IB Diagnostic Tools (IBDIAG)

User’s Manual

Rev 0.21

IB Diagnostic Tools (IBDIAG) User's Manual

Document Number: 2378

Mellanox Technologies, Inc.
2900 Stender Way
Santa Clara, CA 95054
U.S.A.
www.Mellanox.com

Tel: (408) 970-3400
Fax: (408) 970-3403

Mellanox Technologies Ltd
PO Box 586 Hermon Building
Yokneam 20692
Israel

Tel: +972-4-909-7200
Fax: +972-4-959-3245

Table of Contents

Revision History

Table 1 - Revision History

Revision	Date	Description
0.21	September 2006	Added description of IBDIAG installation and run on Windows OS
0.20	March 2006	- The tool package was renamed from IBGDIAG (IBGranite Diagnostics) to IBDIAG (IB Diagnostic). - ibdiscover was renamed to ibdiagnet - ibtrace was renamed to ibdiagpath
0.10	September 2005	First revision

About this Manual

The IB Diagnostic Tools (IBDIAG) package described in this *User's Manual* provides the means for debugging the connectivity of an InfiniBand (IB) fabric.

This manual is organized as follows:

> Chapter 1 (page 7) provides an overview of the IB Diagnostic Tools package, software dependencies, and installation instructions.

> Chapter 2 (page 10) describes the various tools in the IB Diagnostic Tools package providing details of operation, synopses, error codes, and examples.

Intended Audience

The audience of this *User's Manual* is System Administrators who possess fair InfiniBand knowledge.

Related Documentation

- InfiniBand Architecture Specification, Volume 1, Release 1.2

1 Overview

The IB Diagnostic Tools (IBDIAG) package described in this *User's Manual* provides means for debugging the connectivity and status of InfiniBand (IB) devices in a fabric. IBDIAG can be installed and run over the Linux and Windows operating systems.

IBDIAG tools are intended to provide the following services:

- Discover the InfiniBand fabric connectivity, whether a subnet manager is running or not
- Identify links which drop packets and/or incur errors
- Identify fabric level mismatches or inconsistencies (such as Subnet Manager vs. IBDIAG fabric analysis), and internal failures (such as duplicate node/port GUIDs)

> Note: Unlike subnet managers, the IBDIAG tools do not change in any way the settings of IB devices in the fabric.

IBDIAG tools can be used either interactively or by a script. Through the tools, the user issues queries to extract real-time connectivity information and performance data about the devices in the IB fabric. The returned data by the tool commands may concern a single or multiple devices.

1.1 IBDIAG Package Contents

The IBDIAG package includes the following stand-alone tools:

- ***ibdiagnet*** - Discovers the entire network providing a listing of the following:
    - All the nodes, ports and links in the fabric
    - Link Forwarding Tables (LFT) dump file
    - Multicast Forwarding Tables (MFT) dump file
    - Fabric Subnet Managers (SMs) query file and a list of all the masked GUIDs found
    - Values of all ports Performance Monitor (PM) counters

  During the discovery process, packets are sent exhaustively, multiple times, across all the links in the fabric. This process reports links on which packets get dropped.
- ***ibdiagpath*** - Traces a path between two nodes, either by specifying the LIDs of the source and destination nodes, or by specifying a directed route. ibdiagpath provides information regarding the nodes and ports traversed. It utilizes device specific "health" queries for the different devices along the path between the source and destination nodes.

1.2 Supported Platforms and Operating Systems

For supported platforms and operating systems, please refer to the *IB Diagnostic Tools for Linux/Windows Release Notes* of your current tools version.

1.3 IBDIAG Installation

1.3.1 Software Prerequisites

On Linux: Prior to installing IBDIAG on a machine, ***ibdm*** and ***ibis*** must be installed.

On Windows: None.

