

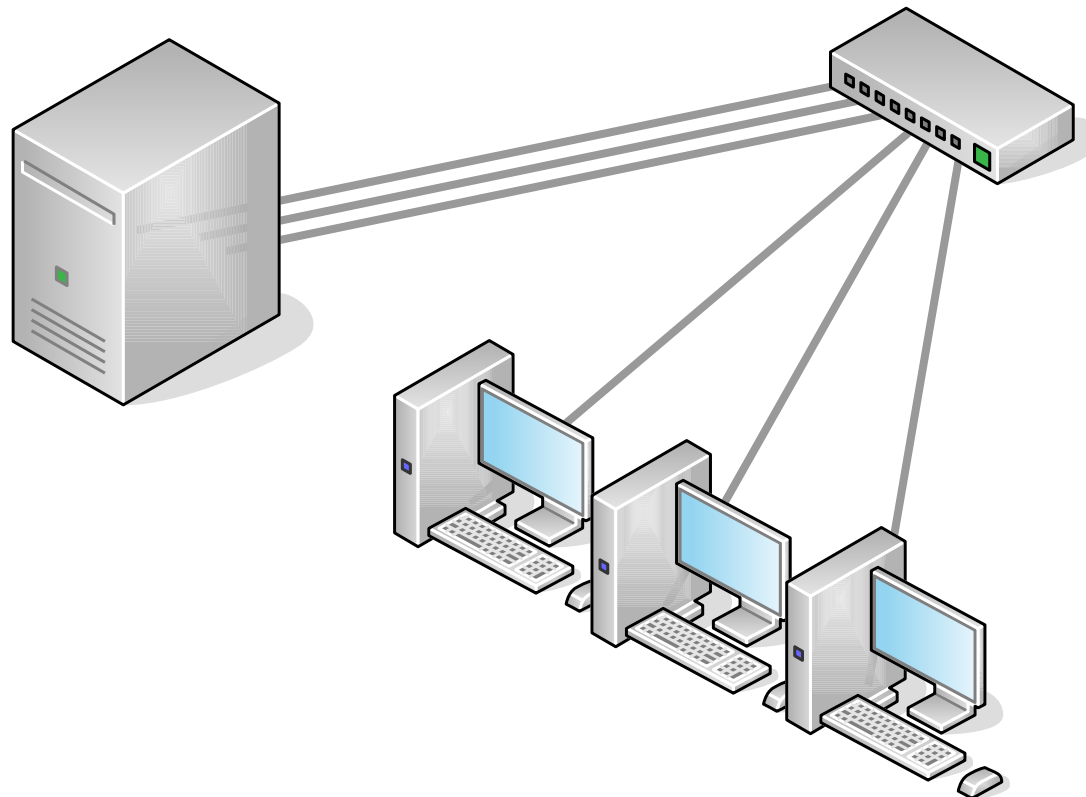
Balance de Carga y Tolerancia a Fallos en Ethernet

BC y TF en Ethernet

- Balance de carga (BC)
 - Distribuir la tarea de comunicación entre interfaces o enlaces
- Tolerancia a fallos (TF)
 - Detectar los problemas de funcionamiento y recuperarse automáticamente
- Técnica de **bonding**
 - Recibe también los nombres de *channel bonding*, *teaming*, o *link aggregation*
 - Acoplamiento de dos o más interfaces de red **esclavas**, creando una interfaz virtual capaz de funcionar en diferentes **modos**
 - Los modos de configuración de bonding definen el comportamiento del conjunto de interfaces y cumplen con diferentes objetivos
 - Algunos modos proporcionan tolerancia a fallos, gracias a la redundancia
 - Otros aumentan el ancho de banda disponible, gracias a la agregación de enlaces
 - Algunos ofrecen ambas cosas a la vez

