TLV-to-MUC Express: Post-quantum MACsec in VXLAN

Joo Yeon Cho[1(✉)] and Andrew Sergeev[2]

[1] ADVA Optical Networking SE, Fraunhoferstrasse 9a, 82152 Martinsried, Germany
JCho@adva.com

[2] ADVA Optical Networking Israel Ltd., 2 Hatidhar Street, 4366105 Ra'anana, Israel
ASergeev@adva.com

Abstract. MACsec in VXLAN is an end-to-end security protocol for protecting Ethernet frames traveling over IP networks. It can provide a high-speed Ethernet encryption while supporting the virtualization of a large network such as data center network. Although MACsec addresses most of security threats, it is not immune against quantum attacks which are a future, yet disastrous threat against public-key cryptography in use. In this paper, we demonstrate a new solution for a MACsec protocol over VXLAN in a post-quantum setting. Instead of a standard MACsec key agreement protocol, we use an ephemeral key exchange protocol and an end-to-end authentication scheme, both of which are based on post-quantum cryptography. To measure the impact on the performance, we established a quantum-secure link between Germany and Israel using MACsec in VXLAN over public IP networks. We verified that the impact on the latency and throughput is minimal. Our experiment confirms that quantum-secure virtualized links can be already established in a long-distance without changing their infrastructure.

Keywords: MACsec · Ethernet · VXLAN · VPN · Quantum security · Authentication

1 Introduction

Due to ever-increasing risk of cyber attacks, data protection is as important as its speed and performance assurance in modern networks. Wide Area Network (WAN) has been driving the industry to innovate to increase security as well as transport speeds. MACsec (Media Access Control Security) is an IEEE 802.1AE standard protocol for secure communication on Ethernet links to ensure confidentiality, integrity and origin authenticity of user data in transit.

While the standard MACsec protocol was developed for Local Area Network (LAN) security, it can be also used for WAN security with additional frame overhead. A common approach is to add a Virtual LAN (VLAN) tag to the MACsec frame [15]. This tag allows MACsec frames to travel multiple hops and enables the end-to-end network encryption over carrier Ethernet. However, it

M. Asplund and S. Nadjm-Tehrani (Eds.): NordSec 2020, LNCS 12556, pp. 127–141, 2021.
https://doi.org/10.1007/978-3-030-70852-8_8

requires MACsec-aware intermediate switches and bridges in the middle. Hence, MACsec frames often do not cross over IP networks, leaving the deployment of VLAN limited in practice.

VXLAN (Virtual Extensible Local Area Network) technology can extend VLAN and overcome the limited capability and scalability posed by VLAN. VXLAN creates a layer 2 tunnel on top of layer 3 by encapsulating Ethernet frames in UDP frames, enabling large-scale virtualized and multitenant data center designs over a shared common physical infrastructure. Hence, VXLAN is commonly used for a site-to-site VPN such as data center networks. VXLAN allows Ethernet frames to travel over IP networks as long as the terminal device is able to decapsulate the VXLAN into MACsec frames.

1.1 Our Contribution

Attacks using quantum computers are a future, yet critical threat against cyber security. Although quantum attacks should not be overstated, it is sensible to prepare new crypto schemes relying on the quantum-resistant mathematical hardness for a long term security.

A MACsec key agreement protocol in a post-quantum setting is investigated in [8]. We extend this approach to a MACsec in VXLAN tunneling. We demonstrated by experiments that a quantum-secure MACsec in VXLAN can be applied in a long distance network without any modification of infrastructure.

The core primitives of the protocol are a key encapsulation mechanism (KEM) and a digital signature scheme, both of which are adapted from the 3rd round finalists of NIST PQC project [2]. We compare their performance in terms of latency and throughput. In order to achieve a forward security, an end-to-end ephemeral key exchange protocol is established, meaning that each session key is independently generated and there is no way to restore the previous keys even though a current key is disclosed. In fact, this approach has been already widely adopted in the industry, especially for WAN or MAN security although this does not comply with a standard MACsec key agreement protocol.

