NVIDIA DOCA

Installation Guide for Linux

Table of Contents

Chapter 1.   Introduction

There are two ways to install the NVIDIA BlueField DPU software:

▶ Using the SDK Manager which provides a GUI/CLI for full DPU software installation

▶ Manual installation with a step-by-step procedure

1.1.   Supported Platforms

NVIDIA SKU	Legacy OPN	PSID	Description
P1004/699210040230	N/A	NVD0000000015	BlueField-2 A30X, P1004 SKU 205, Generic, GA100, 24GB HBM2e, PCIe passive Dual Slot 230W GEN4, DPU Crypto ON W/ Bkt, 1 Dongle, Black, HF, VCPD
900-9D219-0086-ST1	MBF2M516A-CECOT	MT_0000000375	BlueField-2 E-Series DPU 100GbE Dual-Port QSFP56; PCIe Gen4 x16; Crypto and Secure Boot Enabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D219-0086-ST0	MBF2M516A-EECOT	MT_0000000376	BlueField-2 E-Series DPU 100GbE/EDR/ HDR100 VPI Dual-Port QSFP56; PCIe Gen4 x16; Crypto and Secure Boot Enabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D219-0056-ST1	MBF2M516A-EENOT	MT_0000000377	BlueField-2 E-Series DPU 100GbE/EDR/ HDR100 VPI Dual-Port QSFP56; PCIe Gen4 x16; Crypto Disabled; 16GB on-board DDR; 1GbE

NVIDIA SKU	Legacy OPN	PSID	Description
			OOB management; FHHL
900-9D206-0053-SQ0	MBF2H332A-AENOT	MT_0000000539	BlueField-2 P-Series DPU 25GbE Dual-Port SFP56; PCIe Gen4 x8; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; HHHL
900-9D206-0063-ST2	MBF2H332A-AEEOT	MT_0000000540	BlueField-2 P-Series DPU 25GbE Dual-Port SFP56; PCIe Gen4 x8; Crypto Enabled; 16GB on-board DDR; 1GbE OOB management; HHHL
900-9D206-0083-ST3	MBF2H332A-AECOT	MT_0000000541	BlueField-2 P-Series DPU 25GbE Dual-Port SFP56; PCIe Gen4 x8; Crypto and Secure Boot Enabled; 16GB on-board DDR; 1GbE OOB management; HHHL
900-9D219-0066-ST0	MBF2M516A-EEEOT	MT_0000000559	BlueField-2 E-Series DPU 100GbE/EDR/HDR100 VPI Dual-Port QSFP56; PCIe Gen4 x16; Crypto Enabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D219-0056-SN1	MBF2M516A-CENOT	MT_0000000560	BlueField-2 E-Series DPU 100GbE Dual-Port QSFP56; PCIe Gen4 x16; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D219-0066-ST2	MBF2M516A-CEEOT	MT_0000000561	BlueField-2 E-Series DPU 100GbE Dual-Port QSFP56; PCIe Gen4 x16; Crypto Enabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D219-0006-ST0	MBF2H516A-CEEOT	MT_0000000702	BlueField-2 DPU 100GbE Dual-Port QSFP56; PCIe Gen4 x16; Crypto; 16GB on-

NVIDIA SKU	Legacy OPN	PSID	Description
			board DDR; 1GbE OOB management; FHHL
900-9D219-0056-ST2	MBF2H516A-CENOT	MT_0000000703	BlueField-2 DPU 100GbE Dual-Port QSFP56; PCIe Gen4 x16; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D219-0066-ST3	MBF2H516A-EEEOT	MT_0000000704	BlueField-2 DPU 100GbE/EDR/HDR100 VPI Dual-Port QSFP56; PCIe Gen4 x16; Crypto Enabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D219-0056-SQ0	MBF2H516A-EENOT	MT_0000000705	BlueField-2 DPU 100GbE/EDR/HDR100 VPI Dual-Port QSFP56; PCIe Gen4 x16; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D250-0038-ST1	MBF2M345A-HESOT	MT_0000000715	BlueField-2 E-Series DPU; 200GbE/HDR single-port QSFP56; PCIe Gen4 x16; Secure Boot Enabled; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; HHHL
900-9D250-0048-ST1	MBF2M345A-HECOT	MT_0000000716	BlueField-2 E-Series DPU; 200GbE/HDR single-port QSFP56; PCIe Gen4 x16; Secure Boot Enabled; Crypto Enabled; 16GB on-board DDR; 1GbE OOB management; HHHL
900-9D218-0073-ST1	MBF2H512C-AESOT	MT_0000000723	BlueField-2 P-Series DPU 25GbE Dual-Port SFP56; integrated BMC; PCIe Gen4 x8; Secure Boot Enabled; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D218-0083-ST2	MBF2H512C-AECOT	MT_0000000724	BlueField-2 P-Series DPU 25GbE Dual-Port SFP56; integrated BMC;