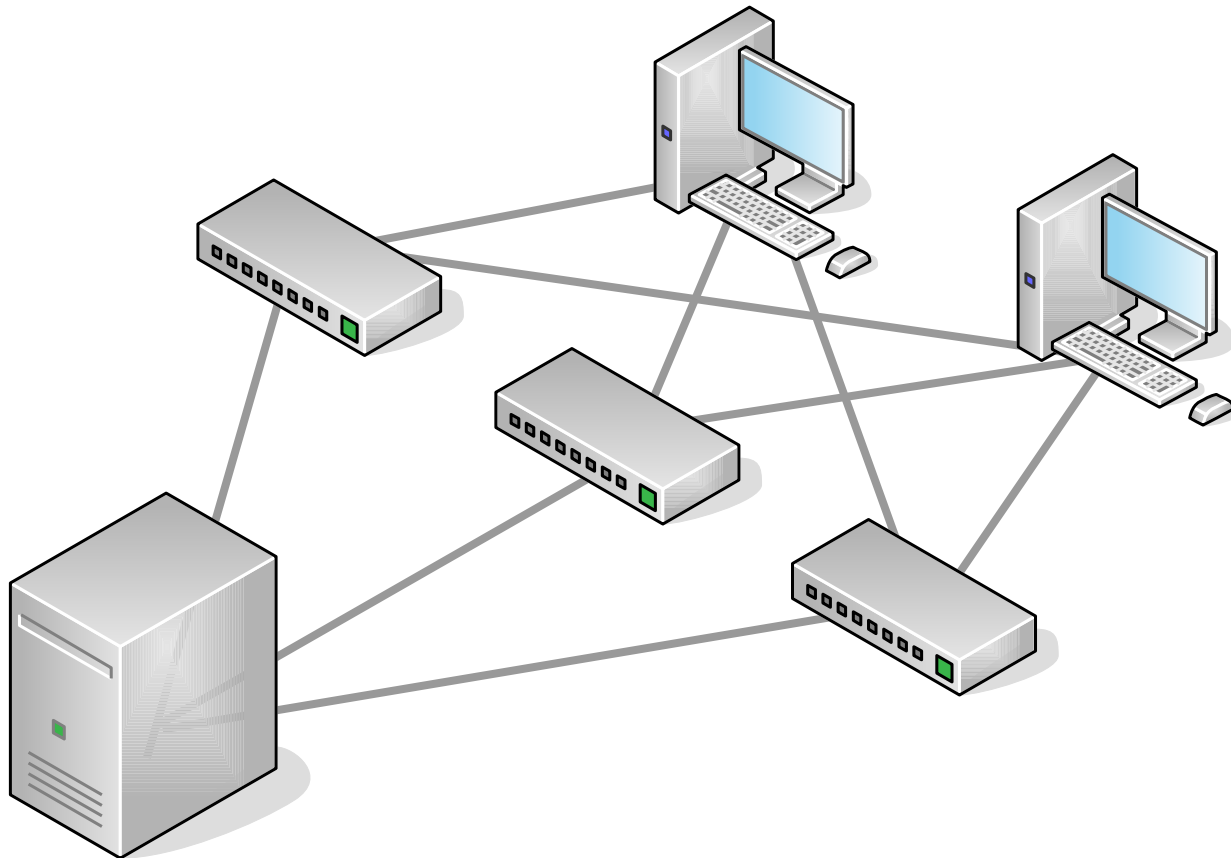
Aplicaciones de bonding

- Multiplicar el ancho de banda para enfrentar altas demandas
- Ofrecer redundancia para tolerancia a fallos



