Future Data Centers: Hardware-Offloaded Post-Quantum Secure Optical IPsec Tunnel for Confidential AI Training

1st D. C. Lawo
Networking Software Architecture
NVIDIA Corporation
2066730 Yokneam Illit, Israel
dlawo@nvidia.com

2nd R. Abu Bakar
TeCIP Institute
Scuola Superiore Sant'Anna
56124 Pisa, Italy
rana.abubakar@santannapisa.it

3rd A. Cano Aguilera
Networking Software Architecture
NVIDIA Corporation
2066730 Yokneam Illit, Israel
acanoaguiler@nvidia.com

4th F. Cugini
PNT Lab
CNIT
56124 Pisa, Italy
filippo.cugini@cnit.it

5th José L. Imaña
Arquitectura de Computadores y Automática
Universidad Complutense de Madrid
28040 Madrid, Spain
jluimana@ucm.es

6th I. Tafur Monroy
Department of Electrical Engineering
Technical University of Eindhoven
5612 AP Eindhoven, The Netherlands
i.tafur.monroy@tue.nl

7th J. J. Vegas Olmos
Networking Software Architecture
NVIDIA Corporation
2066730 Yokneam Illit, Israel
juanj@nvidia.com

***Abstract*—We propose a $100\,\mathrm{Gbit/s}$ line rate post-quantum cryptography (PQC) secured IPsec tunnel for optical intra-data center confidential AI training. Our IPsec tunnel also provides mobile client confidential training AI PQC-secure with $0.486\,\mathrm{Gbit/s}$ speed.**

***Index Terms*—post-quantum cryptography, data processing unit, IPsec, data center, AI training.**

I. Introduction

As artificial intelligence (AI) models grow larger, the demands on the networking infrastructure of data centers increase significantly, especially during the training phase. At the same time, the data used for training these models is becoming increasingly sensitive, including for example financial or medical information. This creates an urgent need for secure, confidential training environments in data centers, without compromising the computational performance. In the meantime, another high-impact technology is being developed: Quantum computers are expected to become commercially available in the coming years. These machines will be able to break our currently used public key cryptography methods. To withstand the quantum thread, the National Institute of Standards and Technology (NIST) chose to standardize different post-quantum cryptography (PQC) algorithms [1]. Porting those protocols to data centers while still satisfying the ever increasing computational demand of modern AI models is a significant technological challenge. This paper presents the first demonstration of Falcon [2], Dilithium [3], and Kyber [4] in (a) a line-rate east-west data center traffic and (b) a hybrid wifi-single mode fiber (SMF) north-south traffic. During a confidential training job for AI, millions of IPsec connections are established per second. Our east-west traffic experiment accounts for this. Accessing the service of an already trained model is usually done from outside of the data center, potentially by a mobile client in a wireless network. We show this with our north-south traffic experiment. In modern AI models, inference is desired, putting additional load on the cloud's network after a client request has been submitted. This scenario can be obtained by combining the both scenarios that we present in this work: first a client connects to the cloud that runs the AI model via north-south traffic; then, within the cloud, the request is handled using out east-west IPsec scenario.

This work was partly funded by the Chips Joint Undertaking (JU), the European Union (EU) HORIZON-JU-IA SMARTY (101140087), QUARC (101073355), and CLEVER (101097560) projects.

II. Experimental Setup

To establish a PQC-secured IPsec channel, we first initiate an OpenSSL session to create an authenticated connection. For research purposes, self-signed certificates are used for authentication, in a real-life deployment, certificates from a trusted certificate authority would be required. The authentication process itself remains the same. Once the authenticated channel is in place, digital signatures are exchanged using the PQC algorithms Falcon [2] and Dilithium [3]. After that, an encryption key is exchanged via Kyber [4]. The PQC-generated keys are then mixed with the OpenSSL key using an XOR operation, ensuring the resulting key remains secure if at least one of the original keys is secure [5]. This resulting shared key is used to establish an IPsec tunnel that uses 256-bit Advanced Encryption Standard (AES) encryption operating in the Galois/Counter mode (GCM). AES-256 is recognized as resistant to quantum computing attacks [6]. Depending on the re-keying interval, the key exchange procedure is repeated every ten minutes or every 2^{30} cipher blocks. Setting up the IPsec tunnel requires superuser privileges. For programming