NVIDIA SKU	Legacy OPN	PSID	Description
			PCIe Gen4 x8; Secure Boot Enabled; Crypto Enabled; 16GB on-board DDR; 1GbE OOB management; FHHL
900-9D208-0086-ST4	MBF2M516C-EECOT	MT_0000000728	BlueField-2 E-Series DPU 100GbE/EDR/HDR100 VPI Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled; Crypto Enabled; 16GB on-board DDR; 1GbE OOB management; Tall Bracket; FHHL
900-9D208-0086-SQ0	MBF2H516C-CECOT	MT_0000000729	BlueField-2 P-Series DPU 100GbE Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled; Crypto Enabled; 16GB on-board DDR; 1GbE OOB management; Tall Bracket; FHHL
900-9D208-0076-ST5	MBF2M516C-CESOT	MT_0000000731	BlueField-2 E-Series DPU 100GbE Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; Tall Bracket; FHHL
900-9D208-0076-ST6	MBF2M516C-EESOT	MT_0000000732	BlueField-2 E-Series DPU 100GbE/EDR/HDR100 VPI Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; Tall Bracket; FHHL
900-9D208-0086-ST3	MBF2M516C-CECOT	MT_0000000733	BlueField-2 E-Series DPU 100GbE Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled; Crypto Enabled; 16GB on-board DDR; 1GbE

NVIDIA SKU	Legacy OPN	PSID	Description
			OOB management; Tall Bracket; FHHL
900-9D208-0076-ST2	MBF2H516C-EESOT	MT_0000000737	BlueField-2 P-Series DPU 100GbE/EDR/HDR100 VPI Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; Tall Bracket; FHHL
900-9D208-0076-ST1	MBF2H516C-CESOT	MT_0000000738	BlueField-2 P-Series DPU 100GbE Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management; Tall Bracket; FHHL
900-9D218-0083-ST4	MBF2H532C-AECOT	MT_0000000765	BlueField-2 P-Series DPU 25GbE Dual-Port SFP56; integrated BMC; PCIe Gen4 x8; Secure Boot Enabled; Crypto Enabled; 32GB on-board DDR; 1GbE OOB management; FHHL
900-9D218-0073-ST0	MBF2H532C-AESOT	MT_0000000766	BlueField-2 P-Series DPU 25GbE Dual-Port SFP56; integrated BMC; PCIe Gen4 x8; Secure Boot Enabled; Crypto Disabled; 32GB on-board DDR; 1GbE OOB management; FHHL
900-9D208-0076-ST3	MBF2H536C-CESOT	MT_0000000767	BlueField-2 P-Series DPU 100GbE Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled; Crypto Disabled; 32GB on-board DDR; 1GbE OOB management; FHHL
900-9D208-0086-ST2	MBF2H536C-CECOT	MT_0000000768	BlueField-2 P-Series DPU 100GbE Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16;

NVIDIA SKU	Legacy OPN	PSID	Description
			Secure Boot Enabled; Crypto Enabled; 32GB on-board DDR; 1GbE OOB management; FHHL
900-9D250-0048-ST0	MBF2M355A-VECOT	MT_0000000786	BlueField-2 E-Series DPU; 200GbE single-port QSFP56; PCIe Gen4 x16; Secure Boot Enabled; Crypto Enabled; 32GB on-board DDR; 1GbE OOB management
900-9D250-0038-ST3	MBF2M355A-VESOT	MT_0000000787	BlueField-2 E-Series DPU; 200GbE single-port QSFP56; PCIe Gen4 x16; Secure Boot Enabled; Crypto Disabled; 32GB on-board DDR; 1GbE OOB management
900-9D218-0073-ST4	MBF2H512C-AEUOT	MT_0000000972	BlueField-2 P-Series DPU 25GbE Dual-Port SFP56; integrated BMC; PCIe Gen4 x8; Secure Boot Enabled with UEFI disabled; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management
900-9D208-0076-STA	MBF2H516C-CEUOT	MT_0000000973	BlueField-2 P-Series DPU 100GbE Dual-Port QSFP56; integrated BMC; PCIe Gen4 x16; Secure Boot Enabled with UEFI disabled; Crypto Disabled; 16GB on-board DDR; 1GbE OOB management
900-9D208-0076-STB	MBF2H536C-CEUOT	MT_0000001008	BlueField-2 P-Series DPU 100GbE Dual-Port QSFP56, integrated BMC, PCIe Gen4 x16, Secure Boot Enabled with UEFI Disabled, Crypto Disabled, 32GB on-board DDR, 1GbE OOB management, Tall Bracket, FHHL

1.2. Hardware Prerequisites

This quick start guide assumes that an NVIDIA® BlueField® DPU has been installed in a server according to the instructions detailed in your DPU's hardware user guide.