The rest of this paper is structured as follows: first, we briefly describe the background on MACsec in VXLAN and post-quantum cryptography. Then, we propose a framework of the MACsec in VXLAN in a post-quantum setting. Next, we describe our test platform and experimental results. Finally, we conclude the paper.

2 Background

In this section, the MACsec protocol is briefly described in terms of encryption, authentication and key management. Then, the benefits of VXLAN are discussed. Later, a brief introduction on the post-quantum crypto algorithms is given.

Table 1. PQC primitives: the 3rd round finalists of NIST PQC project [2] and hash-based signatures from IETF [9]

SDO	Family	*KEM*	*Signature*
NIST	Lattice-based	CRYSTALS-KYBER [28]	CRYSTALS-DILITHIUM [23]
		NTRU [6]	FALCON [26]
		SABER [10]	
	Code-based	Classic McEliece [5]	–
	Multivariate	–	Rainbow [11]
IETF	Hash-based	–	XMSS [14]
			LMS [24]

2.1 MACsec

MACsec is an IEEE standard protocol for Layer-2 security [17]. A MACsec packet is formed with an Ethernet frame by adding a SecTAG (Security TAG) and an ICV (Integrity Check Value). A SecTAG conveys information on the protocol, the cipher suites, as well as the PN (packet number) for replay protection. An ICV is a compressed value of the MAC address, SecTAG, and secure data to ensure the integrity of a packet. Note that payload encryption is optional. If a packet-authentication-only mode is configured, MACsec can verify only the integrity of a transmitted packet.

MACsec supports a limited number of symmetric-key cipher suites: AES-GCM-128 and AES-GCM-256 with a usage of XPN (eXtended PN) as an option [17]. AES-GCM-128 is a default cipher suite. IEEE 802.1AEbn-2011 [18] adds GCM-AES-256 as an optional cipher suite to allow a 256-bit key. IEEE 802.1AEbw-2013 [19] adds GCM-AES-XPN-128 and GCM-AES-XPN-256 for further optional cipher suites that make use of a 64-bit (PN) to allow more than 2^{32} MACsec protected frames to be sent with a single SAK (Secure Association Key). MACsec is now part of the Linux kernel from the version 4.6 [20]. Note that the National Security Agency (NSA) designed the Ethernet Security Specification (ESS) on top of MACsec for providing a hardened layer 2 encryption scheme [25].

Although MACsec was developed for LAN security, a MACsec frame can transverse across local networks by applying VLAN tags defined in IEEE 802.1Q [15]. This technique allows MACsec to be used for WAN (wide area network) security and provide the end-to-end network encryption over carrier Ethernet.

2.2 Virtual Extensible LAN

There are common ways to virtualize Layer 2 networks; VLAN and VXLAN. VLAN is widely used for traffic separation and network segmentation in the enterprise environment. According to the IEEE 802.1Q standard, traditional VLAN identifiers are 12 bits long. This limits network segments to 4094 VLANs.

The VXLAN protocol overcomes this limitation by using a longer logical network identifier that allows more VLANs. The VXLAN protocol uses a 24-bit logical network identifier that allows 2^{24} network virtualizations in total. In addition, VxLAN frames are encapsulated with UDP thus can be transported across IP networks. Hence, VXLANs become dominant for large networks such as cloud environments that typically include many virtual machines. In data centers, VXLAN is commonly used to create overlay networks on top of the physical network, enabling the use of a virtual network of devices. The VXLAN protocol supports the virtualization of the data center network and addresses the needs of multi-tenant data centers by providing the necessary segmentation on a large scale. Note that VXLAN encapsulation by itself does not provide any security features, hence, networks must be protected by other means e.g. as introduced in [21].