1.3.2 IBDIAG Installation

1.3.2.1 Install/Uninstall IBDIAG on Linux

To check if a previous version of IBDIAG exists on your machine, enter: ‘`ibdiagnet -V`’ *or* ‘`ibdiagpath -V`’

To install IBDIAG:

1. Download the IBDIAG package for http://www.mellanox.com/products/management_tools.php
2. cd to the download directory
3. tar -zxvf IBDIAG-<revision_number>.tgz
4. Follow the instructions in the provided README file (under the download directory)

To uninstall IBDIAG, remove its installed directories: *ibdiagnetX.Y, ibdiagpathX.Y.* [1]

1.3.2.2 Install/Uninstall IBDIAG on Windows

IBDIAG for Windows is provided as part of the Mellanox Technologies WinIB software package. If WinIB is installed on your machine[2] and you wish to check whether IBDIAG is installed too, open a console window and run: ‘`ibdiagnet -V`’ *or* ‘`ibdiagpath -V`’.

In case an error is returned by the command, then IBDIAG is not installed on your machine. To install it, activate the WinIB Install Wizard (.msi file), choose ‘Custom’ installation, and select the following components:

- Driver
- IBDIAG

In case IBDIAG *is* installed but you wish to update your machine with a newer version, you need to first remove *ibdiagnetX.Y, ibdiagpathX.Y* [3] using the ‘Add or Remove Programs’ utility of Windows, then perform the ‘Custom’ installation steps described above.

In case WinIB is not installed on your machine, download it via www.mellanox.com and perform either a ‘Custom’ or a ‘Complete’ installation. If you choose a complete installation of WinIB will install IBDIAG in the process. If you choose a custom installation, make sure you select the components as described above.

1.4 Run-time Software Dependency

The IBDIAG tools use the “OpenSM vendor” library to send and receive InfiniBand Management Datagram packets (MADs). Therefore, OpenSM should be installed (but not necessarily running) on the machine. OpenSM can be installed on top of the VAPI driver and the Linux OpenFabrics distributions (e.g., IBGD, IBG2/IBED). In order to send and receive IB MADs from a user level application, the driver should load its user level MAD access modules. These modules are loaded by default by all the aforementioned drivers, except for VAPI. For the VAPI driver, the ib_mgt module should be loaded.

1.5 Key Concepts And Terms

Throughout this manual, there is frequent reference to various concepts and terms which are common to the general audience of System Administrators. However, there is also reference to InfiniBand-specific concepts and terms, a part of which are listed and briefly defined in the following list and throughout the document.

---

[1] X.Y stands for the tools versions.
[2] That is, you can locate it using the ‘Add or Remove Programs’ utility of Windows.
[3] X.Y stands for the tools versions.

- **IB devices:** Integrated Circuits implementing InfiniBand compliant communication.
- **IB Fabric/Cluster/Subnet:** A set of IB devices connected by IB cables.
- **In-Band:** A term assigned to administration activities traversing the IB connectivity only.
- **Channel Adapter (CA):** An IB device that terminates an IB link and executes transport functions. This may be an HCA (Host CA) or a TCA (Target CA).
- **Local device/node/system:** The IB Host Channel Adapter (HCA) Card installed on the machine running IBDIAG tools.
- **Local port:** The IB port of the HCA through which IBDIAG tools connect to the IB fabric.

2 Tools Usage

This chapter first describes common configuration, interface, and addressing for all the tools in the package (Section 2.1). Then it provides detailed descriptions of the tools themselves including: operation, synopsis and options descriptions, error codes, and examples (Section 2.2).

2.1 Common Configuration, Interface and Addressing

2.1.1 Topology File (Optional)

An InfiniBand fabric is composed of switches and channel adapter (HCA/TCA) devices. To identify devices in a fabric (or even in one switch system), each device is given a GUID (a MAC equivalent). Since a GUID is a non-user-friendly string of characters, it is better to alias it to a meaningful, user-given name. For this objective, the IB Diagnostic Tools can be provided with a "topology file", which is an *optional* configuration file specifying the IB fabric topology in *user-given* names.

> Note: For IBDIAG tools to fully support the topology file, the user may need to provide the local system name (if the local hostname is not used in the topology file).