1.3. DOCA Packages

Device	Component	Version	Description
Host	DOCA SDK	1.5.1	Software development kit package for developing host software
	DOCA Runtime	1.5.1	Runtime libraries required to run DOCA-based software applications on host
	DOCA Tools	1.5.1	Tools for developers and administrators on host
	Arm emulated (QEMU) development container	3.9.3.1	Linux-based BlueField Arm emulated container for developers
Target BlueField-2 DPU (Arm)	BlueField BSP	3.9.3.1	BlueField image and firmware
	DOCA SDK	1.5.1	Software development kit packages for developing Arm software
	DOCA Runtime	1.5.1	Runtime libraries requied to run DOCA-based software applications on Arm
	DOCA Tools	1.5.1	Tools for developers and administrators for Arm target

1.4. Supported Operating System

The operating system supported on the BlueField DPU is Ubuntu 20.04.

The following operating systems are supported on the host machine:

▶ Ubuntu 18.04/20.04/22.04

- ▶ CentOS/RHEL 7.6/8.0/8.2
- ▶ Rocky 8.6
- ▶ Debian 10.8

1.5. Supported Kernel Versions

> 📝 **Note:** Only the following generic kernel versions are supported for DOCA local repo package for host installation (whether by SDKM or manually).

Host Operation System	Kernel Support	Arch Support
CentOS 7.6	4.14.0-115.el7a.aarch64	aarch64
	3.10.0-957.el7.x86_64	x86
CentOS 8.0	4.18.0-80.el8.x86_64	
CentOS 8.2	4.18.0-193.el8.x86_64	
RHEL 7.6	3.10.0-957.el7.x86_64	
RHEL 8.0	4.18.0-80.el8.x86_64	
RHEL 8.2	4.18.0-193.el8.x86_64	
Rocky 8.6	4.18.0-372.9.1.el8.x86_64	
Ubuntu 18.04	4.15.0-20-generic	
Ubuntu 20.04	5.4.0-26-generic	
Ubuntu 22.04	5.15.0-52-generic	
Debian 10.8	4.19.0-14-amd64	

Chapter 2. SDK Manager

NVIDIA SDK Manager (SDKM) supports DOCA installation, including software packages on the host and the BlueField-2 target. The SDKM automates the process of DOCA installation and other related configuration of the system.

▶ To use the SDK Manager CLI (recommended), please refer to NVIDIA SDK Manager CLI installation guide for DOCA for detailed instructions.
▶ To use the SDK Manager GUI, please refer to NVIDIA SDK Manager GUI installation guide for DOCA for detailed instructions.

> **Note:** If installing DOCA using SDKM, please skip the remaining sections and follow the wizard instead.

> **Note:** SDKM installation requires Internet connection through out-of-band (OOB) port.

The following is an example for installing SDKM using CLI:
```
# sdkmanager --cli install --logintype devzone --product DOCA --version 1.5.1 --
targetos Linux --host --target BLUEFIELD2_DPU_TARGETS --flash all
```

Chapter 3. Manual BlueField Image Installation

This guide provides the minimal first-step instructions for setting up DOCA on a standard system.

3.1. Installation Files

Device	Component	Arch and OS	Link
Host	These files contain the following components suitable for their respective OS version. ▶ DOCA SDK v1.5.3 ▶ DOCA Runtime v1.5.3 ▶ DOCA Tools v1.5.3	CentOS/RHEL 7.6 on aarch64	doca-host-repo-rhel76-1.5.3-0.0.2.1.5.3003.1.el7a.5.8.4.1.5
		CentOS/RHEL 7.6 on x86	doca-host-repo-rhel76-1.5.3-0.0.2.1.5.3003.1.el7.5.8.4.1.5.
		CentOS/RHEL 8.0 on x86	doca-host-repo-rhel80-1.5.3-0.0.2.1.5.3003.1.el8.5.8.4.1.5.
		CentOS/RHEL 8.2 on x86	doca-host-repo-rhel82-1.5.3-0.0.2.1.5.3003.1.el8.5.8.4.1.5.
		Rocky/RHEL 8.6 on x86	doca-host-repo-rhel86-1.5.3-0.0.2.1.5.3003.1.el8.5.8.4.1.5.
		Ubuntu 18.04 on x86	doca-host-repo-ubuntu1804_1.5.3-0.0.2.1.5.3003.1.5.8.4.1.
		Ubuntu 20.04 on x86	doca-host-repo-ubuntu2004_1.5.3-0.0.2.1.5.3003.1.5.8.4.1.
		Ubuntu 22.04 on x86	doca-host-repo-ubuntu2204_1.5.3-0.0.2.1.5.3003.1.5.8.4.1.
		Debian 10.8 on x86	doca-host-repo-debian108_1.5.3-0.0.2.1.5.3003.1.5.8.4.1.5
	Arm Emulated Development Container	Arm container v3.9.7 on aarch64	doca_devel_ubuntu_20.04-inbox-5.5.tar
Target BlueField-2 DPU (Arm)	BlueField Software v3.9.7	Ubuntu 20.04 on aarch64	doca_1.5.3_bsp_3.9.7_ubuntu_20.04-2.23-lts.prod.bfb
	DOCA SDK v1.5.3		doca-dpu-repo-ubuntu2004-local_1.5.2001-1.5.8.3.0.5.0.bf.3.9.5.12786
	DOCA Runtime v1.5.3		
	DOCA Tools v1.5.3		

