Release Notes

OpenSM

Rev 1.7.0

OpenSM Release Notes

Document Number:

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

1 Overview

This document describes the contents of the OpenSM Rev 1.7.0 release. OpenSM is an InfiniBand compliant Subnet Manager and Administrator, and runs on top of OpenIB. It is provided in two flavors: a fixed flow executable named ***opensm***, and a configureable flow and policy under Tcl **osm package**. The two are accompanied by a testing application named ***osmtest***. Further documentation of the tools is provided in *OpenSM User's Manual, Document no. 2277UM*.

The document includes the following sections:

1.1 New Features

- OpenSM now reports the IB link speed (2.5Gbps/5Gbps/10Gbps) in the subnet.lst file and the verbose display of the subnet inventory.
- OpenSM supports Client-Reregistration as defined in the latest *InfiniBand Architecture Specification, Vol. 1, Release 1.2*.
- OpenSM retry timeout is now an exponential function of the retry iteration. Thus it is 1*ui_timeout for the first retry, 4*ui_timeout for the second, 9*ui_timeout for the third, and 16*ui_timeout for the fourth (and last).

1.2 Software Dependencies

OpenSM depends on the installation of OpenIB stack (pointed at by the TSHOME and MTHOME environment variables). The qualified driver versions are provided in Table 1, "Software Dependencies".

Table 1 - Software Dependencies

Task	Software	Supported Versions
HCA Driver & Special QP Management	OpenIB Stack	0.0.1 and later
HCA Driver & Special QP Management	Mellanox Infiniband HCA Driver	3.2 and later

1.3 Supported Platforms And Operating Systems

The following table lists all supported platforms and operating systems by the tools included in this package.

Table 2 - Supported Platforms and Operating Systems

Platform	Operating System	Kernel
X86	Red Hat Enterprise Linux AS 3.0	2.4.21-20.ELsmp
	Red Hat Linux 9.0	kernel.org: 2.4.27 (smp)
	Red Hat Linux 9.0	2.4.20-8 (smp; bigmem)
	SuSe SLES 9.0	Update (2.6.5-7.111.xx-smp)
	SuSE Linux 9.1 Pro	2.6.9 / 2.6.10
	SuSE Linux 9.1 Pro	Update (2.6.5-7.111.xx-smp)
	Rocks 3.3.0	2.4.21-20.ELsmp
	Fedora Core 3	vanilla 2.6.9 (from kernel.org)
IA-64	Red Hat Enterprise Linux AS 3.0	2.4.21-15.EL
	SuSE SLES 9.0	2.6.5-7.97-default
AMD64 (X86_64)	Red Hat Enterprise Linux AS 3.0	2.4.21-20.ELsmp
	SuSE SLES 9.0	2.6.5-7.111.xx-smp
	SuSE 9.1 Pro	Update (2.6.5-7.111.xx-smp)
	SuSE 9.1 Pro	2.6.9/2.6.10
	Rocks 3.3.0	2.4.21-20.ELsmp
	Fedora Core 3	2.6.9-1.667smp
Intel EM64T	Red Hat Enterprise Linux AS 3.0	2.4.21-20.EL
	SuSE SLES 9.0 RC5	2.6.5-7.97-smp / 2.6.5-7.111.xx-smp
	SuSE 9.1 Pro	2.6.10
	SuSE 9.1 Pro	Update (2.6.5-7.111.xx-smp)
	Rocks 3.3.0	2.4.21-20.ELsmp
	Fedora Core 3	2.6.9-1.667smp

1.4 Supported Firmware

The main task of OpenSM is to initialize InfiniBand devices. The devices and their corresponding firmware versions which were qualified using OpenSM are listed in Table 3 below.

Table 3 - Devices and Corresponding Firmware Qualified with OpenSM

Device	FW versions qualified
MT43132 InfiniScale	5.2.0 (and later)
MT47396 InfiniScale III	0.3.2 (and later)
MT23108 InfiniHost	3.2.0
MT25208 InfiniHost III Ex	4.5.0 (and later)

2 Known Issues And Limitations

Known issues and limitations of OpenSM are described in the following table.

