NVIDIA DOCA Programming Guides

Overview Guide

MLNX-15-060541 _v2.2.1 | October 2023

Table of Contents

Chapter 1. Introduction 1

Chapter 2. DOCA Libraries 2

2.1. DOCA App Shield 2

2.2. DOCA Arg Parser 2

2.3. DOCA Comm Channel 2

2.4. DOCA Compress 3

2.5. DOCA Core 3

2.6. DOCA DMA 3

2.7. DOCA DPA 3

2.8. DOCA DPI 3

2.9. DOCA Erasure Coding 3

2.10. DOCA Ethernet 4

2.11. DOCA Flow 4

2.12. DOCA GPUNetIO 4

2.13. DOCA IPsec 4

2.14. DOCA Programable Congestion Control 5

2.15. DOCA RDMA 5

2.16. DOCA RegEx 5

2.17. DOCA Rivermax 5

2.18. DOCA SHA 5

2.19. DOCA Telemetry 5

2.20. DOCA UCX 6

Chapter 1. Introduction

DOCA programming guides provide the full picture of DOCA libraries and their APIs. Each guide includes an introduction, architecture, API, and many more library-specific information with the aim of making DOCA libraries easy to use.

Chapter 2. DOCA Libraries

DOCA libraries are designed to serve DOCA-based software such as the provided example applications. For optimal performance, it is recommended to run these applications on the DPU. However, if necessary, DOCA libraries can be run on the host.

In addition, built-in gRPC support for DOCA allows certain libraries to be used by gRPC clients running on the host that communicates with a matching gRPC server which implements the library's functionality on the DPU.

2.1. DOCA App Shield

This library offers intrusion detection capabilities using the built-in hardware services of the DPU to collect data from the host's memory space. App Shield makes it possible to detect attacks on critical services in the host system. This library leverages the DPU's direct memory access (DMA) capability to monitor the host's memory space directly without involving the host's operating system nor CPU.

2.2. DOCA Arg Parser

This library offers DOCA-based programs an easy and simple command-line interface. Arg Parser supports both regular command-line arguments and a JSON mode that accepts a JSON file containing the required arguments.

2.3. DOCA Comm Channel

This library creates a secure, network-independent communication channel between the host and the DPU. Comm Channel provides a client-server API.

Comm Channel is reliable and message-based. It offers a notification mechanism that can be used by Linux system calls (e.g., `epoll`, `poll`, `select`) and support for multiple connections on the server-side.

2.4. DOCA Compress

This library offers a hardware-accelerated way to compress and decompress data on both DPU and host.

2.5. DOCA Core

This library provides a unified interface to construct standardized DOCA workflows that other libraries and applications can build upon.

2.6. DOCA DMA

This library offers an API for copying data buffers between the host and the DPU using hardware acceleration, supporting both local and remote copy. DMA allows the execution of complex memory operations in an optimized, hardware-accelerated manner.

2.7. DOCA DPA

This library offers a programming model for offloading communication-centric user code to run on the DPA processor on NVIDIA® BlueField®-3 DPU.

DOCA DPA provides a high-level programming interface to the DPA processor.

2.8. DOCA DPI

This library offers a deep examination of data packets as they traverse a monitored network checkpoint. DPI provides a robust mechanism for enforcing network packet filtering, as it can be used to identify or block a range of complex threats due to efficient data stream inspection.

DPI leverages the RegEx engine on the DPU which can very efficiently parse regular expressions found in packets.

DOCA DPI has built-in gRPC support.

2.9. DOCA Erasure Coding

This library provides an API to encode and decode data using hardware acceleration, supporting both the host and NVIDIA® BlueField® DPU memory regions.

DOCA Erasure Coding recovers lost data fragments by creating generic redundancy fragments (backup). Each redundancy block that the library creates can help recover any block in the original data should total loss of a fragment occur.

DOCA Erasure Coding increases data redundancy and reduces data overhead.

2.10. DOCA Ethernet

This library provides two APIs for receiving Ethernet packets on an RX queue and for sending Ethernet packets on a TX queue respectively.

The library collects the user configuration data on the host CPU side, creates TX/RX objects, and exports them to the GPU side for execution in the data-path.

2.11. DOCA Flow

This library is the most fundamental API for building generic execution pipes in hardware. The main building block of the library is a pipe. Each pipe consists of match criteria, monitoring, and a set of actions. Pipes can be chained to create a set of complex actions to be performed on ingress packets.

This library serves as an abstraction layer API for network acceleration and should be used by applications intended to offload packet processing from the operating system Kernel directly to the user space.

DOCA Flow has a built-in gRPC-support.

2.12. DOCA GPUNetIO

This library offers building blocks to create a GPU-centric packet processing network application where CUDA kernels are capable of directly interacting with the network card without involving the CPU in the main critical path.

This library provides CUDA device functions to send and receive packets. Additionally, an object named semaphore is provided to allow message passing across CUDA kernels or a CUDA kernel and a CPU thread.

This library also allow allocating memory on the GPU that would be accessible from the CPU and vice versa.

2.13. DOCA IPsec

This library provides an API to create the security association (SA) objects required for DOCA Flow's hardware-accelerated encryption and decryption.

2.14. DOCA Programable Congestion Control

This library provides a high-level programming interface that allows users to implement their own customized congestion control (CC) algorithm.

2.15. DOCA RDMA

DOCA RDMA enables direct access to the memory of remote machines, without interrupting the processing of their CPUs or operating systems. Avoiding CPU interruptions reduces context switching for I/O operations, leading to lower latency and higher bandwidth compared to traditional network communication methods.

2.16. DOCA RegEx

This library provides regular expression pattern matching to DOCA programs. It provides access to the regular expression processing (RXP) engine, a high-performance hardware-accelerated engine available on the DPU.

RegEx allows the execution of complex regular expression operations in an optimized, hardware-accelerated manner.

2.17. DOCA Rivermax

This library provides an API for using NVIDIA® Rivermax®, an optimized networking SDK for media and data streaming applications. Rivermax leverages the DPU hardware streaming acceleration technology which allows data to be transferred to and from the GPU to deliver best-in-class throughput and latency.

2.18. DOCA SHA

This library provides a flexible and unified API to leverage the secure hash algorithm offload engine present in the NVIDIA® BlueField®-2 DPU. The SHA hardware engine supports SHA-1, SHA-256, and SHA-512 algorithms either as "single shot" or stateful calculations.

2.19. DOCA Telemetry

This library offers a fast and convenient way to transfer user-defined data to the DOCA Telemetry Service (DTS). Telemetry API provides the user a choice between several

different outputs including saving the data directly to storage, NetFlow, Fluent Bit forwarding, or Prometheus endpoint.

2.20. DOCA UCX

Unified Communication X (UCX) is an optimized point-to-point communication framework. UCX exposes a set of abstract communication primitives that makes the best use of available hardware resources and offloads. UCX facilitates rapid development by providing a high-level API, masking the low-level details, while maintaining high performance and scalability.

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