

Guía: Configurar canal bonding a servidor SAN

Índice de contenido

DESCRIPCIÓN DE LA GUÍA	2
1.1 CONSIDERACIONES PARA LA GUÍA	2
a. <i>Objetivo:</i>	2
b. <i>Requerimientos para el desarrollo de la guía</i>	2
c. <i>Escenario para la red</i>	2
d. <i>Consideraciones para la guía</i>	2
ESCENARIO 1 – AGREGAR ALTA DISPONIBILIDAD EN SERVIDOR OPENFILER 2.9 PARA LA CONEXIÓN LAN	3
SOLUCIÓN ESCENARIO 1	4
I – VERIFICAR LA CONFIGURACIÓN DE RED DEL SERVIDOR	4
<i>Paso 1. Ingrese a la consola del servidor Openfiler</i>	4
<i>Paso 2. Verificar los archivos de configuración de red</i>	4
II – CREACIÓN DEL BONDING (MÉTODO GRÁFICO).....	6
<i>Paso 0 – Pasos previos</i>	6
<i>Paso 1 – Conectar el cable de red de la segunda interfaz de red</i>	7
<i>Paso 2 – Ingresar a la herramienta de administración gráfica (web)</i>	7
<i>Paso 3 – Crear el Bonding</i>	8
<i>Paso 4 – Verificar los archivos de configuración</i>	11
III – PRUEBAS DE DISPONIBILIDAD	13
<i>Paso 1 – Enviar paquetes ICMP desde otro host</i>	13
<i>Paso 2 – Mantener conectado eth0 y eth1</i>	13
<i>Paso 3 – Desconectar eth0 y mantener conectado eth1</i>	14
<i>Paso 4 – Mantener desconectado eth0 y Desconectar eth1</i>	15
<i>Paso 5 – Mantener desconectado eth1 y conectar eth0</i>	16
<i>Paso 6 – Conectar eth0 y mantener conectado eth1</i>	17
ESCENARIO 2 – (TAREA) CONFIGURAR BALANCEO ROUND ROBIN	18
SOLUCIÓN ESCENARIO 2	18
ESCENARIO 3 – (TAREA) CONFIGURAR BALANCEO ALB	19
SOLUCIÓN ESCENARIO 3	19
ANEXOS	20
ANEXO 1 – COMANDO IPERF.....	20
ANEXO 2 – COMANDOS ÚTILES.....	22

Descripción de la guía

1.1 Consideraciones para la guía

a. Objetivo:

El objetivo de esta sección del módulo es preparar las funciones de alta disponibilidad del servidor SAN para un entorno virtualizado.

b. Requerimientos para el desarrollo de la guía

Para esta práctica se utilizará el siguiente software:

- VMWare Workstation versión 9 ó 10. (recomendado)
- VMWare Player, si no tiene Workstation.
- Archivo con sistema operativo Openfiler 2.99 instalado (64 bits)

Es muy conveniente contar con:

- Kitty o Puty para conectarse al servidor SAN
- Iperf para el cliente que probará la conexión hacia el servidor SAN
- Conexión a Internet.

c. Escenario para la red

Para esta práctica el escenario de red será el siguiente, una máquina virtual con Openfiler 2.9 (64 bits), dos tarjetas de red para colocar alta disponibilidad.

d. Consideraciones para la guía

- La guía ha sido desarrollada por Víctor Cuchillac
- Cuando se utiliza información tomada alguna fuente, se hace la correspondiente referencia, caso contrario es autoría propia.
- Con color rojo se destacan indicaciones o comandos a seguir en la guía Ejemplo: **service network restart**
- Los comandos a digitar se muestran en negritas Ejemplo: [root@datastore1~]# **route -n**
- Cuando se utiliza información tomada alguna fuente, se hace la correspondiente referencia.
- Las notas se identifican con el siguiente texto: 📌 **Nota:** Si aparece "unknown!" en la eth1 . . .
- Los resultados al ejecutar un comando se destacan con un marcado amarillo: Ejemplo: **active-backup 1**

Escenario 1 – Agregar alta disponibilidad en servidor Openfiler 2.9 para la conexión LAN

Objetivo del escenario:

Agregar alta disponibilidad para la conexión Ethernet en el servidor SAN.

Descripción del escenario:

- En la empresa se tiene un servidor SAN con Openfiler 2.9 instalado
- El servidor posee dos tarjetas de red Ethernet con soporte para Ethtool¹ (herramienta para administrar parámetros avanzados de las interfaces de red, instalada en Openfiler)
- Uno o dos switches según implementación.