Table 4 - OpenSM Known Issues And Limitations

Issue/Limitation Description	Impacted Platforms	Impact
No Pkey update policy	All	OpenSM does not enable the configuration of Pkey Tables on the subnet.
IB "trusted" concept is unsupported	All	Queries that should be classified according to the trustworthiness of their sources will not be handled correctly.
No Service / Key associations	All	There is no way to manage Service access by Keys.
No SM to SM SMDB synchronization	All	Puts the burden of re-registering services, multicast groups, and inform-info on the client application.
No support for multiple HCA cards on the same host	All	When using more than one HCA card on the same host, OpenSM is not able to recognize IB port state correctly. Multiple HCA card support will be fixed in the next release.
NPTL problem under Red Hat 9.0, Red Hat AS 3.0	All	There are problems with thread handling when using the dynamic Native POSIX Thread Library (/lib/tls) of Red Hat 9.0 & Red Hat AS 3.0 OSs. To overcome the problems, the OpenSM installation places the wrapper scripts *opensm* and *osmtest* in the /usr/bin directory, which preload the standard libc and libpthread before invoking the executables. If using the osm package, a similar workaround is possible by putting the LD_PRELOAD setting in .tclshrc file, for example: set env (LD_PRELOAD) "/lib/libc.so.6:/lib/libpthread.so.0"
InformInfo failure over Mellanox HCA driver	SUSE SLES 9, SUSE 9.1, Red Hat AS 3 Update 2, 2.4.27 (kernel.org)	OpenSM might not respect a valid InformInfo unsubscribe request when running over Mellanox's IBMGT user level MAD interface (not on IBGD). This will be fixed in the next release.
Changing the switch port through which OpenSM connects to the IB fabric may cause wrong operation	All	On a cluster with at least one switch system: If during OpenSM operation it gets disconnected from one switch port and connected to another, OpenSM may fail to correctly setup the fabric. Please restart OpenSM whenever such a connectivity change is made.

3 Unsupported IB Compliancy Statements

The following table lists all the IB compliancy statements which OpenSM does not support. Please refer to IB specification for detailed information on compliancy.

Table 5 - OpenSM Unsupported Compliancy Statements

Flow	Compliancy	Description
Authentication	C14-22	M_Key M_KeyProtectBits and M_KeyLeasePeriod shall be set in one SubnSet method. As a work-around, an OpenSM option is provided for defining the protect bits.
Authentication	C14-67	On SubnGet(SMInfo) and SubnSet(SMInfo) - if M_Key is not zero then the SM shall generate a SubnGetResp if the M_Key is matching or silently drop the packet if M_Key is not matching
Authentication	C15-0.1.23.1	PortInfoRecords shall always be provided with the M_Key component set to 0, except in the case of a trusted request, in which case the actual M_Key component contents shall be provided.
Authentication	C15-0.1.23.2	P_KeyTableRecords and ServiceAssociationRecords shall only be provided in responses to trusted requests.
Authentication	C15-0.1.23.4	InformInfoRecords shall always be provided with the QPN set to 0, except for the case of a trusted request, in which case the actual subscriber QPN shall be returned.
Event FWD	o13-17.1.2	If no permission to forward, the subscription should be removed and no further forwarding should occur
Handover	C14-37.1.2	Priority should be kept in non-volatile memory.
Handover	C14-38.1.1	Support AttributeModifier values in SubnSet(SMInfo). If the state transition requested is invalid - return with status code 7
Initialization	C14-24.1.1.5	GUIDInfo - SM should enable assigning Port GUIDInfo
Initialization	C14-44	If the SM discovers that it is missing an M_Key to update CA/RT/SW, it should notify the higher level.
Initialization	C14-62.1.1.11	PortInfo:VLHighLimit should match the configured VLArb on this port
Initialization	C14-62.1.1.12	PortInfo:M_Key - Set the M_Key to a node based random value
Initialization	C14-62.1.1.13	PortInfo:P_KeyProtectBits - set according to an optional policy
Initialization	C14-62.1.1.14	PortInfo:M_KeyLeasePeriod - set according to optional policy
Initialization	C14-62.1.1.22	GUIDInfo - SM should enable assigning Port GUIDInfo
Initialization	C14-62.1.1.24	SwitchInfo:DefaultPort - Not relevant, works only for random FDB
Initialization	C14-62.1.1.32	RandomForwardingTable
Multicast	o15-0.1.12	If the JoinState is SendOnlyNonMember = 1 (only), then the endport should join as sender only
Multicast	o15-0.1.13	If a Join request using unrealistic parameters is received, return ERR_REQ_INVALID
Multicast	o15-0.1.8	If a request for creating an MCG with fields that cannot be met, return ERR_REQ_INVALID (SL FlowLabelTclass)