3.2. Uninstalling Software from Host

If an older DOCA software version is installed on your host, make sure to uninstall it before proceeding with the installation of the new version:

▶ For Ubuntu/Debian:

```
host# for f in $( dpkg --list | grep doca | awk '{print $2}' ); do echo $f ; apt
 remove --purge $f -y ; done
host# sudo apt-get autoremove
```

▶ For CentOS/RHEL/Rocky:

```
host# for f in $(rpm -qa |grep -i doca ) ; do yum -y remove $f; done
host# yum autoremove
host# yum makecache
```

3.3. Installing Prerequisites on Host for Target DPU

Install `doca-tools` to manage and flash the BlueField DPU.

▶ For Ubuntu/Debian

1. Download the DOCA Tools package from Installation Files section for the host.
2. Unpack the deb repo. Run:
   ```
   host# sudo dpkg -i doca-host-repo-ubuntu<version>_amd64.deb
   ```
3. Perform apt update. Run:
   ```
   host# sudo apt-get update
   ```
4. Run `apt install` for DOCA Tools.

   ▶ For DPU:
      ```
      host# sudo apt install doca-tools
      ```
   ▶ For ConnectX on Ubuntu 20.04:
      ```
      host# sudo apt install doca-cx-tools
      ```

▶ For CentOS/RHEL 8.x or Rocky 8.6

1. Download the DOCA Tools package from Installation Files section for the x86 host.
2. Unpack the RPM repo. Run:
   ```
   host# sudo rpm -Uvh doca-host-repo-rhel<version>.x86_64.rpm
   ```
3. Enable new dnf repos. Run:
   ```
   host# sudo dnf makecache
   ```
4. Run `dnf install` to install DOCA Tools.

   ▶ For DPU:
      ```
      host# sudo dnf install doca-tools
      ```
   ▶ For ConnectX:
      ```
      host# sudo dnf install doca-cx-tools
      ```

▶ For CentOS/RHEL 7.x

1. Download the DOCA Tools package from <u>Installation Files</u> section for the x86 host.
2. Unpack the RPM repo. Run:
   ```
   host# sudo rpm -Uvh doca-host-repo-rhel<version>.x86_64.rpm
   ```
3. Enable new yum repos. Run:
   ```
   host# sudo yum makecache
   ```
4. Run `yum install` to install DOCA Tools.

   ▶ For DPU:
   ```
   host# sudo yum install doca-tools
   ```
   ▶ For ConnectX:
   ```
   host# sudo yum install doca-cx-tools
   ```

3.4. Installing Software on Host

1. Make sure to follow the instructions under <u>Installing Prerequisites on Host for Target DPU</u>.
2. Install DOCA local repo package for host:

   For Ubuntu/Debian Host

   a). Run `apt install` for DOCA runtime, tools, and SDK.

   ▶ For DPU:
   ```
   host# sudo apt install -y doca-runtime doca-sdk
   ```
   ▶ For ConnectX on Ubuntu 20.04:
   ```
   host# sudo apt install -y doca-cx-runtime doca-cx-sdk
   ```

   b). Extra package:
   ```
   host# sudo dnf install -y doca-extra
   ```

   `doca-extra`, located under `/opt/mellanox/doca/tools/`, contains:

   ▶ `doca-info` – displays details of all installed dependencies in DOCA

   ▶ `doca-kernel-support` – running it adds support on existing kernel to support DOCA

   For CentOS Host

   a). Install the following software dependencies. Run:
   ```
   host# sudo yum install -y epel-release
   ```
   b). For CentOS 8.2 only, also run:
   ```
   host# yum config-manager --set-enabled PowerTools
   ```
   c). Enable new yum repos. Run:
   ```
   host# sudo yum makecache
   ```
   d). Run `yum install` for DOCA runtime, tools, and SDK.
   ```
   host# sudo yum install -y doca-runtime doca-sdk
   ```
   e). Extra package:
   ```
   host# sudo dnf install -y doca-extra
   ```

   `doca-extra`, located under `/opt/mellanox/doca/tools/`, contains:

   ▶ `doca-info` – displays details of all installed dependencies in DOCA

▶ `doca-kernel-support` – running it adds support on existing kernel to support DOCA