To specify a topology file to an IBDIAG tool use one of the following two options:

1. On the command line, specify the file name using the option (see Section 2.2, "Tools Descriptions," on page 11):
   `-t <topology file name>`
2. Define the environment variable IBDIAG_TOPO_FILE

To specify the local system name to an IBDIAG tool use one of the following two options:

1. On the command line, specify the system name using the option (see Section 2.2, "Tools Descriptions," on page 11):
   `-s <local system name>`
2. Define the environment variable IBDIAG_SYS_NAME

> Note: For a detailed description of the topology file and for instructions on how to create it, please refer to the *IBADM User's Manual, Document no. 2130UM.*

2.1.2 IB Interface Definition

The IBDIAG tools installed on a machine connect to the IB fabric by means of an HCA (host channel adapter) IB port through which they send MADs. To specify this port to an IBDIAG tool use one of the following options:

1. On the command line, specify the port number using the option (see Section 2.2, "Tools Descriptions," on page 11):
   `-p <local port number>`
2. Define the environment variable IBDIAG_PORT_NUM

In case more than one HCA device is installed on the local machine, it is necessary to specify the device's index to the tool as well. For this use on of the following options:

1. On the command line, specify the index of the local device using the following option (see Section 2.2, “Tools Descriptions,” on page 11):

```
-i <index of local device>
```

2. Define the environment variable IBDIAG_DEV_IDX

2.1.3 Addressing

Note: This section applies to the **ibdiagpath** tool only.

A tool command may require defining the destination device or port to which it applies. The following addressing modes can be used to define the IB ports:

- Using a Directed Route to the destination: (Tool option ‘-d’)

  This option defines a directed route of *output* port numbers from the local port to the destination.
- Using port LIDs: (Tool option ‘-l’):

  In this mode, the source and destination ports are defined by means of their LIDs. If the fabric is configured to allow multiple LIDs per port, then using any of them is valid for defining a port. (See Section 2.2.2 on page 16 for details).
- Using port names defined in the topology file: (Tool option ‘-n’)

  This option refers to the source and destination ports by the names defined in the topology file. (Therefore, this option is relevant only if a topology file is specified to the tool.) In this mode, the tool uses the names to extract the port LIDs from the matched topology, then the tool operates as in the ‘-l’ option.

2.2 Tools Descriptions

2.2.1 ibdiagnet

ibdiagnet scans the fabric using directed route packets and extracts all the available information regarding its connectivity and devices. It then produces the following files in the output directory defined by the ‘-o’ option (see synopsis below):

`ibdiagnet.lst`	`List of all the nodes, ports and links in the fabric`
`ibdiagnet.fdbs`	`A dump of the unicast forwarding tables of the fabric switches`
`ibdiagnet.mcfdbs`	`A dump of the multicast forwarding tables of the fabric switches`
`ibdiagnet.log`	`A log of ibdiagnet command output messages (the same messages displayed to stdout)`
`ibdiagnet.masks`	`List of all assigned Mask GUIDs to the duplicate GUIDs in the IB fabric`
`ibdiagnet.sm`	`List of all the fabric Subnet Managers (SMs) including SM state and priority`
`ibdiagnet.pm`	`List of all the fabric ports PM counters values (with ‘-pm’ option only)`

In addition to generating the files above, the discovery phase also checks for the following internal errors and reports them to the standard output:

- Duplicate node GUIDs
- Duplicate port GUIDs
- Duplicate LIDs
- LIDs with value zero

- System GUIDs with value zero
- The InfiniBand connection link is in INIT state which does not permit data transfer

After the discovery phase is completed, **ibdiagnet** performs the following steps:

1. Reads PM counters values of all the IB fabric ports
2. Sends directed route packets multiple times (see the ‘-c’ option below) on all the fabric links
3. Uses the results from step 2 to detect possible problematic paths on which packets may be lost. Those paths are explored, and a report of the suspected bad links is displayed on the standard output.
4. PM counters values are read again and compared to the values read in step 1. **ibdiagnet** reports each PM counter that reached its limit (overflowed) or increased significantly.