Table 5 - OpenSM Unsupported Compliancy Statements

Flow	Compliancy	Description
SA Query	C15-0.1.11	Query response should use only base lids (as the feature has not been qualified yet).
SA Query	C15-0.1.19	Respond to SubnGetMulti(MultiPathRec)
SA Query	C15-0.1.8.6	Respond to SubnAdmGetTraceTable
SA Query	C15-0.1.8.7	SubnAdmGetMulti SubnAdmGetMultiResp - Only in case of a MultiPath
SA Query		SubAdmGet/GetTable(GUIDInfo)
Services	C15-0.1.13	Reject ServiceRecord create, modify or delete if the given ServiceP_Key does not match the one included in the ServiceGID port and the port that sent the request
Services	C15-0.1.14	Provide means to associate service name and ServiceKeys

4 Major Bug Fixes

Table 6 - Bug Fixes Summary

Index	Desciption	Details
1.	Re-calculation of the multicast tree and switch re-programming upon a new join/ leave request adds a large overhead to run-time of large IB fabrics	OpenSM now lumps re-calculation and re-programming into groups. All join/leave requests arriving while a certain group of requests is being processed are lumped into the ‘next’ group to be processed.
2.	Default HOQ Lifetime Limit changed from infinity to 1sec	To prevent cases of fabric deadlock due to routing changes, host software failure, or hardware failure, a finite 1sec Head Of Queue Lifetime Limit is now the default used by OpenSM. If a packet is stalled for more than this time at the head of any IB transmit queue, it will be dropped.
3.	SM LID is now updated periodically by the SA Client interface	The OpenSM vendor SA client interface did not update the SM LID after it was started. This impacted client applications using the interface after a change in the SM LID.
4.	OpenSM vendor layer does not exit and an application using it remains stuck	This was due to races during the destruction of the OpenSM Vendor Layer
5.	PKey lookup may cause OpenSM to hang	This was due to a PKey lookup problem that causes an infinite loop
6.	Memory leaks in Up/Down router	Fixed

5 Main Verification Flows

OsmTest is the main automated verification tool used for OpenSM testing. Its verification flows are described in Table 7 below.

Table 7 - OsmTest Verification Flows

Test Flow	Verified Compliancy Statement
Plugfest SM Compli-ancy Tests	Plugfest SM compliancy tests per IBTA Test Specification www.infinibandta.org
Inventory File	All port info, node info, and path records parameters
Service Record	- Register a service - Register another service (with a lease period) - Register another service (with service p_key set to zero) - Get all services by name - Delete the first service - Delete the third service. - Added bad flows of get/delete non valid service - Add / Get same service with different data - Add / Get / Delete by different component mask values (services by Name & Key / Name & Data / Name & Id / Id only)
Multicast Member Record	- Query of existing Groups (IPoIB) - BAD Join with insufficient comp mask (o15.0.1.3) - Create given MGID=0 (o15.0.1.4) - Create given MGID=0xFF12A01C,FE800000,00000000,12345678 (o15.0.1.4) - Create BAD MGID=0xFA. (o15.0.1.6) - Create BAD MGID=0xFF12A01B w/ link-local not set (o15.0.1.6) - New MGID with invalid join state (o15.0.1.9) - Retry of existing MGID - See JoinState update (o15.0.1.11) - BAD RATE when connecting to existing MGID (o15.0.1.13) - Partial JoinState delete request - removing FullMember (o15.0.1.14) - Full Delete of a group (o15.0.1.14) - Verify Delete by trying to Join deleted group (o15.0.1.14) - BAD Delete of IPoIB membership (no prev join) (o15.0.1.15)
Event Forwarding	- Register for information - Send a trap and wait for report - Unregister non-existing
Stress Testing	Flood the SA with queries from multiple channel adapters to check the robustness of the mechanism
Trap 64/65 Flow	Register to Trap 64-65, create traps (by disconnect/connect ports) and wait for report, then unregister.
Dynamic Changes	Dynamic Topology changes , through randomlly droping SMP packets used to test OpenSM adaptation to unstable network & verify DB correctness.

In addition to using osmtest to verify the functionality of OpenSM, it is possible to further verify OpenSM by using it to setup an actual cluster. Once that is done, an automated check is performed where it is verified that the resulting port info and node info parameters are as expected. Furthermore, it is checked that the resulting routing tables are credit loop-free, and that they cover all point-to-point connectivity.

Another method for verifying OpenSM involves the use of an interactive shell which supports SM and SA configuration and querying. Current tests include: Component Mask, Trap 64-65, Mutiple SMs, External Muticast IF (under osm package), OpenSM stability.