For Rocky 8.6 Host

a). Install the following software dependencies. Run:
```
host# sudo dnf install -y yum-utils
host# sudo yum-config-manager --enable PowerTools
```

b). Clean cache. Run:
```
host# sudo dnf clean dbcache
```

c). Run `dnf install` for DOCA SDK, DOCA runtime, DOCA tools.
```
host# sudo dnf install -y doca-runtime doca-sdk doca-tools
```

d). Extra package:
```
host# sudo dnf install -y doca-extra
```

`doca-extra`, located under `/opt/mellanox/doca/tools/`, contains:

▶ `doca-info` – displays details of all installed dependencies in DOCA

▶ `doca-kernel-support` – running it adds support on existing kernel to support DOCA

For RHEL Host

> 📝 **Note:** For RHEL 7.6, only perform step d. from the following procedure.

a). Open a RedHat account.

   i. Log into RedHat website via the <u>developers tab</u>.

   ii. <u>Create a developer user</u>.

b). Run:
```
host# subscription-manager register --username=<username> --password=PASSWORD
```

To extract pool ID:
```
host# subscription-manager list --available --all
...
Subscription Name:    Red Hat Developer Subscription for Individuals
Provides:             Red Hat Developer Tools (for RHEL Server for ARM)
                      ...
                      Red Hat CodeReady Linux Builder for x86_64
...
Pool ID:              <pool-id>
...
```

And use the pool ID for the `Subscription Name` and `Provides` that include `Red Hat CodeReady Linux Builder for x86_64`.

c). Run:
```
host# subscription-manager attach --pool=<pool-id>
host# subscription-manager repos --enable codeready-builder-for-rhel-8-x86_64-rpms
host# sudo yum makecache
```

d). Install the DOCA local repo package for host, enable new yum repos, and install DOCA runtime and SDK. Run:
```
host# sudo yum makecache
host# sudo yum install -y doca-runtime doca-sdk
```

e). Sign out from your RHEL account. Run:

```
host# subscription-manager remove --all
host# subscription-manager unregister
```

f). Extra package:

```
host# sudo dnf install -y doca-extra
```

`doca-extra`, located under `/opt/mellanox/doca/tools/`, contains:

- ▶ `doca-info` – displays details of all installed dependencies in DOCA
- ▶ `doca-kernel-support` – running it adds support on existing kernel to support DOCA

3. Initialize MST. Run:

```
host# sudo mst start
```

4. Reset the `nvconfig` params to their default values:

```
host# sudo mlxconfig -d /dev/mst/mt41686_pciconf0 -y reset

Reset configuration for device /dev/mst/mt41686_pciconf0? (y/n) [n] : y
Applying... Done!
-I- Please reboot machine to load new configurations.
```

5. Skip this step if your BlueField DPU is Ethernet only. Please refer to Supported Platforms to learn your DPU type.

   If you have a VPI DPU, the default link type of the ports will be configured to IB. To verify your link type, run:

```
host# sudo mst start
host# sudo mlxconfig -d /dev/mst/mt41686_pciconf0 -e q | grep -i link_type
Configurations:                                 Default        Current        Next
 Boot
*        LINK_TYPE_P1                            IB(1)          ETH(2)
 IB(1)
*        LINK_TYPE_P2                            IB(1)          ETH(2)
 IB(1)
```

> 📑 **Note:** If your DPU is Ethernet capable only, then the `sudo mlxconfig -d <device>` command will not provide an output.

   If the current link type is set to IB, run the following command to change it to Ethernet:

```
host# sudo mlxconfig -d /dev/mst/mt41686_pciconf0 s LINK_TYPE_P1=2
 LINK_TYPE_P2=2
```

6. Verify that RShim is active.

```
host# sudo systemctl status rshim
```

This command is expected to display `active (running)`. If RShim service does not launch automatically, run:

```
host# sudo systemctl enable rshim
host# sudo systemctl start rshim
```

7. Assign a dynamic IP to `tmfifo_net0` interface (RShim host interface).

> 📑 **Note:** Skip this step if you are installing the DOCA image on multiple DPUs.

```
host# ifconfig tmfifo_net0 192.168.100.1 netmask 255.255.255.252 up
```

3.5. Installing Software on DPU

Users have two options for installing DOCA on the DPU:

▶ Upgrading the full DOCA image on the DPU (recommended) - this option overwrites the entire boot partition.

▶ Upgrading DOCA local repo package on the DPU – this option upgrades DOCA components without overwriting the boot partition. Use this option to preserve configurations or files on the DPU itself.

3.5.1. Installing Full DOCA Image on DPU

> **Note:** This installation sets up the OVS bridge.

> **Note:** If you are installing DOCA on multiple DPUs, skip to section Installing Full DOCA Image on Multiple DPUs.

> **Note:** This step overwrites the entire boot partition.

3.5.1.1. Option 1 - No Pre-defined Password

> **Note:** To set the password in advance, proceed to Option 2.