2.3 Post-quantum Cryptography

The goal of post-quantum cryptography is to develop cryptographic systems that are secure against both quantum and classical computers, and can interoperate with existing communications protocols and networks [7]. Post-quantum cryptography is usually classified into five families: code-based, lattice-based, multivariate, symmetric-based, and supersingular isogeny-based. Each family is based on a different mathematical problem that is not feasible so far to solve both with traditional computers as well as quantum computers. Recently, post-quantum cryptography has drawn lots of attention from the community mainly due to the NIST PQC project [7]. Code-based crypto has strength on KEM. It has been studied for a long time and, the theory is well developed and understood. However, the key size is usually quite large, compared to other families. It seems not suitable for signature schemes. Lattice-based crypto is the most popular among other families. It is applicable to both KEM and signature. However, selecting security parameters is challenging since their security is still not

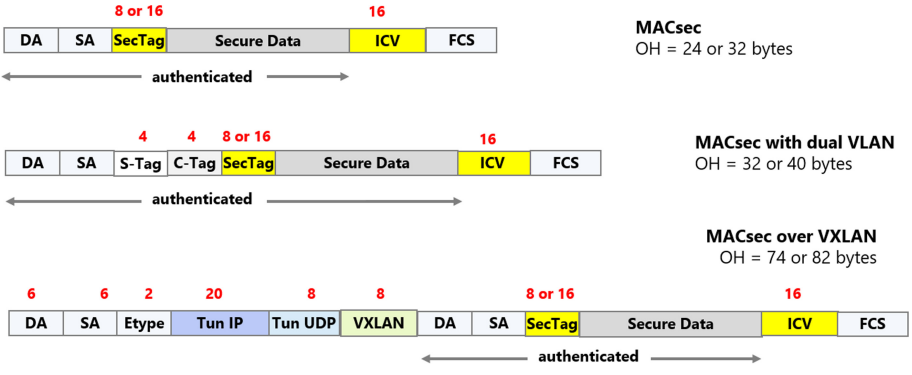

Fig. 1. Comparison of overhead: MACsec encapsulation in VXLAN

well-understood. Multivariate crypto is suitable for signature but not for KEM. Isogeny-based crypto is relatively new but very promising for KEM in terms of the key size. The NIST announced the third round finalists in July 2020 [2]. We sorted out the finalists in Table 1. In addition, NIST supports hash-based signatures that have been already published in IETF [9]. Based on another document from NIST, it is likely that future post-quantum cryptographic standards will specify multiple algorithms for different applications because of differing implementation constraints (e.g., sensitivity to large signature size or large keys) [4].

3 Protocols

In this section, we describe a frame format of MACsec in VXLAN. Then, we present an authenticated post-quantum key exchange protocol in details, together with several aspects of practical considerations.

3.1 Encapsulating MACsec in VXLAN

Due to the minimal requirement of overhead and simple configuration, MACsec is often used for high-speed connectivity at low power and low cost. The disadvantage of MACsec is that all traffic traversing the link requires matching and verifying secret keys at each node. In reality, this downside is avoided by adding a VLAN tag defined in IEEE 802.1Q [15] and re-locating some header fields. However, most of public nodes do not accept the VLAN tag, leaving the MACsec with VLAN tag suitable for a dedicated private link requiring high security. In VXLAN, a UDP header is placed in front of the frame, as shown in Fig. 1. Hence, if a MACsec packet is encapsulated in VXLAN, it can travel over public nodes as long as UDP is accepted. The downside is of course the increased size of overhead. The comparison of frame format is given in Fig. 1.

3.2 Authenticated PQ Key Exchange

Limits on Data Usage. There are cryptographic limits on the amount of data which can be safely encrypted under a single key. For example, TLS 1.3 specifies limits on the number of data to be encrypted by AES-GCM up to $2^{24.5}$ full-size records with a safety margin of approximately 2^{-57} [27]. A new IETF RFC is initiated for the detailed formulation [13]. For this reason, an authenticated key exchange (AKE) protocol is periodically executed and a session key is refreshed before such data limit is reached.