En la siguiente figura se ilustran los dos tipos de implementación para la redundancia requerida en el escenario. En el esquema de la izquierda se utiliza sólo un switch Ethernet y la redundancia se aplica a las tarjetas y cable. En el esquema de la derecha la redundancia se aplica a la tarjeta de red, cable y switch. Debido a que en ambos casos la configuración en el servidor SAN es la misma, es indiferente cual opción se tome (para efectos del laboratorio el esquema de la izquierda estará bien)

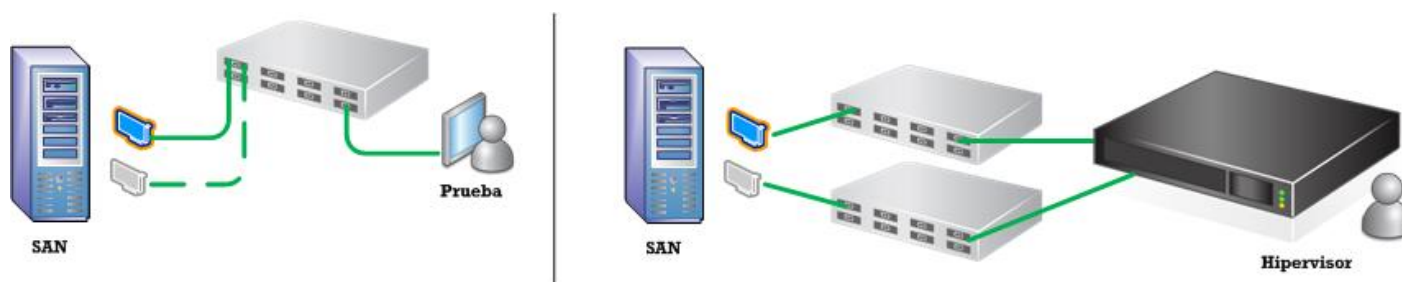


Figura 1 – Esquema de conexión para alta disponibilidad de la red

Requerimiento del escenario:

Se necesita agregar alta disponibilidad (tolerancia a fallos) en la conexión de la red LAN, de forma que cuando se “caiga” la conexión de una tarjeta de red, la segunda se active automáticamente continuando con el enlace.

- La tarjeta *eth0* deberá ser la interfaz activa.
- La tarjeta *eth1* deberá ser la interfaz pasiva.
- El tiempo para verificar que el enlace este activo será de 100 ms
- El tiempo para considerar la tarjeta activa “caída” será de 0 ms
- El tiempo para considerar la tarjeta activa “arriba” será de 0 ms

Desde cualquier máquina de prueba o desde el hipervisor (ESXi 5.X) se enviarán paquetes ICMP para crear el flujo de datos continuo

¹ Para mayor información de Ethtool en español puede consultar en: <http://wiki.elhacker.net/redes/administracion-de-redes-gnu-linux/introduccion-rapida-a-ethtool>

Solución escenario 1

I – Verificar la configuración de red del servidor

Paso 1. Ingrese a la consola del servidor Openfiler

- Las credenciales para las máquinas virtuales son:
 - Usuario administrador: **root**
 - Contraseña: **123456**

Paso 2. Verificar los archivos de configuración de red

Nota el objetivo de este paso es familiarizarse con las configuraciones de red, por si es necesario hacer una modificación manualmente

En general el procedimiento para modificar los archivos será:

1. Editar los archivos con el comando: **nano**
2. Reiniciar el servicio de red con el comando: **service network restart**

2.1 Configuración IPv4 Dinámica

```
[root@datastore1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
# Intel Corporation 82545EM Gigabit Ethernet Controller (Copper)
DEVICE=eth0
BOOTPROTO=dhcp
HWADDR=00:50:56:21:02:99
ONBOOT=yes
DHCP_HOSTNAME=datastore1.empresay.com.sv
```

2.2 Configuración IPv4 Estática

```
DEVICE=eth0
BOOTPROTO=static
HWADDR=00:50:56:21:02:99
ONBOOT=yes
```

```
IPADDR=192.168.65.132
NETMASK=255.255.255.0
NETWOK=192.168.65.0
BROADCAST=192.168.65.255
```

2.3 Configuración de la IPv4 del Gateway

```
[root@datastore1 ~]# cat /etc/sysconfig/network
```

```
NETWORKING=yes
HOSTNAME=datastore1.empresay.com.sv
GATEWAY=192.168.65.2
```

2.4 Configuración de la IPv4 del DNS

```
[root@datastore1 ~]# cat /etc/resolv.conf
```

```
; generated by /sbin/dhclient-script
search localdomain empresay.com.sv
nameserver 192.168.65.2
```

2.5 Configuración de la resolución local

```
[root@datastore1 ~]# cat /etc/hosts
```

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1          datastore1.empresay.com.sv datastore1 localhost.localdomain
localhost
```

2.6 Comprobación de los valores de las tarjetas de red

```
[root@datastore1 ~]# ifconfig
```

```
eth0      Link encap:Ethernet  HWaddr 00:50:56:21:02:99
          UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
          RX packets:16439 errors:0 dropped:0 overruns:0 frame:0
          TX packets:18148 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3124619 (2.9 MiB)  TX bytes:4270339 (4.0 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:560 (560.0 b)  TX bytes:560 (560.0 b)
```

2.7 Comprobación de la tabla de ruteo