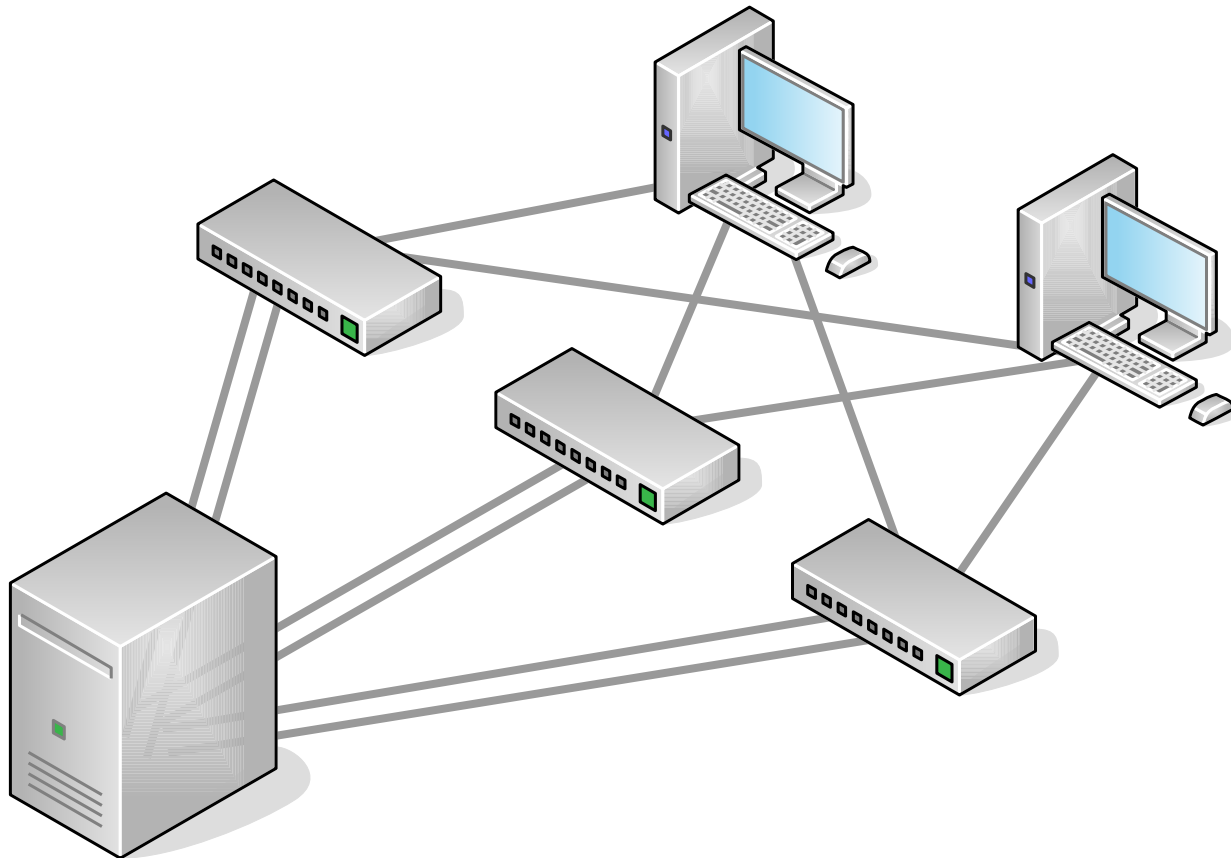
Aplicaciones de bonding

- Multiplicar el ancho de banda para enfrentar altas demandas
- Ofrecer redundancia para tolerancia a fallos