Let us remind that a payload of MACsec frame is encrypted using AES-GCM. A limit on the amount of data that are encrypted by MACsec without needing a key change is determined on the volume of data transmission. Here is an example. Suppose the transmission rate of MACsec packets is 1 Gbps (= 0.125 Gbps). According to [22], the amount of data that can be safely encrypted

with a single key is around 0.3887 terabytes with 2^{-60} success probability, which leads us to calculate its re-key rate as

$$\frac{0.3887 \text{ TB/key}}{0.125 \text{ GB/s}} = 31096 \text{ sec/key} \approx 51.8 \text{ min/key}.$$

This means that a key exchange protocol should be executed every 52 min or less to achieve such error probability. More information are given in Table 2.

Table 2. Max data vs. Re-key rate for an 1 Gbps link

Attack success prob.	Max data (terabytes) [22]	Re-key rate
2^{-60}	0.3887	52 min
2^{-50}	12.44	28 h
2^{-40}	398.1	37 days
2^{-30}	12,738	3.2 years

Ephemeral PQ AKE Protocol. The standard MACsec key agreement (MKA) protocol is a centralized key derivation mechanism based on a hierarchical key structure [16]. Even though MKA is efficient and suitable for LAN, it has a non-negligible risk of being hacked by a root key disclosure, which may lead a severe security breach in entire networks. Especially this risk become high if MACsec is used for WAN and MAN security. Hence, as stated earlier, an ephemeral key exchange has been widely adopted in the industry. That is, each session key is derived independently from the previous session keys so that the disclose of current session key does not reveal any data encrypted in the past. This is called forward secrecy.

Suppose Initiator and Responder perform an AKE protocol. Both peers are assumed to have generated a pair of public and secret key. To agree upon a new session key, two peers execute an AKE protocol using PQ crypto primitives listed in Table 1. The detailed protocol is depicted in Fig. 2.

Hybrid AKE Protocol. A hybrid AKE protocol is a combination of post-quantum authenticated key exchange with classical standard crypto schemes. For instance, McEliece KEM with Falcon signature can be combined with DH (Diffie-Hellman) key exchange protocol with RSA signature. In this way, keys derived by a hybrid AKE scheme remain secure if at least one of the component schemes is secure. This is called crypto agility. Note that post-quantum crypto protocols are added independently on top of the standard protocol, rather than being merged together because the security proof of each protocol should be preserved. Recently NIST revised SP 800-56C to permit the use of a hybrid key establishment construction in FIPS 140 validation [3]. The use of hybrid key exchange and dual signature scheme is an on-going research topic e.g. [29].

Initiator(I)		*Responder*(R)
Required: pk_I, sk_I		Required: pk_R, sk_R
Initiate AKE		Respond AKE
	$\overset{Start}{\longleftrightarrow}$	
Choose random number r_I, k_I		Choose random number r_R, k_J
$X_I = \{pk_I, \text{PQSIGN}_{sk_I}(pk_I, r_I)\}$		$X_R = \{pk_R, \text{PQSIGN}_{sk_R}(pk_R, r_R)\}$
	$\overset{X_I, X_R}{\longleftrightarrow}$	
Verify X_R		Verify X_I
$Y_I = \text{PQENC}_{pk_R}(k_I)$		$Y_R = \text{PQENC}_{pk_I}(k_R)$
	$\overset{Y_I, Y_R}{\longleftrightarrow}$	
$k_R = \text{PQDEC}_{sk_I}(Y_R)$		$k_I = \text{PQENC}_{sk_R}(Y_I)$
$K = k_I \Vert k_R \Vert r_I \Vert r_R$		$K = k_I \Vert k_R \Vert r_I \Vert r_R$
$M_I = MAC(K)$		$M_R = MAC(K)$
	$\overset{M_I, M_R}{\longleftrightarrow}$	
Verify M_R		Verify M_I
Accept / Reject K		Accept / Reject K

Fig. 2. A PQ authenticated key exchange protocol

Practical Considerations. During a PQ key exchange protocol, public keys need to be fragmented in multiple Ethernet packets because the maximum size of an Ethernet packet is much smaller than that of a PQ public-key. For instance, Classical McEliece needs more than 1M bytes of a public-key for the security of category 5, while maximum transmission unit (MTU) of Ethernet pack is around 1500 bytes. Note that a single MTU fits well for classical crypto schemes such as RSA or Elliptic Curve Diffie-Hellman (ECDH).