After scanning the fabric, and if the ‘-r’ option is provided, a full report of the fabric qualities is displayed. This report includes:

- FDBS and MCFDBS files analysis
- Number of nodes and systems
- Hop-count information: maximal hop-count, an example path, and a hop-count histogram
- All CA-to-CA paths traced

  Note: In case the IB fabric includes only one CA, then CA-to-CA paths are not reported.
- Multicast groups table

Furthermore, if a topology file is provided, ibdiagnet uses the names defined in it for the output reports.

Synopsis:

ibdiagnet [-c <count>] [-v] [-r] [-t <topo-file>] [-s <sys-name>] [-i <dev-index>] [-p <port-num>] [-o <out-dir>] [-lw <1x|4x|12x>] [-ls <2.5|5|10>] [-pm] [-pc]

where:

```
-c                Defines the minimal number of packets to be sent across each
                  link. Default: 10
-v                Instructs the tool to run in verbose mode
-r                Provides a complete report of the fabric qualities
-t                Specified the topology file name (see Section 2.1.1 on
                  page 10)
-s                Specifies the local system name. Meaningful only if a topology
                  file is specified (see Section 2.1.1 on page 10).
-i                Specifies the index of the device of the port used to connect
                  to the IB fabric. To be used in case of multiple devices on
                  the local system (see Section 2.1.2 on page 10).
-p                Specifies the local device’s port number used to connect to
                  the IB fabric (see Section 2.1.2 on page 10).
-o                The directory where the output files are placed. Default: /tmp
-lw               Specifies the expected link width
-ls               Specifies the expected link speed
-pm               Dumps all PM counters values into ibdiagnet.pm
-pc               Resets all fabric ports PM counters
-h|--help         Prints this help information
-V|--version      Prints the version of the tool
--vars            Prints the tool’s environment variables and their values
```

Error Codes:

Table 2 - Error Codes Table

Error Code	Description
1	Failed to fully discover the fabric
2	Failed to parse the command line options
3	Failed to interact with IB fabric
4	Failed use local device or local port
5	Failed to use topology file
6	Failed to load required package

Run Examples:

Example 1: No Topology File Specified

```
> Ibdiagnet -o outputDirectory
Loading IBDIAGNET from: /usr/lib/tools/ibdiag/app-ibdiagnet1.0
Loading IBDM from: /usr/lib/ibdm1.0
-W- Topology file is not specified.
    Reports regarding cluster links will use direct routes.
-I- Using port 2 as the local port.
-I- Discovering the subnet ... 30 nodes (10 Switches & 20 CA-s) discovered.

-I---------------------------------------------------
-I- Bad Guids Info
-I---------------------------------------------------
-I- No bad Guids were found

-I---------------------------------------------------
-I- Links With Logical State = INIT
-I---------------------------------------------------
-I- No bad Links (with logical state = INIT) were found

-I---------------------------------------------------
-I- PM Counters Info
-I---------------------------------------------------
-I- No illegal PM counters values were found

-I---------------------------------------------------
-I- Links With links width != 4x (as set by -lw option)
-I---------------------------------------------------
```

```
-I- No unmatched Links (with width != 4x) were found

-I---------------------------------------------------
-I- Links With links speed != 5 (as set by -ls option)
-I---------------------------------------------------
-I- No unmatched Links (with speed != 5) were found

-I---------------------------------------------------
-I- Bad Links Info
-I---------------------------------------------------
-I- No bad link were found

-I- Done. Run time was 1 seconds.
```

Example 2: A Topology File is Specified - Output Has No Errors

```
> ibdiagnet -t /etc/ibadm.topo -s H-1 -p 2
Loading IBDIAGNET from: /usr/local/lib/ibdiagnet1.0
Loading IBDM from: /usr/local/lib/ibdm1.0
-I- Parsing topology definition:/etc/ibadm.topo
-I- Defined 14/29 systems/nodes
-I- Discovering the subnet ... 29 nodes (24 Switches & 5 CA-s) discovered.

-I- Parsing Subnet file:/tmp/ibdiagnet.lst
-I- Defined 14/29 systems/nodes

-I---------------------------------------------------
-I- Bad Guids Info
-I---------------------------------------------------
-I- No bad Guids were found

-I---------------------------------------------------
-I- Links With Logical State = INIT
-I---------------------------------------------------
-I- No bad Links (with logical state = INIT) were found

-I---------------------------------------------------
-I- PM Counters Info
-I---------------------------------------------------
-I- No illegal PM counters values were found

-I---------------------------------------------------
-I- Bad Links Info
-I---------------------------------------------------
-I- No bad link were found
```

```
-I---------------------------------------------------
-I- Topology matching results
-I---------------------------------------------------
-I- The topology defined in /etc/ibadm.topo perfectly matches the discovered
    fabric.

-I- Done. Run time was 1 seconds.
```