```
[root@datastore1 network-scripts]# route -n
```

```
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
192.168.65.0     0.0.0.0         255.255.255.0  U        0      0      0 eth0
169.254.0.0     0.0.0.0         255.255.0.0    U       1004   0      0 eth0
0.0.0.0         192.168.65.1   0.0.0.0        UG        0      0      0 eth0
```

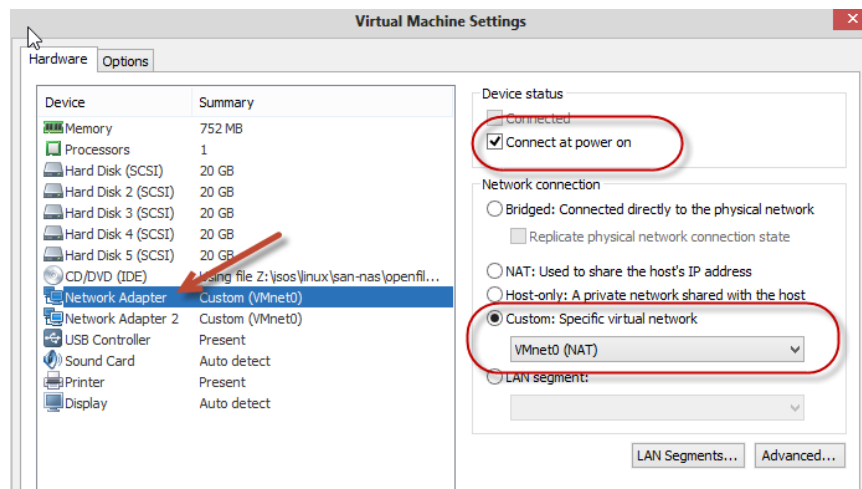
II – Creación del bonding (método gráfico)

Paso 0 – Pasos previos

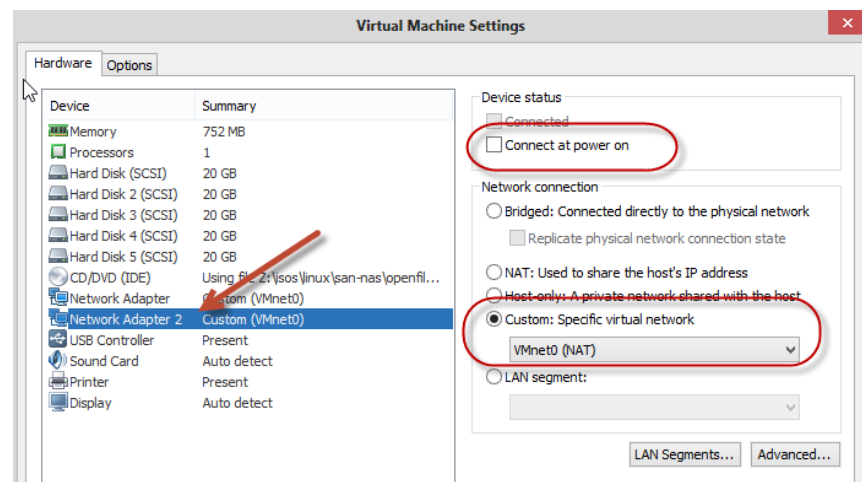
0.1 Copie la MV provista por el instructor a su directorio de trabajo

0.2 Verifique las conexiones de las dos interfaces de red

- La primera interfaz deberá estar conectada al iniciar la MV
- El tipo de conexión deberá ser NAT o “personalizada a VMNet0(NAT)”



- La primera segunda interfaz deberá estar desconectada al iniciar la MV
- El tipo de conexión deberá ser NAT o “personalizada a VMNet0(NAT)”



Nota: Ambas interfaces deben tener el mismo tipo de conexión, se recomienda utilizar NAT, para trabajar en una computadora donde no haya servicio DHCP de la red o haya varias MV con la misma dirección MAC. Sin embargo puede utilizarse Bridge.

0.3 Inicie la máquina virtual y verifique que el sistema operativo arranca correctamente

Utilice las credenciales:

- Usuario administrador: **root**
- Contraseña: **123456**

0.4 Verifique que se obtiene una dirección IPv4 de forma automática y hay comunicación con la red

- Utilice el comando: **ifconfig |more**
- Utilice el comando: **route -n**
- Utilice el comando: **ping IP_router**

Paso 1 – Conectar el cable de red de la segunda interfaz de red

Paso 2 – Ingresar a la herramienta de administración gráfica (web)

2.1 Abra una pantalla en el navegador Web

Utilice de preferencia Firefox u Opera.

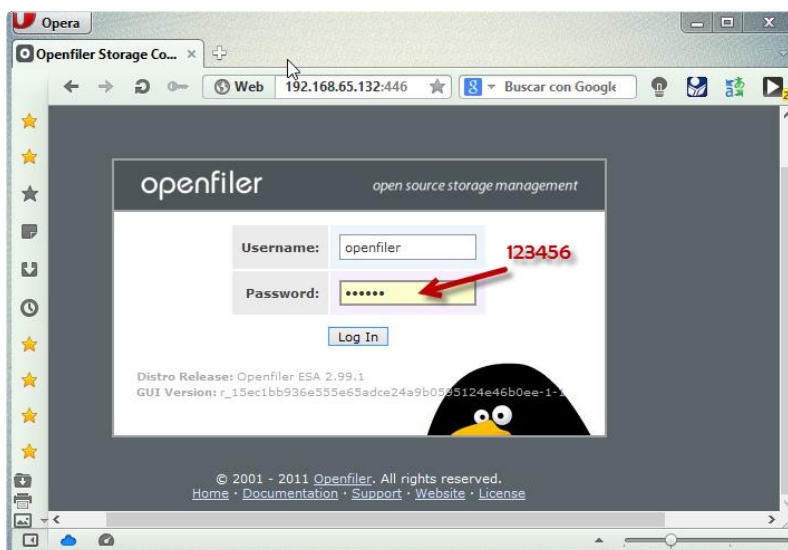
Digite la dirección IPv4 que posee el servidor Openfiler para la tarjeta eth0 con el protocolo https

- Por ejemplo: **https://192.168.65.132:446**

2.2 Acepte el certificado que envía el servidor openfiler para cifrar la comunicación

2.3 Digite las credenciales para la herramienta web de Openfiler

- Usuario: **openfiler**
- Contraseña: **123456**





Paso 3 – Crear el Bonding

3.1 Seleccionar opción de menú "System"

- Seleccionar submenú "Network setup" (seleccionado predeterminadamente)
- Ubique la sección "Network Interface Configuration"
- Verifique que ambas interfaces tengan la misma velocidad y MTU.


 **Nota:** Si aparece "unknown!" en la eth1, deberá conectar el cable de la interfaz.

3.2 Dé un clic en el enlace "Create bonded interface"

Network Interface Configuration							
Interface	Boot Protocol	IP Address	Network Mask	Speed	MTU	Link	Edit
eth0	DHCP	192.168.65.132	255.255.255.0	1000Mb/s	1500	Yes	 Configure
eth1	DHCP			1000Mb/s	1500	Yes	 Configure
Create bonded interface							

3.3 Defina las interfaces que formarán el canal bonding


- Seleccione las interfaces **eth0** y **eth1**
- Dé un clic en el botón "Continue"

Network Bonding Configuration				
 It is highly recommended that a bond be configured only if direct terminal access is possible to reconfigure if a problem arises.				