For transferring a large PQ public-key, we used the simplest method – secured file copy (SCP) over IP networks. The public key transfer could be done via a plain RCP Linux command, but the RCP port 512 is disgraced by the security policy and it is usually blocked by firewalls. So SCP is our preferable solution, although security here is not necessary since the protocol is assumed to be executed in an insecure channel. Another argument in favor of SCP is the presence of management network. Most of deployed network elements have a separate management channel over IP network. Hence, SCP can be run over this channel for an out-of-band key exchange protocol.

SCP can be configured in a password-less mode; peers generate a standard SSH key pairs and share their public keys each other. SSH pubic keys can be easily distributed between peers with the help of network management system. SCP uses the standard Linux IP stack, so it can handle a packet loss, which is very handy for general internet. SCP can be used for either out-of-band or in-band communication.

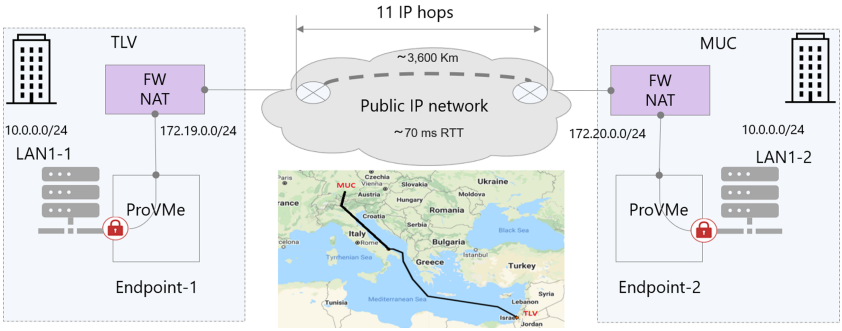

Fig. 3. A site-to-site VPN between TLV and MUC over public IP networks

For a large key transfer, we also considered using a "fragmentation-over-VXLAN" option. However, this requires more implementation efforts and the potential benefit is not so clear. Also, while SCP is suitable for every case, VXLAN-specific implementation would not work for the regular MACsec use case.

4 Experiments

In [8], a post-quantum MACSec key agreement scheme was proposed. For experiments, the authors performed a post-quantum key exchange protocol between two MACsec nodes connected back-to-back in the lab. However, this technique is not directly applicable to a site-to-site VPN where the distance is large and the bandwidth is limited. The reason is that MACsec packets (even with VLAN tag) cannot pass through the nodes in the middle which accept only IP packets. To overcome this limit, MACsec packets are encapsulated in UDP frames. Only end nodes need to decapsulate an ingress packet and retrieve a MACsec frame.

We established a VPN link between Tel Aviv (TLV) in Israel and Munich (MUC) in Germany. A overview of the link is drawn in Fig. 3. The VPN link between TLV and MUC is about 3600 km. The largest part of the path is the JONAH link which is a submarine optical cable spanning 2,300 km. There are 11 traceable IP hops on the way; ICMP ping reports a round-trip delay of around 70 ms. Assuming that a round-trip fiber propagation delay is $2 \times 3600 \times 5\,\mu s = 36$ ms, we can estimate an average delay introduced by a single IP hop: $(70-36)/2/11 = 1.54$ ms.