Example 3: A Topology File is Specified - Output Has Errors

```
> ibdiagnet -t topology/ibadm.topo -s swlab178
Loading IBDIAGNET from: /usr/lib/ibdiag/app-ibdiagnet1.0
Loading IBDM from: /usr/lib/ibdm1.0
-I- Parsing topology definition:topology/ibadm.topo
-I- Defined 14/31 systems/nodes

I- Discovering the subnet ... 30 nodes (24 Switches & 6 CA-s) discovered.

-I- Parsing Subnet file:/tmp/ibdiagnet.lst
-I- Defined 11/30 systems/nodes

-I---------------------------------------------------
-I- Bad Guids Info
-I---------------------------------------------------
-E- #2 Devices with LID = 0x0000 found in the fabric:
a HCA "mtlmd12/U1/P1" at direct path="1,13,1,10,8,8,17"
a HCA "mtlmd12/U1/P2" at direct path="1,13,1,10,8,8,18"
-W- #4 Devices with SystemGUID = 0x0000000000000000 found in the fabric:
a Switch "Buffalo1/U1/P0" at direct path="1,13,1,10"
a Switch "Stallion1/U1/P0" at direct path="1,13,18,10"
a Switch "Reindeer2/U1/P0" at direct path="1,13,1,10,8"
a Switch "Reindeer1/U1/P0" at direct path="1,13,1,10,8,8"
-E- #2 Devices with identical NodeGUID = 0x1234567812345678 found in the fabric:
a Switch "Reindeer1/U1/P0" at direct path="1,13,1,10,8,8"
a HCA "mtlmd12/U1/P1" at direct path="1,13,1,10,8,8,17"
-E- #2 Devices with identical PortGUID = 0x1234567812345678 found in the fabric:
a Switch "Reindeer1/U1/P0" at direct path="1,13,1,10,8,8"
a HCA "mtlmd12/U1/P1" at direct path="1,13,1,10,8,8,17"

-I---------------------------------------------------
-I- Link Logical State Info
-I---------------------------------------------------
W- link with LOG=INI found at direct path "1,13,1,10,8,8,17"
From : a Switch "Reindeer1/U1"  GUID=0x1234567812345678 Port=2
To : a HCA "mtlmd12/U1" GUID=0x1234567812345678 Port=1

-I---------------------------------------------------
```

```
-I- Bad Fabric SM Info
-I---------------------------------------------------
-E- Missing master SM in the discover fabric



-I---------------------------------------------------
-I- Bad Links Info
-I---------------------------------------------------
-I- Errors have occurred on the following links
(for errors details, look in log file /tmp/ibdiagnet.log):
Link at the end of direct route "1,13,18,10,8" "swlab223/U1"

-I---------------------------------------------------
-I- Topology matching results
-I---------------------------------------------------
-I- Note that some "bad" links and the part of the fabric to which they lead (in
the BFS discovery of the fabric, starting at the local node) are not
discovered and therefore will be reported as "missing".

Missing System:swlab223(MT25208)
Should be connected by cable from port: P2(swlab223/U1/P2)
to:Stallion1/P8(Stallion1/U1/P8)

-I- Done. Run time was 7 seconds.
```

2.2.2 ibdiagpath

ibdiagpath traces a path between two end-points and provides information regarding the nodes and ports traversed along the path. It utilizes device specific health queries for the different devices along the path.