Select interfaces to bond				
X	Device	MAC Address	Mii Compatible	Current IP
<input checked="" type="checkbox"/>	eth0	00:50:56:21:02:99	No	192.168.65.132
<input checked="" type="checkbox"/>	eth1	00:50:56:22:02:99	No	
<input type="button" value="Continue"/> <input type="button" value="Cancel"/>				

3.4 Defina las opciones del canal bonding

- Los parámetros de la configuración IPv4
 - IPv4 (estática): 192.168.65.132 (La provista por el servidor DHCP)
 - Máscara de red: 255.255.255.0
- Opciones del Bond
 - Tipo: 1 (Active-Backup)
 - Tarjeta primaria: eth0
 - MII link: 100 ms

Network Bonding Configuration




It is highly recommended that a bond be configured only if direct terminal access is possible to reconfigure if a problem arises.


IP Configuration	
IP Address:	<input type="text" value="192.168.65.132"/>
Netmask:	<input type="text" value="255.255.255.0"/>

Bond Options	
Bonding Mode:	<input type="text" value="Active Backup"/>
Primary Interface:	<input type="text" value="eth0"/>
Alternate Link Detection:	<input type="text" value="False (default)"/>
MII link monitoring:	<input type="text" value="100 (default)"/>
Down Delay:	<input type="text" value="0 (default)"/>
Up Delay:	<input type="text" value="0 (default)"/>

- Dé un clic en el botón "**Continue**"

 **Nota:** El asistente gráfico sólo configura la IPv4 de forma estática para este paso, si desea configurar la IPv4 de forma dinámica deberá seguir al próximo paso y modificar el bond0, o hacerlo manualmente en el archivo /etc/sysconfig/network-scripts/bond0.

3.5 Verifique los valores del bounding




 **Nota:** Si la dirección IPv4 que se digitó en el paso anterior es diferente a la utilizada para iniciar sesión en la herramienta gráfica, la conexión será cerrada y se deberá iniciar sesión web con la nueva dirección IPv4.

- Los valores de la red IPv4 son:
 - Hostname: **datastore1.empresay.com.sv** (Donde Y es el número de grupo)
 - DNS primario: La IPv4 del DNS de la red (El NAT de VMWare se utiliza 192.168.65.2)
 - Gateway: La IPv4 del gateway de la red (El NAT de VMWare se utiliza 192.168.65.2)
- El bond0 deberá tener la siguiente configuración:
 - MTU = **1,500**
 - Enlace activo = **Yes**


Network Configuration

Hostname:	<input type="text" value="datastore1.empresay.com.sv"/>
Primary DNS:	<input type="text" value="192.168.65.2"/>
Secondary DNS:	<input type="text"/>
Gateway:	<input type="text" value="192.168.65.2"/>

Network Interface Configuration

Interface	Boot Protocol	IP Address	Network Mask	Speed	MTU	Link	Edit
bond0	Static	192.168.65.132	255.255.255.0		1500	Yes	 Configure
eth0	Configured as slave to bond: bond0					Yes	 Configure
eth1	Configured as slave to bond: bond0					Yes	 Configure

[Create bonded interface](#)

 **Nota:** Si desea modificar los valores del bond0 dé un clic en el enlace "Configure" y podrá definir si la IPv4 será dinámica, estática o el tamaño del MTU. Si se desea utilizar IPv6 la configuración se hará de forma manual.

Paso 4 – Verificar los archivos de configuración

4.1 Ingrese a la consola y digite los siguientes comandos