4.1 System Setup

We set up a FSP 150 ProVMe edge device [1] on both TLV and MUC sites. They are connected to the public internet through corresponding firewall and NAT on site. A FSP 150 ProVMe is composed of a FPGA and a Linux host. The FPGA facilitates an embedded traffic generator and a packet analyzer, allowing the

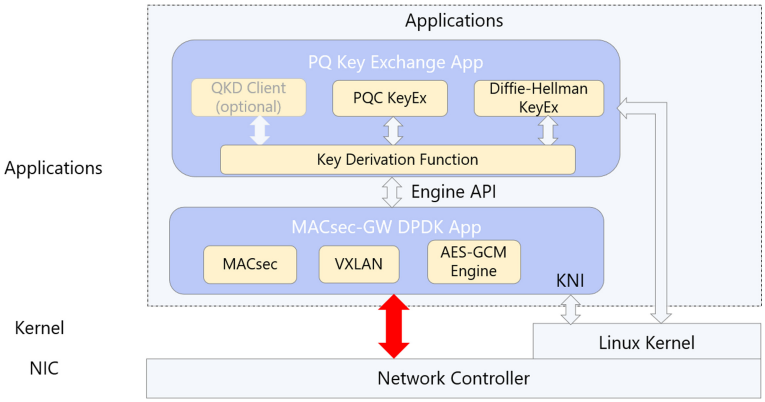

Fig. 4. A block diagram of software architecture on FSP 150 ProVMe

latency measurement with resolution of 50 ns. The Linux host runs a macsec-gw application for tunneling MACsec in VXLAN.

In our experiment, two transmission channels have been established: one is for tunneling MACsec in VXLAN and the other is for performing a PQ key exchange protocol. For this purpose, firewalls on the path must allow the following flows:

- VXLAN for MACsec traffic (UDP port 4789)
- SCP for key exchange (TCP port 22)
- ICMP for a connectivity test
- iperf3 (TCP/UDP port 5200) for a basic IP forwarding test.

Software Design. To achieve the best performance from x86 CPU, we implemented DPDK [12] with the aes-ni-gcm driver for symmetrical encryption. As a result, our macsec-gw DPDK application can process MACsec-in-VXLAN packets up to 9 Gbps on single CPU core. The measurement has been done using a packet size of 1420 bytes and AES-GCM-256 encryption. Our software also includes a DPDK KNI (Kernel NIC Interface) feature, so any application can send and receive IP packets through the port which are used for the VXLAN traffic. An overview of software architecture is given in Fig. 4.

A PQ key exchange engine is implemented in a separated application. It periodically derives a new session key and provides it to the macsec-gw application via an engine API. In fact, a PQ AKE protocol can be performed over either out-of-band or in-band channel through regular kernel interfaces or a dedicated DPDK KNI. This allows users to choose the best suitable communication path for their key exchange protocols. In our experiment, we chose an out-of-band key exchange using DPDK KNI.

Tunneling over IP Networks. Nowadays UDP is widely used for encapsulating Ethernet packets which need to travel over IP network. The reason is that

Table 3. Encapsulation options for PQ MACsec

Encapsulation methods	*Overhead (bytes)*	*Feature*
MACsec in VLAN	28	Can't pass IP network
MACsec in VXLAN	24 + 54 = 74	Stripped VLAN
MASsec over GRE in UDP	66 + 8 = 74	Kept VLAN
VXLAN over ESP in UDP	88	Stripped VLAN and duplicated UDP

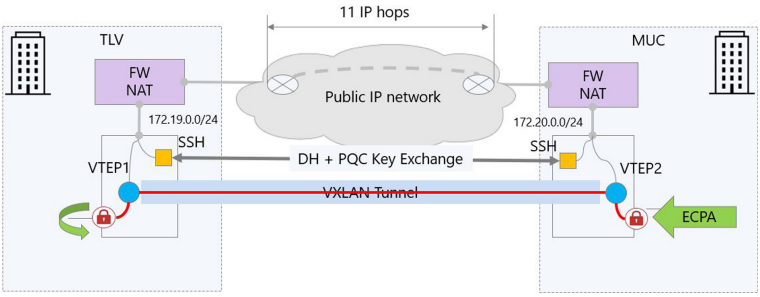

Fig. 5. Site-to-Site VPN setup using post-quantum MACsec over VXLAN

the UDP traffic can take full advantage of equal-cost multi-path (ECMP) routing. UDP is used in VXLAN as well as GRE (or NV-GRE). However, VXLAN gained some popularity over GRE because it is slightly better to be integrated with modern networks e.g. in cloud environment.