BFB installation is executed as follows:

```
host# sudo bfb-install --rshim <rshimN> --bfb <image_path.bfb>
```

Where `rshimN` is `rshim0` if you only have one DPU. You may run the following command to verify:

```
host# ls -la /dev/ | grep rshim
```

3.5.1.2. Option 2 - Set Pre-defined Password

Ubuntu users can provide a unique password that will be applied at the end of the BlueField software image installation. This password needs to be defined in a `bf.cfg` configuration file.

To set the password for the "ubuntu" user:

1. Create password hash. Run:
   ```
   host# openssl passwd -1
   Password:
   Verifying - Password:
   $1$3B0RIrfX$TlHry93NFUJzg3Nya00rE1
   ```

2. Add the password hash in quotes to the `bf.cfg` file:
   ```
   host# sudo vim bf.cfg
   ubuntu_PASSWORD='$1$3B0RIrfX$TlHry93NFUJzg3Nya00rE1'
   ```

When running the installation command, use the `--config` flag to provide the file containing the password:

```
host# sudo bfb-install --rshim <rshimN> --bfb <image_path.bfb> --config bf.cfg
```

> 📄 **Note:** If `--config` is not used, then upon first login to the BlueField device, users will be asked to update their password.

The following is an example of Ubuntu installation assuming the "pv" Linux tool has been installed (to view the installation progress).

```
host# sudo bfb-install --rshim rshim0 --bfb DOCA_<version>-aarch64.bfb --config
 bf.cfg
Pushing bfb
1.08GiB 0:00:57 [19.5MiB/s] [        <=>     ]
Collecting BlueField booting status. Press Ctrl+C to stop…
INFO[BL2]: start
INFO[BL2]: DDR POST passed
INFO[BL2]: UEFI loaded
INFO[BL31]: start
INFO[BL31]: runtime
INFO[UEFI]: eMMC init
INFO[UEFI]: eMMC probed
INFO[UEFI]: PCIe enum start
INFO[UEFI]: PCIe enum end
INFO[MISC]: Ubuntu installation started
INFO[MISC]: Installation finished
INFO[MISC]: Rebooting...
```

3.5.2.   Installing Full DOCA Image on Multiple DPUs

On a host with multiple DPUs, the BFB image can be installed on all of them using the <u>multi-bfb-install</u> script.

```
host# ./multi-bfb-install --bfb <bfb-file> --password <password>
```

This script detects the number of RShim devices and configures them statically.

▶ For Ubuntu – the script creates a configuration file `/etc/netplan/20-tmfifo.yaml`

▶ For CentOS/RHEL 7.6 – the script creates a configuration file `/etc/sysconfig/network-scripts/ifcfg-br_tmfifo`

▶ For CentOS/RHEL 8.0 and 8.2 – the script installs bridge-utils package to use the command `brctl`, creates bridge `tm-br` and connects all RShim interfaces to it

After the installation is complete, the configuration of the bridge and each RShim interface can be observed using `ifconfig`. The expected result is to see the IP on the bridge `tm-br` configured to `192.168.100.1` with subnet `255.255.255.0`.

> 📄 **Note:** To log into BlueField with `rshim0`, run:
> ```
> ssh ubuntu@192.168.100.2
> ```
> For each RShim after that, add 1 to the fourth octet of the IP address (e.g., `ubuntu@192.168.100.3` for `rshim1`, `ubuntu@192.168.100.4` for `rshim2`, etc).

The script burns a new MAC address to each DPU and configures a new IP, 192.168.100.x, as described earlier.

3.5.3.   Installing DOCA Local Repo Package on DPU

> 📃 **Note:** If you have already installed BlueField image, be aware that the DOCA SDK, Runtime, and Tools are already contained in the BFB, and this installation is not mandatory. If you have not installed the BlueField image and wish to update DOCA Local Repo package, proceed with the following procedure.

> 📃 **Note:** Before installing DOCA on the target DPU, make sure the out-of-band interface (mgmt) is connected to the Internet.

1. Download the DOCA SDK, DOCA Runtime, and DOCA Tools package from section Installation Files.

2. Copy deb repo package into BlueField. Run:
```
host# sudo scp -r doca-repo-aarch64-ubuntu2004-local_<version>_arm64.deb
 ubuntu@192.168.100.2:/tmp/
```

3. Unpack the deb repo. Run:
```
dpu# sudo dpkg -i doca-dpu-repo-ubuntu2004-local_<version>_arm64.deb
```

4. Run apt update:
```
dpu# sudo apt-get update
```

5. Check for any DOCA package content upgrade. Run:
```
dpu# sudo apt install doca-runtime
dpu# sudo apt install doca-tools
dpu# sudo apt install doca-sdk
```

3.5.4.   Updating DOCA Local Repo Package on DPU

> 📃 **Note:** Do not perform the following if you have already performed the steps under Installing DOCA Local Repo Package on DPU.