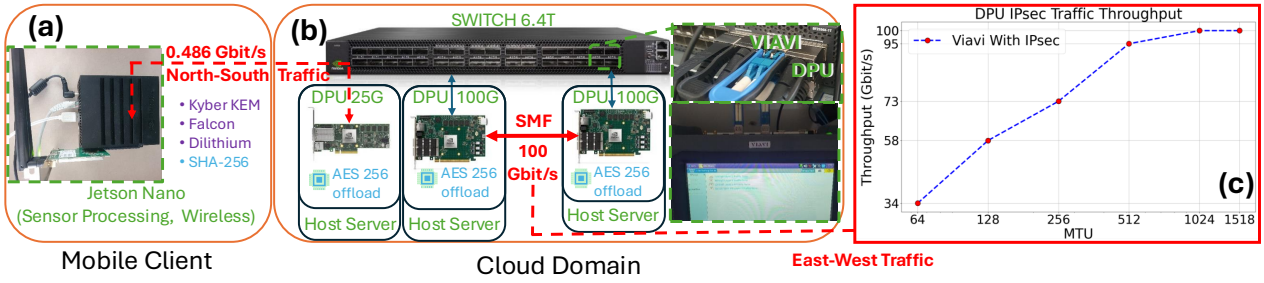

Fig. 1. Experimental Setup: (a) A mobile client communicates with the cloud using a PQC-encrypted north-south IPsec tunnel at 0.486 Gbit/s for confidential AI training. (b) The servers in the data center for confidential AI training use an PQC-based east-west IPsec tunnel, offloading AES-256 to the DPUs achieving (c), encryption at 100 Gbit/s traffic.

the data processing unit (DPU), we use the APIs provided by NVIDIA's DOCA (Data-Center-on-a-Chip) SDK[1] to take advantage of the smartNIC's flow steering. We perform this process on various devices to simulate different scenarios, as depicted in Fig. 1. To achieve statistically relevant results, we test each algorithmic procedure 10.000 times. In the first setup, shown in Fig. 1 (a), we used a NVIDIA Jetson Nano as a mobile client with a 1 Gbit/s-capable Wi-Fi antenna to connect to the wireless network, which then connects to a 25G DPU in the cloud. The PQC-secured IPsec tunnel achieved an encrypted wireless throughput of 0.468 Gbit/s. This setup represents north-south traffic in a data center. The second scenario, illustrated in Fig. 1 (b), involves an east-west channel within a data center, where multiple servers on the same network communicate by establishing PQC-secured channels. During the training of an AI model, this happens millions of times a second. In this setup, we establish a PQC IPsec tunnel between two 100 Gbit/s DPUs connected via standard optical SMF. We tested the throughput of our mobile client PQC-IPsec tunnel using Iperf[2]. The east-west traffic throughput between the two 100G DPU was tested using the VIAVI traffic generator. While the VIAVI generator is capable of achieving high data rates of up to 400 Gbit/s per port, our tests were conducted using 100 Gbit/s DPUs. The VIAVI traffic generator was connected between two DPUs via a QSFP cable for the testing.

III. Experimental Results and Discussion

Using the Iperf traffic generator, we analyzed the performance of the north-south IPsec tunnel established between the Jetson as a mobile client and the 25G DPU in the cloud. We achieved an AES-256 GCM-encrypted wireless throughput of 0.486 Gbit/s. Since this scenario simulates a mobile client communicating with the cloud, the signal traverses multiple hops along the way. Consequently, we did not set the maximum transmission unit (MTU), as any device in the chain between the mobile device and the 25G DPU in the cloud can modify the MTU size.

