



Fabric Computing That Works™

Installation Guide for Intel® Omni-Path Fabric Suite - Debian Release

IntelOPA-IFS.DEBIAN8-x86_64.10.5.1.0.2

**Prepared for Intel, Inc.
by System Fabric Works, Inc.**

September 29, 2017

Contact Information:

<http://www.systemfabricworks.com>

e-mail opasupport@systemfabricworks.com

Table of Contents

1	Overview.....	2
2	Requirements.....	2
3	IFS Installation.....	2
4	IFS Uninstallation.....	3
5	IFS Source Code.....	4
6	GPUDirect.....	4
7	HFI ROM Image Location.....	5
8	RDMA Module Loading.....	5
9	IPoIB Interface Configuration.....	5
10	TID-RDMA.....	6
11	Use of PSM and /dev/ipath, MPI.....	6
12	openmpi and Debian.....	7
13	Fabric Manager Service.....	7
14	Build mpi_apps.....	7
15	References.....	8

1 Overview

This guide is a supplement to the standard documentation for Intel® Omni-Path Fabric Suite (IFS). These guides include the README for Intel® IFS 10.5.1.0.2, Intel® Omni-Path Architecture (OPA) Fabric Administrator's Guide and [Intel® Omni-Path Fabric Suite FastFabric User Guide](#). The focus of this guide is to document special procedures and considerations for installation and configuration of the software on Debian Linux.

This guide is also relevant to the IntelOPA-BASIC distribution, which does not include opa-fm or opa-fastfabric packages.

2 Requirements

This software release is intended to be installed on a Debian 8.x host. Third-party InfiniBand software such as OFED or Mellanox OFED should not be installed on the host.

IFS only supports the x86_64 architecture.

3 IFS Installation

Installation of IFS is performed via the INSTALL script provided within the software distribution. This script installs dependencies from the Debian distribution via apt-get, installs the IFS packages, creates configuration files for RDMA and OPA, rebuilds the initramfs and enables the opa service.

Note: Execute "INSTALL -h" first in order to see command-line options, such as "enable fabric manager".

Note: The activities of the installer are recorded in `/var/log/opa.log`.

1. Unpack the software distribution.
`tar xf IntelOPA-IFS.DEBIAN8-x86_64.10.5.1.0.2.tgz`
2. Change your working directory to the distribution folder.
`cd IntelOPA-IFS.DEBIAN8-x86_64.10.5.1.0.2`
3. Execute the installer as root
`sudo ./INSTALL`
4. Reboot the host.
`sudo reboot`

The installer may report an error after it installs the required packages from apt-get. If that occurs, execute INSTALL again and it should execute without errors. During installation, the Debian packaging rules will not overwrite existing configuration files from a previous install. The packager will output messages which indicate "Keeping old config file as default" and the packaged configuration file will be installed with a `dpkg-dist` suffix.

Once the installer completes, the administrator should reboot the host. After reboot, the `hfi1` and `ib` modules should be loaded and the fabric may be tested according to procedures specified within the Administrator's Guide.

The `.deb` packages are all located under `packages` in the distribution folder. This folder also includes Debian build artifacts and source packages.

4 IFS Uninstallation

IFS is uninstalled via the INSTALL script in the software distribution. This script removes the installed IFS packages. It does not remove the dependencies that were installed via apt-get.

1. Change your working directory to the distribution folder.
`cd IntelOPA-IFS.DEBIAN8-x86_64.10.5.1.0.2`
Execute the installer as root, with the `-u` argument
`sudo ./INSTALL -u`

Uninstallation does not modify `/etc/security/limits.conf`. The administrator may remove the OPA entries from this file before rebooting the host.

If you wish to remove all packages and their configurations (i.e. "purge"), execute `./INSTALL -u -p`.

5 IFS Source Code

Source code packages are included in the distribution tarball. The Debian packages and associated source code are under the `packages` directory.

To determine which source package is used to create a particular package, use the `dpkg --info` command and look for the `Source` attribute:

```
dpkg --info kmod-ifs-kernel-updates_3.16.0-616-1ifs_amd64.deb
Package: kmod-ifs-kernel-updates
Source: ifs-kernel-updates
```

In this example, `ifs-kernel-updates` is the source package. The related source code and build artifacts are:

- `ifs-kernel-updates_3.16.0-616.orig.tar.gz` - tarball of upstream code
- `ifs-kernel-updates_3.16.0-616-1ifs_amd64.build` - output of the build process that created the binary packages
- `ifs-kernel-updates_3.16.0-616-1ifs_amd64.changes` - summary of changes since the previous version of the package
- `ifs-kernel-updates_3.16.0-616-1ifs.dsc` - Debian source package description
- `ifs-kernel-updates_3.16.0-616-1ifs.debian.tar.xz` - tarball of debian source

To build a binary package, the `devscripts` package must be installed. Individual packages each have their own build requirements, which are specified in `debian/control` by the `Build-Depends` statement. Extract the debian source package and use `debuild` to create the binary:

```
# cd ~
# mkdir tmp
# cd IntelOPA-IFS.DEBIAN8-x86_64.10.5.1.0.2/packages
# dpkg -x ifs-kernel-updates_3.16.0-616-1ifs.dsc \
  ~/tmp/ifs-kernel-updates
# cd ~/tmp/ifs-kernel-updates
# debuild -us -uc
```

After completion of the build, the generated package and artifacts will be present in the parent directory of the build process.

6 GPUDirect

GPUDirect (aka CUDA) is not currently supported by the IFS release for Debian. Please contact opasupport@systemfabricworks.com if this functionality is needed.

