

# Bonding

## Généralités

Le bonding sous Linux sert à créer une seule carte réseau virtuelle avec deux cartes réseaux physiques ou plus. Ensuite s'applique un type de bonding dit actif-passif ou actif-actif. L'actif-passif ajoute de la redondance et l'actif-actif de la redondance et des performances.

[Bonding.](#)

## Installation

Le bonding est directement disponible sous RHEL4 et 5. C'est un module nommé bonding qu'il suffit de charger.

## Configuration

### Documentation

- Tutoriel complet : <http://linux.developpez.com/bonding/>
- Autre tutoriel : <http://www.linuxfoundation.org/collaborate/workgroups/networking/bonding>
- Information complètes : <http://www.kernel.org/doc/Documentation/networking/bonding.txt>

### Configuration générale

Emplacement	Fonction
/etc/modprobe.conf	Fichier des modules chargés au démarrage.
/etc/sysconfig/network-scripts/ifcfg-eth0	Fichier de config interface physique eth0.
/etc/sysconfig/network-scripts/ifcfg-eth1	Fichier de config interface physique eth1.
/etc/sysconfig/network-scripts/ifcfg-bond0	Fichier de config interface virtuelle bond0.

### Configuration Spécifique

Exemple avec un serveur qui dispose de quatre cartes réseaux. Ajouter la ligne bond0 qui utilise le module bonding. Exemple avec un bonding sur la même carte réseau, mais il est encore plus intéressant d'utiliser un port de chaque carte sur le serveur afin d'assurer une tolérance de panne d'une carte réseau.

/etc/modprobe.conf

```
alias eth0 bnx2
alias eth1 bnx2
alias eth2 e1000e
```

```
alias eth3 e1000e
alias bond0 bonding
```

/etc/sysconfig/network-scripts/ifcfg-eth0

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
USERCTL=no
```

/etc/sysconfig/network-scripts/ifcfg-eth1

```
DEVICE=eth1
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
USERCTL=no
```

/etc/sysconfig/network-scripts/ifcfg-bond0

```
DEVICE=bond0
BOOTPROTO=none
ONBOOT=yes
IPADDR=172.20.50.100
NETMASK=255.255.0.0
BROADCAST=172.20.255.255
GATEWAY=172.20.50.1
USERCTL=no
BONDING_OPTS="mode=1 miimon=100"
```

## miimon

Specifies the MII link monitoring frequency in milliseconds. This determines how often the link state of each slave is inspected for link failures. A value of zero disables MII link monitoring. A value of 100 is a good starting point. The use\_carrier option, below, affects how the link state is determined. See the High Availability section for additional information. The default value is 0.

Miimon est le temps de rafraichissement pour tester la "vie" de la carte réseau. Le type de bonding se choisit avec le numéro de mode.

## Vérifier le statut du bond