[1]https://developer.nvidia.com/networking/doca
[2]https://iperf.fr/

In our intra-data center east-west traffic scenario, we have control over the MTU size. After setting up the east-west IPsec tunnel between the DPUs, we measured the tunnel's throughput with various MTU sizes using the VIAVI traffic generator. The results are shown in Fig. 1 (c). With 64 B MTU sized packets we achieved a throughput of 34 Gbit/s. Doubling the MTU to 128 B increased the throughput to 58 Gbit/s. Setting the MTU to 256 B resulted in a throughput of 73 Gbit/s. At 512 B MTU, the throughput reached 95 Gbit/s. Finally, from 1024 B MTU the throughput converges to 100 Gbit/s line rate. This holds true for all MTU sizes greater than or equal to 1024 B, including jumbo-sized packets.

Figure 2 shows the latency in CPU clock cycles introduced by cryptography operations executed on different devices and processors. The different variants of the algorithms account for different NIST security levels. The first row represents the operations that are performed by the client machine. During the execution of a signature algorithm (Falcon and Dilithium), the client has to perform one step only that is called verification. For the key exchange (Kyber), the client has to perform the key encapsulation. The second row represents the operations that are performed by the server machine. To successfully execute a signature algorithm, the server needs to do two steps: the key generation and the sign process. For the key exchange, the server must perform the key generation and the key decapsulation. The results shown in the second row of Fig. 2 are a sum of the CPU cycles required for the execution of each single step (keygen+sign, keygen+key decapsulation). Each column represents one class of devices: The first column shows the operations performed on the 25G DPU. The second column shows the same for the Jetson, our mobile client device. The third column presents the results for an Intel Xeon CPU. The devices shown in column one and row three of Fig. 2 represent two scenarios within the data center. The first setting is that the machine does not offload any operation to the network interface card (NIC) and uses the NIC for outgoing and incoming communication only. Therefore, every time a new connection needs to be established, the host has to perform all cryptography operations itself which leads to a penalty. This can be seen in the third column of Fig. 2. In the second setting, depicted in the first column of Fig. 2,