Table 3 shows several options for encapsulating MACsec packets in UDP. Among those, MACsec in VXLAN is chosen for our experiments since encapsulation can be done efficiently and the overhead size is acceptable for a limited bandwidth. Alternatively, it is possible to encapsulate VXLAN in ESP-in-UDP tunnel, but in this case the encapsulation includes two UDP headers, which is not necessarily required.

4.2 Test Results

Baseline IP Connectivity. We run an iperf3 application over Linux Kernel IP stack. The test results show that bandwidth allocations are not symmetrical, depending on their traffic directions. See Table 4 for details. It is not clear why the throughput of TLV to MUC link is much worse than that of MUC to TLV link. This is also in line with a packet loss rate which will be presented later.

Table 4. Throughput measured by an iperf3 application

Link	*iperf3 throughput*
MUC to a public iperf3 Server	574 Mbits/s
MUC to TLV	89.5 Mbits/s
TLV to MUC	6.15 Mbits/s

VXLAN Throughput. To measure a VXLAN throughput and a round-trip latency we used the following setup shown in Fig. 5. A traffic generator sends a test stream to a protected port at MUC site. Packets are encrypted by MACsec, encapsulated in VXLAN (VTEP2) and sent towards TLV site. On TLV site the VXLAN traffic is decapsulated by VTEP1, decrypted by MACsec and transmitted via a protected port. We configured a terminal loopback on the protected port, so the traffic is looped back, encrypted by MACsec, encapsulated in VXLAN and sent back to MUC site. On MUC site the decrypted traffic is transmitted via the protected port back to the traffic analyser.

VXLAN Configuration. On TLV and MUC sites the VXLAN endpoint configuration is as follows:

```
#TLV
vxlan vni 5005 src 172.19.252.107 dst 192.0.2.1
#MUC
vxlan vni 5005 src 172.20.140.8 dst 198.51.100.1
```

To allow an in-band SCP file transfer over DPDK KNI interface (vEth0), we need to add the following routes to our nodes:

```
#TLV
ip route add 192.0.2.1 via 172.19.252.1 dev vEth0
#MUC
ip route add 198.51.100.1 via 172.20.140.1 dev vEth0
```

We measured the throughput and latency of traffic with 60 s interval repeatedly. The tested packet size was either 64 or 1412 bytes. Test results show that the packet loss is different depending on the packet sizes. However, the round trip delay does not depend on packet size – average value is approximately 71 ms for either short (64 byte) packets or long (1412 byte) packets. See Table 5 for details.

Packet Loss. A packet loss over public Internet is well expected due to a long distance connection. This is not observable in the lab setup with a back-to-back connection. In Fig. 6, a packet loss in the direction of TLV → MUC link is

Table 5. Latency and packet loss using 64 and 1412 bytes packets

Item	*Long packets (1412 bytes)*	*Short packets (64 bytes)*
Min. delay	70,628 µs	69,912 µs
Max. delay	70,658 µs	78,810 µs
Avg. delay	71,170 µs	70,471 µs
Tx frames	26,470	448,236
Rx frames	26,466	448,159
Rx avg. bit rate	5,055,325 bps	5,055,209 bps
Rx frame success prob	99.98	99.98

compared with a round-trip packet loss, depending on the usage of short and long packets. We observed that the packet loss mainly occurs in the TLV → MUC direction, especially when running a test using large packets. We guess it is due to the channel characteristics but could not find a root cause of this unbalanced packet loss. Note that all IP flows were set by a default DSCP value (0) in order to make the configuration as simple as possible. Hence, no special QoS setting is done.