```
[root@datastore1 ~]# ifconfig |more

bond0      Link encap:Ethernet  HWaddr 00:50:56:21:02:99
            inet addr:192.168.65.132  Bcast:192.168.65.255  Mask:255.255.255.0
            inet6 addr: fe80::250:56ff:fe21:299/64 Scope:Link
            UP BROADCAST RUNNING MASTER MULTICAST  MTU:1500  Metric:1
            RX packets:16453 errors:0 dropped:0 overruns:0 frame:0
            TX packets:18147 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:3125459 (2.9 MiB)  TX bytes:4270105 (4.0 MiB)

eth0       Link encap:Ethernet  HWaddr 00:50:56:21:02:99
            UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
            RX packets:16439 errors:0 dropped:0 overruns:0 frame:0
            TX packets:18148 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:3124619 (2.9 MiB)  TX bytes:4270339 (4.0 MiB)

eth1       Link encap:Ethernet  HWaddr 00:50:56:21:02:99
            UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
            RX packets:14 errors:0 dropped:0 overruns:0 frame:0
            TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:1000
            RX bytes:840 (840.0 b)  TX bytes:0 (0.0 b)

lo         Link encap:Local Loopback
            inet addr:127.0.0.1  Mask:255.0.0.0
            inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING  MTU:16436  Metric:1
            RX packets:8 errors:0 dropped:0 overruns:0 frame:0
            TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:560 (560.0 b)  TX bytes:560 (560.0 b)
```

Para ver el Gateway del servidor:

```
[root@datastore1 network-scripts]# route -n

Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
192.168.65.0    0.0.0.0         255.255.255.0  U        0      0      0 bond0
169.254.0.0     0.0.0.0         255.255.0.0    U       1004    0      0 bond0
0.0.0.0         192.168.65.2   0.0.0.0        UG        0      0      0 bond0
```

Para verificar el tipo de bonding:

```
[root@datastore1 ~]# more /sys/class/net/bond0/bonding/mode

active-backup 1
```

4.2 Visualice los archivos de configuración del bonding

Este paso es muy útil si tiene problemas en la configuración del bonding o desea hacer cambios manuales. Cada vez que se haga cambios en los archivos de configuración se debe reiniciar la red con el comando: **service network restart**

```
[root@datastore1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
DEVICE=eth0
USERCTL=no
ONBOOT=yes
BOOTPROTO=static
SLAVE=yes
MASTER=bond0
```

```
[root@datastore1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-eth1
```

```
DEVICE=eth1
USERCTL=no
ONBOOT=yes
BOOTPROTO=static
SLAVE=yes
MASTER=bond0
```

```
[root@datastore1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-bond0
```

```
DEVICE=bond0
MTU=1500
USERCTL=no
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.65.132
NETMASK=255.255.255.0
BONDING_OPTS="mode=1 miimon=100 downdelay=0 updelay=0"
```

```
[root@datastore1 network-scripts]# cat /etc/resolv.conf
```

```
search datastore1.empresay.com.sv
nameserver 192.168.65.2
```

```
[root@datastore1 network-scripts]# cat /etc/hosts
```

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1      datastore1.empresay.com.sv datastore1 localhost.localdomain localhost
```

```
[root@datastore1 sysconfig]# cat /etc/sysconfig/network
```

```
NETWORKING=yes
HOSTNAME=datastore1.empresay.com.sv
GATEWAY=192.168.65.2
```

III – Pruebas de disponibilidad

Paso 1 – Enviar paquetes ICMP desde otro host

En Windows: abra una sesión de comandos y digite: `ping -t 192.168.65.132`

En Linux: abra una terminal y digite: `ping 192.168.65.132`

Paso 2 – Mantener conectado eth0 y eth1

2.1 Verificar estado de los enlaces

```
[root@datastore1 ~]# ethtool eth0 |grep "Link detected"
Link detected: yes
```

```
[root@datastore1 ~]# ethtool eth1 |grep "Link detected"
Link detected: yes
```

2.2 Verificar cantidad de enlaces caídos

```
[root@datastore1 ~]# cat /proc/net/bonding/bond0
```

Ethernet Channel Bonding Driver: v3.5.0 (November 4, 2008)

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

Primary Slave: None

Currently Active Slave: eth0

MII Status: up

MII Polling Interval (ms): 100

Up Delay (ms): 0

Down Delay (ms): 0

Slave Interface: eth0

MII Status: up

Link Failure Count: 0

Permanent HW addr: 00:50:56:21:02:99

Slave Interface: eth1

MII Status: up

Link Failure Count: 0

Permanent HW addr: 00:50:56:22:02:99

2.3 Verificar que se están enviando paquetes.

```
C:\>ping -t 192.168.65.132
```

Haciendo ping a 192.168.65.132 con 32 bytes de datos:

Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64

Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64

Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64

Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64

2.4 Comprobar que interfaz está transmitiendo y recibiendo

```
[root@datastore1 ~]# watch -d -n1 netstat -i
```

```
Every 1.0s: netstat -i      Sat Jul 12 17:33:10 2014
```

```
Kernel Interface table
```

Iface	MTU	Met	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
bond0	1500	0	17020	0	0	0	18662	0	0	0	BMmRU
eth0	1500	0	16998	0	0	0	18662	0	0	0	BMsRU
eth1	1500	0	22	0	0	0	0	0	0	0	BMsRU
lo	16436	0	8	0	0	0	8	0	0	0	LRU

Paso 3 – Desconectar eth0 y mantener conectado eth1

3.1 Verificar estado de los enlaces

```
[root@datastore1 ~]# ethtool eth0 |grep "Link detected"  
Link detected: no
```

```
[root@datastore1 ~]# ethtool eth1 |grep "Link detected"  
Link detected: yes
```

3.2 Verificar cantidad de enlaces caídos

```
[root@datastore1 ~]# cat /proc/net/bonding/bond0
```

```
Every 2.0s: cat /proc/net/bonding/bond0
```

```
Sat Jul 12 17:43:11 2014
```

```
Ethernet Channel Bonding Driver: v3.5.0 (November 4, 2008)
```

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

```
Primary Slave: eth0
```

```
Currently Active Slave: eth1
```

```
MII Status: up
```

```
MII Polling Interval (ms): 100
```

```
Up Delay (ms): 0
```

```
Down Delay (ms): 0
```

```
Slave Interface: eth0
```

```
MII Status: down
```

```
Link Failure Count: 1
```

```
Permanent HW addr: 00:50:56:21:02:99
```

```
Slave Interface: eth1
```

```
MII Status: up
```

```
Link Failure Count: 0
```

```
Permanent HW addr: 00:50:56:22:02:99
```

2.3 Verificar que se están enviando paquetes.

```
C:\>ping -t 192.168.65.132
```

```
Haciendo ping a 192.168.65.132 con 32 bytes de datos:  
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64  
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64  
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64  
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64
```

3.4 Comprobar que interfaz está transmitiendo y recibiendo

```
[root@datastore1 ~]# watch -d -n1 netstat -i
```

```
Every 1.0s: netstat -i          Jul 12 17:41:58 2014
```

```
Kernel Interface table
```

Iface	MTU	Met	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
bond0	1500	0	19043	0	0	0	20608	0	0	0	BMmRU
eth0	1500	0	18919	0	0	0	20522	0	0	0	BMsU
eth1	1500	0	124	0	0	0	86	0	0	0	BMsRU
lo	16436	0	8	0	0	0	8	0	0	0	LRU

Paso 4 – Mantener desconectado eth0 y Desconectar eth1

4.1 Verificar estado de los enlaces

```
[root@datastore1 ~]# ethtool eth0 |grep "Link detected"  
Link detected: no
```

```
[root@datastore1 ~]# ethtool eth1 |grep "Link detected"  
Link detected: no
```

4.2 Verificar cantidad de enlaces caídos

```
[root@datastore1 ~]# cat /proc/net/bonding/bond0
```

```
Every 2.0s: cat /proc/net/bonding/bond0  
Sat Jul 12 17:43:11 2014  
Ethernet Channel Bonding Driver: v3.5.0 (November 4, 2008)
```

```
Bonding Mode: fault-tolerance (active-backup)  
Primary Slave: eth0  
Currently Active Slave: eth1  
MII Status: up  
MII Polling Interval (ms): 100  
Up Delay (ms): 0  
Down Delay (ms): 0
```

```
Slave Interface: eth0
```

Módulo: *Servicios de almacenamiento en nube privada*

MII Status: down

Link Failure Count: 1

Permanent HW addr: 00:50:56:21:02:99

Slave Interface: eth1

MII Status: up

Link Failure Count: 1

Permanent HW addr: 00:50:56:22:02:99

4.3 Verificar que se no están enviando paquetes.

```
C:\>ping -t 192.168.65.134
```

Haciendo ping a 192.168.65.134 con 32 bytes de datos:
Respuesta desde 192.168.65.1: Host de destino inaccesible.
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.

4.4 Comprobar que interfaz está transmitiendo y recibiendo

```
[root@datastore1 ~]# watch -d -n1 netstat -i
```

```
Every 1.0s: netstat -i          Jul 12 17:41:58 2014
```

Kernel Interface table

Iface	MTU	Met	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
bond0	1500	0	19044	0	0	0	20618	0	0	0	BMmRU
eth0	1500	0	18919	0	0	0	20542	0	0	0	BMsU
eth1	1500	0	126	0	0	0	86	0	0	0	BMsRU
lo	16436	0	8	0	0	0	8	0	0	0	LRU

Paso 5 – Mantener desconectado eth1 y conectar eth0

5.1 Verificar que se están enviando paquetes de nuevo.

```
C:\>ping -t 192.168.65.132
```

Haciendo ping a 192.168.65.132 con 32 bytes de datos:
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64