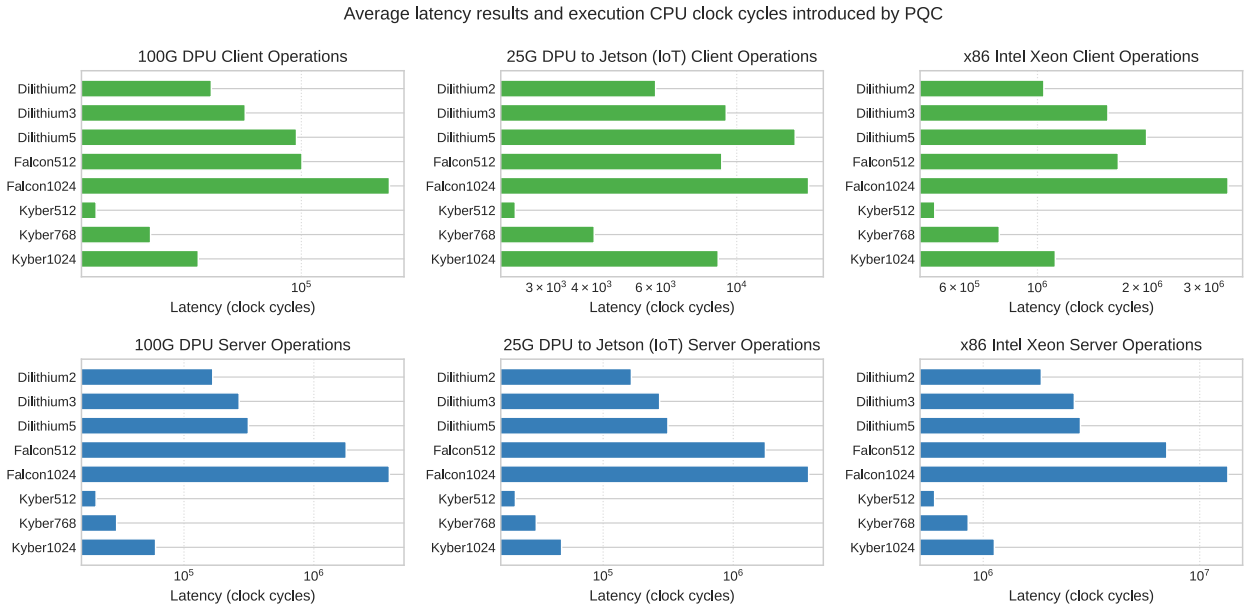

Fig. 2. Average latency results in CPU clock cycles introduced by the execution of PQC cryptography. The first row shows the latency the client machine is penalized with. The second row represents the tax in CPU cycles introduced by PQC operations when performed on the server machine.

the host offloads the cryptography functions to its DPU. The IPsec tunnel is established between the two DPUs. The host sends unencrypted information as plaintext to the DPU via the PCI interface. The DPU handles encryption and decryption.

Kyber is yet the only key encapsulation mechanism (KEM) chosen to be standardized by the NIST and is therefore expected to play a fundamental role in future network communication stacks. The algorithm performs similarly well in terms of execution speed compared to classical key exchange mechanisms (KEMs). Regardless, Kyber needs to transfer more data over the network. Comparing Falcon and Dilithium yields the following results: Falcon's key generation is three orders of magnitude slower than Dilithium's. Falcon's sign process, as well as its verification, is slightly slower but within the same order of magnitude compared to Dilithium. However, Falcon's signature is smaller than Dilithium's. The choice which security level to use has to be determined by the developer ultimately. Which signature algorithm to choose in addition to Kyber depends on the use-case. In a data center, Dilithium proofs advantageous due to its higher performance. With limited network capabilities, Falcon is more advantageous than Dilthium for its signature size is smaller and thus, fewer bytes need to be sent over the network.

IV. Conclusion

In this work, we presented intra-data center PQC secure communications and composite WiFi-fiber deep-edge connections using PQC algorithms for confidential AI training. We achieved a PQC-encrypted north-south WiFi-fiber hybrid throughput of $0.486\,\mathrm{Gbit/s}$. Post-quantum security was established through a PQC key exchange, with the resulting ephemeral keys used to secure an IPsec tunnel employing AES-256 encryption. Our implementation demonstrates the integration of PQC in two critical data center scenarios for confidential AI training: high-speed east-west traffic within the data center and north-south traffic between the data center and external networks. For intra-data center communication, we achieved an east-west throughput of $100\,\mathrm{Gbit/s}$ by offloading cryptographic tasks from the host to the DPUs. Our findings indicate that Dilithium is the preferred signature algorithm for data centers, while Falcon outperforms in low-power applications. Both scenarios that we presented in this work show that it is possible to achieve PQC-secured AI requests via north-south traffic, as well as confidential AI training in the data center using east-west traffic. This way, we accomplish resilience against the arising quantum thread.

References

[1] Gorjan Alagic et al. Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process. Technical report, National Institute of Standards and Technology (U.S.), 2022.

[2] Pierre-Alain Fouque et al. Fast-Fourier Lattice-based Compact Signatures over NTRU. https://falcon-sign.info/.

[3] Léo Ducas et al. CRYSTALS-Dilithium: A Lattice-Based Digital Signature Scheme. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2018(1):238–268, Feb. 2018.

[4] Joppe Bos et al. Crystals - kyber: A CCA-Secure Module-Lattice-Based KEM. In *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 353–367, 2018.

[5] Kunal Meher and Divya MidhunChakkaravarthy. New Approach to Combine Secret Keys for Post-Quantum (PQ) Transition. *Indian Journal of Computer Science and Engineering*, 12(3):629–633, 2021.

[6] Xavier Bonnetain et al. Quantum Security Analysis of AES. *IACR Transactions on Symmetric Cryptology*, 2019(2):55–93, June 2019.