3-2022

Securing Infiniband Networks with End-Point Encryption

Noah B. Diamond

SECURING INFINIBAND NETWORKS WITH END-POINT ENCRYPTION

THESIS

Noah B. Diamond, 2d Lt, USAF

AFIT-ENG-MS-22-M-024

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-MS-22-M-024

SECURING INFINIBAND NETWORKS WITH END-POINT ENCRYPTION

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Computer Engineering

Noah B. Diamond, B.S.C.E.

2d Lt, USAF

March 2022

AFIT-ENG-MS-22-M-024

SECURING INFINIBAND NETWORKS WITH END-POINT ENCRYPTION

THESIS

Noah B. Diamond, B.S.C.E.
2d Lt, USAF

Committee Membership:

Scott R. Graham, Ph.D.
Chair

Barry E. Mullins, Ph.D., P.E.
Member

Gilbert J. Clark III, M.S.
Member

Abstract

InfiniBand is increasingly used in applications outside the high performance computing domain, generating interest in securing InfiniBand networks with encryption and packet inspection. However, the performance benefit realized by the InfiniBand hardware transport protocols is at odds with many kernel, stack-based Internet Protocol (IP) datagram encryption and network monitoring technologies. Kernel bypass approaches make it necessary for new security applications to be developed.

The NVIDIA-Mellanox Bluefield-2 is a 100 Gbps high-performance network interface which offers hardware offload and acceleration features that can operate directly on network traffic without routine involvement from the ARM CPU. This allows the ARM multi-core CPU to orchestrate the hardware to perform operations on both Ethernet and Remote Direct Memory Access (RDMA) traffic at high rates rather than processing all the traffic directly.

A testbed called TNAP was created for performance testing and a Man-in-the-Middle verification process called MiTMVP is used to ensure proper network configuration. The hardware accelerators of the Bluefield-2 support a throughput of nearly 86 Gbps when using IP Security (IPsec) to encrypt and authenticate RDMA over Converged Ethernet Version 2 (RoCEv2) traffic.

This research closes by providing operational security recommendations to defend against presented vulnerabilities, and secure InfiniBand with the Bluefield-2 and similar network adapters. Security and performance implications are discussed, and the need for ongoing evaluation of InfiniBand is emphasized.

This work is dedicated to my family for their unfailing love and support.

Acknowledgements

I am grateful to AFIT and the USAF for letting me participate in this unique and rare opportunity. The world class curriculum and instructors at AFIT have taught me much about being an engineer in today's Air Force. I will carry the knowledge and lessons I have gained at AFIT with me throughout my career.

I would like to thank my research advisor Dr. Scott Graham for his constant support and advice during my time at AFIT. He has always gone the extra mile to help me succeed, and has provided priceless insights and words of wisdom. I am forever grateful for his leadership.

I would also like to thank Mr. Gilbert Clark for providing his expertise and vast knowledge of computer networks to help focus this research. He volunteered many hours of his time to help me understand InfiniBand networks.

Finally, I would like to thank Dr. Barry Mullins for his support, guidance, encouragement. His thesis process and courses provided me with a knowledge foundation that allowed me to complete this research.

Noah B. Diamond

Table of Contents

Page

Page

List of Figures

List of Tables

List of Acronyms

AOC	Active Optical Cable
API	Application Programming Interface
ARM	Advanced Reduced Instruction Set Computer Machines
ASAP^2	Accelerated Switching and Packet Processing
ASIC	Application Specific Integrated Circuit
BTH	Base Transport Header
CA	Channel Adapter
CE	Converged Ethernet
CPU	Central Processing Unit
CQ	Completion Queue
CUT	component under test
DC	Dynamically Connected
DCB	Data Center Bridging
DDR4	Double Data Rate 4
DETH	Datagram Extended Transport Header
DMA	Direct Memory Access
DoD	Department of Defense
DoS	Denial-of-Service
DPDK	Data Plane Development Kit
DPU	Data Processing Unit
ECPF	Embedded CPU Function
ESP	Encapsulating Security Payload
eSwitch	Embedded Switch
GID	Global IDentifier

GRH Global Route Header

GUID Globally Unique IDentifier

HCA Host Channel Adapter

HPC High Performance Computing

I/O Input/Output

IBA InfiniBand Architecture

IBTA InfiniBand Trade Association

IP Internet Protocol

IPsec IP Security

LID Local IDentifier

LRH Local Router Header

MAC Media Access Control

MiTM Man-in-the-Middle

MiTMVP MiTM Verification Process

MSS Maximum Segment Size

MTU Maximum Transmission Unit

NIC Network Interface Card

OFED OpenFabrics Enterprise Distribution

OS Operating System

OSI Open Systems Interconnection

OvS Open vSwitch

PCIe Peripheral Component Interconnect Express

PF Physical Function

PMD Poll Mode Driver

QOS Quality of Service

QP Queue Pair

RAM Random-Access Memory

RC Reliable Connection

RDMA Remote Direct Memory Access

RFC Request for Comments

RoCE RDMA over Converged Ethernet

RoCEv2 RDMA over Converged Ethernet Version 2

RoE RDMA over Ethernet

SerDes Serializer/Deserializer

SL Service Level

SM Subnet Manager

SoC System on Chip

SPCL Scalable Parallel Computing Laboratory

SR-IOV Single Root I/O Virtualization

SUT system under test

TC Traffic Classification

TCP Transmission Control Protocol

TLS Transport Layer Security

TNAP Testbed for Network Adapter Performance

UDP User Datagram Protocol

VF Virtual Function

VI Virtual Interface

VL Virtual Lane

VM Virtual Machine

VPI Virtual Protocol Interconnect

WQ Work Queue

WQE Work Queue Entry

I. Introduction

1.1 Background

In recent years, the InfiniBand interconnect family has become one of the most popular in most major industries [11]. InfiniBand is installed in six of the top ten supercomputers in the world. It accounts for 35.6% of the interconnect family system share, and 44.5% of the interconnect family performance share across the top 500 supercomputers in the world [12]. InfiniBand is currently in use in thousands of data centers, High Performance Computing (HPC) clusters, and embedded applications.

Widespread demand for high-performance, scalable, and reliable networks in a diverse set of applications has promoted interest in InfiniBand networks. Amidst the rapid development of kernel bypass networks, developers have paid more attention to performance and cost efficiency than to security [13]. Lee and Kim [14] and Rothenberger et al. [10] state that there are numerous security loopholes within the InfiniBand Architecture (IBA) that have been revealed and, consequently, the design of secure InfiniBand networks has recently surfaced as a critical issue. The increased prevalence of InfiniBand reveals the need for investigation into its vulnerabilities and potential defenses.

1.2 Problem Statement

Remote Direct Memory Access (RDMA) is a hardware transport protocol that allows both Ethernet and InfiniBand network adapters to transfer data to and from host

memory with minimal involvement from the host processor. RDMA is increasingly important in modern networks because it alleviates computational loads placed on host Central Processing Units (CPUs) by virtualization, storage, and network management applications by offloading packet processing to dedicated hardware. Despite some built-in security features of RDMA, existing RDMA network protocols do not provide any mechanisms for authentication or encryption of the header and payload of RDMA packets. This allows an adversary to spoof any field in packet headers or alter the payload of RDMA messages. These packet injections are undetectable if packet checksums are recalculated using the algorithms and seeds specified by the IBA [10]. This is a well documented vulnerability of RDMA network protocols which reveals the need for robust encryption and authentication solutions to be integrated into the IBA. Further, this vulnerability highlights the need for novel network monitoring solutions for RDMA network protocols and kernel bypass technologies. This thesis identifies and characterizes the capabilities provided by cutting-edge channel adapters, such as the Bluefield-2 Data Processing Unit (DPU), to defend against vulnerabilities present in RDMA fabric architectures.

1.3 Research Goals

This work characterizes the security capabilities of the Bluefield-2 DPU and its ability to perform line-rate encryption on RDMA traffic. This research also offers operational security recommendations to defend against vulnerabilities in the IBA.

1.4 Hypothesis

This research hypothesizes that the hardware accelerators of the Bluefield-2 DPU are capable of providing near line-rate encryption of RDMA traffic when using Ethernet at the data link-layer. It also theorizes that the Advanced Reduced Instruction

Set Computer Machines (ARM) CPU and memory of the Bluefield-2 DPU are quickly overwhelmed by custom link-layer encryption schemes implemented in software.

1.5 Approach

The Testbed for Network Adapter Performance (TNAP) testbed was designed to characterize the capability of the Bluefield-2 DPU to perform hardware and software based encryption. The TNAP consists of a pair of Bluefield-2 DPUs each installed into an HP Z840 workstation via a 16 lane Peripheral Component Interconnect Express (PCIe) Gen 3 slot. The workstation CPUs are used by the testbed to generate traffic for performance tests. Furthermore, a MiTM Verification Process (MiTMVP) was developed for debugging and verifying proper end-to-end encryption configuration. This approach uses a Bluefield-1 DPU as a Man-in-the-Middle (MiTM) which allows the Bluefield-1 DPU to passively sniff Ethernet and RDMA traffic in the TNAP.

1.6 Assumptions/Limitations

The following assumptions/limitations are understood when performing device characterization tests:

- This research does not use available optimizing and tuning tools. While additional performance improvement is expected given further configuration changes, the results presented by this research are assumed to be representative of the potential impact encryption and hardware acceleration could have on system performance. While it is true that using the Nvidia-Mellanox tuning tool for the Bluefield-2 DPU could optimize network performance, this approach introduces hidden system changes that would be difficult to identify and may not be reversible.

- The comparison of `Testpmd` and Open vSwitch (OvS) in the results of this research is assumed to be representative of the performance difference of traditional and kernel bypass network monitoring applications. Although, `Testpmd` is a Data Plane Development Kit (DPDK) application intended for forwarding traffic between ports on an Ethernet interface, and does not provide any monitoring capabilities.

1.7 Contributions

This thesis contributes to the field of InfiniBand security with a focus on channel adapter software and hardware encryption capabilities:

1. **TNAP:** The Bluefield-2 DPU must be the limiting factor in the network in order to validate the performance limits of the card. This can only be achieved by generating traffic at a greater rate than the card can handle. This research illustrates how this can be achieved using the DPDK `Pktgen` traffic generator.

2. **MiTMVP:** End-to-end encryption must be verified by a third device. A 100 Gbps Ethernet switch was not available for this research effort, so a Bluefield-1 DPU acts as a software bridge capable of sniffing traffic and verifying that end-to-end encryption is properly configured prior to performance testing.

3. **Pre-Existing Application Performance:** This work characterizes the built-in hardware and software based encryption capabilities of the Bluefield-2 DPU.

4. **Synthesis:** This work stresses the importance of securing InfiniBand and demonstrates that commercially available devices are capable of near line-rate encryption.

1.8 Thesis Overview

This thesis document is arranged in six chapters. Chapter II provides a brief summary of relevant technologies, an outline of tools used, and relevant research. Chapter III presents the system design details, TNAP, MiTMVP, and encryption methods tested to add confidentiality to RDMA traffic. The experiment methodology and the analysis of results are presented in Chapter IV and Chapter V respectively, while Chapter VI summarizes the research and discusses opportunities for future work in this domain.

II. Background and Related Work

2.1 Overview

This chapter provides a technical summary of the IBA, highlighting characteristics that may enable vulnerabilities in critical infrastructure. It follows with an outline of the current state of InfiniBand security, a survey of open-source tools used in this work, and a discussion of related research.

2.2 InfiniBand Architecture

Networks are limited by the speed of either processors, Input/Output (I/O) interfaces, or network protocols. The achievable performance of network devices has steadily improved as manufacturers are able to create chip-sets with smaller feature sizes, more efficient computer architectures, faster clock rates, etc. As these improvements materialize, governing bodies of network protocols must make careful decisions with respect to future protocols, considering the effects of compatibility with established network protocols. Growing demand for improved network performance and awareness of the limitations of legacy technologies in the high-performance computing domain led to the formation of the InfiniBand Trade Association (IBTA). The IBTA is led by a steering committee that includes Broadcom, HPE, IBM, Intel Corporation, Marvell Technology Group, Mellanox Technologies and Microsoft [15]. A notional InfiniBand network is shown in Figure 1.

2.3 InfiniBand vs Ethernet

Differences between InfiniBand and Ethernet go beyond the data-link layer. InfiniBand is a complete network architecture with its own set of network protocols,

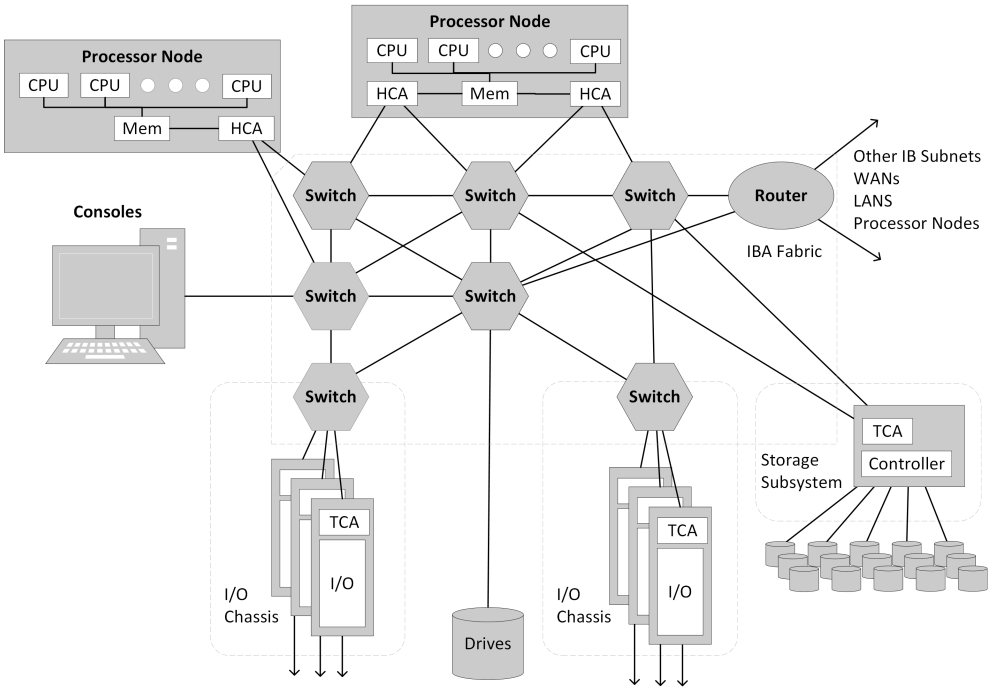

Figure 1. InfiniBand Fabric Overview (adapted from [1])

communication models, security features, and components. As a result, most Ethernet applications do not work natively with InfiniBand. This section seeks to identify differences between the IBA and Ethernet beyond the data-link layer.

2.3.1 Performance

InfiniBand and Ethernet use the same 50 Gbps Serializer/Deserializer (SerDes) elements that convert bi-directional network traffic at the physical layer. Despite having the same throughput per SerDes lane, the current InfiniBand specification allows up to 12 SerDes links to be packed together in a single link, whereas Ethernet only allows eight [11]. Therefore, the maximum throughput supported by the current Ethernet and InfiniBand specifications are 400 and 600 Gbps respectively.

2.3.2 Network Stack

Figure 2 shows a side-by-side comparison of the network stack for Ethernet and InfiniBand, using the 5-layer TCP/IP stack as a reference. Between the application and transport layers, InfiniBand uses verbs in place of Ethernet sockets. InfiniBand verbs are the basis for specifying the Application Programming Interfaces (APIs) that an application uses [16]. Additionally, InfiniBand has a number of transport services. The two primary types are reliable and unreliable connections, analogous to Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). Native InfiniBand employs Local IDentifiers (LIDs), Global IDentifiers (GIDs), and Globally Unique IDentifiers (GUIDs) addresses, analogous to, but in place of Internet Protocol (IP) and Media Access Control (MAC) addresses. Lastly, InfiniBand uses an Subnet Manager (SM) to configure local subnets. There must be at least one SM present in the subnet to manage all switch and router setups, and for subnet reconfiguration when a link drops or a new link appears [17].

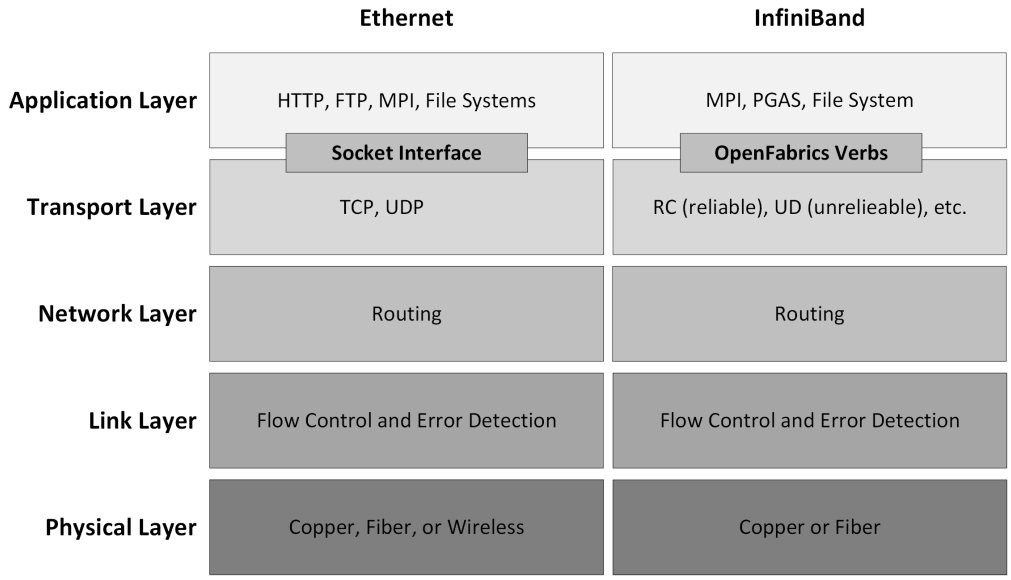

Figure 2. Comparison of Ethernet and InfiniBand Network Stacks (adapted from [2])

2.3.2.1 Addressing and Packet Format

Ethernet and InfiniBand message formats are illustrated in Figure 3.

- **Data Link-Layer:** InfiniBand uses a Local Router Header (LRH) in place of the MAC header in an ethernet frame. InfiniBand uses LID addresses at Layer 2 while Ethernet uses MAC addresses. The LRH also specifies the Virtual Lane (VL) and Service Level (SL) the packet is using.

- **Network Layer:** InfiniBand uses the Global Route Header (GRH) at the network layer. The GRH contains GID addresses for routing between subnets. Each GID is 128-bits and provides a very large address space.

- **Transport Layer:** The Base Transport Header (BTH) is used in InfiniBand to specify the IBA packet type, partition key, destination Queue Pair (QP), and packet sequence number. Partition and QP keys are security measures built into the InfiniBand Layer 4. Key management is discussed in more detail in Section 2.3.5.

2.3.3 Communication Model

Traditional network architectures use an Operating System (OS) to virtualize network hardware into a set of logical communication endpoints available to network consumers. The OS multiplexes access to hardware among these endpoints. The OS also implements protocols that provide reliable connections. This model permits the interface between the network hardware and the OS to be very lightweight. However, a significant drawback is that all communication operations require a call or trap into the OS kernel; and interaction from the host CPU can be computationally expensive.

In 1997, Intel paved the way to improving the traditional network model with the introduction on the Virtual Interface (VI) Architecture. The VI Architecture

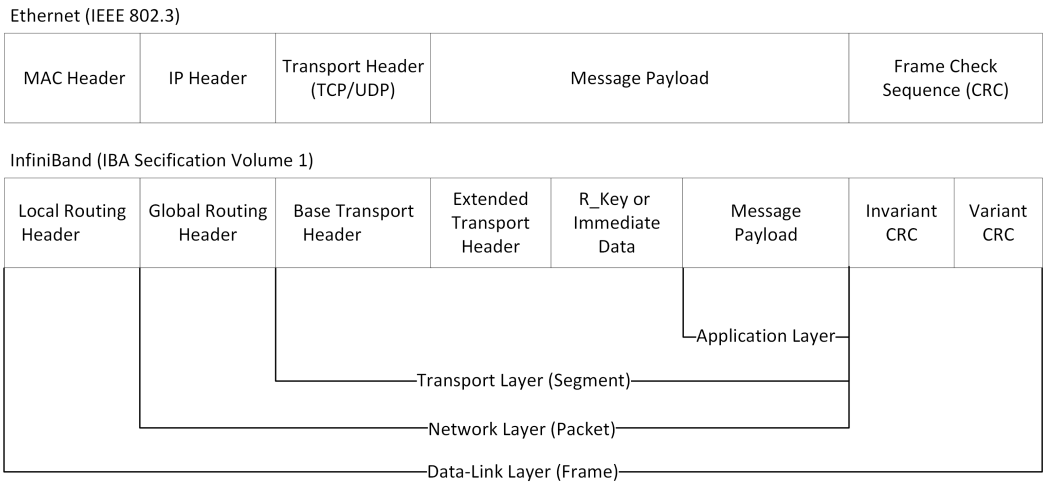

Figure 3. Ethernet and Infiniband Message Formats

eliminates the system-processing overhead of the traditional model by providing each consumer process with a protected, directly accessible interface to the network hardware. Each VI represents a communication endpoint. The VI model reduces CPU interaction in tasks of multiplexing, de-multiplexing, and data transfer scheduling [3].

Many concepts in the VI Architecture are incorporated in the IBA Specification. The IBA has a number of enhanced features compared to the VI. Queues are the VI of the IBA. InfiniBand offloads traffic control from the software client through the use of execution queues [17]. Figure 4 illustrates the InfiniBand communication stack, where control is offloaded from the software client to a Work Queue (WQ) for InfiniBand to manage. Each communication channel is assigned a QP, consisting of a send and receive queue being assigned at the corresponding end nodes. QPs are unidirectional, and bi-directional packet transmission requires the creation of two QPs. The client places transactions into the WQ in the form of a Work Queue Entry (WQE) so that it can be processed by the Channel Adapter (CA). When the transaction is finished, the CA notifies the client by placing an entry into the

Completion Queue (CQ) [17]. Complete hardware implementations of the InfiniBand network stack streamline InfiniBand communication models, and allow applications to interface with InfiniBand solely through the use of InfiniBand verbs.

