NVIDIA BlueField DPU Modes of Operation

User Guide

MLNX-15-060472 _v2.2.1 | October 2023

Table of Contents

Chapter 1. Introduction 1

Chapter 2. DPU Mode 2

Chapter 3. Zero-trust Mode 4

3.1. Enabling Zero-trust Mode 4

3.2. Disabling Zero-trust Mode 4

Chapter 4. NIC Mode 6

4.1. NIC Mode for BlueField-3 6

4.1.1. Configuring NIC Mode on BlueField-3 from Linux 6

4.1.1.1. Enabling NIC Mode from Linux 6

4.1.1.2. Disabling NIC Mode from Linux 7

4.1.2. Configuring NIC Mode on BlueField-3 from UEFI 7

4.2. NIC Mode for BlueField-2 7

4.2.1. Enabling NIC Mode on BlueField-2 7

4.2.2. Disabling NIC Mode on BlueField-2 8

Chapter 1. Introduction

The NVIDIA® BlueField® DPU has several modes of operation:

- DPU mode, or embedded function (ECPF) ownership, where the embedded Arm system controls the NIC's resources and data path (default)
- Zero-trust DPU mode which is an extension of the ECPF ownership with additional restrictions on the host side
- NIC mode where the DPU behaves exactly like an adapter card from the perspective of the external host

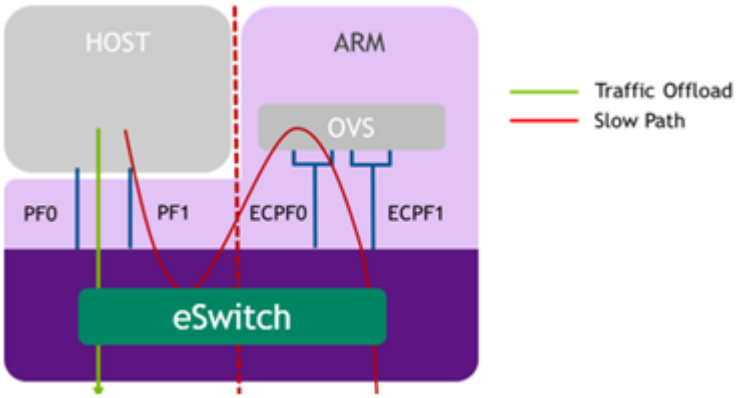
Chapter 2. DPU Mode

This mode, also known as ECPF or embedded mode, is the default mode for the BlueField DPU.

In DPU mode, the NIC resources and functionality are owned and controlled by the embedded Arm subsystem. All network communication to the host flows through a virtual switch control plane hosted on the Arm cores, and only then proceeds to the host. While working in this mode, the DPU is the trusted function managed by the data center and host administrator—to load network drivers, reset an interface, bring an interface up and down, update the firmware, and change the mode of operation on the DPU device.

A network function is still exposed to the host, but it has limited privileges. In particular:

1. The driver on the host side can only be loaded after the driver on the embedded side has loaded and completed NIC configuration.
2. All ICM (interface configuration memory) is allocated by the ECPF and resides in the embedded host memory.
3. The ECPF controls and configures the NIC embedded switch which means that traffic to and from the host interface always lands on the Arm side.



When the server and DPU are initiated, the networking to the host is blocked until the virtual switch on the DPU is loaded. Once it is loaded, traffic to the host is allowed by default.

There are two ways to pass traffic to the host interface: Either using representors to forward traffic to the host (every packet to/from the host would be handled also by the

network interface on the embedded Arm side), or push rules to the embedded switch which allows and offloads this traffic.

In DPU mode, OpenSM must be run from the DPU side (not the host side). Also, management tools (e.g., sminfo, ibdev2netdev, ibnetdiscover) can only be run from the DPU side (not from the host side).

Chapter 3. Zero-trust Mode

Zero-trust mode is a specialization of DPU mode which implements an additional layer of security where the host system administrator is prevented from accessing the DPU from the host. Once zero-trust mode is enabled, the data center administrator should control the DPU entirely through the Arm cores and/or BMC connection instead of through the host.

For security and isolation purposes, it is possible to restrict the host from performing operations that can compromise the DPU. The following operations can be restricted individually when changing the DPU host to zero-trust mode:

- Port ownership – the host cannot assign itself as port owner
- Hardware counters – the host does not have access to hardware counters
- Tracer functionality is blocked
- RShim interface is blocked
- Firmware flash is restricted

3.1. Enabling Zero-trust Mode

To enable host restriction:

1. Start the MST service.
   ```
   $ mst start
   ```
2. Set zero-trust mode. From the Arm side, run:
   ```
   $ mlxprivhost -d /dev/mst/<device> r --disable_rshim --disable_tracer --
   disable_counter_rd --disable_port_owner
   ```

   Note: Power cycle is required if any `--disable_*` flags are used.

3.2. Disabling Zero-trust Mode

To disable host restriction and set the mode to privileged, run:

```
$ mlxprivhost -d /dev/mst/<device> p
```

The configuration takes effect immediately.