The way ibdiagpath operates depends on the addressing mode used on the command line. If directed route addressing is used (-d flag), the local node is the source node and the route to the destination port is known apriori. On the other hand, if LID-route (or by-name) addressing is employed, then the source and destination ports of a route are specified by their LIDs (or by the names defined in the topology file). In this case, the actual path from the local port to the source port, and from the source port to the destination port, is defined by means of Subnet Management Linear Forwarding Table queries of the switch nodes along that path. Therefore, the path cannot be predicted as it may change.

ibdiagpath should not be supplied with contradicting local ports by the -p and -d flags (see synapsis descriptions below). In other words, when ibdiagpath is provided with the options -p and -d together, the first port in the direct route must be equal to the one specified in the “-p” option. Otherwise, an error is reported.

> Note: When ibdiagpath queries for the performance counters along the path between the source and destination ports, it always traverses the LID route, even if a directed route is specified. If along the LID route one or more links are not in the ACTIVE state, ibdiagpath reports an error.

Using **ibdiagpath** with contradicting local ports to be used	If ibdiagpath is provided with the options **-p** and **-d** together, the first port in the direct route must be equal to the one specified in the "**-p**" option. Otherwise, an error is reported.

Moreover, the tool allows omitting the source node in LID-route addressing, in which case the local port on the machine running the tool is assumed to be the source.

Synopsis:

ibdiagpath {-l <src-lid,dst-lid>|-d <p1,p2,p3…>|-n <src-name,dst-name>} [-c] [-o <out-dir>] [-v] [-t <topo-file>]
[-s <sys-name>] [-i <hca-idx>] [-p <port-num>] [-lw <1x|4x|12x>] [-ls <2.5|5|10>] [-pm] [-pc]
[-h|--help] [-V|--version] [--vars]

where:

`-l`	`Source and destination LIDs (source may be omitted --> local port is assumed to be the source)`
`-d`	`Directed route of output ports from the local node (which is considered the source) to the destination`
`-n`	`Names of the source and destination ports (as defined in the topology file)`
`-c`	`Number of iterations to perform. Default: 100`
`-o`	`The directory where the output files are placed. Default: /tmp`
`-v`	`Instructs the tool to run in verbose mode`
`-t`	`Specifies the topology file name (see Section 2.1.1 on page 10)`
`-s`	`Specifies the local system name. Meaningful only if a topology file is specified (see Section 2.1.1 on page 10).`
`-p`	`Specifies the local device's port number used to connect to the IB fabric (see Section 2.1.2 on page 10).`
`-i`	`Specifies the index of the device of the port used to connect to the IB fabric. To be used in case of multiple devices on the local system (see Section 2.1.2 on page 10).`
`-lw`	`Specifies the expected link width`
`-ls`	`Specifies the expected link speed`
`-pm`	`Dumps all PM counters values into ibdiagnet.pm`
`-pc`	`Resets all fabric ports PM counters`
`-h|--help`	`Prints this help information`
`-V|--version`	`Prints the version of the tool`
`--vars`	`Prints the tool's environment variables and their values`

Error Codes:

Table 3 - Error Codes Table

Error Code	Description
1	The path traced is unhealthy
2	Failed to parse command line options

Table 3 - Error Codes Table

Error Code	Description
3	More than 64 hops are required for traversing the local port to the source port and then to the destination port
4	Unable to traverse the LFT data from source to destination
5	Failed to use topology file
6	Failed to load required package

Run Examples:

Note: The verbose format provided below may be truncated from the original longer format.

