprimer une route il existe la commande : route delete [0.0.0.0 pour la route par default ou l'IP de destination]

Exemple :



Après tout ça il y a d'autres vérifications faisables

Il existe beaucoup de commandes possibles pour configurer un routeur, ici on va prendre l'exemple de la configuration du routeur RB750GR3 de chez Mikrotik. ([ici le document de configuration](#))

Besoin de faire des tests ?

Il existe pas mal d'outils de diagnostic, en voici quelques uns :

- Ping
 - Permet de tester la communication entre deux machines, en local ou sur internet.
 - Envoie des paquets en ICMP.
 - Wireshark
 - Permet de regarder tout les paquets qui passent par la carte réseau ce qui peut nous aider à trouver la panne.
 - Tcpdump
 - C'est une version en ligne de commande de Wireshark.
 - Trace route ou tracert sur Windows
 - Ce sont à peu près les mêmes outils, ils envoient des paquets IP avec un TTL de plus en plus long ce qui permet de retracer la route qu'emprunte un paquet.
 - Il permet donc de retrouver quel élément du réseau pose problème.
-