```
cat /proc/net/bonding/bond0
```

# Les différents modes de bonding

Nous allons voir dans ce chapitre les différents modes proposés par le module bonding. Le mode par défaut est le 0 (balance-rr = round robin).

## Mode 0 balance-rr, l'équilibrage de charge

Round-robin policy: Transmit packets in sequential order from the first available slave through the last. This mode provides load balancing and fault tolerance.

Grâce à l'équilibrage de charge, les paquets transitent sur une carte réseau active, puis sur une autre, séquentiellement. Le débit de la bande passante est augmenté. Si une des cartes réseaux vient à tomber en rade, l'équilibre de charge saute cette carte et continue à tourner de manière cyclique.

## Mode 1 active-backup, la sauvegarde active

Active-backup policy: Only one slave in the bond is active. A different slave becomes active if, and only if, the active slave fails. The bond's MAC address is externally visible on only one port (network adapter) to avoid confusing the switch.

In bonding version 2.6.2 or later, when a failover occurs in active-backup mode, bonding will issue one or more gratuitous ARPs on the newly active slave. One gratuitous ARP is issued for the bonding master interface and each VLAN interfaces configured above it, provided that the interface has at least one IP address configured. Gratuitous ARPs issued for VLAN interfaces are tagged with the appropriate VLAN id. This mode provides fault tolerance. The primary option, documented below, affects the behavior of this mode.

Ce mode est une simple redondance avec basculement. Une seule interface est active. Dès que sa panne est détectée, une autre interface est activée et prend le relais. Votre bande passante ne change pas.

## Mode 2, balance-xor

XOR policy: Transmit based on the selected transmit hash policy. The default policy is a simple `( {source} + {destination} ) % n_{slaves}`

Alternate transmit policies may be selected via the `xmit_hash_policy` option. This mode provides load balancing and fault tolerance.

Une interface est affectée à l'envoi vers une même adresse MAC. Ainsi les transferts sont parallélisés et le choix de l'interface suit la règle : (Adresse MAC de la source XOR Adresse MAC de la destination) modulo nombre d'interfaces.

## Mode 3, broadcast

Broadcast policy: transmits everything on all slave interfaces. This mode provides fault tolerance.

Aucune particularité dans ce cas, toutes les données sont transmises sur toutes les interfaces actives. Aucune autre règle.

## Mode 4, la norme 802.3ad

IEEE 802.3ad Dynamic link aggregation. Creates aggregation groups that share the same speed and duplex settings. Utilizes all slaves in the active aggregator according to the 802.3ad specification.

Slave selection for outgoing traffic is done according to the transmit hash policy, which may be changed from the default simple XOR policy via the `xmit_hash_policy` option, documented below. Note that not all transmit policies may be 802.3ad compliant, particularly in regards to the packet mis-ordering requirements of section 43.2.4 of the 802.3ad standard. Differing peer implementations will have varying tolerances for noncompliance.

Prerequisites:

1. Ethtool support in the base drivers for retrieving the speed and duplex of each slave.
2. A switch that supports IEEE 802.3ad Dynamic link aggregation.

Most switches will require some type of configuration to enable 802.3ad mode.

La norme 802.3ad permet l'agrégation des liens, élargissant dynamiquement la bande passante. Les groupes sont créés dynamiquement sur la base d'un paramétrage commun.

## Mode 5, balance-tlb

Adaptive transmit load balancing: channel bonding that does not require any special switch support. The outgoing traffic is distributed according to the current load (computed relative to the speed) on each slave. Incoming traffic is received by the current slave. If the receiving slave fails, another slave takes over the MAC address of the failed receiving slave.

Prerequisite:

1. Ethtool support in the base drivers for retrieving the speed of each slave.

“TLB” pour Traffic Load Balancing Le trafic sortant est distribué selon la charge courante (calculée relativement à la vitesse) de chaque interface. Le trafic entrant est reçu par l'interface courante. Si l'interface de réception devient inactive, une autre interface prend l'adresse MAC de l'interface inactive.

## Mode 6, la balance ALB

Adaptive load balancing: includes balance-tlb plus receive load balancing (rlb) for IPV4 traffic, and does not require any special switch support. The receive load balancing is achieved by ARP negotiation.

The bonding driver intercepts the ARP Replies sent by the local system on their way out and

overwrites the source hardware address with the unique hardware address of one of the slaves in the bond such that different peers use different hardware addresses for the server. Receive traffic from connections created by the server is also balanced. When the local system sends an ARP Request the bonding driver copies and saves the peer's IP information from the ARP packet. When the ARP Reply arrives from the peer, its hardware address is retrieved and the bonding driver initiates an ARP reply to this peer assigning it to one of the slaves in the bond. A problematic outcome of using ARP negotiation for balancing is that each time that an ARP request is broadcast it uses the hardware address of the bond. Hence, peers learn the hardware address of the bond and the balancing of receive traffic collapses to the current slave. This is handled by sending updates (ARP Replies) to all the peers with their individually assigned hardware address such that the traffic is redistributed. Receive traffic is also redistributed when a new slave is added to the bond and when an inactive slave is re-activated. The receive load is distributed sequentially (round robin) among the group of highest speed slaves in the bond. When a link is reconnected or a new slave joins the bond the receive traffic is redistributed among all active slaves in the bond by initiating ARP Replies with the selected mac address to each of the clients. The updelay parameter (detailed below) must be set to a value equal or greater than the switch's forwarding delay so that the ARP Replies sent to the peers will not be blocked by the switch.

#### Prerequisites:

1. Ethtool support in the base drivers for retrieving the speed of each slave.
2. Base driver support for setting the hardware address of a device while it is open. This is required so that there will always be one slave in the team using the bond hardware address (the `curr_active_slave`) while having a unique hardware address for each slave in the bond. If the `curr_active_slave` fails its hardware address is swapped with the new `curr_active_slave` that was chosen.

“ALB” pour Adaptive load balancing. C'est un mode étendu de la balance tlb, qui inclut du load balancing en réception. L'équilibrage de charge de réception est réalisé au niveau des réponses ARP. Le module intercepte les réponses ARP et change l'adresse MAC par celle d'une des interfaces.

From:

<https://wiki.ouieuhoutca.eu/> - **kilsufi de noter**

Permanent link:

<https://wiki.ouieuhoutca.eu/bonding-carte-reseau>

Last update: **2021/01/21 21:42**