Example 1: Using Directed Route

```
> ibdiagpath -d 1,14,1,10,8,8,15
Loading IBDIAGPATH from: /usr/lib/ibdiag/app-ibdiagpath1.0
Loading IBDM from: /usr/lib/ibdm1.0
-W- Topology file is not specified.
    Reports regarding cluster links will use direct routes.
-I- Using port 1 as the local port (since this is the output port of the direct
    route).

-I-----------------------------------------------------
-I- Traversing the path from local to destination port
-I-----------------------------------------------------
-I- From: swlab178/P1 lid=0x0005 guid=0x0002c90200200000 dev=25204
-I- To:   lid=0x0030 guid=0x0000008880abcd0 dev=47396 port=8

-I- From: lid=0x0030 guid=0x0000008880abcd0 dev=47396 port=14
-I- To:   lid=0x0014 guid=0x0002c9010afcfec8 dev=47396 port=14

-I- From: lid=0x0014 guid=0x0002c9010afcfec8 dev=47396 port=1
-I- To:   lid=0x002b guid=0x00000003330abcd0 dev=47396 port=14

-I- From: lid=0x002b guid=0x00000003330abcd0 dev=47396 port=10
-I- To:   lid=0x0002 guid=0x0002c901051d1c01 dev=43132 port=3

-I- From: lid=0x0002 guid=0x0002c901051d1c01 dev=43132 port=8
-I- To:   lid=0x005a guid=0x0000000000dadada dev=47396 port=23

-I- From: lid=0x005a guid=0x0000000000dadada dev=47396 port=8
-I- To:   lid=0x000b guid=0x1234567812345678 dev=47396 port=21

-I- From: lid=0x000b guid=0x1234567812345678 dev=47396 port=15
-I- To:   mtlmd11/P1 lid=0x0001 guid=0x0000000000000011 dev=23108
```

```
-I---------------------------------------------------
-I- Validating path health
-I---------------------------------------------------

-I- Done. Run time was 0 seconds..
```

Example 2: Non-verbose Mode Using a LID Route and Port #2 as Exit Port (Sets Both Source LID and Destination LID):

```
> ibdiagpath -l 0x05,0x010 -p 2
Loading IBDIAGPATH from: /usr/lib/ibdiag/app-ibdiagpath1.0
Loading IBDM from: /usr/lib/ibdm1.0
-W- Topology file is not specified.
    Reports regarding cluster links will use direct routes.

-I---------------------------------------------------
-I- Traversing the path from local to source port
-I---------------------------------------------------
-I- From: mtlmd11/P2 lid=0x000a guid=0x0000000000000011 dev=23108
-I- To:   lid=0x000b guid=0x1234567812345678 dev=47396 port=13

-I- From: lid=0x000b guid=0x1234567812345678 dev=47396 port=21
-I- To:   lid=0x005a guid=0x0000000000dadada dev=47396 port=8

-I- From: lid=0x005a guid=0x0000000000dadada dev=47396 port=23
-I- To:   lid=0x0002 guid=0x0002c901051d1c01 dev=43132 port=8

-I- From: lid=0x0002 guid=0x0002c901051d1c01 dev=43132 port=3
-I- To:   lid=0x002b guid=0x0000003330abcd0 dev=47396 port=10

-I- From: lid=0x002b guid=0x0000003330abcd0 dev=47396 port=13
-I- To:   lid=0x0015 guid=0x0002c9010afcfed0 dev=47396 port=1

-I- From: lid=0x0015 guid=0x0002c9010afcfed0 dev=47396 port=11
-I- To:   lid=0x0030 guid=0x00000008880abcd0 dev=47396 port=16

-I- From: lid=0x0030 guid=0x00000008880abcd0 dev=47396 port=8
-I- To:   swlab178/P1 lid=0x0005 guid=0x0002c90200200000 dev=25204


-I---------------------------------------------------
-I- Traversing the path from source to destination port
-I---------------------------------------------------
-I- From: lid=0x0030 guid=0x00000008880abcd0 dev=47396 port=13
-I- To:   lid=0x0015 guid=0x0002c9010afcfed0 dev=47396 port=14

-I- From: lid=0x0015 guid=0x0002c9010afcfed0 dev=47396 port=16
```

```
-I- To:   lid=0x002c guid=0x00000004440abcd0 dev=47396 port=16

-I- From: lid=0x002c guid=0x00000004440abcd0 dev=47396 port=10
-I- To:   lid=0x000d guid=0x12345678aa123456 dev=47396 port=24

-I- From: lid=0x000d guid=0x12345678aa123456 dev=47396 port=18
-I- To:   swcl40/P2 lid=0x0010 guid=0x0002c90200203f0c dev=25218



-I---------------------------------------------------
-I- Validating path health
-I---------------------------------------------------
-E-  lid:0x0010 port:2: symbol_error_counter=0xffff(=overflow)

-I- Done. Run time was 0 seconds.
```