5.2 Verificar estado de los enlaces

```
[root@datastore1 ~]# ethtool eth0 |grep "Link detected"  
Link detected: yes
```

```
[root@datastore1 ~]# ethtool eth1 |grep "Link detected"  
Link detected: no
```


5.3 Comprobar que interfaz está transmitiendo y recibiendo

```
[root@datastore1 ~]# watch -d -n1 netstat -i
```

```
Every 1.0s: netstat -i
```

```
Sat Jul 12 19:00:42 2014
```

```
Kernel Interface table
```

Iface	MTU	Met	RX-OK	RX-ERR	RX-DRP	RX-OVR	TX-OK	TX-ERR	TX-DRP	TX-OVR	Flg
bond0	1500	0	33191	0	0	0	34079	0	0	0	BMmRU
eth0	1500	0	25898	0	0	0	27839	0	0	0	BMsRU
eth1	1500	0	7293	0	0	0	6240	0	0	0	BMsU
lo	16436	0	80	0	0	0	80	0	0	0	LRU

Paso 6 – Conectar eth0 y mantener conectado eth1

6.1 Verificar estado de los enlaces

```
[root@datastore1 ~]# ethtool eth0 |grep "Link detected"  
Link detected: no
```

```
[root@datastore1 ~]# ethtool eth1 |grep "Link detected"  
Link detected: yes
```

6.2 Verificar que se están enviando paquetes.

```
C:\>ping -t 192.168.65.132
```

```
Haciendo ping a 192.168.65.132 con 32 bytes de datos:
```

```
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64
```

```
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64
```

```
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64
```

```
Respuesta desde 192.168.65.132: bytes=32 tiempo<1m TTL=64
```

Instalar iperf

Para openfiler

Ya está instalado.

- Necesitaría conexión hacia Internet y
- Digitar: [root@datastore1 ~]# conary install iptraf

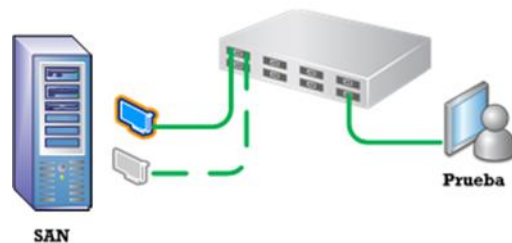
Escenario 2 – (Tarea) Configurar balanceo round robin

Objetivo del escenario:

Agregar balanceo de carga Ethernet tipo Round Robin al tráfico saliente en la conexión Ethernet en el servidor SAN.

Descripción del escenario:

- En la empresa se tiene un servidor SAN con Openfiler 2.9 instalado
- El servidor posee dos tarjetas de red Ethernet con soporte para Ethtool (herramienta para administrar parámetros avanzados de las interfaces de red, instalada en Openfiler)
- Uno o dos switches según implementación.



Requerimiento del escenario:

Se necesita que el tráfico que se envía desde el servidor Openfiler sea distribuido por las dos tarjetas de red Ethernet.

Solución escenario 2

Recomendaciones

- Realice el escenario 1
- Cambie el valor del modo del canal bonding
- Utilice el siguiente comando para visualizar la asignación round robin
`[root@datastore1 ~]# watch -d -n1 netstat -i`

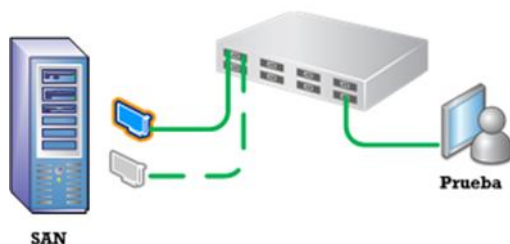
Escenario 3 – (Tarea) Configurar balanceo alb

Objetivo del escenario:

Agregar balanceo de carga al tráfico entrante y saliente de la conexión Ethernet en el servidor SAN.

Descripción del escenario:

- En la empresa se tiene un servidor SAN con Openfiler 2.9 instalado
- El servidor posee dos tarjetas de red Ethernet con soporte para Ethtool (herramienta para administrar parámetros avanzados de las interfaces de red, instalada en Openfiler)
- Uno o dos switches según implementación.



Requerimiento del escenario:

- Se necesita que el tráfico que se envía y recibe en el servidor Openfiler sea distribuido por las dos tarjetas de red Ethernet utilizando balanceo equitativo.
- Se desea comparar el ancho de banda sin bonding, active-backup y balance-alb.
- Compare los resultados y explique si hay o no variación y ¿por qué ocurre?

Solución escenario 3

Recomendaciones

- Realice el escenario 1
- Cambie el valor del modo del canal bonding
- Utilice el siguiente comando para visualizar la asignación en el flujo entrante y saliente
`[root@datastore1 ~]# watch -d -n1 netstat -i`
- Utilice el siguiente comando para medir el ancho de banda (vea Anexo 1)
`[root@datastore1 ~]# iperf`

Anexos

Anexo 1 – Comando iperf

La siguiente información ha sido tomada de: <http://es.wikipedia.org/wiki/Iperf>

- Iperf es una herramienta que se utiliza para realizar pruebas en redes de computadoras. El funcionamiento habitual es crear flujos de datos TCP y UDP y medir el rendimiento de la red.
- Iperf fue desarrollado por el Distributed Applications Support Team (DAST) en el National Laboratory for Applied Network Research (NLNR)
- Iperf permite al usuario ajustar varios parámetros que pueden ser usados para hacer pruebas en una red, o para optimizar y ajustar la red.
- Iperf puede **funcionar como cliente o como servidor** y puede medir el rendimiento entre los dos extremos de la comunicación, unidireccional o bidireccionalmente. Es software de código abierto y puede ejecutarse en varias plataformas incluyendo Linux, Unix y Windows.