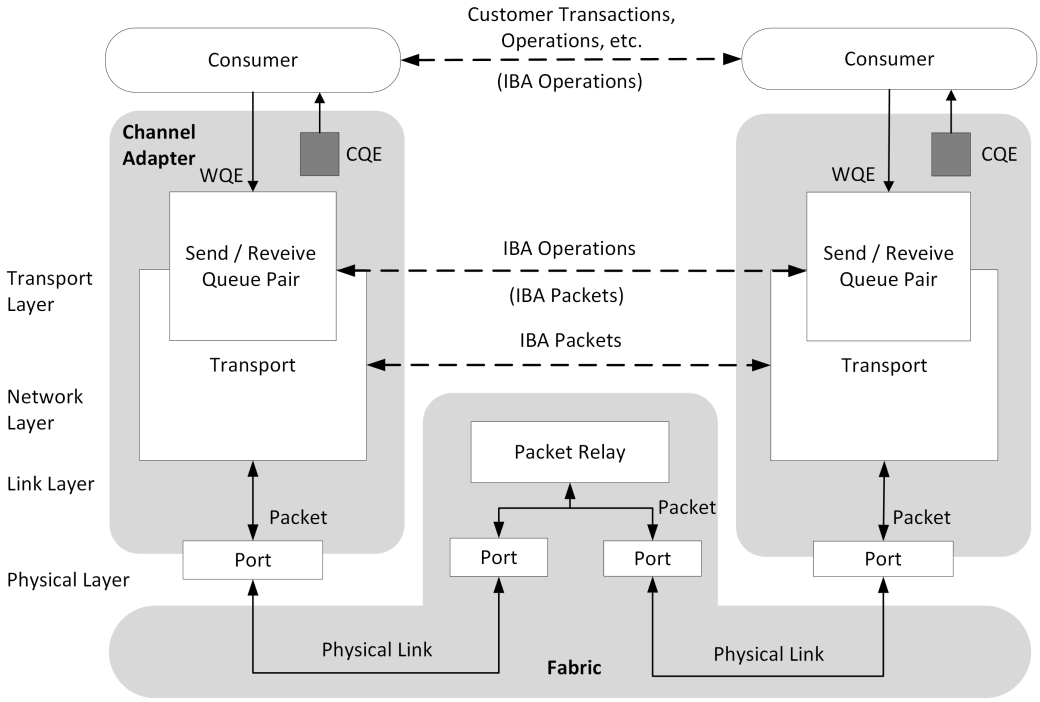

Figure 4. InfiniBand Architecture Transactions (adapted from [1])

2.3.4 Transport Functions

InfiniBand transport functions offload the computational load placed on data center CPUs by allowing data to be transferred with minimal host processor involvement. InfiniBand implements five distinct types of transport functions, each of which use QPs in hardware to minimize intervention from host processors. Figure 5 illustrates how RDMA traffic moves between applications and avoids latencies incurred from buffers in the OS kernel. Although the host processor authorizes the transfer, the hardware based RDMA implementation bypasses the host CPU for execution.

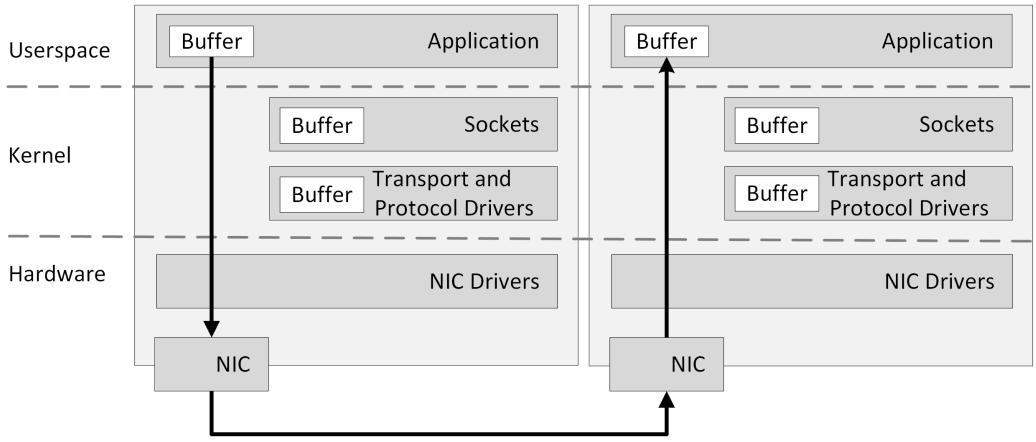

Figure 5. RDMA Traffic Flow (adapted from [1])

Transport functions are initiated when a QP provides the client of the transport layer (e.g. the verbs layer in an Host Channel Adapter (HCA)) with a specific transport service. Each transport service has a corresponding reliability level for connected or connectionless communication. Transport functions are the underlying messaging methods utilized by each transport service. There are five distinct transport functions defined in the IBA specification. SEND, RDMA READ, and RDMA WRITE are the only three investigated by this research.

- **SEND:** The SEND operation is sometimes referred to as a "Push" operation. With a SEND operation, the client pushes data to the remote server QP. The client does not specify where the data is going on the server. The CA of the server simply places the data into the next available receive buffer for the corresponding QP. On an HCA, the receive buffer is pointed to by the WQE at the head of the QP receive queue [1]. The data is tagged with a discriminator which consists of the destination LID and QP number. Once received, the server chooses where to place the data based on the discriminator [1].

- **RDMA WRITE:** Prior to RDMA WRITEs, the destination node allocates

a memory range for access by the destination QP(s). The destination CA associates a 32-bit R_Key with this memory region or window. This is know as registering a memory region for an HCA [1].

The destination communicates the virtual address, length, and R_Key to any other host it wishes to grant access to its memory region through a client upper level protocol. For example, an application program might embed the address, length, and R_Key into a private data structure that it in turn pushes to other application programs using the SEND Operation [1].

A set of memory locations that have been registered are referred to as a memory region. Memory region verbs produce a handle that is used to identify specific memory regions for application use through memory management verbs. When registering a memory region, the consumer also specifies the maximum number of memory locations that are to be reserved for future use. This allows writing end nodes to know which memory regions are available on remote end nodes [1].

- **RDMA READ:** RDMA READs are very similar to RDMA WRITEs. They allow the requesting node to read a virtually contiguous block of memory on a remote node. As with RDMA WRITEs, the responding node first allows the requesting node permission to access its memory by passing a virtual address, length, and R_Key to use in the RDMA READ request packet [1]. The RDMA READ transport function requires the requesting node to first send a read request to the responding node before data is transferred. Figure 6 is a ladder diagram that illustrates the delay caused by an RDMA READ as compared to RDMA Write and SEND.

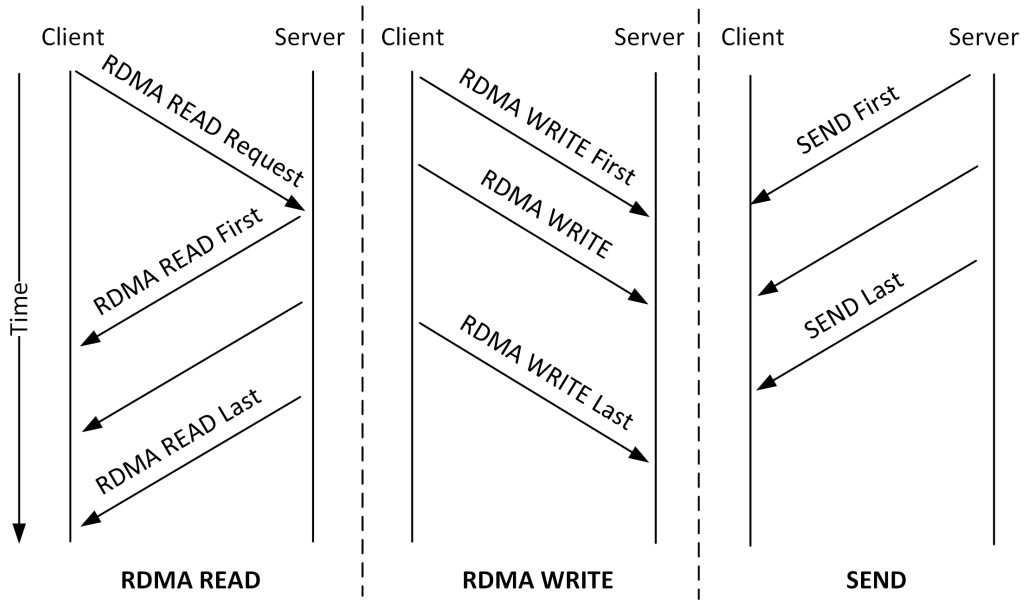

Figure 6. Comparison of Send/Receive & RDMA Read/Write (adapted from [1])

2.3.5 Built-in Security Features

There are several security features built into the InfiniBand transport layer intended to filter unauthorized network traffic and protect the memory of each end node.

2.3.5.1 Partitions

The BTH of each InfiniBand segment includes a 16 bit partition key (P_Key) that indicates which logical partition is associated with a packet. Partitioning enforces isolation among systems sharing an InfiniBand fabric by establishing sets of end nodes that may communicate [1]. Each port of an end node is a member of at least one partition, and each partition is represented by a unique P_Keys. Reception of an invalid P_Key causes packets to be dropped. Partition keys are sent in the clear within the BTH of InfiniBand packets.

Switches and routers can be configured to enforce partitioning in which case the switch or router populates a P_Key Table and inspects the P_Key of all received packets.

2.3.5.2 Memory Registration

As mentioned in Section 2.3.3, QPs are virtual, communication interfaces provided to InfiniBand consumers by hardware. Memory regions and memory windows are registered for QPs using a four step process:

1. **Registration Request:** The client application sends a virtual address and length to the OS kernel.

2. **Virtual to Physical Mapping**: The kernel handles memory mapping and reserves regions of physical memory for RDMA transactions. This process adds a level of security because a process cannot map memory that it does not own.

3. **CA Cache Mapping:** The CA caches the virtual to physical mapping and QP. Each QP is issued an alpha-numeric handle which includes a local key (L_Key) and remote key (R_Key).

4. **Handle Returned:** The QP handle is returned to the client application.

QP memory is protected against inadvertent and unauthorized access through the use of QP Keys (Q_Key), memory keys (L_Keys and R_Keys), and Protection Domains.

First, Q_Keys are 32 bit keys used by datagram transport service QPs to validate the right of a remote sender to access a local receive queue [1]. Q_Keys are placed in the Datagram Extended Transport Header (DETH).

Second, memory keys enable the use of virtual addresses and provide end nodes with a mechanism to restrict access to their physical memory. Memory keys are 32 bit

keys administered by the CA during the four step registration process. The consumer registers a region of memory with the CA and receives an L_Key and R_Key. The consumer uses the L_Key in work requests to describe local memory to the QP and passes the R_Key to the remote consumer in the RDMA Extended Transport Header of an RDMA request packet. As illustrated in Figure 7, a consumer receives an R_Key from the remote consumer when it queues an RDMA operation. The R_Key validates that a sending end node has access to the memory of the destination end node. Further, the R_Key provides the destination channel adapter with the means to translate the virtual to physical address [1].

Third, Protection domains allow a consumer to limit access to memory regions and memory windows. A consumer creates one or more protection domains before a consumer allocates a QP or registers memory. QPs and memory are allocated to that protection domain. L_Keys and R_Keys are only valid for QPs created for the same protection domain [1].

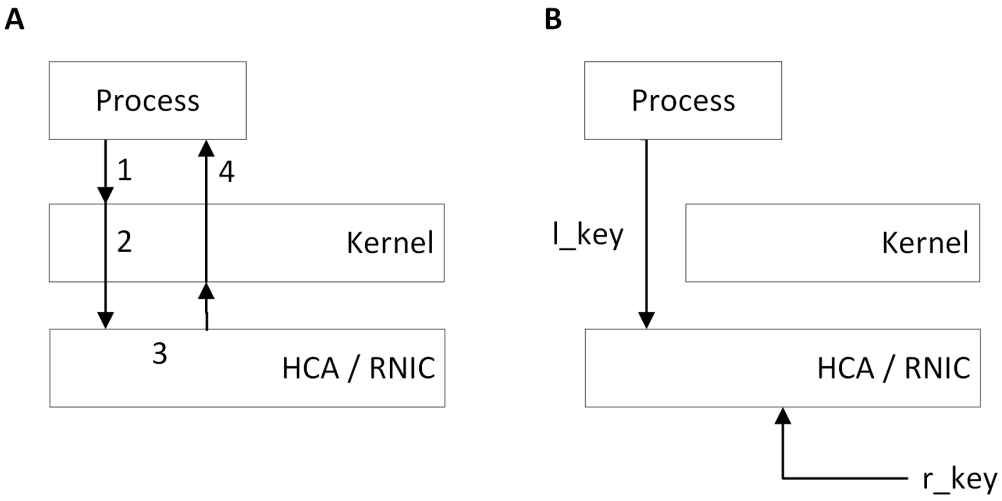

Figure 7. (A) Memory Registration (B) Memory Protection (adapted from [2])

2.4 Components

2.4.1 Channel Adapter

The terms VI Network Interface Card (NIC) and CA both refer to network interfacing hardware capable of supporting VIs. VI NIC typically refers to Ethernet compatible hardware, whereas CA refers to InfiniBand compatible hardware. As previously mentioned, VI refers to the virtualization of hardware interfaces which allows a single physical link to be split into many VIs. Figure 8 shows how context for each VI is stored in memory [3]. Each VI is typically given a time slice for execution on a physical link. The common hardware is controlled by swapping out VI contexts.

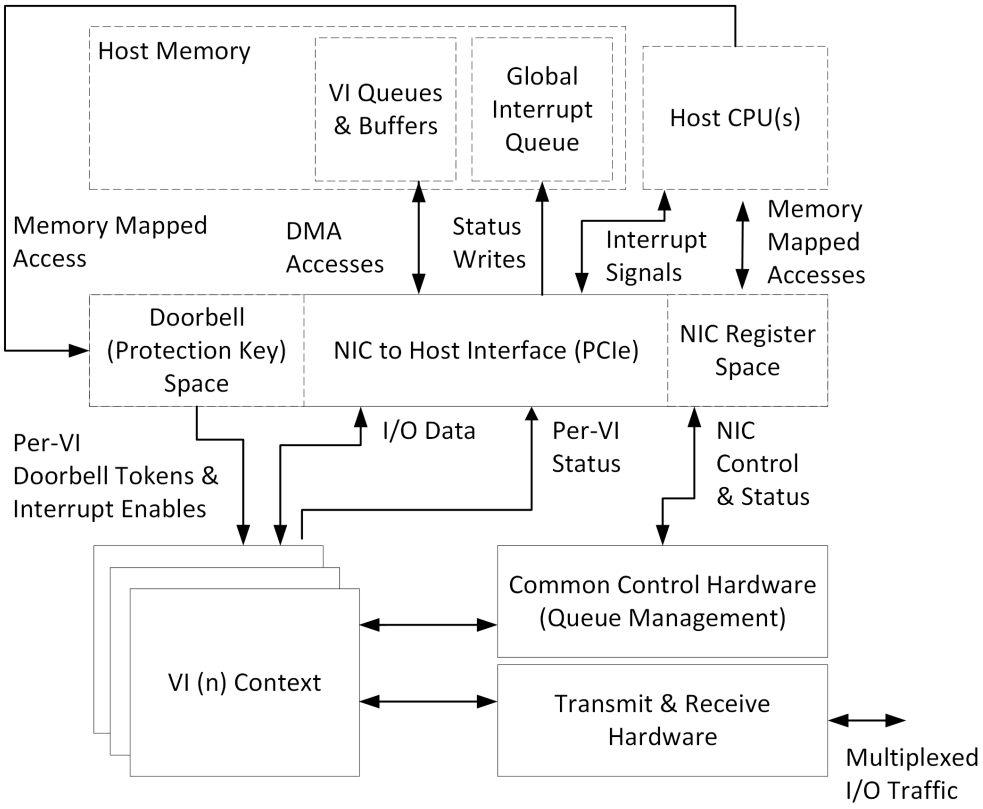

Figure 8. VI NIC Hardware Architecture (adapted from [3])

Every end node must have a CA in InfiniBand networks. Figure 9 shows an overview of a CA. CAs typically have a few physical links that are multiplexed into independent data streams called VLs. Each virtual lane is assigned a Quality of Service (QOS) on a packet-boundary basis. Most CAs support up to 16 VLs per physical link [17]. Use of VLs and QPs allow a significant portion of the functionality to be implemented in CA hardware to minimize communication latency and offload computational demands from the host CPU.

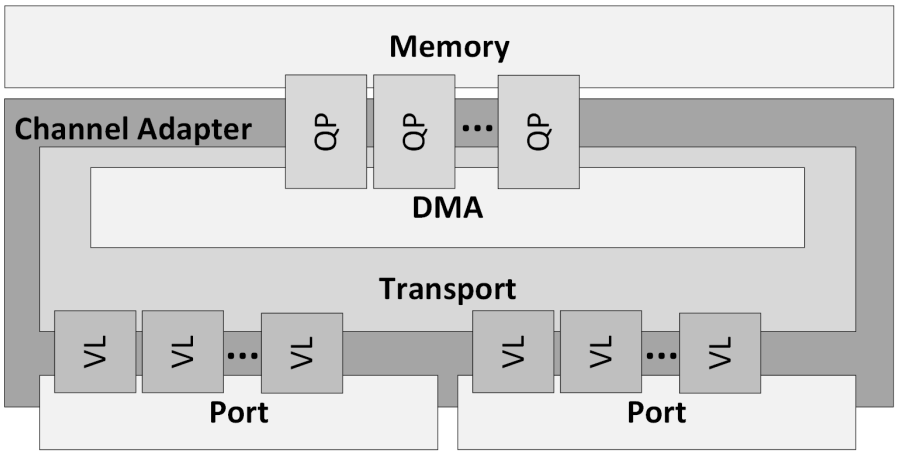

Figure 9. Channel Adapter (adapted from [2])

2.4.1.1 Single Root I/O Virtualization (SR-IOV)

SR-IOV technology allows a physical PCIe device, a Physical Function (PF), to present itself multiple times through the PCIe bus. SR-IOV enables multiple virtual instances, Virtual Functions (VFs), to be supported with separate resources. Each port of Mellanox ConnectX adapters are capable of supporting up to 127 VFs [18]. VFs can be provisioned separately, and can be seen as an additional device connected to the PF. SR-IOV enabled hypervisors provide Virtual Machines (VMs) with direct hardware access to network resources [18].

2.4.2 Subnet Manager

An SM is an entity attached to a subnet that is responsible for configuring and managing network devices including, switches, routers, and CAs. An SM can be supported by either a switch or a CA. The IBA is capable of supporting multiple subnet managers per subnet, but each subnet may only have one Master SM [1].

2.4.3 Switch

InfiniBand switches are the fundamental routing component for intra-subnet routing. Switches forward packets based on the destination LID address in the LRH of the packet. InfiniBand switches support unicast forwarding and may support multicast forwarding. An InfiniBand subnet manager configures switches by populating their forwarding tables [1]. Switches may be optionally configured to enforce partitions.

2.4.4 Router

InfiniBand routers are the fundamental routing component for inter-subnet routing. Routers forward packets based on their destination GID address in the GRH. Routers replace each packet LRH as the packet passes between subnets. Therefore, routers are not completely transparent to the end nodes.

Subnet Prefixes are used to distinguish each subnet. The subnet manager programs all ports with the corresponding Subnet Prefix and populates routing tables. The GID of each port is created by combining the subnet prefix with the Port GUID. The subnet prefix portion of each GID represents the path through the router [1].

2.5 Convergent Technologies

The layered abstraction of the Open Systems Interconnection (OSI) network model allows for the integration of novel network protocols with legacy systems. The

IBA was developed with that in mind, and today, InfiniBand is very flexible and backwards-compatible with the conventional five layer network-stack. In fact, most CAs are compatible with InfiniBand and Ethernet.

2.5.1 NVIDIA-Mellanox Virtual Protocol Interconnect

Virtual Protocol Interconnect (VPI) is a distributed messaging technology that supports both InfiniBand and Ethernet. VPI is auto-sensing of Layer-2 protocols and may be configured to work with either InfiniBand or Ethernet. This allows multi-port CAs to use one port for InfiniBand and the other for Ethernet. Integration of VPI into data centers and clusters allows InfiniBand and Ethernet networks to be hosted on the same hardware [17].

2.5.2 RoE, RoCE, and RoCEv2

RDMA over Ethernet (RoE), RDMA over Converged Ethernet (RoCE), and RDMA over Converged Ethernet Version 2 (RoCEv2) are the product of the native convergence of the InfiniBand network and transport layers with the Ethernet link layer. RoE encapsulates InfiniBand packets in Ethernet frames. RoE works natively in Ethernet environments and has all the benefits of InfiniBand verbs. Congestion control, multicast, prioritization, and fixed-bandwidth QOS are optional in (regular) Ethernet, but are required in the native InfiniBand link-layer. RoE, RoCE, and RoCEv2 are often used interchangeably, but Converged Ethernet (CE) is a lossless link-layer. CE uses all the features of the link layer of native InfiniBand [2].

Figure 10 provides a comparison of the InfiniBand, RoCE, and RoCEv2 network stacks. RoCE does not carry an IP header so it cannot be routed across boundaries of Ethernet L2 subnets using regular IP routers. RoCEv2 is a straightforward extension of the RoCE protocol that replaces the InfiniBand GRH with an IP header. This

allows RoCEv2 packets to traverse IP L3 routers [4]. The UDP transport header serves as a stateless encapsulation layer for the RDMA Tansport Protcol Packets over IP.

These convergent communication approaches exclusively affect the packet format on the wire because RDMA packets are generated and consumed below the API. Therefore, applications can operate over any form of RDMA service in a completely transparent way [4].

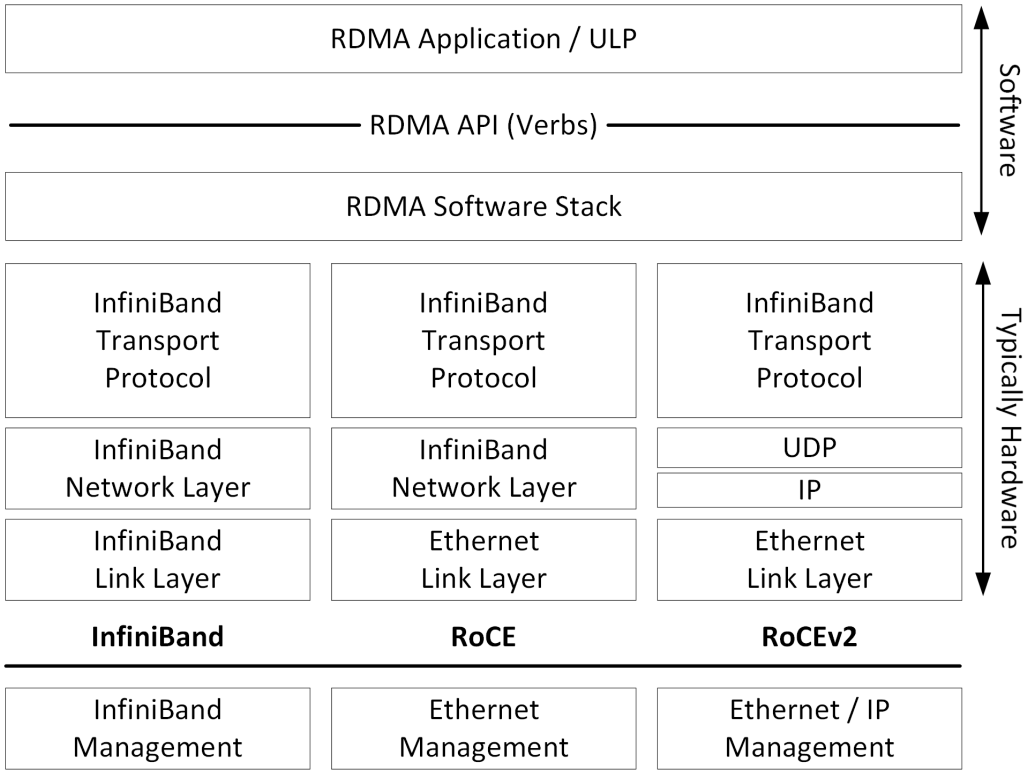

Figure 10. Comparison of Network Stacks (adapted from [4])

2.5.2.1 Comparison of RoCE and InfiniBand

RoCE delivers many of the advantages of RDMA using an Ethernet switched fabric instead of InfiniBand adapters and switches. This allows RoCE to be added

to legacy Ethernet switched fabric networks [19]. From an application perspective, both RoCE and InfiniBand present the same API and provide about the same set of services. There are two primary differences between Ethernet and InfiniBand beyond their use of different link-layers:

- **Fabric Management:** There is a fundamental difference between an RDMA fabric built on Ethernet using RoCE and one built on native InfiniBand [19]. InfiniBand relies on a central fabric management scheme in contrast to the distributed management system used commonly used by traditional Ethernet switched fabrics. Centralized management provides InfiniBand fabric managers with a high level view of the entire network fabric and facilitates several advanced features like partitioning and QOS. Management implications are an important difference between RDMA implementations based on RoCE and native InfiniBand.