Example 3: Using Named Route in Verbose Mode (Sets Both Source LID and Destination LID)

```
> ibdiagpath -t topology/ibadm.topo -n Buffalo1,swlab178 -v
Loading IBDIAGPATH from: /usr/lib/ibdiag/app-ibdiagpath1.0
Loading IBDM from: /usr/lib/ibdm1.0
-I- Parsing topology definition:topology/ibadm.topo
-I- Defined 14/31 systems/nodes
-W- Since local system name was not specified (-s flag), its set to mtlmd11/U1.
-W- A few ports of {node Buffalo1/U1} (specified by -n flag) are connected to
    the fabric; one of them - port 3 - is used.
-I- Obtaining source and destination LIDs:
    Buffalo1/P3         LID=0x0002
    swlab178/P1         LID=0x0005

-I---------------------------------------------------
-I- Traversing the path from local to source port
-I---------------------------------------------------
-I- From: "mtlmd11/U1/P1" lid=0x0001 guid=0x0000000000000011 dev=23108
-I- To:   "Reindeer1/U1/P15" lid=0x000b guid=0x1234567812345678 dev=47396

-I- From: "Reindeer1/U1/P21" lid=0x000b guid=0x1234567812345678 dev=47396
-I- To:   "Reindeer2/U1/P8" lid=0x005a guid=0x0000000000dadada dev=47396

-I- From: "Reindeer2/U1/P23" lid=0x005a guid=0x0000000000dadada dev=47396
-I- To:   "Buffalo1/U1/P8" lid=0x0002 guid=0x0002c901051d1c01 dev=43132



-I---------------------------------------------------
-I- Traversing the path from source to destination port
-I---------------------------------------------------
-I- From: "Buffalo1/U1/P3" lid=0x0002 guid=0x0002c901051d1c01 dev=43132
```

```
-I- To:   "Rhino_SW/leaf3/U1/P10" lid=0x002b guid=0x00000003330abcd0 dev=47396

-I- From: "Rhino_SW/leaf3/U1/P13" lid=0x002b guid=0x00000003330abcd0 dev=47396
-I- To:   "Rhino_SW/spine2/U3/P1" lid=0x0015 guid=0x0002c9010afcfed0 dev=47396

-I- From: "Rhino_SW/spine2/U3/P11" lid=0x0015 guid=0x0002c9010afcfed0 dev=47396
-I- To:   "Rhino_SW/leaf8/U1/P16" lid=0x0030 guid=0x00000008880abcd0 dev=47396

-I- From: "Rhino_SW/leaf8/U1/P8" lid=0x0030 guid=0x00000008880abcd0 dev=47396
-I- To:   "swlab178/U1/P1" lid=0x0005 guid=0x0002c90200200000 dev=25204


-I----------------------------------------------------
-I- Validating path health
-I----------------------------------------------------
-V- PM  lid:0x0002 port:3  0 0 0 0 0 0 0 0 0 0 0 0 0 0 504 504 7 7
-V- PM  lid:0x002b port:10  0 0 0 0 0 0 0 0 0 0 0 0 0 0 504 504 7 7
-V- PM  lid:0x002b port:13  0 0 0 0 0 0 0 0 0 0 0 0 0 0 72 144 1 2
-V- PM  lid:0x0015 port:1  0 0 0 0 0 0 0 0 0 0 0 0 0 0 144 72 2 1
-V- PM  lid:0x0015 port:11  0 0 0 0 0 0 0 0 0 0 0 0 0 0 72 0 1 0
-V- PM  lid:0x0030 port:16  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 72 0 1
-V- PM  lid:0x0030 port:8  0 0 0 0 0 0 0 0 0 0 0 0 0 0 72 72 1 1
-V- PM  lid:0x0005 port:1  0 0 0 0 0 0 0 0 0 0 0 0 0 0 72 72 1 1
-E-  lid:0x0002 port:3: excesive_buffer_errors=0xf(=overflow)

-I- Done. Run time was 1 seconds.
```