To upgrade the DPU software to DOCA_1.5.1_BSP_3.9.3_Ubuntu_20.04-4.2211-LTS version from DOCA_1.5.0_BSP_3.9.3_Ubuntu_20.04-11:

1. Run the following:
```
# wget -qO - https://linux.mellanox.com/public/repo/doca/lts/latest/ubuntu20.04/
aarch64/GPG-KEY-Mellanox.pub | sudo apt-key add -
# sudo apt update
# sudo apt-mark hold linux-tools-bluefield linux-image-bluefield linux-bluefield
 linux-headers-bluefield linux-libc-dev linux-tools-common
# sudo apt upgrade
```

2. Download and install the `mlxbf-bootimages` DEB file which includes the DPU's UEFI/ATF and set the right image type ("dev" vs "prod"):
```
# IMAGE_TYPE=dev
# wget -P /tmp -r --no-verbose --no-directories -l1 --no-parent -A 'mlxbf-
bootimages_*_arm64.deb' https://linux.mellanox.com/public/repo/bluefield/latest/
bootimages/${IMAGE_TYPE}/
# dpkg -i /tmp/mlxbf-bootimages_*_arm64.deb
```

3. Upgrade UEFI/ATF (included in `mlxbf-bootimages` DEB package) on the boot partition, run:
```
# bfrec --bootctl --policy dual
```

```
# bfrec --capsule /lib/firmware/mellanox/boot/capsule/boot_update2.cap --policy
 dual
# reboot
```

4. Update NIC firmware according to Upgrading Firmware.

3.6.   Upgrading Firmware

> 📝 **Note:** If multiple DPUs are installed, the following steps must be performed on all of them after BFB installation.

To upgrade firmware:

1. SSH to your BlueField device via 192.168.100.2 (preconfigured).

> 📝 **Note:** If multiple DPUs are installed, the tmfifo IP interface does not have to be 192.168.100.2. The last octate changes and depends on the RShim number.

The default credentials for Ubuntu are as follows:

▸ Username: `ubuntu`

▸ Password: `ubuntu` or a unique password that you set in `bf.cfg`

For example:
```
host# ssh ubuntu@192.168.100.2 Password: <configured-password>
```

2. Upgrade firmware in BlueField DPU. Run:
```
dpu# sudo /opt/mellanox/mlnx-fw-updater/mlnx_fw_updater.pl --force-fw-update
```
Example output:
```
Device #1:
----------

  Device Type:      BlueField-2
  [...]
  Versions:         Current       Available
    FW              <Old_FW>      <New_FW>
```

3. For the firmware upgrade to take effect:

a). Run the following command on the BlueField DPU and host:
```
dpu# sudo mst start
```

b). Query the available reset flows:
```
dpu# sudo mlxfwreset -d /dev/mst/mt41686_pciconf0 q
```

Example output:
```
Reset-levels:
...

Reset-types (relevant only for reset-levels 3,4):
...

Reset-sync (relevant only for reset-level 3):
0: Tool is the owner                      -Supported  (default)
1: Driver is the owner               -Supported
```

c). **If** `reset-sync 1` **is not supported or if** `mlxfwreset` **failed, perform host power cycle.**
Otherwise, trigger reset by running the following:

```
dpu# sudo mlxfwreset -d /dev/mst/mt41686_pciconf0 --sync 1 -y reset
```

> 🗨 **Note:** The entire DPU will experience reset.

3.7.   Post-installation Procedure

1. Restart the driver. Run:
```
host# sudo /etc/init.d/openibd restart
Unloading HCA driver:                                    [  OK  ]
Loading HCA driver and Access Layer:                     [  OK  ]
```

2. Configure the physical function (PF) interfaces.
```
host# sudo ifconfig <interface-1> <network-1/mask> up
host# sudo ifconfig <interface-2> <network-2/mask> up
```
For example:
```
host# sudo ifconfig p2p1 192.168.200.32/24 up
host# sudo ifconfig p2p2 192.168.201.32/24 up
```
Pings between the source and destination should now be operational.

Chapter 4. Building Your Own BFB Installation Image

Users wishing to build their own customized BlueField OS image can use the BFB build environment. Please refer to the bfb-build project in this GitHub webpage for more information.

> **Note:** For a customized BlueField OS image to boot on the UEFI secure-boot-enabled DPU (default DPU secure boot setting), the OS must be either signed with an existing key in the UEFI DB (e.g., the Microsoft key), or UEFI secure boot must be disabled. Please refer to the Secure Boot section and its subpages of the *NVIDIA BlueField DPU Platform Operating System Documentation* for more details.

Chapter 5. Setting Up Build Environment for Developers

For full instructions about setting up a development environment, refer to the NVIDIA DOCA Developer Guide.

Chapter 6.   Additional SDKs for DOCA

6.1.   Installing CUDA on NVIDIA Converged Accelerator

NVIDIA® CUDA® is a parallel computing platform and programming model developed by NVIDIA for general computing GPUs.