- **Link Level flow control vs Data Center Bridging (DCB):** RDMA requires a lossless fabric. A lossless fabric is one where packets are not routinely dropped. Ethernet is mostly considered a lossy fabric because it frequently drops packets. Traditional Ethernet relies on TCP to provide reliable connections. InfiniBand uses a link level flow control to ensure packets are not dropped. RoCE accomplishes flow control similarly using DCB which adds five new specifications to the IEEE Ethernet specification [19].

2.6 NVIDIA-Mellanox Bluefield-2 Data Processing Unit

2.6.1 Hardware Architecture

The NVIDIA-Mellanox's Bluefield-2 DPU combines a ConnectX-6 DX network adapter with an array of ARM cores and IP Security (IPsec)/Transport Layer Security

(TLS) enabled hardware accelerators. The Bluefield-2 operates as an independent system that communicates with its host over 16 lanes of third/fourth generation PCIe, offering a theoretical transfer rate of 128/256 Gbps respectively. The card has two multi-function 100 Gbps ports, 16 GB of local Double Data Rate 4 (DDR4) Random-Access Memory (RAM), 8 ARM Cortex A72 pipeline processors, and local persistent storage. Each core has 48KB I-cache and 32KB D-cache. The ARM CPU also features 1 MB L2 cache per two cores and 6 MB L3 cache with plurality of eviction policies. The transfer rate of the Bluefield-2's DDR4 RAM is 3200 transfers per second (T/s). The card uses a tailored version of Ubuntu 20.04 provided by NVIDIA-Mellanox allowing developers to both develop new applications and deploy existing applications directly onto the card itself. These applications can process and modify traffic before it is ever seen on the host [20]. The Bluefield-2s, therefore, can host a wide variety of applications and services for networking, storage, and security [21]. Figure 11 shows the high-level hardware architecture of the Bluefield-2.

2.6.2 Software Architecture

The Bluefield-2 DPU software architecture is a combination of two preexisting standard off-the-shelf-components: 8 ARM Cortex A72 general-purpose processors as well as the ConnectX-6 Dx CA chipset. Each of these components has its own software ecosystem. As a result, the programmable software interfaces in the Bluefield-2 DPU come from existing standard interfaces for the respective components [6].

The ARM interfaces are standard Linux interfaces that are enabled by drivers and low-level code provided by NVIDIA. The ConnectX-6 Dx network controller related instances are identical to those of standalone network controllers. These interfaces take advantage of the Mellanox OpenFabrics Enterprise Distribution (OFED) software stack and InfiniBand verb-based interfaces to support software (Figure 12).

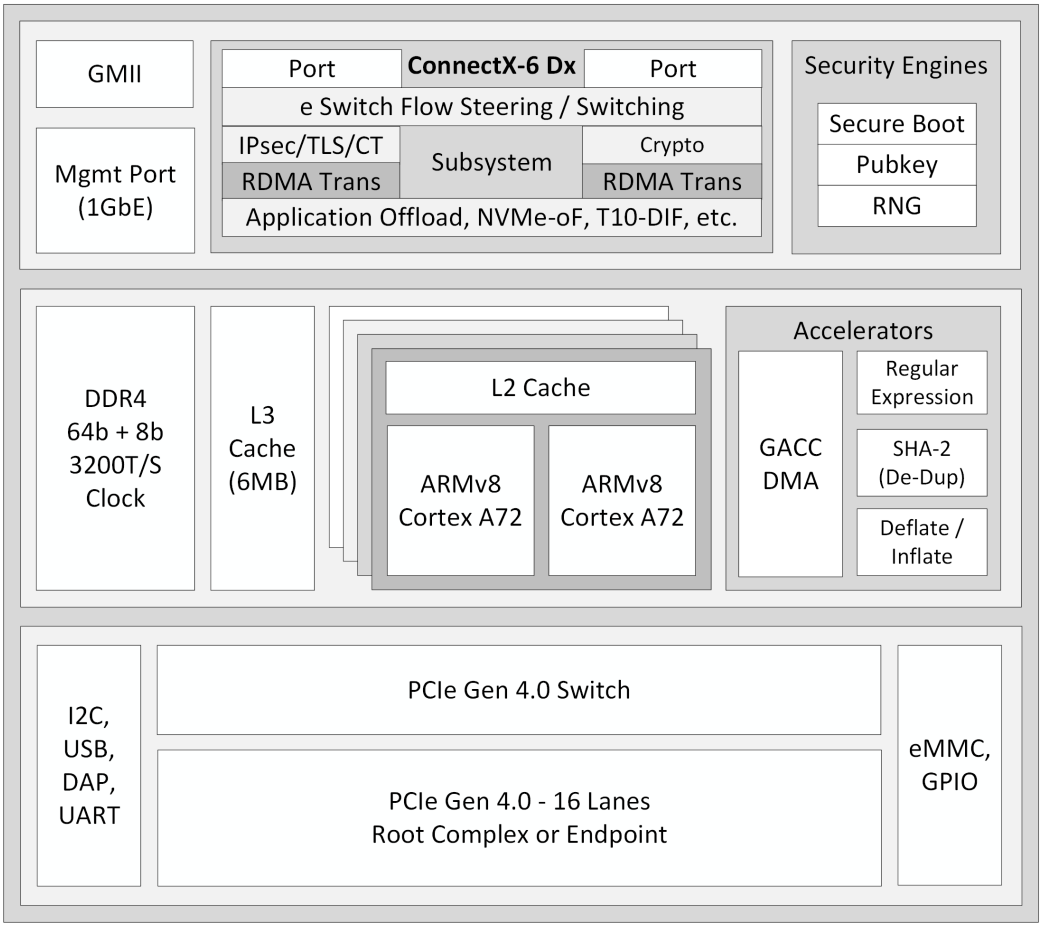

Figure 11. Bluefield-2 DPU Hardware Architecture (adapted from [5])

2.6.2.1 Cryptodev Linux Module

Cryptodev is a Linux device that allows access to Linux kernel cryptographic drivers. Cryptodev is a standalone Linux module [22].

The Bluefield-2 DPU Linux images provided by NVIDIA-Mellanox comes preloaded with cryptodev and several cryptology libraries (e.g., OpenSSL). Cryptodev is used to give userspace applications access to the hardware accelerators, and the cryptography libraries allow software encryption to be performed using the suite of ARM cores of the Bluefield-2.

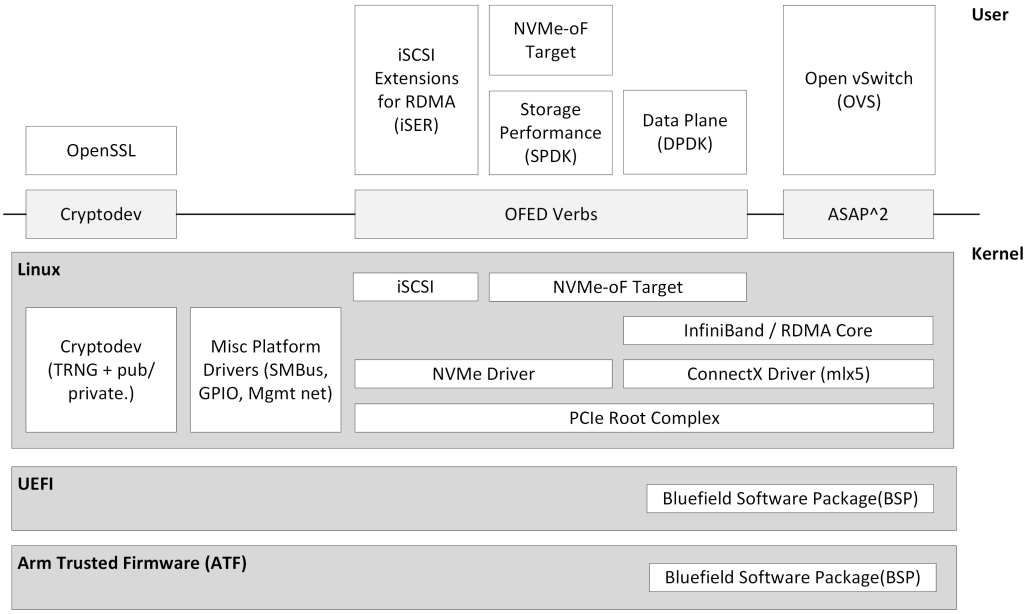

Figure 12. Bluefield-2 DPU Software Architecture (adapted from [6])

2.6.2.2 Mellanox OpenFabrics Enterprise Distribution for Linux

Mellanox OFED is a single VPI software stack that operates across all Mellanox network adapter solutions. The Mellanox version of OFED supports InfiniBand and Ethernet using an RDMA and kernel bypass APIs called OFED verbs [6]. Up to 100 Gbps Ethenet and InfiniBand are supported. Figure 13 shows the Mellanox OFED software stack.

2.6.2.3 Kernel Representors Model

The BlueField 1 and 2 DPUs use netdev representors to map each host side physical and virtual functions. Representors provide a tunnel for the Bluefield to pass traffic from the virtual switch or application running on the Arm cores to the relevant PF or VF on the Arm side. Representors can also create a channel for configuring the embedded switch of the Bluefield [23].

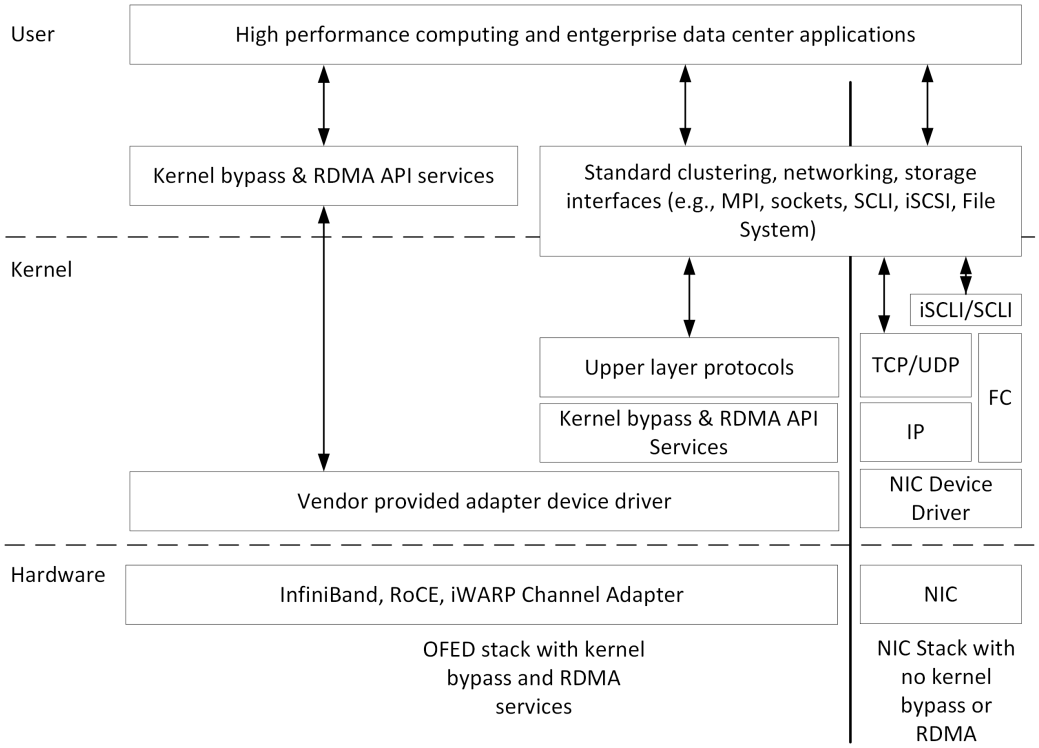

Figure 13. OFED Software Architecture (adapted from [7])

Representors connect virtual ports to OvS or any other virtual switch running on the Arm cores [23]. Each physical port of the Bluefield is typically assigned two representors. One representor is assigned to the uplink and the other is assigned to the host side PF. A representor is also created on the Arm side for each VF created on the host. The following naming convention is used for representors [23]:

1. Uplink representors: p<port_number>
2. PF representors: pf<port_number>hpf
3. VF representors: pf<port_number>vf<function_number>

2.6.2.4 Modes of Operation

The Bluefield-2 DPU has two main modes of operation:

- **Separated Host:** This is the default configuration. The Embedded CPU Function (ECPF) and the function exposed to the host are symmetric in this mode. Each function has its own MAC address and is able to send and receive Ethernet and RoCE traffic [8].
- **ECPF Ownership Mode (SmartNIC Mode):** The ARM subsystem owns and controls the NIC resources and functionality in this mode. There is still a network function exposed to the host in ECPF Mode, but it has limited privileges. There are two ways to pass traffic to the host interface in ECPF Mode. Representors can be used to forward traffic to the host. This method forces every packet to be handled by the network interface on the embedded Arm side. Handling traffic in software is computationally expensive. In order to improve performance, traffic can alternatively be pushed to an embedded switch which offloads this traffic to hardware [8].

Figure 14 shows how traffic is handled in Separated Host and ECPF Mode. Traffic is most commonly forwarded by a virtual switch when the Bluefield-2 DPU is configured in ECPF mode.

2.6.2.5 Accelerated Switching and Packet Processing

The ARM subsystem takes full control of the Bluefield-2 DPU when it is configured in ECPF Mode. In this mode, a virtual switch is typically required to forward traffic between the host and arm core facing interfaces. The Bluefield-2 DPU supports accelerated virtual switching through the use of Accelerated Switching and Packet Processing (ASAP^2) [24].

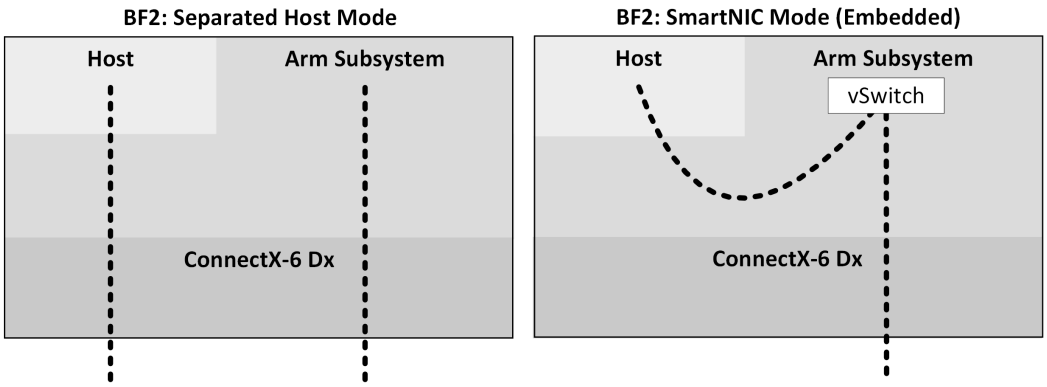

Figure 14. Bluefield-2 DPU Modes (adapted from [8])

Early router and switch implementations processed packets with CPUs. This has since become known as the slow path. Modern routers, switches, and NICs instead offload packet processing and forwarding to a hardware fast path. The hardware fast path is typically implemented using Application Specific Integrated Circuits (ASICs) or network processors [24].

Offloading packet forwarding to the NIC significantly improves network performance. However, not all NICs offloads are compatible with compute and network virtualization. ASAP^2 is the proprietary solution used by Mellanox to solve this issue. ASAP^2 supports accelerated virtual switching in server NIC hardware. This capability is enabled by an Embedded Switch (eSwitch) in the hardware that implements switching between virtual NICs. This pipeline-based programmable eSwitch is built into the NIC, and enables the NIC to handle a large portion of packet processing and forwarding in hardware [24].

2.7 Relevant Technologies

2.7.1 Data Plane Development Kit

The DPDK is a set of software libraries and drivers that run in userspace in order to accelerate packet-processing workloads. The DPDK is an open-source project that supports all major CPU architectures. Interestingly, DPDK has been instrumental in driving the use of general-purpose CPUs in modern networks [25].

Architecturally, DPDK sits alongside the OS kernel. As a result, DPDK rides directly above the hardware in the network stack and is capable of accelerating specific networking functions [25]. The Bluefield-2 software package provided by NVIDIA-Mellanox comes with a tailored version of DPDK pre-installed. The Bluefield-2 DPDK package only supports a few applications and is very limited. Additional DPDK applications and features can be added to the Bluefield-2 DPU by cloning the online DPDK repository directly onto the card. This research uses DPDK version 20.11 in addition to the version provided by Mellanox.

2.7.1.1 MLX5 Poll Mode Driver

DPDK uses the MLX5 Poll Mode Driver (PMD) to facilitate kernel bypasss for send and receive queues and allocate system resources to DPDK processes. DPDK PMDs achieve fast packet processing and low-latency by avoiding the overhead of interrupt processing. The MLX5 PMD is dependent on the libiverbs library which allows programs to use RDMA verbs for direct access to RDMA hardware from userspace [26].

2.7.1.2 iPerf3 vs DPDK Pktgen

`iPerf3` is an open-source traffic generator that is intended for use in Ethernet networks using the traditional TCP/IP network stack. Preliminary tests using `iPerf3`

indicated that `iPerf3` was unable to generate Ethernet traffic fast enough to saturate the PCIe bus between the workstations and Bluefield-2 DPU. In response, the `Pktgen` DPDK application was installed on each workstation and the performance of the two applications was directly compared.

`Pktgen` achieved significantly better performance than `iPerf3` achieving a throughput peaking near 100 Gbps. These preliminary tests provide an example of the performance benefit realized by using the fast data path provided by DPDK applications.

2.7.1.3 Running Pktgen

Once `Pktgen` is installed on each workstation, the following set of commands can be used to configure and run `Pktgen`:

```
$ pktgen -c fffff -n 4 --socket-mem 1024 -w 0000:03:00.1
   -- -T -p 1 -P -m "[1:2-3].0"
```

2.7.2 IPsec

Figure 15 shows the format of an IPsec datagram using Encapsulating Security Payload (ESP) and tunnel mode. The IPsec datagram still meets the requirements of an IPv4 datagram. Within the IPsec datagram, the payload consists of an ESP header, the original IP datagram, an ESP trailer, and an authentication field.

IPsec headers and trailers create additional overhead and must be accounted for when configuring the Maximum Transmission Unit (MTU) of network interfaces. In total, the protocol suite can add over 100 bytes of overhead to IP datagrams. As a result, care must be taken to ensure that the payload, when combined with the IPsec headers, does not exceed the MTU of the network link. If it does, the resulting packets could be fragmented or dropped.

IPsec is compatible with RoCEv2. RoCEv2 uses IP at the network layer, and RoCEv2 packets can be encapsulated in an Ethernet frame. Conversely, IPsec is not

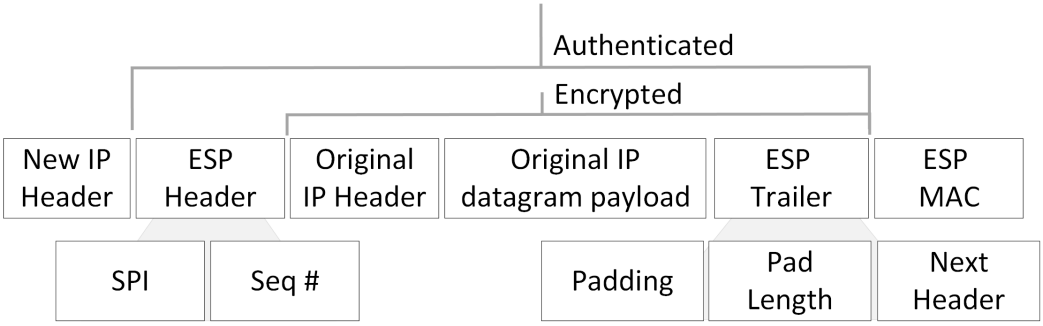

Figure 15. IPsec Datagram Format (adapted from [9])

compatable with RoCE or native InfiniBand packets because they use the InfiniBand network layer (Figure 10).

2.8 Tools

This research used the tools listed in Table 1 to conduct throughput tests and verify network configurations:

2.9 Related Research

2.9.1 Vulnerabilities

In 2020, Rothenberger and colleagues [10] performed a cyber vulnerability assessment of the IBA. In their assessment, Rothenberger et al. created an adversary model and analyzed existing security mechanisms in RDMA fabric architectures including memory protection key generation, QP number generation, memory regions, memory windows, and protection domains. Following their analysis, Rothernberger et al. identified ten vulnerabilities in the IBA, and proposed eight mitigation mechanisms that are readily deployable by RDMA applications without requiring changes to hardware or InfiniBand itself.

Table 1. Data Gathering and Analysis Tools

Tool Name	Description
top	Linux command line tool used to show real-time view of the system. Top lists the CPU utilization, virtual memory use, task priority, and more for each process running on the system [27]
numactl	Linux command line tool used to run processes with a specific non-uniform memory access (NUMA) scheduling or memory placement policy [28]
scapy	Interactive packet manipulation tool used to send or receive Ethernet packets [29]
vmstat	Linux command line tool used to collect information about processes, memory, paging, block IO, traps, and cpu activity [30]
netstat	Linux command line tool used to print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships [31]
iPerf3	Linux command line tool used for active measurements of the maximum achievable bandwidth on IP networks [32]
libreswan	Open-source, software implementation of IPsec [33]
Open vSwitch	Production quality, multilayer virtual switch. One of the most popular implementations of OpenFlow [34]
InfiniBand Fabric Utilities	NVIDIA-Mellanox library which includes a variety of diagnostic and performance utilities [35]
tcpdump	Open-source command line packet analyzer used for sniffing traffic [36]
Wireshark	Open-source GUI packet analyzer used for decryption [37]

The cybersecurity vulnerability assessment conducted by Rothenberger et al. considered four attacker models. First, this assessment considered an adversary that has rightfully obtained access to a different end node than the victim (e.g., renting an instance in a public cloud). Figure 16 shows that the attacker can communicate with other end nodes through the use of RDMA services. The second adversary model considers attackers that actively compromise end nodes (Figure 16). Having gained root administrative access, these attackers are capable of fabricating and injecting messages.