7 HFI ROM Image Location

In order to comply with [Debian Filesystem Hierarchy Standard](#), HFI ROM images are not located in `/opt/opa/bios_images`. The images are installed under `/usr/share/hfi1-uefi/bios_images`.

`scripts/update_eprom.sh` may be used by the administrator to update the HFI EPROM and firmware.

8 RDMA Module Loading

On the RedHat platform, RDMA kernel modules are usually loaded via `/usr/libexec/rdma-init-kernel`. `rdma-init-kernel` is a RedHat package, and the prescribed method for loading these modules on Debian is via a configuration file in `/etc/modules-load.d`. The installer creates `/etc/modules-load.d/rdma.conf` to load the RDMA modules at boot time.

A consequence of this technique is that the settings in `/etc/rdma/rdma.conf` are not used for the RDMA modules (e.g. "IPOIB_LOAD=no" does not prevent `ib_ipoib` from loading). However, the `opa` service does use this file for its configuration.

9 IPoIB Interface Configuration

An IPoIB interface is configured using the standard Debian network interface configuration techniques. An overview is provided at <https://pkg-ufed.aliOTH.debian.org/howto/infiniband-howto-5.html>

To temporarily bring up the `ib0` interface, execute the following as root, using the appropriate CIDR:

```
ifconfig ib0 10.20.30.2/24
```

The `ib0` device should then be configured with an IP address.

```
ifconfig ib0
ping 10.20.30.2
```

To automatically bring up the `ib0` interface at boot time, add the following to `/etc/network/interfaces`:

```
auto ib0
iface ib0 inet static
    address 10.20.30.2
    netmask 255.255.255.0
    broadcast 10.20.30.255
```

If connected mode is desired, and/or an MTU should be set, use a configuration that resembles the following:

```
auto ib0
iface ib0 inet static
    address 10.20.30.2
    netmask 255.255.255.0
    broadcast 10.20.30.255
    post-up echo connected > /sys/class/net/ib0/mode && ifconfig ib0 mtu 65520
```

The post-up directive in that configuration sets ib0's IPoIB mode to "connected" and sets the MTU to 65520. The directive may be modified to meet the system requirements.

Test the configuration:

```
ifup ib0
ifconfig ib0 #view and verify IP address
ping 10.20.30.2
```

The ib0 interface should come up automatically and be configured with the IP address when the host is rebooted.

10 TID-RDMA

TID-RDMA enables hardware offload for RDMA processing of messages sized 2MB or greater. It is not enabled, by default.

To enable TID-RDMA:

1. If this is not a virtual machine, add `intel_iommu=off` to the `GRUB_CMDLINE_LINUX_DEFAULT` setting in `/etc/default/grub`
2. Execute `update-grub` to update settings in `/boot/grub/grub.cfg`. **WARNING:** this will overwrite any customizations previously made to `/boot/grub/grub.cfg`. If `grub.cfg` has been manually edited, then `intel_iommu=off` needs to be added to the linux kernel statements in `grub.cfg`.
3. In `/etc/modprobe.d/hfi1.conf`, add `cap_mask=0x4c09a01cbba` to the options for `hfi1`.
4. Reboot.

To determine if TID-RDMA is enabled, examine the `cap_mask` module parameter. If TID-RDMA is enabled, it should be `0x4c09a01cbba`.

```
# cat /sys/module/hfi1/parameters/cap_mask
0x4c09a01cbba
```

11 Use of PSM and /dev/ipath, MPI

If an MPI implementation has not been compiled with PSM2 support, and PSM is desired instead of ibverbs, please review section 4 of:

http://www.intel.com/content/dam/support/us/en/documents/network-and-i-o/fabric-products/Intel_OP_Fabric_Host_Software_UG_H76470_v4_0.pdf

`/dev/opath` is created by udev according to `/lib/udev/rules/40-psm.rules` when the `hfi1` module is loaded by the operating system.

12 openmpi and Debian

There are two issues when running openmpi on Debian:

1. openmpi does not specify absolute paths for certain utilities that it executes via `ssh` (e.g. `orted`).
2. Debian's `.bashrc` terminates early when executed in a non-interactive shell (e.g. a remote `ssh` command).

The consequence of this is that `mpirun` will report an error "bash: orted: command not found". In order to correct this:

1. Use `mpi-selector-menu` to select openmpi on each node.
2. Add `". /etc/profile.d/mpi-selector.sh"` to the beginning of `$HOME/.bashrc` on all nodes.

13 Fabric Manager Service

The Fabric Manager installed by `opa-fm` is not enabled by default. This service is managed via the `systemctl` utility.

Note: `INSTALL` will enable `opafm` if the `-f` flag is specified.

To start the Fabric Manager:

```
systemctl start opafm
```

To stop the Fabric Manager:

```
systemctl stop opafm
```

To enable the Fabric Manager on boot:

```
systemctl enable opafm
```

14 Build mpi_apps

In order to build `mpi_apps`, which are required for certain OPA utilities (e.g. `opacabletest`), the administrator should install the necessary development tools and execute the make process.

```
# sudo bash
# apt-get install build-essential gfortran
# ln -s /usr/bin/make /usr/bin/gmake
# cd /usr/src/opa-mpi-apps
# make
```

After the build completes, these final messages should appear on the console:

```
build base sample applications
Built subset of sample applications
Built sample applications
```

15 References

- Intel® Omni-Path Architecture (OPA) Fabric Administrator's Guide
- [Intel® Omni-Path Fabric SuiteFastFabric User Guide](#)
- [Infiniband HOWTO: Setting up a basic infiniband network](#)
- [Infiniband HOWTO: IP over Infiniband \(IPoB\)](#)
- [Intel® Omni-Path Fabric Suite FabricManager User Guide](#)