This section details the necessary steps to set up CUDA on your environment. This section assumes that a BFB image has already been installed on your environment.

To install CUDA on your converged accelerator:

1. Download and install the latest NVIDIA Data Center GPU driver.
2. Download and install CUDA.

> 📄 **Note:** Downloading CUDA includes the latest NVIDIA Data Center GPU driver and CUDA toolkit. For more information about CUDA and driver compatibility please refer to NVIDIA CUDA Toolkit Release Notes.

6.1.1.   Configuring Operation Mode

There are two modes that the NVIDIA Converged Accelerator may operate in:

▶ Standard mode (default) – the BlueField DPU and the GPU operate separately

▶ BlueField-X mode – the GPU is exposed to the DPU and is no longer visible on the host

To verify which mode the system is operating in, run:
```
host# sudo mst start
host# sudo mlxconfig -d /dev/mst/mt41686_pciconf0 q PCI_DOWNSTREAM_PORT_OWNER[4]
```

Standard mode output:
```
Device #1:
[…]
Configurations:                            Next Boot
        PCI_DOWNSTREAM_PORT_OWNER[4]       DEVICE_DEFAULT(0)
```

BlueField-X mode output:
```
Device #1:
[…]
```

```
Configurations:                                    Next Boot
        PCI_DOWNSTREAM_PORT_OWNER[4]               EMBEDDED_CPU(15)
```

To configure BlueField-X mode, run:

```
host# mlxconfig -d /dev/mst/mt41686_pciconf0 s PCI_DOWNSTREAM_PORT_OWNER[4]=0xF
```

To configure standard mode, run:

```
host# mlxconfig -d /dev/mst/mt41686_pciconf0 s PCI_DOWNSTREAM_PORT_OWNER[4]=0x0
```

Power cycle is required for configuration to take effect. To power cycle the host run:

```
host# ipmitool power cycle
```

6.1.2. Downloading and Installing CUDA Toolkit and Driver

This section details the necessary steps to set up CUDA on your environment. It assumes that a BFB image has already been installed on your environment.

1. Install CUDA by visiting the CUDA Toolkit 11.6.2 Downloads webpage.

> 📝 **Note:** Select the Linux distribution and version relevant for your environment.

2. Test that the driver installation completed successfully. Run:

```
nvidia-smi

Tue Apr  5 13:37:59 2022
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 510.47.03    Driver Version: 510.47.03    CUDA Version: 11.6     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|          Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA BF A10      Off  | 00000000:06:00.0 Off |                    0 |
| 0%   43C    P0    N/A / 225W |      0MiB / 23028MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

3. Verify that the installation completed successfully.

    a). Download CUDA samples repo. Run:

    ```
    dpu# git clone https://github.com/NVIDIA/cuda-samples.git
    ```

    b). Build and run `vectorAdd` CUDA sample. Run:

    ```
    dpu# cd cuda-samples/Samples/0_Introduction/vectorAdd
    dpu# make
    ```

```
dpu# ./vectorAdd
```

> **Note:** If the `vectorAdd` sample works as expected, it should output "Test Passed".

> **Note:** If it seems that the GPU is slow or stuck, stop execution and run:
> ```
> dpu# sudo setpci -v -d ::0302 800.L=201 # CPL_VC0 = 32
> ```

6.1.3. GPUDirect RDMA

To enable GPUDirect RDMA with a network card on NVIDIA Converged Accelerator, you need an additional kernel module. Run:

```
dpu# sudo modprobe nvidia-peermem
```

6.1.4. DPDK GPUDEV

To enable CPU map GPU memory feature in DPDK's gpudev library, you need the GDRCopy library and driver to be installed on your system.

1. Install GDRCopy library. Run:
   ```
   dpu# git clone https://github.com/NVIDIA/gdrcopy.git
   ```
2. Install dependencies.

   ▶ For RHEL:
   ```
   # DKMs can be installed from epel-release. See https://fedoraproject.org/wiki/
   EPEL.
   dpu# sudo yum install dkms check check-devel subunit subunit-devel
   ```
   ▶ For Debian:
   ```
   dpu# sudo apt install check libsubunit0 libsubunit-dev
   ```
3. Build the library and install the driver. Run:
   ```
   dpu# cd gdrcopy
   dpu# make
   # Launch gdrdrv kernel module on the system
   dpu# ./insmod.sh
   ```
4. Setup GDRCopy path. Run:
   ```
   dpu# export GDRCOPY_PATH_L=/path/to/libgdrapi
   ```

> **Note:** In general, the path to libgdrapi is `/path/to/gdrcopy/src/`.

6.2. Installing Rivermax on DPU

NVIDIA® Rivermax® offers a unique IP-based solution for any media and data streaming scenario.

DOCA supports compatible Rivermax libraries that can be installed via SDKM to provide the best user experience.

For additional details and guidelines, please visit the NVIDIA Rivermax SDK product page.

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
http://www.nvidia.com