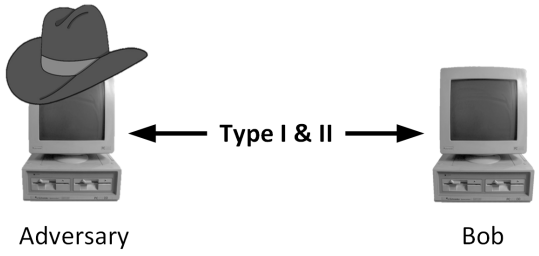

Figure 16. Type I and II InfiniBand Adversary (adapted from [10])

Third, Rothenberger et al. considered network-based attackers where the attacker is located on the path between the victim and the service. Figure 17 shows that on-path attacks can be conducted by attackers that have compromised routers, switches, or are able to tap a link between victims (e.g., malicious bump-in-the-wire devices). These adversaries are capable of passively eavesdropping, injecting, dropping, delaying, replaying, or altering messages.

Lastly, Rothenberger et al. considered an adversary that makes use of RDMA as a covert channel for exfiltrating data (FIgure 18). Rothenberger et al. demonstrate that a Type IV adversary is capable of manipulating code or libraries executed by the victim (e.g., using malware) such that it establishes an RDMA connection to an RDMA capable attacker in the same network. This attack allows that adversary to

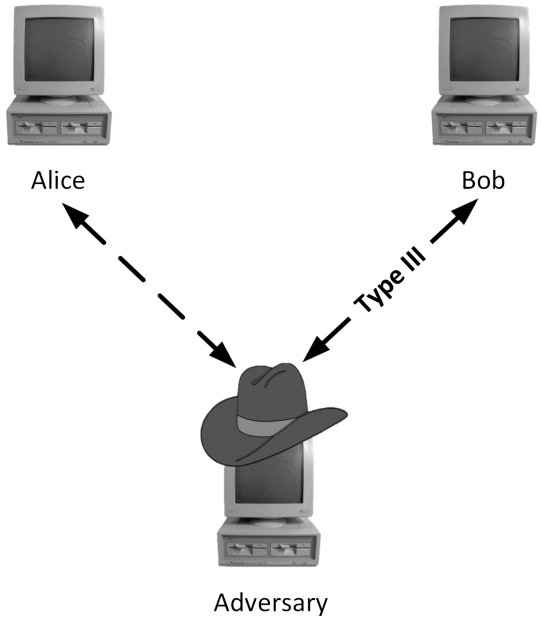

Figure 17. Type III InfiniBand Adversary (adapted from [10])

"silently" read and write to the memory of the victim process.

Rothenberger et al. suggested that the existing IBA security mechanisms can be circumvented due to the lack of endpoint and packet authentication. As current RDMA systems enforce no source authentication, an adversary can impersonate any endpoint by injecting packets that seem to belong to an established connection by another client. Further, connections using the Reliable Connection (RC) transport service QPs are sensitive to content request headers. Memory errors, such as incorrect operation numbers, or an inconsistency between payload length and Direct Memory Access (DMA) length immediately lead to unrecoverable errors. These errors will cause the CA to transit the QP to the error state and the QP to disconnect [10]. These unrecoverable error states present an opportunity for Denial-of-Service (DoS) attacks on InfiniBand networks.

Rothenberger et al. state that the aforementioned vulnerabilities to packet injec-

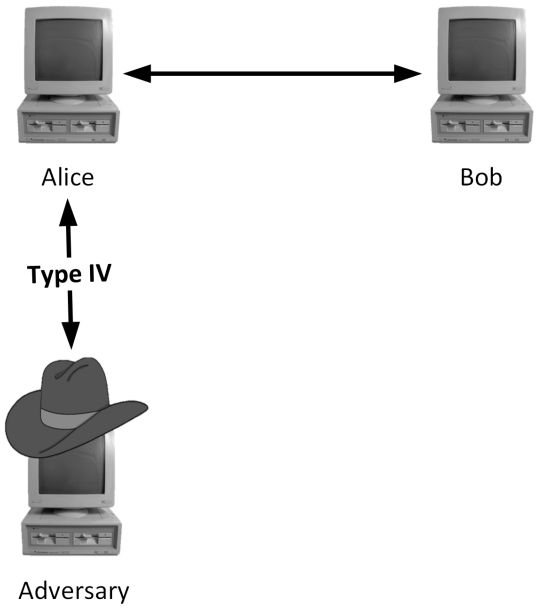

Figure 18. Type IV InfiniBand Adversary (adapted from [10])

tion and DoS attacks can be mitigated through the use of encryption and authentication at any layer of the protocol stack or in-network filtering. Rothenberger et al. suggest that network administrators could deploy a filtering mechanism at the ingress of the network and attempt to effectively prevent an attacker from injecting spoofed packets from outside the InfiniBand network. Encryption and authentication integrated into the IBA can prevent information from leaking to attackers and prevent message tampering as the RDMA header is authenticated. With these mitigations in place, it becomes difficult for an attacker to spoof RDMA header fields and prevents attacks based on packet injection [10].

Table 2 lists all of the attacks and proposed mitigations offered as a result of this vulnerability assessment. Of the proposed mitigation techniques, encryption and authentication pose the most significant challenges. The unique network stack of native InfiniBand reveals the need for a new encryption method that is not dependent

on IP addresses or Ethernet frames. Further, the computational requirements of cryptographic ciphers are at odds with the high data rates supported by InfiniBand networks. This research explores the capabilities of hardware offload and acceleration technologies to support encryption and authentication in high-performance networks.

2.9.2 sRDMA

Also in 2020, Taranov and colleagues [38] proposed sRDMA, a protocol that extends the IBA by designing a connection mode that provides encryption and authentication for RDMA based symmetric cryptography. Benchmark testing performed by Taranov et al. shows that software implementations of sRDMA are computationally demanding due to the data movement overhead in the current implementation. Taranov et al. suggest that the datapath could be optimized with a different architecture using specialized programmable packet processing units. An open-source implementation of sRDMA is available for download from Scalable Parallel Computing Laboratory (SPCL), which is an organization that performs research in all areas of scalable computing. Unlike IPsec, sRDMA is compatable with InfiniBand and RoCE because it encrypts at the transport layer. Encrypting at the transport layer preserves InfiniBand network layer headers that are necessary for packets to be routable once they are encrypted.

2.9.3 IPsec over RoCEv2

In 2005, Romanow and colleagues [39] wrote Request for Comments (RFC) 4297 and stated, "RDMA protocols must permit integration with Internet security standards, such as IPsec and TLS". Romanow et al. explain that native convergence of RDMA and IP necessitates that RDMA protocols permit integration with Internet security standards, such as IPsec and TLS.

Table 2. InfiniBand Vulnerabilities and Proposed Mitigation (adapted from [10])

Attack	Attack Model	Mitigations
Packet Injection by Impersonation	Type II & III	(1) Encryption / Authentication (2) In-Network Filtering (3) Random QP numbers
DoS by Transiting QPs to an Error State	Type II & III	(1) Encryption / Authentication (2) In-Network Filtering (3) Random QP numbers
Unauthorized Memory Access	Type I & III	(1) Random R_Keys (2) Multiple Protection Domains (PDs) (3) Type 2 Mem Windows
Resource Exhaustion DoS	Type I	(1) Per-Client Resource Constraints
RDMA Covert Channel	Type IV	(1) Hardware Counters

Fast forward to 2020, and the ConnectX-6 Dx Channel adapter is the first System on Chip (SoC) in its class to offer full IPsec acceleration for both Ethernet and RoCEv2. The RoCEv2 protocol uses UDP and IP at the transport and network layers respectively, thus, RoCEv2 is compatible with IPsec. With IPsec full offload, the IPsec encryption/decryption and ESP header encapsulation/decapsulation are done in hardware. Offloading IPsec operations to hardware significantly reduces the computational overhead of IPsec [6].

2.9.4 AFIT: Securing InfiniBand

As mentioned in Section 2.3.3, several communication models, like RoCE, combine features of InfiniBand and Ethernet. As a result, most CAs and DPUs on the market today support both InfiniBand and Ethernet at the data link-layer. RoCEv2, unlike native InfiniBand, uses IP addresses at the network layer, and is compatible with IPsec encryption. Mireles and colleagues [40] sought to characterize the capabilities of NVIDIA Mellanox's Innova Flex SmartNIC and Innova IPsec Ethernet Adapter to offload and encrypt RoCEv2 traffic with IPsec-enabled hardware. Mireles et al. found

that the Innova Flex SmartNIC and Innova IPsec Ethernet Adapter were unable to offload RoCEv2 traffic to the IPsec-enabled hardware.

Hintze and colleagues [41] sought to demonstrate offloading and encrypting RoCEv2 traffic using the suite of IPsec enabled hardware accelerators on-board the NVIDIA-Mellanox Bluefield-1 DPU. Hintze et al. found that the Bluefield-1 DPU was also unable to encrypt RoCEv2 traffic in hardware.

2.9.5 Encryption and Authentication Trade-Offs

The research efforts mentioned above have identified the necessity of adding encryption and authentication to RDMA traffic. Table 3 lists the various encryption methods discussed in this section, and specifies the compatibility of each method with various forms of RDMA traffic. Interestingly, no single method is capable of supporting every implementation of RDMA.

2.10 Background Summary

This chapter presents a brief technical summary of the IBA and how its security features relate to those of comparable interconnect technologies. It provides background on key InfiniBand technologies and open-source tools as they pertain to this work. It observes related research into the development of DPU hardware offloading

Table 3. Encryption and Authentication Methods

	Ethernet	InfiniBand	RoCE	RoCEv2
Application/Transport (TLS)	Applicable	Not Applicable	Not Applicable	Applicable
Transport (sRDMA)	Not Applicable	Applicable	Applicable	Not Applicable
Network (IPsec)	Applicable	Not Applicable	Not Applicable	Applicable
Link (Custom)	Applicable	Possible	Possible	Applicable

capabilities, encryption and authentication schemes, and current efforts in securing RDMA fabric architectures. While research has been conducted on the performance of different encryption ciphers and methods, little work has provided insight into the utility of using hardware acceleration to provide line-rate encryption. This thesis contributes to the field of securing the IBA, specifically encryption of RDMA traffic, by characterizing the capabilities of the Bluefield-2 DPU to perform encryption in hardware and software.

III. TNAP and MiTMVP Design

3.1 Overview

This research introduces the TNAP and MiTMVP for characterizing the capability of the Bluefield-2 to perform end-to-end encryption in hardware and software. The TNAP is a testbed capable of generating Ethernet and RDMA traffic at rates exceeding 100 Gbps. Saturating the Bluefield-2 DPUs in the TNAP allows the performance of network adapters to be characterized. The MiTMVP provides a monitoring solution capable of passively sniffing Ethernet and RoCEv2 traffic between the end nodes of the TNAP.

Readily available sniffing tools running on the workstations or network adapters are incapable of sniffing RDMA traffic in the TNAP. Kernel bypass traffic like RDMA does not pass through the Linux kernel, and is inaccessible to TCP/IP monitoring tools. Additionally, the TNAP network topology does not allow encrypted traffic to be monitored if encryption and decryption are handled by the network adapters. Traffic is only available to the TNAP workstations and network adapters after the traffic has passed through a decryptor. Conventional approaches verify end-point encryption using fast switches configured with port mirroring. This allows the switch to duplicate port traffic and forward it to a third device. However, a 100 Gbps Ethernet switch was not available for this research.

The MiTMVP provides the same capabilities as a switch by integrating a readily available Bluefield-1 as a hot pluggable MiTM. The Bluefield-1 directs network traffic through the TCP/IP network stack so traditional monitoring tools running on the card are able to passively sniff encrypted traffic. This MiTM solution allows both kernel bypass and encrypted traffic to be monitored, but the MiTMVP is only used for verification purposes because it introduces significant latencies in the connection

between the end nodes of the TNAP. This chapter provides a detailed description of the TNAP, the MiTMVP, and their respective roles within the experiment.

3.2 Testbed for Network Adapter Performance

TNAP is used to facilitate performance testing of 100 Gbps network adapters. The Bluefield-2 DPU is the subject of the throughput tests in this research. As depicted in Figure 19, TNAP includes an identical pair of HP Z840 workstations each with its own Bluefield-2 DPU installed. The Bluefield-2 DPUs are connected in tandem with a 100 Gbps fiber optic link.

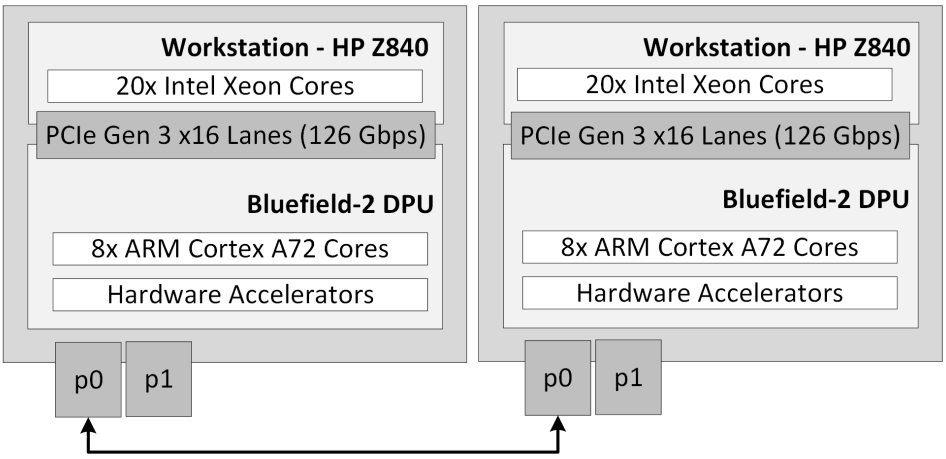

Figure 19. Diagram of TNAP Components

3.2.1 TNAP workstations

HP Z840 workstations have up to PCIe Gen 3 which is capable of generating 126 Gbps (using sixteen lanes, and after accounting for encoding overhead). Thus, PCIe Gen 3 provides sufficient throughput to overwhelm the system under test, namely the ConnectX-6 Dx in the Bluefield-2s, which are only capable of 100 Gbps. The HP Z840s used in this research have 20 Intel Xeon Cores, 256 GB of RAM, a 1 TB hard

drive, and Ubuntu 20.04 installed. In order to interface with the Bluefield-2 DPU via the PCIe bus, the Mellanox edition of OFED is installed on each workstation.

3.2.2 Optical cable connections

The Bluefield-2 DPUs are connected in tandem using an NVIDIA-Mellanox 100 Gbps QSFP28 MMF Active Optical Cables (AOCs) which are VCSEL-based (Vertical Cavity Surface-Emitting Laser) active optical cables designed for use in 100 Gbps systems [42]. These links are hot pluggable, so they are easy to install and replace.

3.3 MiTM Verification Process

The network topology used for the MiTMVP inserts an intermediate workstation installed with a Bluefield-1, between the TNAP endpoints as shown in Figure 20. Each Bluefield in this configuration rides on sixteen lanes of PCIe Gen 3, and the DPU ports are connected by 100 Gbps AOCs.

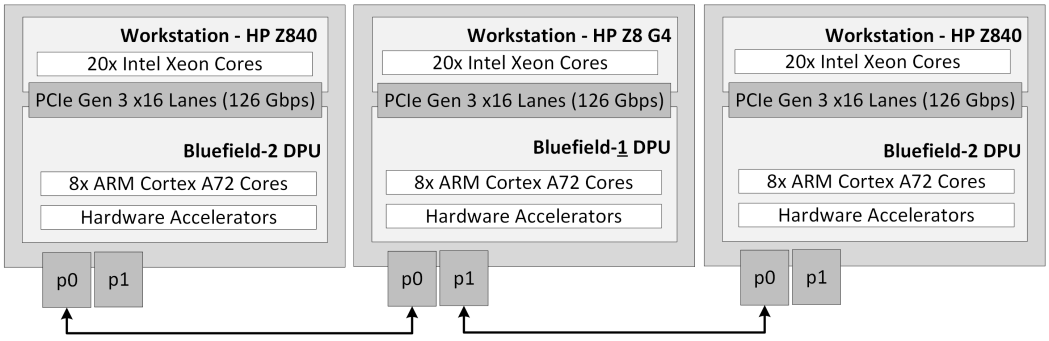

Figure 20. Diagram of MiTMVP Components

3.3.1 MiTMVP workstation

Similar to the HP Z840 workstations, the HP Z8 G4 used as the intermediate workstation in the MiTMVP has up to PCIe Gen 3 which is also capable of 126 Gbps

using sixteen lanes. The HP Z8 G4 also has 20 Intel Xeon Cores, 256 GB of RAM, a 1 TB hard drive, and Ubuntu 18.04 installed.

3.3.2 MiTMVP DPU

The NVIDIA-Mellanox's Bluefield-1 DPU combines a ConnectX-5 DX network adapter with an array of ARM cores and hardware accelerators. The Bluefield-1 operates as an independent system that communicates with its host over 16 lanes of third/fourth generation PCIe, offering a theoretical transfer rate of 126/252 Gbps respectively. The card itself includes two multi-function 100 Gbps ports, 16 GB of local DDR4 RAM, 16 Cortex A72 ARM cores, and local persistent storage. Each core has 48KB I-cache and 32KB D-cache. The ARM CPU also features 1 MB L2 cache per two cores and two banks of 6 MB L3 cache with sophisticated eviction policies. The card uses a tailored version of Ubuntu 18.04 provided by NVIDIA-Mellanox.

3.3.3 Passive Sniffing

Passive sniffing is used to capture Ethernet and RoCEv2 traffic from the Bluefield-2s. Sniffing occurs on the Bluefield-1 DPU acting as a MiTM using `Tcpdump`. `Tcpdump` can be used to sniff traffic on either physical port on the Bluefield-1. When operating `Tcpdump`, the interface ("p1") and write capture to file ("<filename>.pcap") options are set. Sniffing is initiated using:

```
$ tcpdump -i p1 -w <filename>.pcap
```

3.3.4 Verification

As previously noted, Ethernet traffic analyzers cannot sniff RDMA traffic in traditional network topologies because kernel bypass packets never traverse the TCP/IP stack [41]. The MiTMVP solves this issue by inserting a Bluefield-1 DPU in-line

between the two endpoints in the TNAP. Using a Bluefield-1 DPU as a bridge forces network traffic thought the traditional TCP/IP network stack, and allows Ethernet traffic analyzers to actively sniff traffic on the card itself. The implementation of the MiTMVP used in this research uses OvS as a virtual bridge between the two physical ports of the Bluefield-1 DPU. Forwarding traffic with this method significantly degrades network performance, but allows Ethernet traffic analyzers to sniff network traffic. This capability is used in this research to monitor network connections and verify properly functioning encryption configurations.

Verifying IPsec encryption in this research follows these steps:

1. Configure each Bluefield-2 DPU with Ethernet at the link-layer and configure the desired encryption settings.

2. Place network in the monitoring configuration (Figure 20).

3. Sniff traffic sent across the network by running `Tcpdump`, and write sniffed traffic to a .pcap file.

4. Run the following Python code to generate an ICMP packet containing a human readable string.

```
#! /usr/bin/env python3
from scapy.all import send, IP, ICMP
send(IP(src="10.0.0.3",dst="10.0.0.4")/ICMP()/"Hello
   World")
```

5. Verify IPsec encryption by uploading the .pcap file to `Wireshark`. The original human readable string should appear as cipher text within the ICMP packet. Proper encryption can be verified by decrypting the cipher text using the known encryption key. `Wireshark` provides an automated feature for decryption.

6. Place network in performance configuration by connecting Bluefield-2s in tandem (Figure 19), i.e., removing the MiTMVP system.

7. Run throughput test with verified network configuration.

3.3.4.1 OvS Configuration

OvS bridges can be configured on the Bluefield-2 DPUs using the following commands:

```
$ ovs-vsctl add-br ovsbr1
$ ovs-vsctl add-port ovsbr1 p1
$ ovs-vsctl add-port ovsbr1 pf1hpf
$ ifconfig ovsbr1 up
```

The example commands above create a virtual bridge between the uplink PF of port 1 (p1) and the host facing PF of port 1 (pf1hpf).

3.4 Design Summary

This chapter describes each component of the TNAP and MiTMVP. The design presented is an effective testbed that can be used for network adapter performance characterization and evaluation.

IV. Research Methodology

4.1 Objective

The Bluefield-2 offers several different hardware offload and acceleration features that can operate directly on network traffic without routine involvement from the ARM CPU. This allows the ARM multi-core CPU to orchestrate the hardware to perform operations on traffic at high rates rather than processing all traffic directly. This research aims to characterize the capabilities of the hardware and software features of the Bluefield-2 DPU. Specifically, the experimentation attempts to accomplish three objectives:

1. Characterize the capability of the Bluefield-2 DPU to offload and accelerate IPsec encryption of Ethernet traffic and RoCEv2 traffic.

2. Characterize the capability of the Bluefield-2 DPU to encrypt traffic in software using its ARM CPU.

3. Build and characterize the performance of DPDK applications for both Ethernet and RoCEv2 traffic.

Exploring pre-configured settings of the Bluefield-2 DPU ports, hardware acceleration, and software technologies reveals the efficacy of the readily available security capabilities offered by this network adapter. Additionally, investigating the programmable capabilities of the card quantifies some of the available performance improvements offed by third party and custom security applications.

4.2 System Under Test

Figure 21 displays the system under test (SUT) and component under test (CUT) diagram. Response variables, or metrics are described in Section 4.3. Uncontrolled

variables are examined in Section 4.5. Section 4.6 discusses parameters that do not change throughout each experiment such as computing parameters. Finally, Section 4.7 describes the purpose of each configuration and treatment.

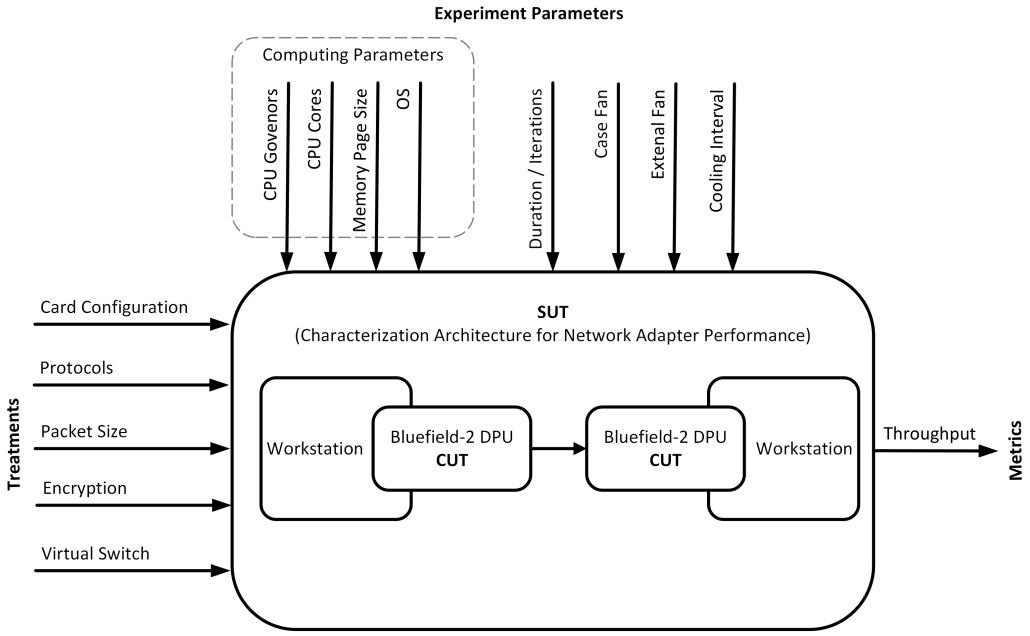

Figure 21. System Under Test and Component Under Test Diagram

4.3 Response Variables

Throughput is the response variable actively measured throughout this research. CPU utilization is also noted throughout this research, however, CPU utilization is used as a subjective metric. The `top` linux command line tool provides a high-level view into how much the workstation and Bluefield-2 DPU CPUs are utilized by the data path during performance tests.