Note: Power cycle is required when reverting to privileged mode if host restriction has been applied using any `--disable_*` flags.

Chapter 4. NIC Mode

In this mode, the DPU behaves exactly like an adapter card from the perspective of the external host.

Important: The following instructions presume the DPU to operate in DPU mode. If the DPU is operating in zero-trust mode, please return to DPU mode before continuing.

4.1. NIC Mode for BlueField-3

WARNING: Update the BFB to version 4.2.0 before updating BlueField-3's firmware version to 32.38.1002. Otherwise, anomalous behavior will occur.

Note: When BlueField-3 is configured to operate in NIC mode, Arm OS will not boot.

NIC mode for BlueField-3 saves power, improves device performance, and improves the host memory footprint.

4.1.1. Configuring NIC Mode on BlueField-3 from Linux

4.1.1.1. Enabling NIC Mode from Linux

Before moving to NIC mode, make sure you are operating in DPU mode by running:

```
host/dpu> sudo mlxconfig -d /dev/mst/mt41692_pciconf0 -e q
```

The output should have `INTERNAL_CPU_MODEL= EMBBEDDED_CPU(1)` and `EXP_ROM_UEFI_ARM_ENABLE = True (1)` (default).

To enable NIC mode from DPU mode:

1. Run the following on the host or Arm:

   ```
   host/dpu> sudo mlxconfig -d /dev/mst/mt41692_pciconf0 s
    INTERNAL_CPU_OFFLOAD_ENGINE=1
   ```

2. Power cycle the host.

4.1.1.2. Disabling NIC Mode from Linux

To return to DPU mode from NIC mode:

1. Run the following on the host:
```
host> sudo mlxconfig -d /dev/mst/mt41692_pciconf0 s
 INTERNAL_CPU_OFFLOAD_ENGINE=0
```
2. Power cycle the host.

4.1.2. Configuring NIC Mode on BlueField-3 from UEFI

1. Access the Arm UEFI menu by pressing the Esc button twice
2. Select "Device Manager".
3. Select "Network Device List".
4. Select the network device that presents the uplink (i.e., select the device with the uplink MAC address).
5. Select "NVIDIA Network adapter - $<uplink-mac>".
6. Select "BlueField Internal Cpu Configuration".
   - To enable NIC mode, set "Internal Cpu Offload Engine" to "Disabled".
   - To switch back to DPU mode, set "Internal Cpu Offload Engine" to "Enabled".

4.2. NIC Mode for BlueField-2

In this mode, the ECPFs on the Arm side are not functional but the user is still able to access the Arm system and update `mlxconfig` options.

Note: When NIC mode is enabled, the drivers and services on the Arm are no longer functional.

4.2.1. Enabling NIC Mode on BlueField-2

To enable DPU NIC mode:

1. Run the following from the x86 host side:
```
$ sudo mst start
$ sudo mlxconfig -d /dev/mst/<device> s INTERNAL_CPU_MODEL=1 \
INTERNAL_CPU_PAGE_SUPPLIER=1 \
INTERNAL_CPU_ESWITCH_MANAGER=1 \
INTERNAL_CPU_IB_VPORT0=1 \
INTERNAL_CPU_OFFLOAD_ENGINE=1
```

Note: To restrict RShim PF (optional), make sure to configure `INTERNAL_CPU_RSHIM=1` as part of the `mlxconfig` command.

2. Power cycle the host.

Note: Multi-host is not supported when the DPU is operating in NIC mode.

To obtain firmware BINs for NVIDIA® BlueField®-2 devices, refer to the BlueField-2 firmware download page.

4.2.2. Disabling NIC Mode on BlueField-2

To switch from NIC mode back to DPU mode:

1. Install and start the RShim driver on the host.
2. Disable NIC mode. Run:

```
$ sudo mst start
$ sudo mlxconfig -d /dev/mst/<device> s INTERNAL_CPU_MODEL=1 \
INTERNAL_CPU_PAGE_SUPPLIER=0 \
INTERNAL_CPU_ESWITCH_MANAGER=0 \
INTERNAL_CPU_IB_VPORT0=0 \
INTERNAL_CPU_OFFLOAD_ENGINE=0
```

Note: If `INTERNAL_CPU_RSHIM=1`, then make sure to configure `INTERNAL_CPU_RSHIM=0` as part of the `mlxconfig` command.

3. Power cycle the host.

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation nor any of its direct or indirect subsidiaries and affiliates (collectively: “NVIDIA”) make no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assume no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer (“Terms of Sale”). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer’s own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer’s sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer’s product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA’s aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

Trademarks

NVIDIA, the NVIDIA logo, and Mellanox are trademarks and/or registered trademarks of Mellanox Technologies Ltd. and/or NVIDIA Corporation in the U.S. and in other countries. The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a world¬wide basis. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

© 2023 NVIDIA Corporation & affiliates. All rights reserved.

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