Iperf puede ser descargado de: <https://iperf.fr/>

Suponga que se tiene la siguiente configuración:


- Servidor iperf en Linux recibiendo datos = 192.168.65.132
- Cliente iperf en Windows enviando datos = 192.168.65.133

Para probar evaluar el ancho de banda se debe descargar iperf en cada equipo y ejecutar un equipo como servidor iperf y el otro equipo como cliente iperf.

De manera predeterminada iperf utiliza el puerto 5001 en el servidor y un puerto mayor a 1024 en el cliente, por lo cual el firewall del servidor debe tener abierto el puerto 5001

Opciones a utilizar en este ejemplo

Opción	Descripción
-s	modo servidor
-t 30	duración del test 30 segundos
-i 5	intervalo de muestra de estadísticas cada 5 segundos
-c 192.168.65.132	conexión hacia el servidor iperf
-f MB	Para ver las respuestas en MBps

 **Nota:** En el servidor Openfiler brindado como máquina virtual en el laboratorio ya tiene instalado iperf. Para instalar iperf en Openfiler se debe: tener conexión a Internet y digitar el siguiente comando **conary install iperf**

Comando a digitar en el servidor Linux:

```
[root@datastore1 ~]# iperf -s -t 30 -i 5
```

```
-----  
Server listening on TCP port 5001  
TCP window size: 85.3 KByte (default)  
-----
```

```
[ 4] local 192.168.65.132 port 5001 connected with 192.168.65.133 port 49195  
[ ID] Interval      Transfer      Bandwidth  
[ 4]  0.0- 5.0 sec  0.00 Bytes   0.00 bits/sec  
[ ID] Interval      Transfer      Bandwidth  
[ 4]  5.0-10.0 sec  0.00 Bytes   0.00 bits/sec  
[ ID] Interval      Transfer      Bandwidth  
[ 4] 10.0-15.0 sec  0.00 Bytes   0.00 bits/sec  
[ ID] Interval      Transfer      Bandwidth  
[ 4] 15.0-20.0 sec  0.00 Bytes   0.00 bits/sec  
[ ID] Interval      Transfer      Bandwidth  
[ 4] 20.0-25.0 sec  0.00 Bytes   0.00 bits/sec  
[ ID] Interval      Transfer      Bandwidth  
[ 4] 25.0-30.0 sec  0.00 Bytes   0.00 bits/sec  
[ ID] Interval      Transfer      Bandwidth  
[ 4]  0.0-30.1 sec  8.38 GBytes  2.39 Gbits/sec
```

comando en cliente Windows:

```
C:\progs\sistema\iperf>iperf -c 192.168.65.132 -t 30 -i 5
```

```
Microsoft Windows [Versión 6.1.7600]  
Copyright (c) 2009 Microsoft Corporation. Reservados todos los derechos.
```

```
C:\progs\sistema\iperf>iperf -c 192.168.65.132 -t 30 -i 5
```

```
-----  
Client connecting to 192.168.65.132, TCP port 5001  
TCP window size: 64.0 KByte (default)  
-----
```

```
[ 3] local 192.168.65.133 port 49195 connected with 192.168.65.132 port 5001  
[ ID] Interval      Transfer      Bandwidth  
[ 3]  0.0- 5.0 sec  1.40 GBytes  2.41 Gbits/sec  
[ 3]  5.0-10.0 sec  1.45 GBytes  2.50 Gbits/sec  
[ 3] 10.0-15.0 sec  1.45 GBytes  2.49 Gbits/sec  
[ 3] 15.0-20.0 sec  1.45 GBytes  2.48 Gbits/sec  
[ 3] 20.0-25.0 sec  1.20 GBytes  2.06 Gbits/sec  
[ 3] 25.0-30.0 sec  1.43 GBytes  2.45 Gbits/sec  
[ 3]  0.0-30.0 sec  8.38 GBytes  2.40 Gbits/sec
```

Anexo 2 – Comandos útiles

Función	Comando a digitar (puede haber más de una forma de realizar la función)
Editar archivos	nano ruta/nombre_archivo F2 -> para guardar Ctl + X -> para salir
Activar tarjeta de red	ifconfig et0 up ifup eth0
Desactivar tarjeta de red	ifconfig eth0 down ifdown eth0
Solicitar IPv4 dinámicamente al DHCP	dhclient eth0
Matar un proceso	killall dhclient
Cambiar contraseña del root	passwd root
Ver direcciones ip	ifconfig more
Ver tabla de ruteo	route -n
Reiniciar la configuración de la red	Service network restart (stop, start)