Pktgen is a DPDK traffic generator that reports network performance in terms of packets transmitted and received. The average throughput, R, of each trial can be calculated using the equation

$$R = \frac{P * S \text{ bytes} * 8\frac{\text{bits}}{\text{byte}}}{T \text{ sec} * 1.0 * 10^9 \frac{\text{bits}}{\text{Gb}}} \tag{1}$$

where P is the number of packets received by the `Pktgen` server receive queue (RX), S is the size of the packets being transmitted, and T is the total duration of the test.

4.4 Control Variables

The primary goal of this experiment is to measure the average throughput supported by various configurations of the Bluefield-2 DPU. The card configuration, network protocols, packet size, virtual switch, and encryption settings are the primary factors in this experiment.

4.5 Uncontrolled Variables

A consequence of generating network traffic using a workstation is that processes will occasionally get evicted from CPU cores by the host OS. During performance testing, this could interrupt the flow of traffic sent to the Bluefield-2 DPU if the traffic generating process is evicted from its CPU core. Although tools exist which may help to mitigate process eviction effects, such as `isolcpus`, these were not employed as part of this data collection.

This research uses three replicates of each treatment to reduce the effect of uncontrolled variables like process eviction on average system performance. Three replicates were used throughout this research because preliminary performance test results had low variance. In fact, the results of many replicants were identical. Therefore it is

reasonable to assume that either process evictions have little effect on the observed throughput or are rare events. In either case, three replicants are sufficient for noting outliers and performing analysis of variance for statistical analysis.

4.6 Experiment Parameters

Throughout the course of experimentation, several factors are held constant to limit the scope of the experiment:

1. **Computing Parameters:** The operating systems, resources (memory, CPU, and disk space), script languages, and hardware are held constant.

2. **Test Duration:** The duration of each trial is held constant. The duration of Ethernet tests are measured in seconds, and RDMA tests are measured in iterations. Ethernet tests are conducted for 60 seconds, and RDMA tests are conducted for 100,000 iterations.

3. **Cooling:** Thermal considerations play a role in performance testing because processors prevent overheating by throttling their clock rates. In order to ensure trials are independent of run order, the case fans in the workstations are held at 75% capacity and a house fan is added to circulate cool air into the testing environment. Additionally, noise introduced by heat is further minimized by performing trials in a random order.

4.7 Experimental Design

The experimentation of this research is comprised of three distinct experiments. First, this research characterizes the hardware offload and acceleration capabilities of the Bluefield-2. Second, this research investigates the capability of OvS and DPDK virtual bridges to forward traffic while using the MiTMVP network topology. Last,

this research characterizes the capability of the Bluefield-2 CPU to encrypt network traffic.

4.7.1 Experiment 1: Hardware Acceleration Characterization

Characterizing the performance of the hardware accelerators of the Bluefield-2 DPU is conducted in two configurations. First, `Pktgen` is used as the traffic generator for testing Ethernet traffic within the TCP/IP network stack. The results of the TCP/IP performance testing provide a baseline for the performance improvement realized by using hardware acceleration. Second, the `NVIDIA-Mellanox Fabric Utilities` are used to characterize the capability of the hardware accelerators to encrypt RDMA traffic using RoCEv2 as the hardware transport protocol.

4.7.1.1 Ethernet

Preliminary tests show that the Pktgen DPDK application is capable of generating TCP/IP, Ethernet traffic at rates exceeding 100 Gbps. This allows the TNAP testbed to characterize the limits of the Bluefield-2 capability to offload and accelerate IPsec encryption.

This set of treatment in Experiment 1 test the capability of the Bluefield-2 to forward plain text Ethernet traffic with and without the use of harware offloads. Table 4 lists all of the factor levels tested in this portion of Experiment 1, and Table 5 lists all of the treatments.

Sending plain text without hardware acceleration: The first set of treatments tested in Experiment 1 measure the baseline performance of the Bluefield-2 DPU using TCP and Ethernet at the transport and link layers respectively. This treatment does not use the fast data path provided by hardware acceleration on the Bluefield-2 DPU. Rather, this treatment relies on a virtual switch to forward traf-

Table 4. Experiment 1: Ethernet Factors and Levels

Factor	Level(s)
Packet Generator	(1) Pktgen
Virtual Switch	(1) OvS
Bluefield-2 Configuration	(1) Plain Text (2) Plain Text HW Offload (3) IPsec HW Acceleration
Maximum Segment Size (Bytes)	(1) 64 (2) 128 (3) 256 (4) 512 (5) 1024 (6) 1518 (MAX)

Table 5. Experiment 1: Ethernet Treatments

Treatment	Bluefield-2 Configuration	Maximum Segment Size (Bytes)
1	Plain Text	64
2	Plain Text	128
3	Plain Text	256
4	Plain Text	512
5	Plain Text	1024
6	Plain Text	1518
7	Plain Text HW Offload	64
8	Plain Text HW Offload	128
9	Plain Text HW Offload	256
10	Plain Text HW Offload	512
11	Plain Text HW Offload	1024
12	Plain Text HW Offload	1518
13	IPsec HW Accelerated	64
14	IPsec HW Accelerated	128
15	IPsec HW Accelerated	256
16	IPsec HW Accelerated	512
17	IPsec HW Accelerated	1024
18	IPsec HW Accelerated	1518

fic from within the ARM subsystem of each card. This configuration independently tests the performance of OvS,DPDK `Testpmd`, and DPDK `L2FWD` virtual switches. End-to-end encryption is verified using the MiTMVP before throughput tests are performed.

Sending plain text with hardware offload: This set of treatments offloads all traffic through the hardware accelerators on-board the Bluefield-2 using Traffic Classification (TC) flowers. TC flowers are managed by OvS which is an open-source `OpenFlow` switch. `OpenFlow` rules are used to configure the TC flower data-forwarding behaviors of OvS. OvS can also be configured to support DPDK hardware offloads. Research performed at Clemson University found that offloading the DPDK datapath improved the maximum achievable throughput by approximately 3 Gbps when compared to offloading using TC flowers [43]. Offloading the DPDK data path is not investigated in this research. Nonetheless, sending traffic through hardware avoids interaction from the CPU of the Bluefield-2 DPU and should significantly improve performance within the TNAP. TC flower hardware offloads are configured by running the following commands on each Bluefield-2 DPU in the TNAP:

```
$ ovs-ofctl dump-flows ovsbr1
$ ovs-ofctl del-flows ovsbr1
$ ovs-ofctl -O OpenFlow12 add-flow ovsbr1 arp,actions=
   FLOOD
$ ovs-ofctl -O OpenFlow12 add-flow ovsbr1 ip,in_port=
   pf1hpf, ip_dst=10.0.0.4,ip_src=10.0.0.3,actions=output:
   p1
$ ovs-ofctl -O OpenFlow12 add-flow ovsbr1 ip,in_port=p1,
   ip_dst=10.0.0.3,ip_src=10.0.0.4,actions=output:pf1hpf
$ ovs-vsctl --no-wait set Open_vSwitch .  other_config:hw-
   offload=true
```

Sending encrypted text (IPsec) with hardware acceleration: This set of Ethernet treatments in Experiment 1 test test the capability of the Bluefield-2 to offload IPsec encryption operations to its hardware accelerators. The Bluefield-2 DPU supports full hardware offload of IPsec encryption in switchdev mode, but not by default, however. The following commands place a Bluefield-2 DPU into legacy

mode and enable full hardware offload before switching the Bluefield-2 DPU back to switchdev mode:

```
$ devlink dev eswitch set pci/0000:03:00.1 mode legacy
$ echo none > /sys/class/net/p1/compat/devlink/ipsec_mode
$ echo dmfs > /sys/bus/pci/devices/0000\:03\:00.1/net/p1/
   compat/devlink/steering_mode
$ echo full > /sys/class/net/p1/compat/devlink/ipsec_mode
$ devlink dev eswitch set pci/0000:03:00.1 mode switchdev
```

Once the card is configured in switchdev mode and has IPsec full offload enabled, IP XFRM rules can be written to configure the IPsec rules and settings. The following commands are an example of how one of the Bluefield-2 DPUs can be configured to support IPsec in hardware using custom Mellanox iproute2 tools:

```
$ ip xfrm state add src 10.0.0.3/24 dst 10.0.0.4/24 proto
   esp spi 0x28f39549 reqid 0x28f39549 mode transport aead
    'rfc4106(gcm(aes))' 0
   x492e8ffe718a95a00c1893ea61afc64997f4732848ccfe6ea
07db483175cb18de9ae411a 128 full_offload dev p1 dir out
   sel src 10.0.0.3 dst 10.0.0.4
$ ip xfrm state add src 10.0.0.4/24 dst 10.0.0.3/24 proto
   esp spi 0x622a73b4 reqid 0x622a73b4 mode transport aead
    'rfc4106(gcm(aes))' 0
   x093bfee2212802d626716815f862da31bcc7d9c44cfe3ab
8049e7604b2feb1254869d25b 128 full_offload dev p1 dir in
   sel src 10.0.0.4 dst 10.0.0.3
$ ip xfrm policy add src 10.0.0.3 dst 10.0.0.4 dir out
   tmpl src 10.0.0.3/24 dst 10.0.0.4/24 proto esp reqid 0
   x28f39549 mode transport
$ ip xfrm policy add src 10.0.0.4 dst 10.0.0.3 dir in tmpl
    src 10.0.0.4/24 dst 10.0.0.3/24 proto esp reqid 0
   x622a73b4 mode transport
$ ip xfrm policy add src 10.0.0.4 dst 10.0.0.3 dir fwd
   tmpl src 10.0.0.4/24 dst 10.0.0.3/24 proto esp reqid 0
   x622a73b4 mode transport
```

Note: The keys mentioned in the IPsec configuration above are notional examples and are not in use in real systems.

4.7.1.2 RoCEv2

The next set of treatments tested in Experiment 1 characterize the capability of the Bluefield-2 to accelerate IPsec encryption of RoCEv2 traffic by using the `NVIVDIA-Mellanox InfiniBand Fabric Utilities`. The `InfiniBand Fabric`

`Utilities` provide several applications for managing and testing RDMA fabric architectures. This study specifically uses the SEND, RDMA READ, and RDMA WRITE bandwidth utilities as traffic generators during performance tests. End-to-end encryption is verified using the MiTMVP before RoCEv2 throughput tests are performed. Table 6 lists all of the factor levels tested this portion of Experiment 1, and Table 7 lists all of the treatments.

The following commands are an example of how the `InfiniBand Fabric Utilities` can be used to create client and server processes on the TNAP workstations for throughput testing:

Server:

```
$ numactl --cpubind=0 ib_write_bw -d mlx5_1 -m 1024 --
   report_gbits --iters=100000
```

Client:

```
$ numactl --cpubind=0  ib_write_bw 10.0.0.3 -d mlx5_1 -m
   1024 --report_gbits --iters=100000
```

Sending Plain Text with hardware acceleration: Similar to Ethernet treatments, the data path is offloaded on the Bluefield-2 DPU using OvS and TC flowers. The SEND, RDMA READ, and RDMA WRITE transport functions are all tested to provide a baseline performance for comparison with encrypted results.

Sending encrypted text (IPsec) with hardware acceleration: IPsec acceleration is configured using the same set of IP XFRM rules previously mentioned in Ethernet configurations. SEND, RDMA READ, and RDMA WRITE transport functions are tested during this treatment after end-to-end encryption is verified using the MiTMVP.

4.7.2 Experiment 2: DPDK Virtual Bridge Characterization

A Bluefield-1 DPU is used in the MiTMVP network topology to monitor network traffic and verify end-to-end encryption. Using the Buefield-1 DPU to monitor RDMA

Table 6. Experiment 1: RoCEv2 Factors and Levels

Factor	Level(s)
Packet Generator	(1) Mellanox InfiniBand Fabric Utilities
Virtual Switch	(1) OvS
Transport Service	(1) Reliable Connection
Transport Function	(1) READ (2) WRITE (3) SEND
Bluefield-2 Configuration	(1) Plain Text HW Offload (2) IPsec HW Acceleration
Maximum Transmission Unit (Bytes)	(1) 256 (2) 512 (3) 1024 (4) 2048 (5) 4096 (MAX)

traffic using conventional TCP/IP sniffing tools presents a bottleneck in the MiTMVP network. This configuration of the MiTMVP seeks to minimize latencies introduced by the software switch during RDMA performance tests.

OvS and the `Testpmd` DPDK application are capable of acting as a virtual bridge in the MiTMVP network topology. This study investigates the capabilities of `Testpmd` to serve as a virtual bridge in place of the `Tcpdump` and OvS instances running on the MiTMVP network. Although `Testpmd` does not inherently support passive traffic sniffing, it is reasonable to expect that a DPDK traffic analyzer would have similar performance to `Testpmd` when sniffing traffic. Capturing traffic is significantly more difficult than sniffing because attempting to write at 100 Gbps rapidly exhausts available RAM and storage resources. This configuration only tests plain text treatments because end-to-end encryption is transparent to virtual bridges forwarding packets based on destination IP addresses.

The virtual bridge needs to bridge the two physical ports of the Bluefield-1 DPU. This is accomplished with OvS using the following configuration:

Table 7. Experiment 1: RoCEv2 Treatments

Treatment	Bluefield-2 Configuration	Transport Function	Maximum Transmission Unit (Bytes)
1	Plain Text HW Offload	RDMA READ	256
2	Plain Text HW Offload	RDMA READ	512
3	Plain Text HW Offload	RDMA READ	1024
4	Plain Text HW Offload	RDMA READ	2048
5	Plain Text HW Offload	RDMA READ	4096
6	Plain Text HW Offload	RDMA WRITE	256
7	Plain Text HW Offload	RDMA WRITE	512
8	Plain Text HW Offload	RDMA WRITE	1024
9	Plain Text HW Offload	RDMA WRITE	2048
10	Plain Text HW Offload	RDMA WRITE	4096
11	Plain Text HW Offload	SEND	256
12	Plain Text HW Offload	SEND	512
13	Plain Text HW Offload	SEND	1024
14	Plain Text HW Offload	SEND	2048
15	Plain Text HW Offload	SEND	4096
16	IPsec HW Acceleration	RDMA READ	256
17	IPsec HW Acceleration	RDMA READ	512
18	IPsec HW Acceleration	RDMA READ	1024
19	IPsec HW Acceleration	RDMA READ	2048
20	IPsec HW Acceleration	RDMA READ	4096
21	IPsec HW Acceleration	RDMA WRITE	256
22	IPsec HW Acceleration	RDMA WRITE	512
23	IPsec HW Acceleration	RDMA WRITE	1024
24	IPsec HW Acceleration	RDMA WRITE	2048
25	IPsec HW Acceleration	RDMA WRITE	4096
26	IPsec HW Acceleration	SEND	256
27	IPsec HW Acceleration	SEND	512
28	IPsec HW Acceleration	SEND	1024
29	IPsec HW Acceleration	SEND	2048
30	IPsec HW Acceleration	SEND	4096

```
$ ovs-vsctl add-br ovsbr1
$ ovs-vsctl add-port ovsbr1 p0
$ ovs-vsctl add-port ovsbr1 p1
```

The `Testpmd` virtual bridge can also be configured on the Bluefield-1 DPU using these commands:

```
$ sysctl -w vm.nr_hugepages=16
$ testpmd -d librte_mempool_ring.so -d librte_pmd_mlx5.so
   -w 03:00.0 -w 03:00.1
```

Once running, the `Testpmd` portmask needs to be set to `0x5` for traffic to be bridged properly between the physical ports of the card.

4.7.2.1 Ethernet

This set of treatments in Experiment 2 use `iPerf3` throughput tests to determine the performance capabilities of OvS and `Testpmd` to forward Ethernet traffic. The average throughput and drop-rate of each virtual bridge is recorded during these treatments. Table 8 lists all of the factor levels tested in this portion of Experiment 2, and Table 9 lists all of the treatments.

4.7.3 RoCEv2

This set of treatments in Experiment 2 use the `NVIDIA-Mellanox InfiniBand Fabric Utilities` to characterize the capability of OvS and `Testpmd` to forward RoCEv2 traffic. Table 10 lists all of the factor levels tested in this portion of Experiment 2, and Table 11 lists all of the treatments.

4.7.4 Experiment 3: Software Encryption Characterization

Experiment 3 characterizes the capability of the Bluefield-2 DPU to perform software encryption using its ARM CPU. The NVIDIA-Mellanox Bluefield-2 DPU software package comes preloaded with OpenSSL libraries and cryptodev drivers. This

Table 8. Experiment 2: Ethernet Factors and Levels

Factor	Level(s)
Packet Generator	(1) iPerf3
Virtual Switch	(1) OvS (2) Testpmd
Traffic Generator Threads	1, 2, 3, 4, 5, 6, 7, and 8
Bluefield-2 Configuration	(1) Plain Text
Maximum Segment Size (Bytes)	(1) 890 (2) 1780 (3) 2670 (4) 3560 (5) 4450 (6) 5340 (7) 6230 (8) 7120 (9) 8010 (10) 8900

allows up to six virtual cryptography devices to be pinned to the available ARM cores of the Bluefield-2 DPU. The Bluefield-2 DPU software package also comes with a NULL cryptography cipher. The NULL virtual cryptography devices operate similarly to the OpenSSL cryptography devices, but the NULL devices do not apply a cipher to packets. NULL cryptography devices are useful for benchmarking the maximum achievable performance.

The following Linux commands were used to configure DPDK applications on the Bluefield-2 DPU for this experiment:

```
$ echo 1024 > /sys/kernel/mm/hugepages/hugepages-2048kB/
   nr_hugepages
```

Testpmd:

```
$ dpdk-testpmd -a 03:00.0,representor=[0,65535] -a
   03:00.1,representor=[0,65535] -- -i -a --total-num-
   mbufs=16384
```

Table 9. Experiment 2: Ethernet Treatments (Repeated for each iPerf3 thread 1-8)

Treatment	MiTM Virtual Switch	Maximum Segment Size (Bytes)
1	OvS	890
2	OvS	1780
3	OvS	2670
4	OvS	3560
5	OvS	4450
6	OvS	5340
7	OvS	6230
8	OvS	7120
9	OvS	8010
10	OvS	8900
11	Testpmd	890
12	Testpmd	1780
13	Testpmd	2670
14	Testpmd	3560
15	Testpmd	4450
16	Testpmd	5340
17	Testpmd	6230
18	Testpmd	7120
19	Testpmd	8010
20	Testpmd	8900

Table 10. Experiment 2: RoCEv2 Factors and Levels

Factor	Level(s)
Packet Generator	(1) Mellanox InfiniBand Fabric Utilities
Virtual Switch	(1) OvS (2) Testpmd
Transport Service	(1) Reliable Connection
Transport Function	(1) READ (2) WRITE (3) SEND
Bluefield-2 Configuration	(1) Plain Text
Maximum Transmission Unit (Bytes)	(1) 256 (2) 512 (3) 1024 (4) 2048 (5) 4096 (MAX)

L2FWD:

```
$ /dpdk-l2fwd -a 03:00.0,representor=[0,65535] -a 03:00.1,
   representor=[0,65535] -- --no-mac-updating -P -p 3f
```

L2FWD-CRYPTO: AES 128 (Encrypt/Decrypt):

```
$ dpdk-l2fwd-crypto --socket-mem 1024,0 --legacy-mem --
   vdev "crypto_openssl_0" --vdev "crypto_openssl_1" --
   vdev "crypto_openssl_2" --vdev "crypto_openssl_3" --
   vdev "crypto_openssl_4" --vdev "crypto_openssl_5" -a
   03:00.0,representor=[0,65535] -a 03:00.1,representor
   =[0,65535] -- -p 0x3f --chain CIPHER_ONLY --cdev_type
   SW --cipher_op <ENCRYPT, DECRYPT> --cipher_algo aes-cbc
    --cipher_key 00:01:02:03:04:05:06:07:08:09:0a:0b:0c:0d
   :0e:0f --cipher_iv 00:01:02:03:04:05:06:07:08:09:0a:0b
   :0c:0d:0e:0f --no-mac-updating
```

L2FWD-CRYPTO: NULL (Encrypt/Decrypt):

```
$ dpdk-l2fwd-crypto --socket-mem 1024,0 --legacy-mem --
   vdev "crypto_null_0" --vdev "crypto_null_1" --vdev "
   crypto_null_2" --vdev "crypto_null_3" --vdev "
   crypto_null_4" --vdev "crypto_null_5" -a 03:00.0,
   representor=[0,65535] -a 03:00.1,representor=[0,65535]
   -- -p 0x3f --cipher_op <ENCRYPT, DECRYPT> --cipher_algo
    null --auth_algo null --no-mac-updating
```

Table 11. Experiment 2: RoCEv2 Treatments

Treatment	MiTM Virtual Switch	Transport Function	Maximum Transmission Unit (Bytes)
1	OvS	RDMA READ	256
2	OvS	RDMA READ	512
3	OvS	RDMA READ	1024
4	OvS	RDMA READ	2048
5	OvS	RDMA READ	4096
6	OvS	RDMA WRITE	256
7	OvS	RDMA WRITE	512
8	OvS	RDMA WRITE	1024
9	OvS	RDMA WRITE	2048
10	OvS	RDMA WRITE	4096
11	OvS	SEND	256
12	OvS	SEND	512
13	OvS	SEND	1024
14	OvS	SEND	2048
15	OvS	SEND	4096
16	Testpmd	RDMA READ	256
17	Testpmd	RDMA READ	512
18	Testpmd	RDMA READ	1024
19	Testpmd	RDMA READ	2048
20	Testpmd	RDMA READ	4096
21	Testpmd	RDMA WRITE	256
22	Testpmd	RDMA WRITE	512
23	Testpmd	RDMA WRITE	1024
24	Testpmd	RDMA WRITE	2048
25	Testpmd	RDMA WRITE	4096
26	Testpmd	SEND	256
27	Testpmd	SEND	512
28	Testpmd	SEND	1024
29	Testpmd	SEND	2048
30	Testpmd	SEND	4096

Table 12 lists all of the factor levels tested Experiment 3, and Table 13 lists all of the treatments.

4.7.4.1 L2FWD-Crypto: OpenSSL AES-CBC 128

This set of treatments in Experiment 3 test the capability of the Bluefield-2 DPU ARM CPU to support software encryption using the AES-CBC 128 cryptography algorithm. Six OpenSSL virtual cryptography devices are pinned to the ARM CPU cores.

4.7.4.2 L2FWD-Crypto: NULL

Similarly, this set of treatments in Experiment 3 test the capability of the Bluefield-2 DPU ARM CPU to support software encryption using the null library. Six null virtual cryptography devices are pinned to the ARM CPU cores. As mentioned before, it is reasonable to assume that the achievable throughput during this treatment represents the maximum achievable throughput for the `L2fwd-Crypto` DPDK application running on the Bluefield-2 DPU.

4.8 Testing Process

Bash scripts are used to perform throughput tests using `iPerf3` and the `NVIVDIA-Mellanox InfiniBand Fabric Utilities`. The bash scripts for those tests write results to text files, and the results can be compiled using a Python script (Appendix A, B, and C).

