



CHAPTER 3

IP over IB Protocol

This chapter describes IP over IB protocol and includes the following sections:

- [Introduction, page 3-1](#)
- [Manually Configuring IPoIB for Default IB Partition, page 3-2](#)
- [Subinterfaces, page 3-2](#)
- [Verifying IPoIB Functionality, page 3-5](#)
- [IPoIB Performance, page 3-6](#)
- [Sample Startup Configuration File, page 3-8](#)
- [IPoIB High Availability, page 3-8](#)



Note

See the “[Root and Non-root Conventions in Examples](#)” section on [page ix](#) for details about the significance of prompts used in the examples in this chapter.

Introduction

Configuring IPoIB requires that you follow similar steps to the steps used for configuring IP on an Ethernet network. When you configure IPoIB, you assign an IP address and a subnet mask to each HCA port. The first HCA port on the first HCA in the host is the ib0 interface, the second port is ib1, and so on.



Note

To enable these IPoIB settings across reboots, you must explicitly add these settings to the networking interface startup configuration file. For a sample configuration file, see the “[Sample Startup Configuration File](#)” section on [page 3-8](#).

See your Linux distribution documentation for additional information about configuring IP addresses.

Manually Configuring IPoIB for Default IB Partition

To manually configure IPoIB for the default IB partition, perform the following steps:

Step 1 Log in to your Linux host.

Step 2 To configure the interface, enter the **ifconfig** command with the following items:

- The appropriate IB interface (**ib0** or **ib1** on a host with one HCA)
- The IP address that you want to assign to the interface
- The **netmask** keyword
- The subnet mask that you want to assign to the interface

The following example shows how to configure an IB interface:

```
host1# ifconfig ib0 192.168.0.1 netmask 255.255.252.0
```

Step 3 (Optional) Verify the configuration by entering the **ifconfig** command with the appropriate port identifier *ib#* argument.

The following example shows how to verify the configuration:

```
host1# ifconfig ib0
ib0      Link encap:Ethernet HWaddr F8:79:D1:23:9A:2B
         inet addr:192.168.0.1 Bcast:192.168.0.255 Mask:255.255.255.0
         inet6 addr: fe80::9879:d1ff:fe20:f4e7/64 Scope:Link
         UP BROADCAST RUNNING MULTICAST MTU:2044 Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:9 overruns:0 carrier:0
         collisions:0 txqueuelen:1024
         RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

Step 4 Repeat [Step 2](#) and [Step 3](#) on the remaining interface(s).

Subinterfaces

This section describes subinterfaces. Subinterfaces divide primary (parent) interfaces to provide traffic isolation. Partition assignments distinguish subinterfaces from parent interfaces. The default Partition Key (p_key), ff:ff, applies to the primary (parent) interface.

This section includes the following topics:

- [Creating a Subinterface Associated with a Specific IB Partition, page 3-3](#)
- [Removing a Subinterface Associated with a Specific IB Partition, page 3-4](#)

Creating a Subinterface Associated with a Specific IB Partition

To create a subinterface associated with a specific IB partition, perform the following steps:

Step 1 Create a partition on an IB SFS. Alternatively, you can choose to create the partition of the IB interface on the host first, and then create the partition for the ports on the IB SFS. See the *Cisco SFS Product Family Element Manager User Guide* for information regarding valid partitions on the IB SFS.

Step 2 Log in to your host.

Step 3 Add the value of the partition key to the file as root user.

The following example shows how to add partition 80:02 to the primary interface ib0:

```
host1# /usr/local/topspin/sbin/ipoibcfg add ib0 80:02
```

Step 4 Verify that the interface is set up by ensuring that ib0.8002 is displayed.

The following example shows how to verify the interface:

```
host1# ls /sys/class/net
eth0 ib0 ib0.8002 ib1 lo sit0
```

Step 5 Verify that the interface was created by entering the **ifconfig -a** command.

The following example shows how to enter the **ifconfig -a** command:

```
host1# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:30:48:20:D5:D1
          inet addr:172.29.237.206  Bcast:172.29.239.255  Mask:255.255.252.0
          inet6 addr: fe80::230:48ff:fe20:d5d1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:9091465 errors:0 dropped:0 overruns:0 frame:0
          TX packets:505050 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1517373743 (1.4 GiB)  TX bytes:39074067 (37.2 MiB)
          Base address:0x3040 Memory:dd420000-dd440000

ib0       Link encap:Ethernet  HWaddr F8:79:D1:23:9A:2B
          inet addr:192.168.0.1  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::9879:d1ff:fe20:f4e7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:2044  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:9 overruns:0 carrier:0
          collisions:0 txqueuelen:1024
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

ib0.8002  Link encap:Ethernet  HWaddr 00:00:00:00:00:00
          BROADCAST MULTICAST  MTU:2044  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1024
          RX bytes:0 (0.0 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:378 errors:0 dropped:0 overruns:0 frame:0
          TX packets:378 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:45730 (44.6 KiB)  TX bytes:45730 (44.6 KiB)

sit0      Link encap:IPv6-in-IPv4
```

```
NOARP MTU:1480 Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

Verify that you see the `ib0.8002` output.

- Step 6** Configure the new interface just as you would the parent interface. (See the “[Manually Configuring IPoIB for Default IB Partition](#)” section on page 3-2.)

The following example shows how to configure the new interface:

```
host1# ifconfig ib0.8002 192.168.12.1 netmask 255.255.255.0
```

Removing a Subinterface Associated with a Specific IB Partition

To remove a subinterface, perform the following steps:

- Step 1** Take the subinterface offline. You cannot remove a subinterface until you bring it down.

The following example shows how to take the subinterface offline:

```
host1# ifconfig ib0.8002 down
```

- Step 2** Remove the value of the partition key to the file as root user.

The following example shows how to remove the partition 80:02 from the primary interface `ib0`:

```
host1# /usr/local/topspin/sbin/ipoibcfg del ib0 80:02
```

- Step 3** (Optional) Verify that the subinterface no longer appears in the interface list by entering the `ifconfig -a` command.

The following example shows how to verify that the subinterface no longer appears in the interface list:

```
host1# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:30:48:20:D5:D1
          inet addr:172.29.237.206  Bcast:172.29.239.255  Mask:255.255.252.0
          inet6 addr: fe80::230:48ff:fe20:d5d1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:9091465 errors:0 dropped:0 overruns:0 frame:0
          TX packets:505050 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1517373743 (1.4 GiB)  TX bytes:39074067 (37.2 MiB)
          Base address:0x3040 Memory:dd420000-dd440000

ib0       Link encap:Ethernet  HWaddr F8:79:D1:23:9A:2B
          inet addr:192.168.0.1 Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::9879:d1ff:fe20:f4e7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:2044  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:9 overruns:0 carrier:0
          collisions:0 txqueuelen:1024
          RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

ib0.8002  Link encap:Ethernet  HWaddr 00:00:00:00:00:00
          BROADCAST MULTICAST  MTU:2044  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
```

```

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1024
RX bytes:0 (0.0 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:378 errors:0 dropped:0 overruns:0 frame:0
          TX packets:378 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:45730 (44.6 KiB) TX bytes:45730 (44.6 KiB)

sit0     Link encap:IPv6-in-IPv4
          NOARP MTU:1480 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

```

Verifying IPoIB Functionality

To verify your configuration and your IPoIB functionality, perform the following steps:

Step 1 Log in to your hosts.

Step 2 Verify the IPoIB functionality by using the **ifconfig** command.

The following example shows how two IB nodes are used to verify IPoIB functionality. In the following example, IB node 1 is at 192.168.0.1, and IB node 2 is at 192.168.0.2:

```

host1# ifconfig ib0 192.168.0.1 netmask 255.255.252.0
host2# ifconfig ib0 192.168.0.2 netmask 255.255.252.0

```

Step 3 Enter the **ping** command from 192.168.0.1 to 192.168.0.2.