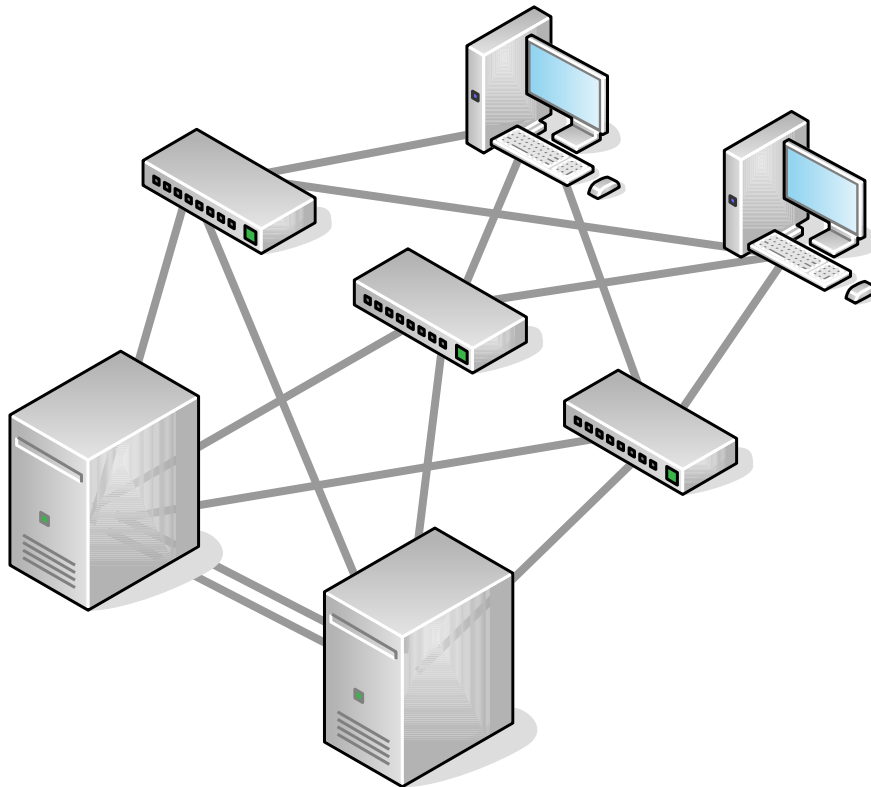
Aplicaciones de bonding

- Multiplicar el ancho de banda para enfrentar altas demandas
- Ofrecer redundancia para tolerancia a fallos



Aplicaciones de bonding

- Multiplicar el ancho de banda para enfrentar altas demandas
- Ofrecer redundancia para tolerancia a fallos



Modos de Bonding

Modo		SW	BC	TF
0	Balance-RR	Sí	Sí	Sí
1	Active-Backup			Sí
2	Balance-XOR	Sí	Sí	Sí
3	Broadcast	Sí		Sí
4	802.3ad	Sí	Sí	Sí
5	Balance-TLB		Sí	
6	Balance-ALB		Sí	

SW = Requiere capacidades especiales del switch

Modo 0 – Balance-RR

- Transmite frames usando los esclavos en orden secuencial, desde el primero disponible hasta el último
 - Si un bond tiene dos interfaces reales, y llegan al bond simultáneamente dos frames a ser enviados, el primero será transmitido por el primer esclavo; el segundo frame, por el segundo esclavo; el tercer frame será transmitido por el primer esclavo, etc.
 - Requiere capacidad especial del switch
- Provee a la vez balance de carga y tolerancia a fallos

Modo 1 – Active-Backup

- Sólo hay un esclavo activo en el bond en cada momento
 - Coloca a las demás interfaces esclavas en estado de backup
 - Sólo se activará una de ellas si se pierde el link de la interfaz activa
 - La dirección MAC del enlace está visible externamente sólo en una de las interfaces para evitar confundir al switch
 - La opción *primary* afecta el comportamiento de este modo
 - No tiene requerimientos especiales de los switches
- Provee tolerancia a fallos

Modo 2 – Balance-XOR

- Elige la interfaz de transmisión basándose en una fórmula XOR
 - Se computa la operación XOR entre la dirección MAC de origen y la de destino, módulo la cantidad de esclavos (es decir, se toma el resto de dividir por la cantidad de esclavos)
 - Este procedimiento tiene el efecto de seleccionar siempre el mismo esclavo para cada dirección MAC destino
 - Requiere capacidad especial del switch
- Provee balance de carga y tolerancia a fallos

Modo 3 – Broadcast

- Repite todos los frames por todas las interfaces esclavas
 - Es el menos usado, sólo para propósitos específicos
 - Requiere capacidad especial del switch
- Sólo provee tolerancia a fallos

Modo 4 – 802.3ad

- Este modo se conoce como el modo de agregación dinámica de enlaces (Dynamic Link Aggregation)
 - Crea grupos de agregación que comparten la misma velocidad y modo de duplexing
 - Requiere capacidad especial del switch, soporte para la norma IEEE 802.3ad (Dynamic Link) y drivers de las interfaces que soporten **ethtool**
- Provee balance de carga y tolerancia a fallos

Modo 5 – Balance-TLB

- Llamado Balance de Carga Adaptativo en Transmisión (TLB = transmission load-balance)
 - El tráfico de salida se distribuye de acuerdo a la carga actual y es encolado en cada interfaz esclava
 - El tráfico entrante es recibido por el esclavo activo en cada momento
- Provee balance de carga

Modo 6 – Balance-ALB

- Balance de carga adaptativo
- Incluye balance de carga en transmisión (Balance-TLB) y balance de carga en recepción (RLB)
 - El driver de bonding intercepta las respuestas ARP enviadas por el servidor para ejecutar balance de carga en recepción
 - Sobreescribe la dirección MAC origen del frame con la dirección MAC única de uno de los esclavos en el bond
 - De esta forma diferentes clientes usarán diferentes direcciones MAC para dirigirse al mismo server
- Provee balance de carga