The `Pktgen` DPDK application allows scripting in the LUA scripting language. Despite this scripting option, `Pktgen` results are collected by hand in this research.

Table 12. Experiment 3: Software Encryption Factors and Levels

Factor	Level(s)
Packet Generator	(1) Pktgen
Bluefield-2 Virtual Switch	(1) L2fwd (2) L2fwd-Crypto:Null (3) L2fwd-Crypto:AES-CBC
Maximum Transmission Unit (Bytes)	(1) 64 (2) 128 (3) 256 (4) 512 (5) 1024 (6) 1518 (MAX)

Table 13. Experiment 3: Software Encryption Treatments

Treatment	Bluefield-2 Virtual Switch	Maximum Transmission Unit (Bytes)
1	L2fwd	64
2	L2fwd	128
3	L2fwd	256
4	L2fwd	512
5	L2fwd	1024
6	L2fwd	1518
7	L2fwd-Crypto:Null	64
8	L2fwd-Crypto:Null	128
9	L2fwd-Crypto:Null	256
10	L2fwd-Crypto:Null	512
11	L2fwd-Crypto:Null	1024
12	L2fwd-Crypto:Null	1518
13	L2fwd-Crypto:AES-CBC	64
14	L2fwd-Crypto:AES-CBC	128
15	L2fwd-Crypto:AES-CBC	256
16	L2fwd-Crypto:AES-CBC	512
17	L2fwd-Crypto:AES-CBC	1024
18	L2fwd-Crypto:AES-CBC	1518

4.9 Statistical Analysis

4.9.1 Kruskal-Wallis Test

The Kruskal-Wallis test is a nonparametric alternative to ANOVA for situations where the normality assumption is unjustified [44]. Kruskal-Wallis uses an F-test analysis of variance that does not require normal residuals. Preliminary throughput tests show that network performance using the TNAP and MiTMVP network designs from Section 3.2 and 3.3 is non-normal. Therefore, the Kruskal-Wallis test is a good fit for analyzing the statistical significance of the data collected in this research.

4.9.2 Full-Factorial Screening Tests

Confounding variables and uncontrolled factors introduce noise into experiment results. This research applies the Kruskal-Wallis analysis of variance test on a full factorial design to identify factors that have a significant effect on the response variable: throughput.

4.9.2.1 Ethernet Factor Screening

The Ethernet full-factorial design tests the significance of packet size (Maximum Segment Size (MSS)), iPerf3 thread count, CPU performance setting, and the direction of the throughput test. Applying the Kruskal-Wallis analysis of variance test on the results gathered during this screening test allows factors that have a significant effect on the response variable, throughput, to be identified. Table 14 lists all of the treatments tested in the preliminary screening tests.

Only two factor levels are required for screening tests. Screening tests often work best when factor levels have large differences. 890 and 8900 were selected for the MSS levels, roughly representing the upper and lower bounds of packet sizes that can be sent across the TNAP. Additionally, one and four were chosen for the iPerf3

Table 14. Ethernet Factor Screening Treatments

Treatment	Maximum Segment Size (Bytes)	Thread	CPU Performance Setting	Test Direction
1	890	1	Ondemand	WS3 to WS4
2	890	1	Ondemand	WS4 to WS3
3	890	1	Performance	WS3 to WS4
4	890	1	Performance	WS4 to WS3
5	890	4	Ondemand	WS3 to WS4
6	890	4	Ondemand	WS4 to WS3
7	890	4	Performance	WS3 to WS4
8	890	4	Performance	WS4 to WS3
9	8900	1	Ondemand	WS3 to WS4
10	8900	1	Ondemand	WS4 to WS3
11	8900	1	Performance	WS3 to WS4
12	8900	1	Performance	WS4 to WS3
13	8900	4	Ondemand	WS3 to WS4
14	8900	4	Ondemand	WS4 to WS3
15	8900	4	Performance	WS3 to WS4
16	8900	4	Performance	WS4 to WS3

thread factor levels. Adding multiple iPerf3 threads appeared to increase the average throughput across the TNAP during preliminary tests.

Applying the Kruskal-Wallis test to the results of the full factorial design described above determines that MSS and thread count significantly affect average Ethernet throughput. The effect of MSS is significant on a 99.9% ($p = 0.00077$) confidence interval, and thread count is significant on a 90.0% ($p = 0.09265$) confidence interval. CPU performance setting and the traffic direction do not have a significant effect on the response variable.

4.9.2.2 RoCEv2 Factor Screening

Table 15 list all of the treatments tested in the RoCEv2 factor screening tests. 16 treatments are tested in a full factorial test of four, two level factors (2^4). Three replicates of each treatment are performed in order to further reduce noise. In total, 48

RoCE throughput tests (16 treatments x 3 replicates) are performed in this screening test using the InfiniBand Fabric Utilities.

Table 15. RoCEv2 Factor Screening Treatments

Treatment	Maximum Transmission Unit (Bytes)	Transport Function	Transport Service	Test Duration (Iterations)
1	512	RDMA READ	RC	1,000
2	512	RDMA READ	RC	100,000
3	512	RDMA READ	DC	1,000
4	512	RDMA READ	DC	100,000
5	512	RDMA WRITE	RC	1,000
6	512	RDMA WRITE	RC	100,000
7	512	RDMA WRITE	DC	1,000
8	512	RDMA WRITE	DC	100,000
9	4096	RDMA READ	RC	1,000
10	4096	RDMA READ	RC	100,000
11	4096	RDMA READ	DC	1,000
12	4096	RDMA READ	DC	100,000
13	4096	RDMA WRITE	RC	1,000
14	4096	RDMA WRITE	RC	100,000
15	4096	RDMA WRITE	DC	1,000
16	4096	RDMA WRITE	DC	100,000

256 and 4096 Bytes were selected for the MTU levels since they are the minimum and maximum MTUs supported by the Bluefield-2 when using RoCE. MTU is tested in this research because network performance is often dependent on packet size. RDMA read and write are foundational operations. RC and Dynamically Connected (DC) transports are tested for the connection types. RC and DC operate similarly to TCP and UDP respectively. Lastly, 1,000 and 100,000 iterations are tested. Increased throughput test duration sometimes improves experimental results because longer tests can dilute noise caused by systems throttling CPU clocks. Many end nodes dynamically throttle clock rates to reduce power consumption. Each of these factors can be configured using the command line arguments of the `InfiniBand Fabric Utilities`.

Applying the Kruskal-Wallis test to the results of the full factorial design described above determines that MTU, RDMA operation type, and iterations significantly affect average RoCE throughput. The effect of MTU is significant on a 99.9% ($p = 2.2\text{x}10^{-16}$) confidence interval; RDMA operation type on a 99% ($p = 0.0077$) confidence interval; and iterations on a 95.0% ($p = 0.0275$) confidence interval. Transport service type does not significantly affect the response variable.

4.10 Randomization

The Kruskal-Wallis test assumes that data is independent of run-order. This research ensures independence of run-order by randomizing factor levels during each throughput test.

4.11 Methodology Summary

This chapter describes the experimentation methodology used to characterize the capability of the Bluefield-2 DPU to encrypt Ethernet and RDMA traffic in hardware and software. Each treatment tests a specific device configuration that adds to the operational capabilities of the Bluefield-2 DPU.

V. Results and Analysis

5.1 Overview

This chapter presents the results of the experimentation described in Chapter IV. Results are discussed for characterizing the performance trade-offs associated with three distinct capabilities of the Bluefield-2 DPU: (i) Hardware accelerated encryption, (ii) virtual bridges, and (iii) software based encryption. The MiTMVP network architecture is used when analyzing the performance of virtual bridges. Section 5.2 discusses the performance capabilities of the Bluefield-2 DPU to encrypt both Ethernet and RoCEv2 traffic. Possible sources of error for the findings are discussed in Section 5.3. Finally, this chapter discusses security benefits, drawbacks, and challenges as they relate to securing the RDMA fabric architectures, like InfiniBand, with the Bluefield-2 DPU in Section 5.4.

5.2 TNAP Performance

This section analyzes the results of throughput tests conducted using the TNAP and Bluefield-2 DPUs. Results are presented for all three capabilities.

5.2.1 Hardware Accelerator Characterization

5.2.1.1 Ethernet

Figure 22 shows the performance curves when OvS is used as the virtual bridge on each Bluefield-2 DPU. The throughput collected during these trials peaks around 99 Gbps when traffic was offloaded to the hardware accelerators of each Bluefield-2 DPU using TC flowers. This result demonstrates that `Pktgen` instances on the host workstations are capable of generating enough Ethernet traffic to saturate the card. The average performance of the hardware accelerated IPsec peaks below 5 Gbps.

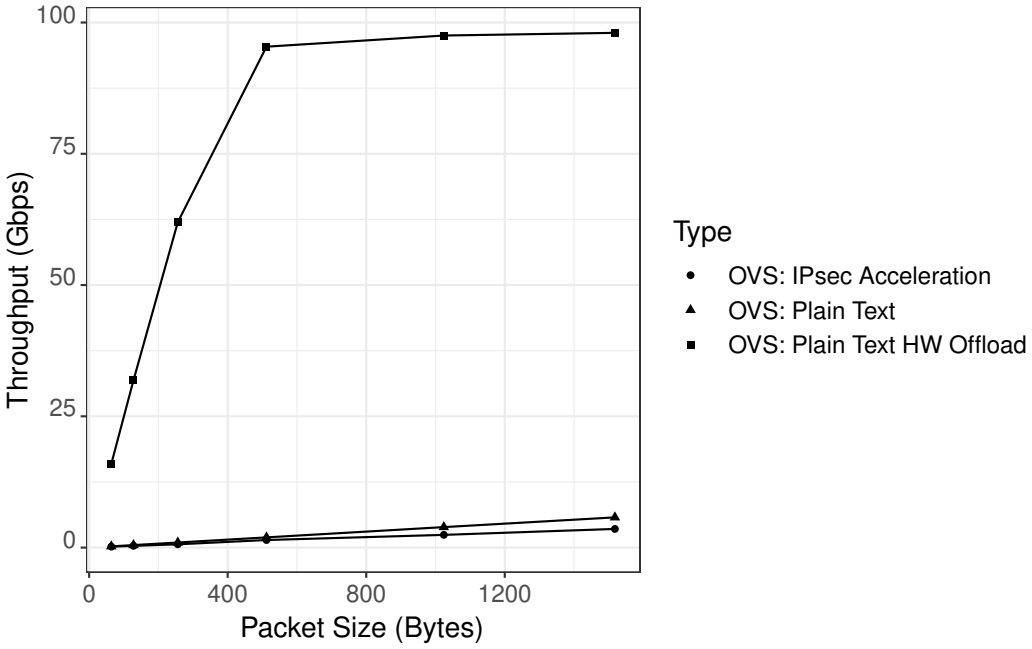

Figure 22. OvS Hardware Acceleration Throughput vs Packet Size

Figure 23 and Table 16 illustrate the differences in performance between the three OvS configurations mentioned above. The Kruskal-Wallis test indicates that offloading plain text traffic to the hardware accelerators of the Bluefield-2 DPU significantly affects performance according to a 99.9% confidence interval. There is no significant difference between the baseline performance of the card and when IPsec is offloaded to the hardware accelerators of the card.

The limited performance of the Bluefield-2 DPUs when offloading encryption of Ethernet traffic is attributable to the limited capabilities of the software switching

Table 16. OvS Hardware Acceleration Statistical Analysis

Treatment 1	Treatment 2	P-value
Plain Text	Plain Text HW Offload	$2.968 * 10^{-07}$
Plain Text	IPsec Acceleration	0.2547
Plain Text HW Offload	IPsec Acceleration	$2.968 * 10^{-07}$

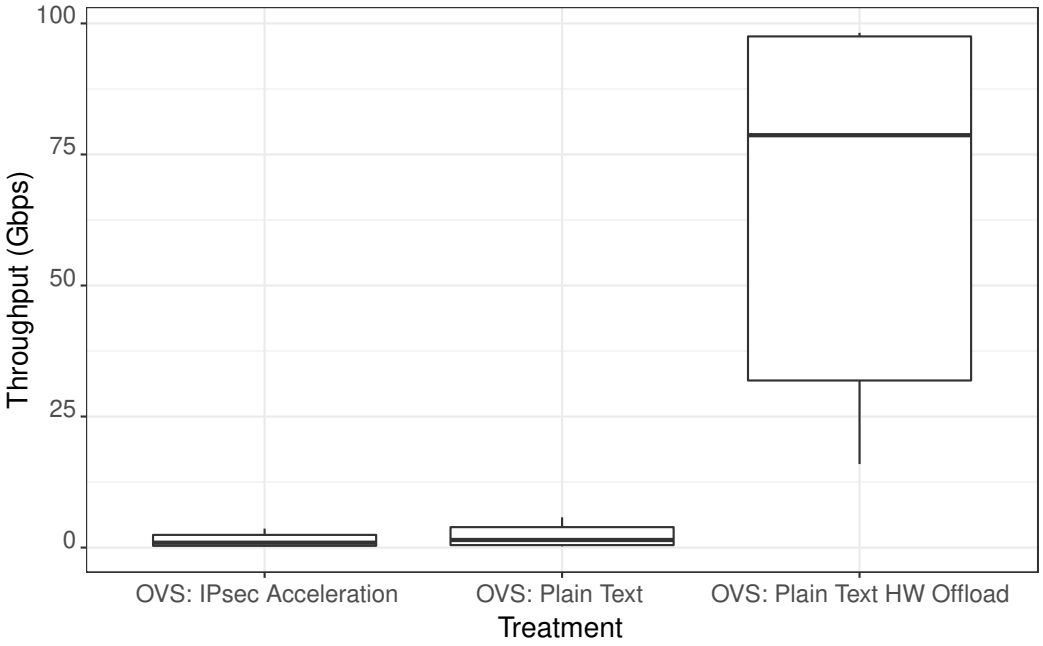

Figure 23. OvS Hardware Acceleration Quartile Ranges

path of the card. IPsec offload is configured by placing the card in switchdev mode, attaching VF representor to OvS, and then writing IP XFRM rules. This configuration forces Ethernet traffic through the TCP/IP stack in the OS kernel before they are handed off to the hardware of the card.

On the other hand, offloading plain text traffic significantly improves performance because the card is able to offload Ethernet frames in the fast data path using TC Flowers. In this configuration, Ethernet frames interact directly with the hardware.

5.2.1.2 RoCEv2

Figure 24 shows the performance curves of the Bluefield-2 DPU when RoCEv2 traffic is offloaded to the hardware accelerators. In total, 45 throughput tests were conducted for each configuration. The hardware accelerators of the Bluefield-2 DPU are capable of encrypting RoCEv2 traffic at a rate of nearly 86 Gbps.

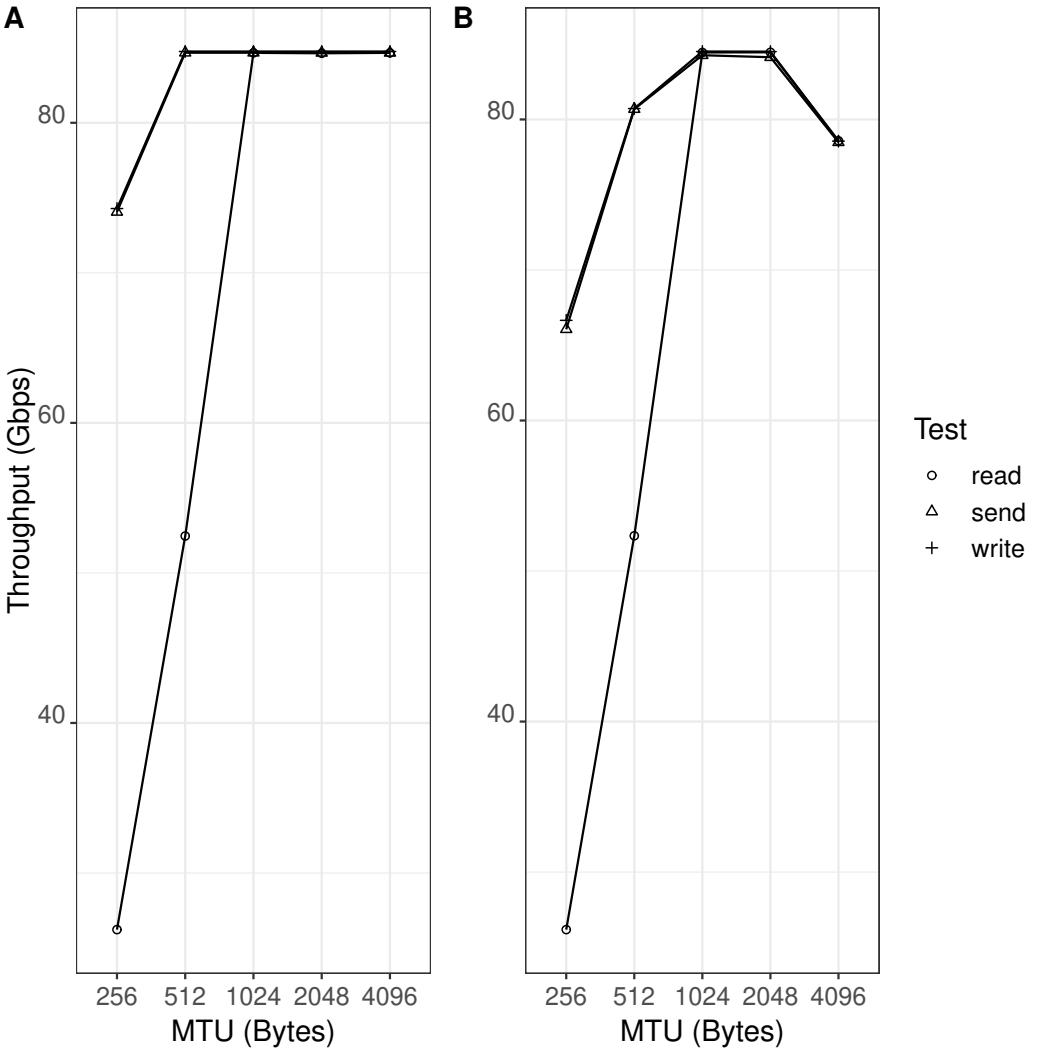

Figure 24. Hardware Accelerator Performance (A) Plain Text (B) IPsec

Figure 25 shows that the hardware accelerators of the Bluefield-2 DPU perform slightly better without encryption according to a 99.9% ($p = 2.3\text{x}10^{-9}$) confidence interval.

IPsec encryption is limited to Ethernet and RoCEv2 traffic because RoCE and native InfiniBand use the InfiniBand network layer. I.e., RoCE and InfiniBand do not use IP addresses and are not compatible with IPsec encryption. Other encryption approaches, such as sRDMA, are needed to encrypt RoCE and InfiniBand traffic.

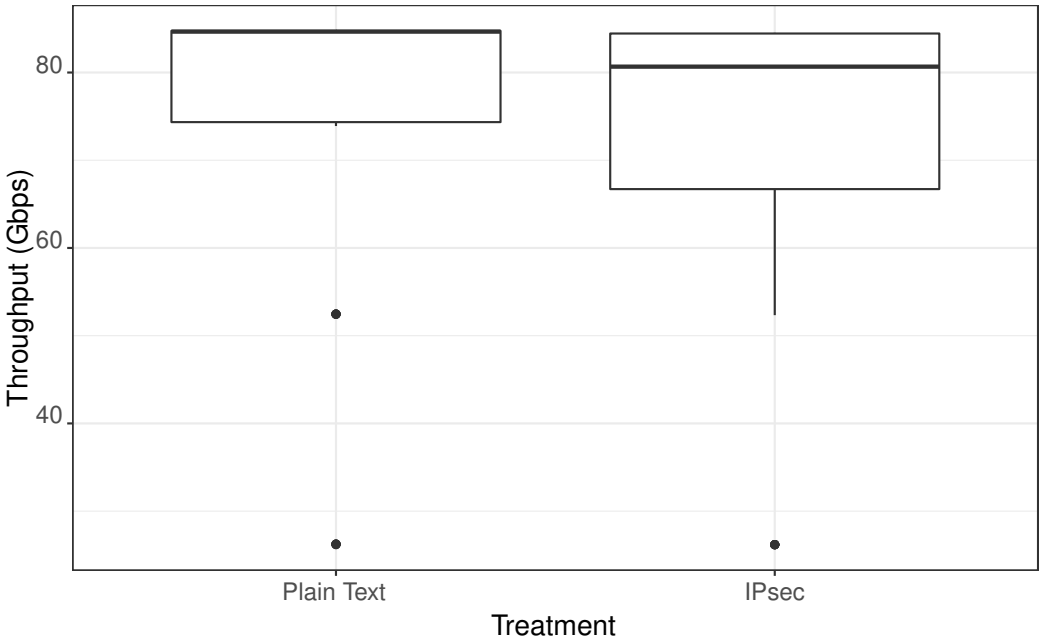

Figure 25. Comparison of Plain Text and IPsec RoCEv2 Performance

5.2.2 DPDK Virtual Bridge Characterization

5.2.2.1 Ethernet

Figure 26 shows the performance curves of the OvS and DPDK `Testpmd` when forwarding Ethernet traffic across the card. The performance of OvS and `Testpmd` peak just under 10 Gbps. Applying the Kruskal-Wallis test to this dataset shows that OvS performs slightly better than `Testpmd` on average based on a 99.9% ($p = 2.053\text{x}10^{-6}$) confidence interval (Figure 27).

5.2.3 RoCEv2

Figure 28 compares the performance curves of the OvS and DPDK `Testpmd` when forwarding RoCEv2 traffic across the card. DPDK `Testpmd` performs better than OvS in this scenario on a 99.9% ($p = 2.2\text{x}10^{-16}$) confidence interval (Figure 29). The

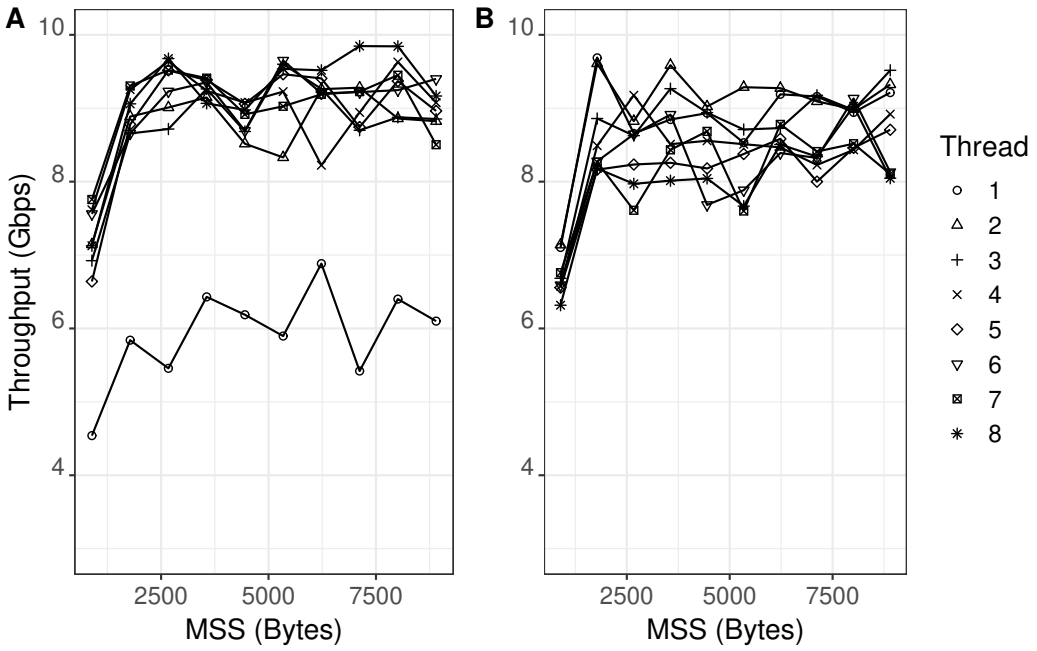

Figure 26. Qualitative Ethernet Bridge Comparison (A) OVS (B) DPDK

performance of DPDK peaks around 70 Gbps.