The following example shows how to enter the **ping** command:

```

host1# ping -c 5 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data.
64 bytes from 192.168.0.2: icmp_seq=0 ttl=64 time=0.079 ms
64 bytes from 192.168.0.2: icmp_seq=1 ttl=64 time=0.044 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=64 time=0.055 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=64 time=0.049 ms
64 bytes from 192.168.0.2: icmp_seq=4 ttl=64 time=0.065 ms

--- 192.168.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms rtt min/avg/max/mdev =
0.044/0.058/0.079/0.014 ms, pipe 2

```

IPoIB Performance

This section describes how to verify IPoIB performance by running the Bandwidth test and the Latency test. These tests are described in detail at the following URL:

<http://www.netperf.org/netperf/training/Netperf.html>

To verify IPoIB performance, perform the following steps:

Step 1 Download Netperf from the following URL:

<http://www.netperf.org/netperf/NetperfPage.html>

Step 2 Compile Netperf by following the instructions at <http://www.netperf.org/netperf/NetperfPage.html>.

Step 3 Start the Netperf server.

The following example shows how to start the Netperf server:

```
host1$ netserver
Starting netserver at port 12865
Starting netserver at hostname 0.0.0.0 port 12865 and family AF_UNSPEC
host1$
```

Step 4 Run the Netperf client. The default test is the Bandwidth test.

The following example shows how to run the Netperf client, which starts the Bandwidth test by default:

```
host2$ netperf -H 192.168.0.1 -c -C -- -m 65536
TCP STREAM TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.0.1 (192.168.0.1) port 0
AF_INET
Recv  Send  Send      Utilization      Service Demand
Socket Socket  Message  Elapsed           Send  Recv  Send  Recv
Size  Size  Size     Time      Throughput  local  remote  local  remote
bytes bytes bytes    secs.    10^6bits/s  % S   % S   us/KB  us/KB

 87380 16384 65536    10.00      2701.06   46.93  48.73   5.694  5.912
```



Note You must specify the IPoIB IP address when running the Netperf client.

The following list describes parameters for the **netperf** command:

-H	Where to find the server
192.168.0.1	IPoIB IP address
-c	Client CPU utilization
-C	Server CPU utilization
--	Separates the global and test-specific parameters
-m	Message size, which is 65536 in the example above

The notable performance values in the example above are as follows:

Throughput is 2.70 gigabits per second.

Client CPU utilization is 46.93 percent of client CPU.

Server CPU utilization is 48.73 percent of server CPU.

Step 5 Run the Netperf Latency test.

Run the test once, and stop the server so that it does not repeat the test.

The following example shows how to run the Latency test, and then stop the Netperf server:

```
host2$ netperf -H 192.168.0.1 -c -C -t TCP_RR -- -r 1,1
TCP REQUEST/RESPONSE TEST from 0.0.0.0 (0.0.0.0) port 0 AF_INET to 192.168.0.1
(192.168.0.1) port 0 AF_INET
Local /Remote
Socket Size   Request Resp.  Elapsed Trans.   CPU    CPU    S.dem  S.dem
Send  Recv   Size   Size   Time    Rate    local  remote local  remote
bytes bytes  bytes  bytes  secs.   per sec % S    % S    us/Tr  us/Tr

16384 87380 1       1       10.00  17228.96  12.98  12.30  30.146  28.552
16384 87380
```

The following list describes parameters for the **netperf** command:

-H	Where to find the server
192.168.0.1	IPoB IP address
-c	Client CPU utilization
-C	Server CPU utilization
-t	Test type
TCP_RR	TCP required response test
--	Separates the global and test-specific parameters
-r 1,1	The request size sent and how many bytes requested back

The notable performance values in the example above are as follows:

Client CPU utilization is 12.98 percent of client CPU.

Server CPU utilization is 12.30 percent of server CPU.

Latency is 29.02 microseconds. Latency is calculated as follows:

$(1 / \text{Transaction rate per second}) / 2 * 1,000,000 = \text{one-way average latency in microseconds}$

Step 6 To end the test, shut down the Netperf server.

```
host1$ pkill netserver
```

Sample Startup Configuration File

IP addresses that are configured manually are not persistent across reboots. You must use a configuration file to configure IPoIB when the host boots. Two sample configurations are included in this section.

