



NVIDIA DOCA DPI

Sample Guide

Table of Contents

- Chapter 1. Introduction..... 1
- Chapter 2. Dependencies..... 2
- Chapter 3. Prerequisites..... 3
- Chapter 4. Running the Sample.....4
- Chapter 5. Samples..... 5
 - 5.1. DPI Scan..... 5
 - 5.2. gRPC DPI Scan..... 5

Chapter 1. Introduction

Deep packet inspection (DPI) is a method of examining the full content of data packets as they traverse a monitored network checkpoint.

DPI provides a more robust mechanism for enforcing network packet filtering as it can be used to identify and block a range of complex threats hiding in network data streams more accurately. This includes:

- ▶ Malicious applications
- ▶ Malware data exfiltration attempts
- ▶ Content policy violations
- ▶ Application recognition
- ▶ Load balancing

For more information about the DOCA DPI library, refer to [NVIDIA DOCA DPI Programming Guide](#).

Chapter 2. Dependencies

N/A

Chapter 3. Prerequisites

DPI-based applications can run either on the host machine or on the DPU target.

As DPI leverages the regular expressions (RegEx) engine, users must make sure it is enabled:

1. Allocate hugepages:

```
sudo echo 2048 > /sys/kernel/mm/hugepages/hugepages-2048kB/nr_hugepages
```

2. Make sure the RegEx engine is active:

```
systemctl status mlx-regex
```

If the status is inactive (Active: failed), run:

```
systemctl start mlx-regex
```

3. DPI compiler must be used since the RegEx engine accepts only .cdo files. The CDO files are constructed by compiling a signature file written in Cta open-source format.

To compile the signature file, run the following:

```
doca_dpi_compiler -i <path to signature file> -o /tmp/signatures.cdo -f suricata
```

The .cdo file is created in the output path flagged as the -o input path of the compiler.

4. For gRPC related samples, assign the IP 192.168.102.2 to the server.

Make sure to deploy the DPI gRPC server and needed setups according to the [NVIDIA DOCA gRPC Infrastructure User Guide](#).

Chapter 4. Running the Sample

1. Refer to the following documents:

- ▶ [NVIDIA DOCA Installation Guide for Linux](#) for details on how to install BlueField-related software.
- ▶ [NVIDIA DOCA Troubleshooting Guide](#) for any issue you may encounter with the installation, compilation, or execution of DOCA samples.

2. To build a given sample:

```
cd /opt/mellanox/doca/samples/doca_dpi/<sample_name>
meson build
ninja -C build
```



Note: The binary `doca_<sample_name>` will be created under `./build/`.

3. Sample (e.g., `dpi_scan`) usage:

```
Usage: doca_dpi_scan [DPDK Flags] -- [DOCA Flags] [Program Flags]
DOCA Flags:
  -h, --help                Print a help synopsis
  -v, --version             Print program version information
  -l, --log-level           Set the log level for the program <CRITICAL=20,
  ERROR=30, WARNING=40, INFO=50, DEBUG=60>
Program Flags:
  -s, --sig-file            Signatures file path
```

4. For additional information per sample, use the `-h` option after the `--` separator:

```
./build/doca_<sample_name> -- -h
```

5. DOCA DPI samples are based on DPDK libraries. Therefore, the user is required to provide DPDK flags. The following is an example from an execution on the DPU:

```
./build/doca_dpi_scan -a 0000:03:00.0,class=regex -a
auxiliary:mIx5_core.sf.4,sft_en=1 -- -s /tmp/signatures.cdo
```

In contrast to DOCA DPI-based applications, the samples require only one network port. This is because they parse the packets they received without forwarding them.



Note: When running on the DPU using the command above sub-functions must be enabled according to the [Scalable Function Setup Guide](#).



Note: When running on the host, virtual functions must be used according to the instructions in the [NVIDIA DOCA Virtual Functions User Guide](#).

Chapter 5. Samples

5.1. DPI Scan

This sample illustrates how to use DPI programming interface on a single packet.

The sample logic includes:

1. Initializing DOCA DPI.
2. Loading the signature .cdo file into the RegEx engine.
3. Creating DPI flow for the incoming packet.
4. Inserting packet into DPI work queue.
5. Polling packet from work queue when processing ends.
6. Retrieving matched signature information if there is a match.
7. Retrieving statistics information.
8. Destroying DPI structures.

Reference:

- ▶ `/opt/mellanox/doca/samples/doca_dpi/dpi_scan/dpi_scan.c`

5.2. gRPC DPI Scan

This sample illustrates how to use gRPC DPI programming interface on a single packet.

The sample logic includes:

1. Initializing DOCA gRPC DPI client.
2. Loading the signature .cdo file into the RegEx engine.
3. Creating DPI flow for the incoming packet.
4. Inserting packet into DPI work queue.
5. Polling packet from work queue when processing ends.
6. Retrieving matched signature information if there is a match.
7. Retrieving statistics information.
8. Destroy DPI structures.

Reference:

- ▶ `/opt/mellanox/doca/samples/doca_dpi/grpc_dpi_scan/grpc_dpi_scan.c`

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 NON-INFRINGEMENT, 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

© 2022 NVIDIA Corporation & affiliates. All rights reserved.