5.2.4 Monitoring Capability

Figure 30 lists the capture rate of `Testpmd` and `Tcpdump`. `Testpmd` hardly dropped any TCP or RoCEv2 packets. On the other hand, `Tcpdump` dropped a significant majority of the packets sent across the network. `Tcpdump` performed the same when forwarding TCP and RoCEv2 traffic.

Testing the performance of virtual bridges is an important part of this research because it highlights the performance benefit realized through the use of user space applications like `Testpmd`. DPDK applications ride directly above the hardware in the network stack, whereas traditional applications operate on top of the OS kernel. Future DPDK applications could provide monitoring and link-layer encryption solutions for RDMA traffic.

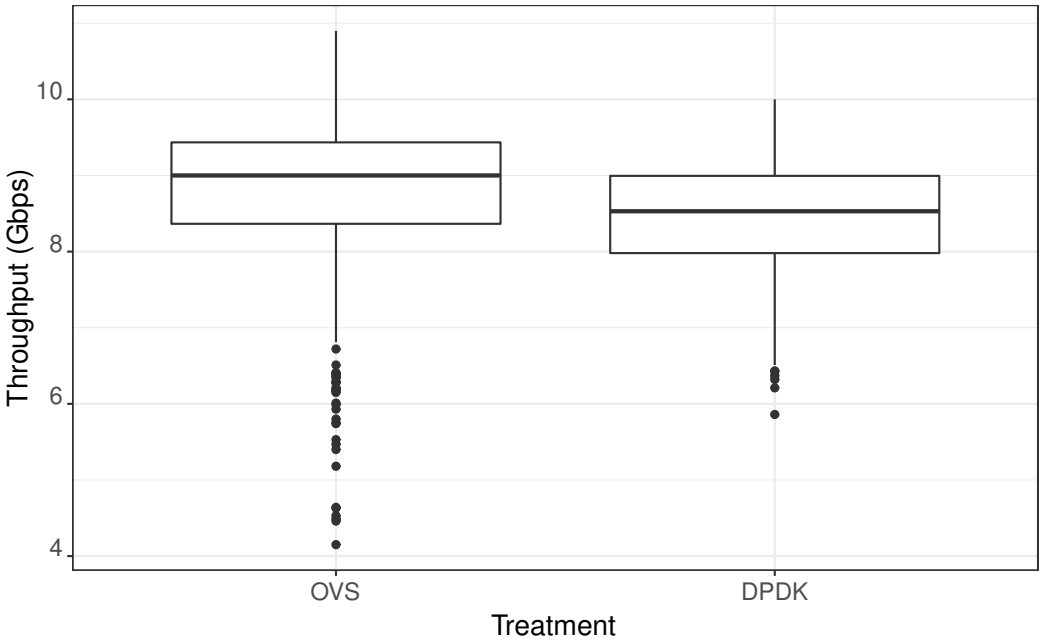

Figure 27. Ethernet Bridge Quartile Ranges

5.2.5 Software Encryption Characterization

Figure 31 shows the performance curves of the `Testpmd` , `L2FWD`, and `L2FWD-CRYPTO` DPDK applications when on each Bluefield-2 DPU in the TNAP. `Testpmd` and `L2FWD` perform very similarly with a performance that peaks around 75 Gbps.

`L2FWD-CRYPTO` is a sample DPDK application that performs a cryptographic operation with a physical or virtual cryptography device. As discussed in the Background of this research, the Mellanox OFED comes preloaded with cryptodev libraries which contain a suite of ciphers. The average throughput supported by `L2FWD-CRYPTO` using six virtual OpenSSL cryptography devices and the AES-CBC 128 cipher peaks just under 10 Gbps.

The upper limit of the performance of L2FWD-CRPYTO is determined using six virtual NULL cryptography devices. The null crypodev Linux module is a basic cryptography device that does not apply a cipher. The average throughput supported

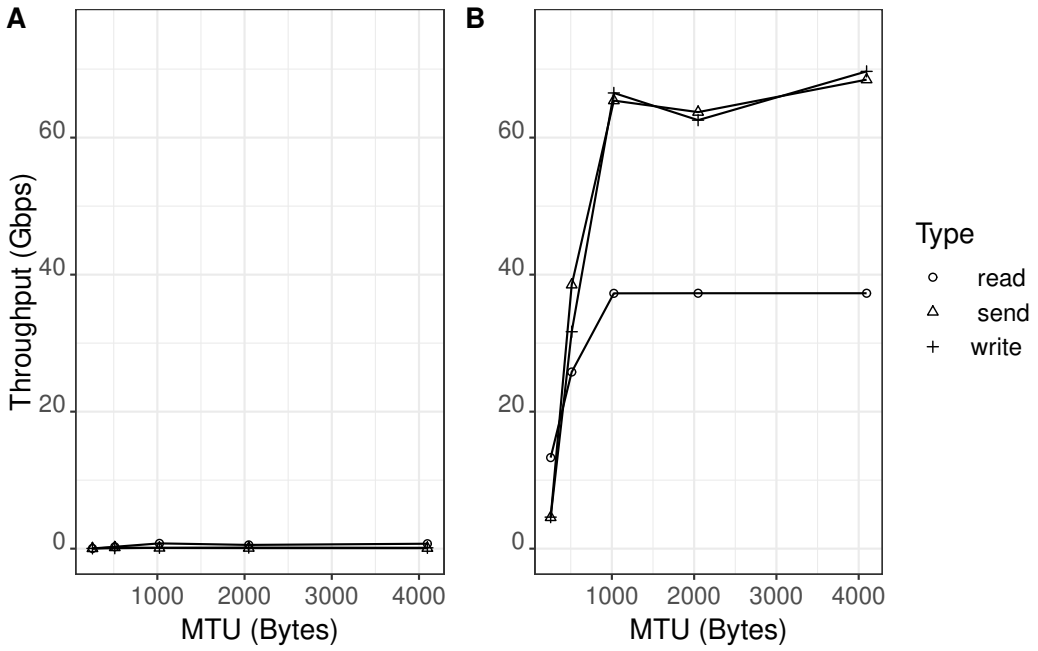

Figure 28. Qualitative RoCEv2 Bridge Comparison (A) OVS (B) DPDK

by the NULL cryptography devices peaks over 40 Gbps. Therefore, it is reasonable to expect all software encryption implementations on the Bluefield-2 DPU to achieve an average throughput less than or equal to 40 Gbps.

Table 17 lists the results of applying the Kruskal-Wallis test to the performance data collected for the four DPDK applications. The results indicate that there is no statistical difference between the performance of `Testpmd` and `L2FWD`, and show that there is a statistical difference between the average throughput achieved when the NULL cipher is used by the L2FWD-CRYTO application in place of the AES-CBC 128 cipher.

Interestingly, the software based encryption implementations perform better than hardware based implementations when using Ethernet traffic. Figure 33 and 34 illustrate the performance curves of each of the Ethernet encryption methods tested in this research. Table 18 lists the results of the Kruskal-Wallis tests applied between

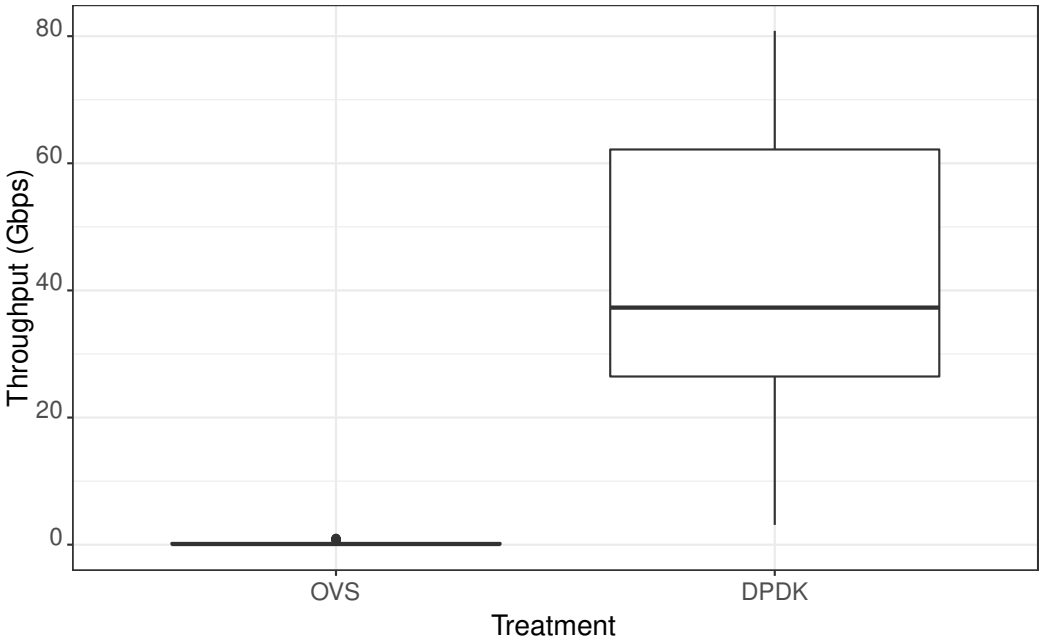

Figure 29. RoCEv2 Bridge Quartile Ranges

each of the three encryption methods. The test results show that offloading IPsec encryption of Ethernet traffic on the Bluefield-2 DPU performs worse than the software implementations using AES-CBC 128 and NULL ciphers according to 99.0% and 99.9% confidence intervals respectively.

Table 17. DPDK Application Statistical Analysis

Treatment 1	**Treatment 2**	**P-value**
L2FWD	Testpmd	0.8743
L2FWD	L2FWD-CRYPTO: NULL	0.01039
L2FWD	L2FWD-CRYPTO: AES-CBC	$5.215 * 10^{-06}$
Testpmd	L2FWD-CRYPTO: NULL	0.04624
Testpmd	L2FWD-CRYPTO: AES-CBC	$9.961 * 10^{-05}$
L2FWD-CRYPTO: NULL	L2FWD-CRYPTO: AES-CBC	0.004407

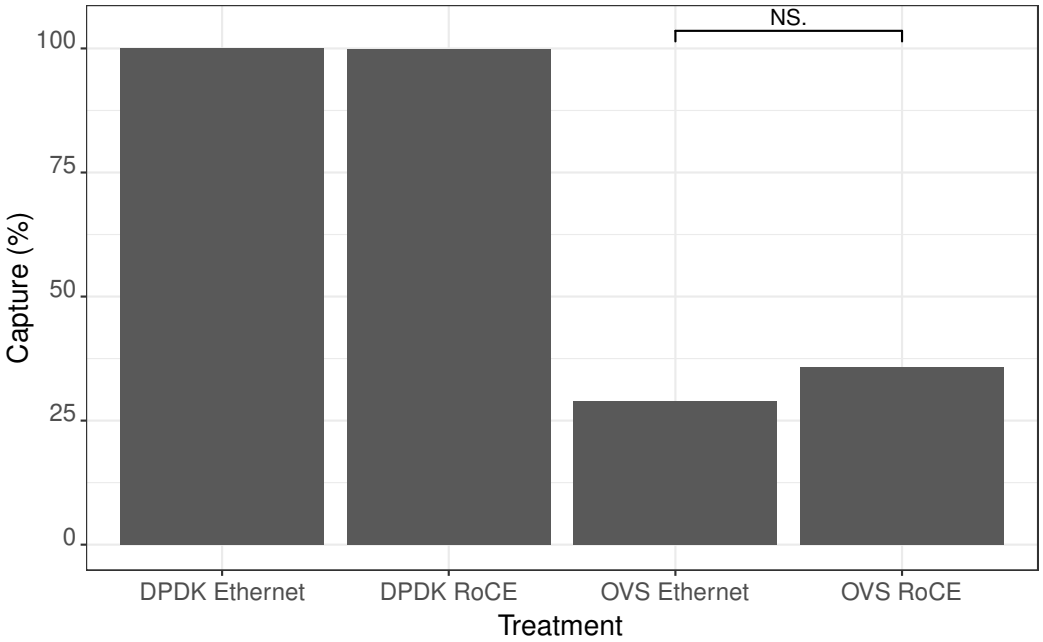

Figure 30. Virtual Bridge Capture Capability

Table 18. Software Encryption Statistical Analysis

Treatment 1	Treatment 2	P-value
L2FWD-CRYPTO: AES-CBC	L2FWD-CRYPTO: NULL	0.004407
L2FWD-CRYPTO: AES-CBC	OVS: IPsec Acceleration	0.001119
L2FWD-CRYPTO: NULL	OVS: IPsec Acceleration	$5.215 * 10^{-06}$

5.3 Possible Sources of Errors

There are many tools and layer implementations that enable TNAP. Consequently, there are many possible factors that may impact the precision or accuracy of the throughput measurements for this set of experiments. The possible sources of error for this data-set are identified and described below:

- **Thermal:** The Bluefield-2 DPU is a high performance network adapter that draws a significant amount of power. Although a large heatsink is attached to the processor of the Bluefield-2 DPU, the card overheats without significant

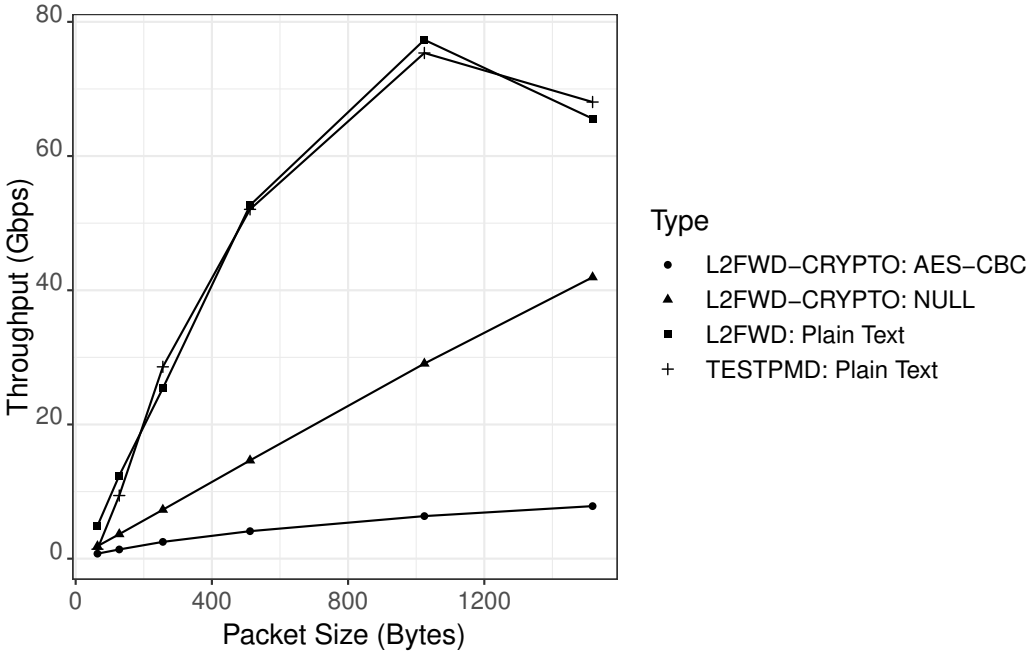

Figure 31. DPDK Application Throughput vs Packet Size

airflow. Sufficient airflow to run the cards is generated in this research using the case fans of the workstations. In addition, a large external fan was used to circulate air throughout the testbed. Nonetheless, chip heating is a possible source of error in this research.

- **Process Eviction:** As mentioned previously, process eviction is a possible source of error in this research. The Linux OS might evict a process from a CPU core during performance tests.

- **Clock Throttling:** Most modern workstations throttle CPU clocks to save power. This is the case for the workstations used in this research. The CPUs have a clock rate of approximately 1.2 GHz when idle and 3.2 GHz when fully utilized. Throttling CPU clocks is a possible source of error in this research because it can potentially lower the average throughput measured during per-

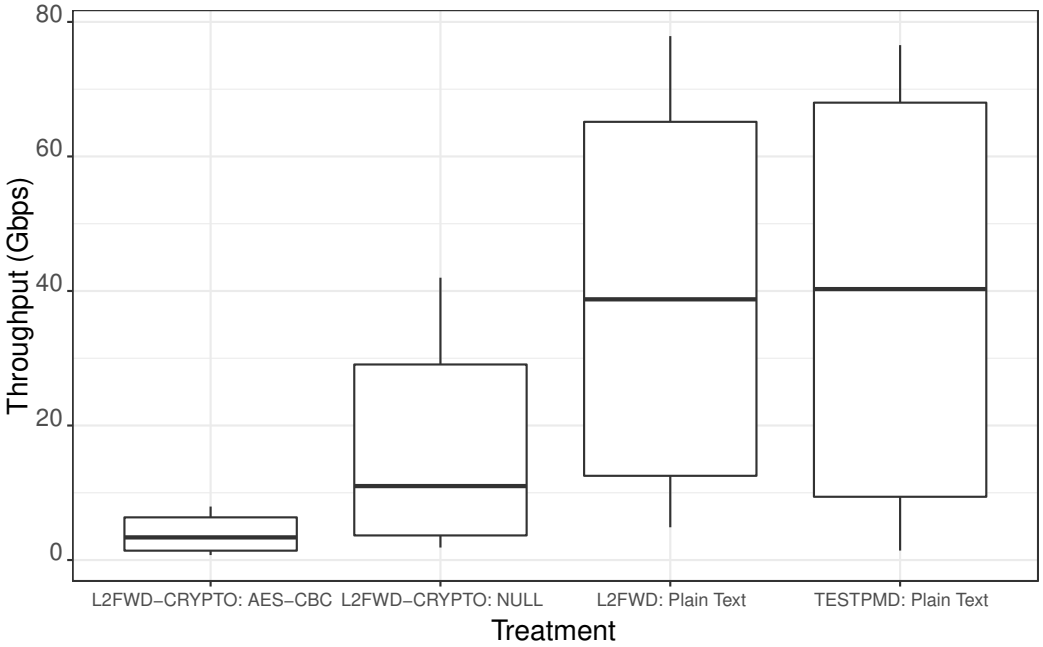

Figure 32. DPDK Application Quartile Ranges

formance tests.

- **Assigned Resources:** System resources are allocated manually to DPDK applications through the command line interface. For example, the amount of memory allocated to the ring buffer of each `Testpmd` instance. Determining optimal configurations of CPU cores and memory allocations is nontrivial. Therefore, it is likely that performance measurements in this research could be improved given time for refined resource allocation.

- **Recording Results by Hand:** As mentioned in Section 4.8, the results of throughput tests performed using `Pktgen` were recorded by hand, possibly introducing errors while compiling results.

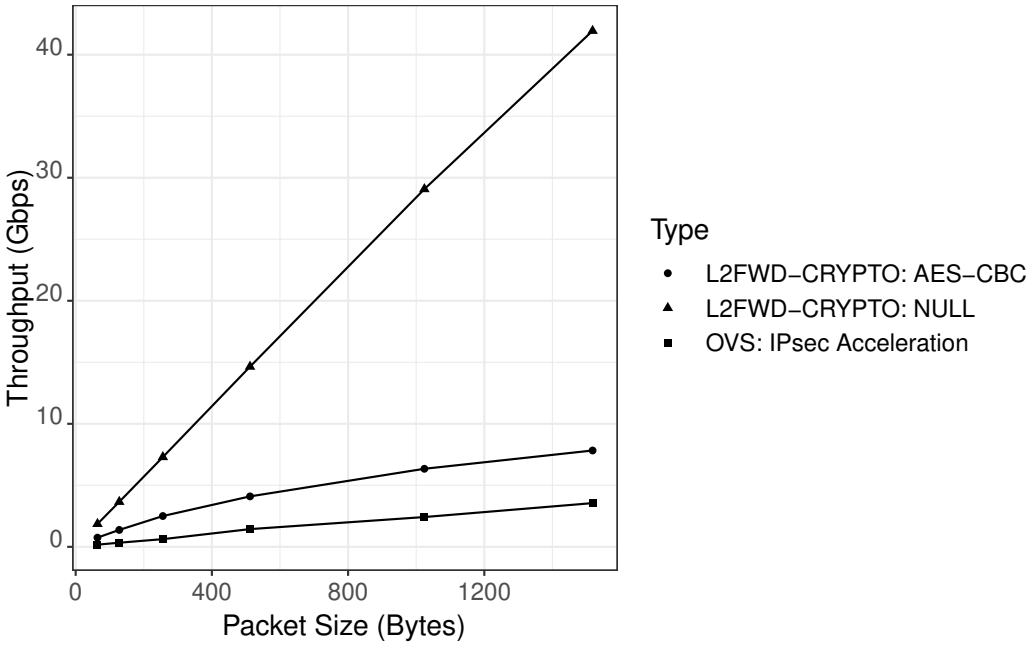

Figure 33. Encryption Method Throughput vs Packet Size

5.4 Drawbacks & Challenges

IPsec encryption is not compatible with RoCE or native InfiniBand. This makes the fast data path offered by the hardware offloads of the Bluefield-2 DPU inaccessible for most RDMA traffic unless it uses IP and Ethernet at the network and link-layers respectively. Exploring other capabilities of the Bluefield-2 DPU showed that software based encryption using DPDK applications is a promising method for encrypting traffic at the link layer. While exploring this area, several limitations were encountered that present a barrier to the development of custom applications aimed at encrypting RoCE and InfiniBand traffic.