The following sample configuration shows an example file named `ifcfg-ib0` that resides on a Linux host in `/etc/sysconfig/network-scripts/` on RHEL3 and RHEL4. The configuration file configures an IP address at boot time.

```
host1# cat > /etc/sysconfig/network-scripts/ifcfg-ib0 << EOF
> DEVICE=ib0
> BOOTPROTO=static
> IPADDR=192.168.0.1
> NETMASK=255.255.255.0
> ONBOOT=yes
> EOF
```

The following sample configuration shows an example file named `ifcfg-ib0` in `/etc/sysconfig/network/` on SLES10. The configuration file configures an IP address at boot time.

```
host1# cat > /etc/sysconfig/network/ifcfg-ib0 << EOF
> DEVICE=ib0
> BOOTPROTO=static
> IPADDR=192.168.0.1
> NETMASK=255.255.255.0
> STARTMODE=auto
> EOF
```

IPoIB High Availability

This section describes IPoIB high availability. IPoIB supports active/passive port failover high availability between two or more ports. When you enable the high availability feature, the ports on the HCA (for example, `ib0` and `ib1`) merge into one virtual port. If you configure high availability between the ports on the HCA(s), only one of the physical ports passes traffic. The other ports are used as standby in the event of a failure. This section includes the following topics:

- [Merging Physical Ports](#)
- [Unmerging Physical Ports](#)

Merging Physical Ports

To configure IPoIB high availability on HCA ports in a Linux host, perform the following steps:

-
- Step 1** Log in to your Linux host.
- Step 2** Display the available interfaces by entering the `ipoibcfg list` command. The following example shows how to configure IPoIB high availability between two ports on one HCA.

The following example shows how to display the available interfaces:

```
host1# /usr/local/topspin/sbin/ipoibcfg list
ib0 (P_Key 0xffff) (SL:255) (Ports: InfiniHost0/1, Active: InfiniHost0/1)
ib1 (P_Key 0xffff) (SL:255) (Ports: InfiniHost0/2, Active: InfiniHost0/2)
```

Step 3 Take the interfaces offline. You cannot merge interfaces until you bring them down.

The following example shows how to take the interfaces offline:

```
host1# ifconfig ib0 down
host1# ifconfig ib1 down
```

Step 4 Merge the two ports into one virtual IPoIB high availability port by entering the **ipoibcfg merge** command with the IB identifiers of the first and the second IB ports on the HCA.

The following example shows how to merge the two ports into one virtual IPoIB high availability port:

```
host1# /usr/local/topspin/sbin/ipoibcfg merge ib0 ib1
```

Step 5 Display the available interfaces by entering the **ipoibcfg list** command.

The following example shows how to display the available interfaces:

```
host1# /usr/local/topspin/sbin/ipoibcfg list
ib0 (P_Key 0xffff) (SL:255) (Ports: InfiniHost0/1, Active: InfiniHost0/1)
```



Note The ib1 interface no longer appears, as it is merged with ib0.

Step 6 Enable the interface by entering the **ifconfig** command with the appropriate port identifier *ib#* argument and the **up** keyword.

The following example shows how to enable the interface with the **ifconfig** command:

```
host1# ifconfig ib0 up
```

Step 7 Assign an IP address to the merged port just as you would assign an IP address to a standard interface.

Unmerging Physical Ports

To unmerge physical ports and disable active-passive IPoIB high availability, perform the following steps:

Step 1 Disable the IPoIB high availability interface that you want to unmerge by entering the **ifconfig** command with the appropriate IB interface argument and the **down** argument.

The following example shows how to unmerge by disabling the IPoIB high availability interface:

```
host1# ifconfig ib0 down
```

Step 2 Unmerge the port by entering the **ipoibcfg unmerge** command with the identifier of the port that you want to unmerge.

The following example shows how to unmerge the port:

```
host1# /usr/local/topspin/sbin/ipoibcfg unmerge ib0 ib1
```



Note After the unmerge, ib1 no longer has an IP address and needs to be configured.

Step 3 Display the available interfaces by entering the **ipoibcfg list** command.

The following example shows how to display the available interfaces:

```
host1# /usr/local/topspin/sbin/ipoibcfg list
ib0 (P_Key 0xffff) (SL:255) (Ports: InfiniHost0/1, Active: InfiniHost0/1)
ib1 (P_Key 0xffff) (SL:255) (Ports: InfiniHost0/2, Active: InfiniHost0/2)
```

Step 4 Enable the interfaces by entering the **ifconfig** command with the appropriate IB interface argument and the **up** argument.

The following example shows how to enable the interfaces:

```
host1# ifconfig ib0 up
```