Configuración en Debian

```
auto lo
    iface lo inet loopback
auto eth0
    iface eth0 inet manual
    bond-master bond0
    bond-primary eth0 eth1
auto eth1
    iface eth1 inet manual
    bond-master bond0
    bond-primary eth0 eth1
```

```
auto bond0
    iface bond0 inet static
    address 192.168.1.15
    netmask 255.255.255.0
    network 192.168.1.0
    gateway 192.168.1.1
    bond-slaves eth0 eth1
    bond-mode active-backup
    bond-miimon 100
    bond-downdelay 200
    bond-updelay 200
```

Archivo [/etc/network/interfaces](#)

Configuración en Red Hat

```
#en ifcfg-eth0
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=nice
USERCTL=no
MASTER=bond0
SLAVE=yes
```

```
#en ifcfg-eth1
DEVICE=eth1
ONBOOT=yes
BOOTPROTO=nice
USERCTL=no
MASTER=bond0
SLAVE=yes
```

```
#en ifcfg-bond0
DEVICE=bond0
BONDING_OPTS="mode=1 miimon=100
downdelay=200 updelay=200"
IPADDR=X.X.X.X
NETMASK=X.X.X.X
NETWORK=X.X.X.X
BROADCAST=X.X.X.X
GATEWAY=X.X.X.X
ONBOOT=yes
BOOTPROTO=nice
USERCTL=no
```

Archivos en </etc/sysconfig/network-scripts>

Configuración y operación

- En `/etc/modprobe.conf`:
`alias bond0 bonding`
`options bond0 miimon=80 mode=5`
- Insertar el módulo correspondiente del kernel
 - Comando `modprobe bonding`
- Desacoplar un esclavo
 - Comando `ifenslave -d bond0 eth1`
- Monitorizar el bond
 - Comando `cat /proc/net/bonding/bond0`

/proc/net/bonding/bond0

Ethernet Channel Bonding Driver: v3.2.5 (March 21, 2008)

Bonding Mode: **fault-tolerance (active-backup)**

Primary Slave: eth0

Currently Active Slave: eth0

MII Status: up

MII Polling Interval (ms): 0

Up Delay (ms): 0

Down Delay (ms): 0

ARP Polling Interval (ms): 2000

ARP IP target/s (n.n.n.n form): 10.1.1.2, 10.1.1.3

Slave Interface: eth0

MII Status: up

Link Failure Count: 0

Permanent HW addr: a6:8a:d7:63:1e:49

Slave Interface: eth1

MII Status: up

Link Failure Count: 0

Permanent HW addr: e6:c9:c8:a3:eb:72

Parámetros opcionales

- mode = 0, mode = balance-rr (equivalentes)
- miimon = 100
 - Frecuencia en ms con la cual el kernel inspecciona si siguen activas las interfaces esclavas
- downdelay, updelay
 - Tiempo en ms hasta declarar caída o nuevamente activa a una interfaz esclava luego de un cambio de estado
- primary = eth0
 - Interfaz preferida para los modos con tolerancia a fallos en stand-by
- packets-per-slave = 5
 - Cantidad de frames a emitir antes de alternar al siguiente esclavo en el modo balance-rr
- Muchos más (ver las Referencias)

Ejercitación

- Configurar un modo cualquiera de bond, observar salida de ifconfig, contenido de /proc/net/bonding/bond0. Cambiar los modos.
- Conectar dos equipos al switch, medir throughput de una transferencia (p.ej. con el comando scp e iftop [<http://pkgs.repoforge.org/iftop/iftop-0.17-1.el5.rf.i386.rpm>]). Repetir el experimento con bonds en otros modos con BC.
- Idem entre dos equipos conectando directamente bond a bond, con dos y con tres interfaces.
- Idem con dos equipos con dos interfaces en bond cada uno, a dos switches, con BC y TF.
- Configurar bond en modos con TF, examinar tráfico en las interfaces y monitorear /proc. Quitar un enlace y observar resultados.

Referencias

- <https://www.kernel.org/doc/Documentation/networking/bonding.txt>
- <https://wiki.centos.org/es/TipsAndTricks/BondingInterfaces>
- <http://www.linuxfoundation.org/collaborate/workgroups/networking/bonding>
- <http://www.cyberciti.biz/tips/debian-ubuntu-teaming-aggregating-multiple-network-connections.html>