5.4.1 Limitations

1. **Multi-process Support:** The Bluefield-2 DPU is not capable of supporting multiple `L2FWD-CRYPTO` processes. Each `L2FWD-CRYPTO` process either performs

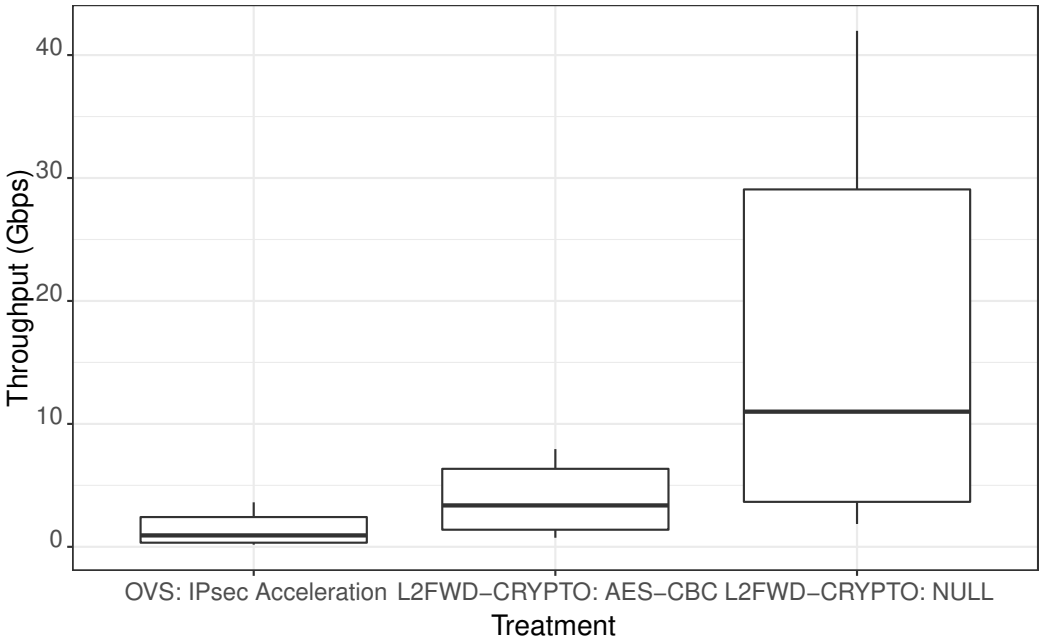

Figure 34. Encryption Method Quartile Ranges

encryption or decryption. As a result, multiple `L2FWD-CRYPTO` processes are required on each card to support bi-directional communication. (This issue was reported to NVIDIA-Mellanox.)

2. **Odd Number of Cryptography Devices:** `L2FWD-CRYPTO` reports a critical error if it is initialized with an odd number of virtual cryptography devices. This further complicates the issue of running two `L2FWD-CRYPTO` processes on the same card because a maximum of six CPU cores can be allocated to DPDK applications. Splitting the available CPU cores is not possible using the latest DPDK version.

5.5 Results Summary

This section summarizes the results of all throughput tests performed using the TNAP and MiTMVP. The first set of treatments test the capability of the Bluefield-2 DPU to offload and accelerate IPsec encryption of Ethernet traffic RDMA. The capability of the Bluefield-2 to accelerate IPsec encryption of Ethernet traffic is limited by the computational capabilities of the processor and memory of the card itself. The average throughput for hardware accelerated IPsec encryption of Ethernet traffic peaks near 5 Gbps. On the other hand, the Bluefield-2 is capable of accelerating IPsec encryption at much higher rates because RDMA traffic bypasses the kernel of the card. When offloading RoCEv2 traffic to the IPsec hardware accelerators of the card, an average throughput of nearly 86 Gbps was achieved.

The second set of treatments test the capabilities of OvS and the `Testpmd` DPDK application to provide a virtual bridge across the MiTMVP network architecture. The combination of `Tcpdump` and OvS performed slightly better than `Testpmd` while forwarding Ethernet traffic. However, `Testpmd` performed much better than OvS when forwarding RoCEv2 traffic. This result suggests that DPDK applications like `Testpmd` could be modified to provide custom monitoring solutions in RDMA fabrics with little degradation of network performance.

The third set of treatments test the capability of the Bluefield-2 to encrypt Ethernet traffic in the software path. Results show that the ARM processor of the Bluefield-2 is capable of encrypting Ethernet traffic at rates up to 8 Gbps using the AES-CBC 128 cipher and no authentication algorithm. These results show that high data rate supported by the Bluefield-2 quickly overwhelms the processor and memory of the chip when the card is expected to encrypt Ethernet traffic. This highlights the advantage of using RDMA fabrics. RoCEv2 traffic bypasses the TCP/IP network stack, and allows traffic to stay within the fast data path of the Bluefield-2.

VI. Conclusion and Recommendations

6.1 Overview

This chapter summarizes the research and results found during experimental evaluation. Section 6.2 reiterates notable conclusions derived from experimentation and statistical analysis. Section 6.3 synthesizes findings to underline InfiniBand security vulnerabilities and provides practical recommendations for improving InfiniBand security. Lastly, Section 6.4 provides possibilities for future work for securing InfiniBand with the Bluefield-2 DPU and similar network adapters.

6.2 Research Conclusions

Convergent InfiniBand and Ethernet communication models like RoCEv2 leverage the superior performance of RDMA and existing TCP/IP network infrastructure. RDMA is a kernel bypass technology that prevents many conventional security applications from being able to sniff network traffic. However, it is imperative that this issue is addressed as these hybrid communication models begin to make their way into critical infrastructure. The Bluefield-2 DPU provides a configurable platform capable of supporting a wide variety of security and network management applications. What separates the Bluefield-2 DPU from other InfiniBand CAs is its high-performance, programmable ARM CPU and suite of cryptography enabled hardware accelerators. This research investigates practical ways of securing the InfiniBand by combining the computational capabilities of the Bluefield-2 DPU with conventional encryption and monitoring technologies.

This research was successful in characterizing the security capabilities of the Bluefield-2 through three contributions: first, designing the TNAP to test the maximum data rates supported by various configurations of the Bluefield-2; second, passive

sniffing using the MiTMVP allowed verification of end-to-end encryption; and third, characterizing the pre-existing hardware and software encryption capabilities of the Bluefield-2.

As hypothesized, the hardware accelerators of the Bluefield-2 DPU are capable of providing near line-rate encryption of RDMA traffic when using Ethernet at the data link-layer (RoCEv2), whereas software encryption implementations quickly overwhelmed the ARM CPU and memory of the Bluefield-2 DPU.

Results show that the Bluefield-2 DPU is capable of accelerating IPsec encryption of RoCEv2 traffic up to 86 Gbps. The capability of the Bluefield-2 DPU to encrypt RoCEv2 traffic at near line-rate is impressive and provides an effective method for adding confidentiality, integrity, and authentication to Remote Direct Memory Access fabrics with minimal interaction from host CPUs. Exploring the capability of the Bluefield-2 DPU to perform software based encryption shows that the Bluefield-2 DPU is capable of supporting up to 5 Gbps IPsec encryption.

6.3 Research Significance and Synthesis

As RDMA fabric architectures like InfiniBand are increasingly used in applications outside the high performance computing domain, they become more susceptible to attacks. Clear text key exchanges, predictable QP numbers, and centralized management make InfiniBand vulnerable to wide variety of attacks. Encryption and authentication can help minimize the threat of packet injection and DoS attacks by adding confidentiality, integrity, and availability to InfiniBand networks. Although encryption and authentication do not resolve all the security vulnerabilities present in the IBA, they provide an important first line of defense.

As seen in the MiTMVP network architecture, a MiTM can passively sniff network traffic at up to 70 Gbps. A similar set-up could be used by an adversary to intercept

confidential information or inject packets of their own into the RDMA fabric. This example is directly relevant to the military or Department of Defense (DoD) as an adversary might seek to falsify information sent across the network.

Organizations seeking to harden the security of RDMA fabric architectures will have to balance the trade-off of network security and performance. The security measures proposed in this research and related research inherently degrade network performance. The degree to which organizations are willing to sacrifice performance for security will likely be dependent on the requirements of the system being built. Organizations seeking to implement RDMA fabric architectures using legacy Ethernet hardware can add confidentiality, integrity, and availability to their networks using chip sets that offer accelerated IPsec encryption like the Bluefield-2 DPU at the end nodes. Other organizations seeking to use native InfiniBand should consider implementing other security mitigation techniques proposed by Rothenberger and colleagues [10] which include randomized QP numbers, hardware counters, randomized memory keys (R_Keys), and sRDMA encryption and authentication. The Bluefield-2 DPU is capable of supporting custom security applications. A custom encryption and authentication solution could be created to support RoCE or native InfiniBand in future research. While these recommendations can improve security of RDMA fabric architectures, none of these ideas completely mitigate the security vulnerabilities within high-performance networks.

6.4 Future Work

There are a number of avenues for extending this research as InfiniBand and other kernel bypass architectures become increasingly prevalent. The following five paragraphs provide options for future work effort based off this research and related research:

1. Custom, link-layer encryption and authentication applications can be developed using DPDK. These applications would likely be limited by the capabilities of the Bluefield-2 DPU to support software. Although the NULL cipher was shown to have a performance ceiling of nearly 40 Gbps, future research should investigate the achievable performance when using light-weight ciphers.

2. The performance benefit realized by kernel bypass technology is at odds with many kernel stack-based network monitoring applications. Future research should investigate methods of performing in-network traffic filtering and monitoring. This research demonstrates the capability of DPDK applications to passively sniff RDMA traffic in userspace. Perhaps, custom filtering or monitoring applications could be developed on top of the existing DPDK applications in future research.

3. The centralized management scheme used by InfiniBand makes the SM a valuable target to adversaries. Future research should investigate vulnerabilities of the SM in order to better protect InfiniBand networks.

4. NVIDIA-Mellanox recently announced the release of the Bluefield-3 DPU. The Bluefield-3 DPU is capable of supporting 200 Gbps Ethernet and InfiniBand. Future research should investigate the security capabilities of the Bluefield-3 and novel ways of securing InfiniBand.

5. The use of machine learning could potentially add security to InfiniBand networks. The high data rate of RDMA fabric architectures makes managing workloads overwhelming. However, applying statistical models can help characterize network performance and AI algorithms could be used to identify and classify irregularities.

6.5 Conclusion

Developers of RDMA architectures like RoCE and InfiniBand have neglected security because security is traditionally associated with degraded network performance. As a result of inherent vulnerabilities in these architectures, adversaries are able to inject packets and gain unauthorized access to memory regions. These attacks can potentially have drastic consequences of exposing confidential information and denying users access to the network. As RDMA architectures become increasingly prevalent, developers must employ mitigations like encryption and authentication. This research shows how the hardware offload and accelerator features offered by programmable network adapters like the Bluefield-2 allow layers of security to be added to RDMA architectures with little interaction from the host CPU or degradation of network performance.

Appendix A. InfiniBand Fabric Utilities Server Bash Script

```
# Author: Noah Diamond, 2d Lt, USAF
# Filename: server.py
# Description: This bash script starts the InfiniBand
# Fabric Utilities server for a series of throughput tests.

#!/bin/bash
echo "Starting server!"
echo "Starting Factorial Design Tests"
for i in 256 4096 2048 1024 512
do
   echo "Testing $i byte MTU"
   declare -a testType=("send" "read" "write")
   for j in ${testType[@]};
   do
      echo "Testing ib_${j}_bw -d mlx5_1 -m $i -n 100000"
      for m in 1 2 3 4 5
      do
         echo "Iteration $m"
         numactl --cpubind=0 ib_${j}_bw --report_gbits -d
            mlx5_1 -m $i -n 100000
         wait
      done
   done
done
echo "Tests Finished"
```

Appendix B. InfiniBand Fabric Utilities Client Bash Script

```
# Author: Noah Diamond, 2d Lt, USAF
# Filename: client.py
# Description: This bash script starts the InfiniBand
# Fabric Utilities client for a series of throughput tests.
# This script waits for one second to ensure the server on
# the other workstation has enough time to start. The results
# of each test are recorded to a text file.

#!/bin/bash
echo "Starting Client!"
echo "Starting Factorial Design Tests"
for i in 256 4096 2048 1024 512
do
   echo "Testing $i byte MTU"
   declare -a testType=("send" "read" "write")
   for j in ${testType[@]};
   do
      echo "Testing ib_${j}_bw -d mlx5_1 -m $i -n 100000"
      for m in 1 2 3 4 5
      do
         echo "Iteration $m"
         numactl --cpubind=0 ib_${j}_bw 10.0.0.4 --
            report_gbits -d mlx5_1 -m $i -n 100000 | tee ./
            IPsec_${i}_${j}_${m}.txt
         wait
         sleep 1
      done
   done
done
echo "Tests Finished"
```

Appendix C. Data Crawler Script

```
"""-------------------------------------------------------------
Author: Noah Diamond, 2d Lt, USAF
Filename: dataCrawler.py
Description: This python script searches textfiles in the
current working directory and will write the average
throughput and filename to a shared results text file. This
script is useful when used after running the server.sh
and client.sh scripts.
-------------------------------------------------------------"""

#! /usr/bin/env python3
import os
import linecache as lc

def main():

    # Create a new text file where results are stored
    results = open("./results.txt", "x")

    # Start compressing results into a single file
    Path = "./"
    filelist = os.listdir(Path)
    for i in filelist:
        if i.endswith(".txt"):
            with open(Path + i, 'r') as f:
                j = 0
                for line in f:
                    if "BW average" in line:
                        k = j + 2
                        # print(k)
                        gfg = lc.getline(i, k)
                        print(gfg)

                        # Add average throughput to a text
                           file.
                        results.write(i + " " + gfg[48:53] +
                           "\n")

                        print("Got here")
                    else:
                        j = j + 1

if __name__ == "__main__":
    main()
```

Bibliography

1. IBTA, "Infiniband Architecture Specification Volume 1 Release 1.4," Tech. Rep., 2015, accessed: 30 Apr 2020 [Online]. Available: https://www.infinibandta.org/ibta-specifications-download/.

2. S. Sur and D. Panda, "Designing Cloud and Grid Computing Systems with InfiniBand and High-Speed Ethernet," 2011. [Online]. Available: http://www.ics.uci.edu/~ccgrid11/files/ccgrid11-ib-hse_last.pdf

3. Intel Corporation, "Virtual Interface Architecture Specification," 1997, accessed: 17 Dec 2021 [Online]. Available: https://cupdf.com/document/virtual-interface-architecture-specification-uml-billcs520vispecpdfvirtual.html.

4. IBTA, "Supplement to InfiniBand Architecture Specification Annex A17: RoCEv2," Tech. Rep., 2014, accessed: 14 Dec 2021 [Online]. Available: https://cw.infinibandta.org.

5. P. Kennedy, "NVIDIA BlueField-2 and BlueField-2X DPU Offerings Launched," 2020, accessed: 9 Dec 2021 [Online]. Available: https://www.servethehome.com/nvidia-bluefield-2-and-bluefield-2x-dpu-offerings-launched/.

6. Mellanox-Technologies, "BlueField Software Overview - BlueField DPU OS 3.8.0 - Mellanox Docs," 2021, accessed: 9 Dec 2021 [Online]. Available: https://docs.mellanox.com/display/BlueFieldDPUOSv380/ BlueField+Software+Overview.

7. Mellanox-Technologies, "Introduction - MLNX_OFED v4.6-1.0.1.1," 2021, accessed: 9 Dec 2021 [Online]. Available: https://docs.mellanox.com/display/MLNXOFEDv461000/Introduction.

8. Mellanox-Technologies, "Modes of Operation - BlueField SW Manual v2.2.0.11000 - Mellanox Docs," accessed: 13 Dec 2021 [Online]. Available: https://docs.mellanox.com/display/BlueFieldSWv22011000/Modes+of+Operation.

9. J. Kurose and K. Ross, *Computer Networking A Top-Down Approach*, 7th ed. Pearson, 2017.

10. B. Rothenberger, K. Taranov, A. Perrig, and T. Hoefler, "ReDMArk : Bypassing RDMA Security Mechanisms," *Usenix Security*, pp. 1–16, 2020.

11. Z. Kerravala, "Despite Predictions of Its Demise, InfiniBand is Still Alive - eWEEK," 9 2020, accessed: 10 Jan 2022 [Online]. Available: https://www.eweek.com/networking/despite-predictions-of-its-demise-infiniband-is-still-alive/.

12. E. Strohmaier, J. Dongarra, H. Simon, and M. Meuer, "TOP 500 The List," 2021, accessed: 7 Jun 2021 [Online]. Available: https://top500.org/statistics/list/.

13. G. F. Pfister, "An introduction to the InfiniBand(TM) architecture," *High Performance Mass Storage and Parallel I/O: Technologies and Applications*, pp. 617–632, 2001.

14. M. Lee and E. J. Kim, "A comprehensive framework for enhancing security in InfiniBand architecture," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 10, pp. 1393–1406, 2007.

15. IBTA, "InfiniBand Trade Association Website," 2021, accessed: 7 Jun 2021 [Online]. Available: https://www.infinibandta.org/.

16. Paul Grun, "Introduction to InfiniBand(TM) for End Users," 2010, accessed: 11 Dec 2021 [Online]. Available: https://www.mellanox.com/pdf/whitepapers/Intro_to_IB_for_End_Users.pdf.

17. Mellanox-Technologies, "Introduction to InfiniBand," Tech. Rep., 2003, accessed: 15 Jul 2021 [Online]. Available: https://www.mellanox.com/pdf/whitepapers/IB_Intro_WP_190.pdf.

18. NVIDIA, "Single Root IO Virtualization," accessed: 10 Jan 2022 [Online]. Available: https://docs.nvidia.com/networking/pages/viewpage.action?pageId=39279752.

19. P. Grun, "RoCE and InfiniBand: Which should I choose?" 2012, accessed: 3 Jan 2022 [Online]. Available: https://www.infinibandta.org/roce-and-infiniband-which-should-i-choose/.

20. NVIDIA, "NVIDIA Bluefield-2 DPU Data Center Infrastructure on a Chip," 2021, accessed: 3 Jun 2021 [Online]. Available: https://www.nvidia.com/content/dam/en-zz/Solutions/Data-Center/documents/datasheet-nvidia-bluefield-2-dpu.pdf.

21. Mellanox-Technologies, "NVIDIA Mellanox BlueField Data Processing Unit (DPU)," pp. 1–4, 2020.

22. C. linux Project, “Cryptodev-linux Module,” 2013, accessed: 12 Jan 2022 [Online]. Available: http://cryptodev-linux.org/.

23. NVIDIA, “Kernel Representors Model - BlueField DPU SW Manual,” accessed: 10 Jan 2022 [Online]. Available: https://docs.nvidia.com/networking/display/BlueFieldSWv35011563/Kernel+Representors+Model.

24. Mellanox-Technologies, “Mellanox ASAPˆ2,” pp. 1–3, 2020. [Online]. Available: https://www.mellanox.com/files/doc-2020/sb-asap2.pdf

25. The Linux Foundation, “Myth-busting DPDK in 2020,” 2020, accessed: 10 Dec 2021 [Online]. Available: https://nextgeninfra.io/dpdk-myth-busting-2020/.

26. Mellanox-Technologies, “MLX5 Poll Mode Driver,” accessed: 14 Dec 2021 [Online]. Available: http://doc.dpdk.org/guides/nics/mlx5.html.

27. M. Kerrisk, “top(1) - Linux manual page,” accessed: 14 Dec 2021 [Online]. Available: https://man7.org/linux/man-pages/man1/top.1.html.

28. A. Kleen, “numactl(8) - Linux man page,” accessed: 14 Dec 2021 [Online]. Available: https://linux.die.net/man/8/numactl.

29. P. Biondi, “Scapy,” accessed: 14 Dec 2021 [Online]. Available: https://scapy.net/.

30. H. Ware and F. Frederick, “vmstat(8) - Linux man page,” accessed: 14 Dec 2021 [Online]. Available: https://linux.die.net/man/8/vmstat.

31. F. Baumgarten, M. Welsh, A. Cox, T. Hoang, and B. Eckenfels, “netstat(8) - Linux man page,” accessed: 14 Dec 2021 [Online]. Available: https://linux.die.net/man/8/netstat.

32. J. Dugan, S. Elliott, B. A. Mah, J. Poskanzer, and K. Prabhu, “iPerf - The ultimate speed test tool for TCP, UDP and SCTP,” 2021, accessed: 14 Dec 2021 [Online]. Available: https://iperf.fr/iperf-download.php.

33. The Libreswan Project, “Libreswan VPN software,” 2021, accessed: 14 Dec 2021 [Online]. Available: https://libreswan.org/.

34. Linux Foundation, “Open vSwitch,” 2016, accessed: 14 Dec 2021 [Online]. Available: https://www.openvswitch.org/.

35. Mellanox-Technologies, “InfiniBand Fabric Utilities - MLNX_OFED v4.6-1.0.1.1 - Mellanox Docs,” accessed: 14 Dec 2021 [Online]. Available: https://docs.mellanox.com/display/MLNXOFEDv461000/InfiniBand+Fabric+Utilities.

36. The Tcpdump Group, "Tcpdump & Libpcap," accessed: 14 Dec 2021 [Online]. Available: https://www.tcpdump.org/.

37. G. Combs, "Wireshark," 2021, accessed: 14 Dec 2021 [Online]. Available: https://www.wireshark.org/.

38. K. Taranov, B. Rothenberger, A. Perrig, and T. Hoefler, "sRDMA - Efficient NIC-based authentication and encryption for remote direct memory access," *Proceedings of the 2020 USENIX Annual Technical Conference, ATC 2020*, pp. 691–704, 2020.

39. A. Romanow, J. Mogul, T. Talpey, and S. Bailey, "rfc 4297," 2005, accessed: 14 Dec 2021 [Online]. Available: https://datatracker.ietf.org/doc/html/rfc4297.

40. L. Mireles, "Implications and Limitations of Securing an InfiniBand Network," Master's thesis, Air Force Institute of Technology, 2020, accessed: 5 Jan 2021 [Online]. Available: https://scholar.afit.edu/etd/3183/.

41. K. Hintze, "Infiniband Network Monitoring: Challenges and Possibilities," Master's thesis, Air Force Institute of Technology, 2021, accessed: 5 Jan 2021 [Online]. Available: https://scholar.afit.edu/etd/4902/.

42. M. Technologies, "100Gb / s QSFP28 MMF Active Optical Cables," pp. 2–3, 2017, accessed: 20 Dec 2021 [Online]. Available: https://www.mellanox.com/related-docs/prod_cables/PB_MFA1A00-Cxxx_100GbE_QSFP28_MMF_AOC.pdf.

43. cs.lev, "NVIDIA Mellanox Bluefield-2 SmartNIC Hands-On Tutorial: "Rig for Dive"," 2021, accessed: 17 Dec 2021 [Online]. Available: https://cslev.medium.com/.

44. D. Montgomery, *Design and Analysis of Experiments*, 10th ed. Wiley, 2019.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
24–03–2022	Master's Thesis	Sept 2020 — Mar 2022

4. TITLE AND SUBTITLE

SECURING INFINIBAND NETWORKS WITH END-POINT ENCRYPTION

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Diamond, Noah B., 2d Lt, USAF

5d. PROJECT NUMBER

18G230

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering an Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-MS-22-M-024

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory
2241 Avionics Circle
WPAFB OH 45433-7765
Attn: Steven Stokes
COMM 937-528-8035
Email: steven.stokes@us.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RYWA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The NVIDIA-Mellanox Bluefield-2 is a 100 Gbps high-performance network interface which offers hardware offload and acceleration features that can operate directly on network traffic without routine involvement from the ARM CPU. This allows the ARM multi-core CPU to orchestrate the hardware to perform operations on both Ethernet and RDMA traffic at high rates rather than processing all the traffic directly. A testbed called TNAP was created for performance testing and a MiTM verification process called MiTMVMP is used to ensure proper network configuration. The hardware accelerators of the Bluefield-2 support a throughput of nearly 86 Gbps when using IPsec to encrypt and authenticate RoCEv2 traffic. This research closes by providing operational security recommendations to defend against presented vulnerabilities, and secure InfiniBand with the Bluefield-2 DPU and similar InfiniBand channel adapters.

15. SUBJECT TERMS

InfiniBand, Cybersecurity, Bluefield-2 DPU, Hardware Acceleration, Encryption

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Scott Graham, AFIT/ENG
U	U	U	UU	111	19b. TELEPHONE NUMBER *(include area code)* (937) 255-6565 x4581; scott.graham@afit.edu