

# BOND VLAN

Dans cette doc on va parler un schouya de bond (agrégation de liens Ethernet), et de vlan (subdivision de lien Ethernet). Ouais, j'en vois certains qui sourient déjà. On pourra donc faire de l'agrégation pour ensuite subdiviser. Voilà ! Trop fort. Heureusement, ce sont des choses simples à mettre en place sous Linux, et Dieu bénisse Linux. Je n'ose même pas imaginer comment le faire sur des fenêtres.

## A – Bonding :

Bondons d'abord.

Il s'agit donc de regrouper 2 ou plusieurs interfaces réseaux d'une machine en une seule interface réseau. Les raisons en sont multiples, mais la raison la plus simple d'augmenter la bande passante d'un lien. Ainsi donc si j'ai 2 cartes réseaux de 1 Giga chacun, et que je voudrais avoir un liaison avec une capacité supérieure au giga, je suis obligé de regrouper mes 2 interfaces. On peut aussi penser à une raison de redondance pour des questions de disponibilité. Bref, nous on bond en bondissant même.

**# modprobe bonding** // cette commande ne doit pas vous donner d'erreur. C'est le module du noyau qui va vous permettre de faire du bonding. Ou bien faites un **lsmod** pour vérifier.

**# aptitude install ifenslave-2.6** // Sur une debian Lenny  
ifenslave est le programme qui permet d'ajouter et supprimer des interfaces réseaux physiques d'une interface bonding.

Ensuite, on passe à la configuration des interfaces bond mais avant il faut causer un petit peu des différents mode de bonding. Il y'en a 7, ces modes, mais personnellement je ne m'intéresse qu'à un seul mode, le mode qui permet d'augmenter la bande passante ; il s'agit du mode 0, les modes allant de 0 à 6. On peut causer aussi du mode 1 qui ne fait que de la tolérance aux pannes de l'une des cartes.

Pour la suite de notre exercice, nous supposons que nous allons créer 2 bonds (bond0, et bond1) avec 5 cartes réseaux (eth0, eth1, eth2, eth3, eth4, eth5).

Les étapes :

0/ **# vi /etc/modprobe.d/bonding.conf** // puis mettre les octets suivant :

```
alias bond0 bonding
options bond0 mode=0 miimon=100 downdelay=200 updelay=200 max_bonds=2
alias bond1 bonding
options bond1 mode=3 miimon=100 downdelay=200 updelay=200 max_bonds=2
```

On lance donc autant de bonding que de bonds, tout en précisant que le nombre max de bonds est de 2 (max\_bonds=2), la valeur par défaut étant 1. On indique aussi le mode pour chaque bond (0 = agrégation de bande passante, 3 = broadcast), le miimon qui correspond au temps de détection de panne (fréquence de checking des cartes), le downdelay qui est le temps entre la découverte de panne d'une carte et sa déconnexion, et le updelay qui est le contraire du downdelay.

1/ La config des interfaces proprement dite :

Ouvrir donc /etc/network/interfaces pour les debian :

```

auto bond0
iface bond0 inet static
    address 10.196.1.196
    netmask 255.255.255.0
    gateway 10.196.1.254
    up /sbin/ifenslave bond0 eth0 eth1 eth2

auto bond1
iface bond1 inet manual
    up /sbin/ifenslave bond1 eth3 eth4

```

La commandes ifenslave peut être utilisée en dehors du fichier de config pour supprimer ou rajouter des interfaces aux bonds.

## **A – VLANisation :**

Les VLANs, c'est des virtuels LAN, des LAN virtuels. Comment ?

Il s'agit d'avoir plusieurs réseaux sur un même switch donc plusieurs réseaux sur un même LAN. Ce qui implique que tu as des cartes réseaux qui appartiendront à des réseaux différents et aussi que tu as des cartes réseaux qui auront les informations de plusieurs réseaux. C'est finalement moins évident à expliquer que je ne le pensais :-). Ensuite, cela implique pour notre machine Linux qui doit faire du VLAN d'avoir une interface physique branchée sur toutes ces interfaces virtuelles. Voilà pourquoi je parlais de subdivision.

Comment faire alors ?

Encore quelque chose de très simple :

Le module utilisé est **8021q** par similitude au numéro du protocole 802.1q). Procéder donc aux vérifications avec **lsmod**.

Puis, attaquons le fichier de config des interfaces :

On installe juste avant le paquet **vlan** et c'est bon.

**# aptitude install vlan**

Ensuite :

**# vi /etc/network/interfaces**

```

auto eth0.12
iface eth0.12 inet dhcp
    vlan-raw-device eth0

auto eth0.20
iface eth0.20 inet static
    vlan-raw-device eth0
    address 192.168.2.3
    netmask 255.255.255.0

```

Ainsi donc, 2 vlans ont été créés (le 12 et le 20) sur l'interface eth0. Dès que le nom d'une interface contient un point (.) suivi d'un numéro, Linux comprend qu'il s'agit d'un vlan. Une commande existe (vconfig qui permet de faire des manipulations sur les vlans.

Sources :

<http://www.system-linux.eu/index.php?post/2010/02/08/Configuration-bonding-avec-deux-cartes-r%C3%A9seaux> C'est beau pour un début

<http://wiki.debian.org/NetworkConfiguration#Howtocreatefaulttolerantbondingwithvlan.28Etch.29>

Très bon, plus sophistiqué

<http://www.debianadmin.com/linux-ethernet-bonding-configuration.html> Pas mal aussi

<http://linux.developpez.com/bonding/> Il est bô

<http://www.queret.net/blog/post/2008/02/07/111-linux-debian-configuration-de-vlan-ieee-8021q-vlan>

<http://www.linux-france.org/prj/inetdoc/cours/routage.inter-vlan/routage.inter-vlan.intro.vlan.html>

vlan

<http://dietpizza.ch/geekomatic/2010/07/26/1280135940000.html>

bon bonding

© Août 2010  
Roger Yerbanga  
www.yerbynet.com