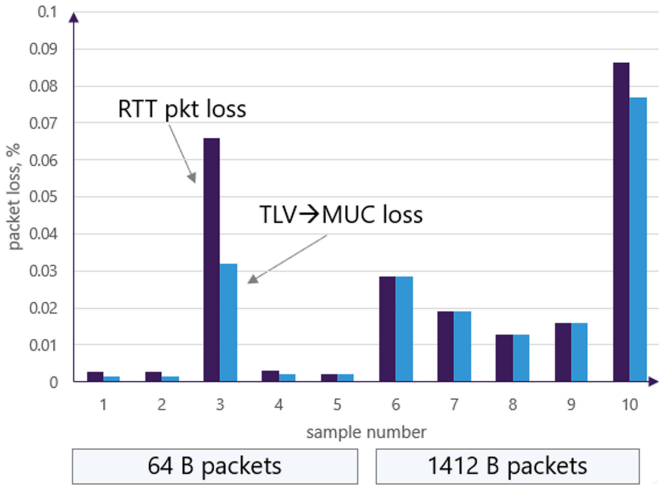

Fig. 6. Comparison of packet loss: round-trip vs. TUV-MUC link

Impact of PQ AKE Protocol. We tested a PQ key exchange protocol with multiple PQ crypto primitives listed in Table 1. Among various parameter sets, we chose those of the category 5; mceliece6960119 (Classic McEliece), ntruhps4096821 (NTRU), kyber1024 (Crystal-Kyber) and FireSaber (Saber) for

KEM, Crystal-Dilithium (Dilithium IV) and Falcon1024 (Falcon) for digital signature. The most challenging primitive is Classic McEliece KEM due to its large key size. Our experiments show that the completion of key exchange protocol takes 17 to 22 s assuming there is no background traffic. If there exists a VXLAN traffic with bidirectional 5 Gbps and a packet size of 1412 bytes, the maximum value increases to 24 s.

To evaluate the impact of the packet loss caused by a PQ key exchange protocol, we monitored Linux kernel TCP counters using *netstat* command once each key exchange is completed. On every key exchange sequence tcp re-transmission counters increased in a range from 2 to 14, regardless of the presence of a background MACsec-in-VXLAN traffic.

```
# netstat -s — grep retrans
269 segments retransmited
235 fast retransmits
```

5 Conclusion

The post-quantum crypto standard by NIST is in the final stage, leaving the final candidates only 7 primitives (4 for KEM and 3 for signature). We implemented those finalists on a commercial product and measured their impacts on the performance over a field trial link. We established a MACsec in VXLAN tunnel between TLV and MUC and performed a PQ AKE protocol over the link. Our test results show that MACsec in VXLAN using post-quantum crypto primitives can be applied to existing networks without significant impact on their performance. For instance, we did not observe any performance degradation on a bidirectional MACsec-in-VXLAN traffic with a range from 5 to 6 Mbps throughput. Therefore, we conclude that PQ MACsec in VXLAN is a practical solution for establishing a quantum-secure site-to-site VPN on existing networks.

For future work, we plan to analyze the security of a PQ AKE protocol by several attack methods, in particular, by timing attacks. Also, we plan to apply our solution to some use cases where a quantum-secure VPN needs to be deployed at a low cost.

Acknowledgment. This research is co-funded by the Federal Ministry of Education and Research of Germany under the QuaSiModO project (Grant agreement No 16KIS1051).

References

1. ADVA Optical Networking. FSP 150 ProVMe Series. https://www.adva.com/en/products/packet-edge-and-aggregation/edge-computing/fsp-150-provme-series
2. Alagic, G., et al.: Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process, July 2020

3. Barker, E., Chen, L., Davis, R.: Recommendation for key-derivation methods in key-establishment schemes. NIST Special Publication 800–56C Revision 2, August 2020. https://csrc.nist.gov/publications/detail/sp/800-56c/rev-2/final
4. Barker, W., Polk, W., Souppaya M.: Getting Ready for Post-Quantum Cryptography: Explore Challenges Associated with Adoption and Use of Post-Quantum Cryptographic Algorithms, May 2020
5. Bernstein, D., et al.: Classic McEliece: conservative code-based cryptography (2019). https://classic.mceliece.org/nist/mceliece-20190331.pdf
6. Chen, C., et al.: NTRU 2019. https://ntru.org/
7. Chen, L., et al.: Report on post-quantum cryptography, NISTIR 8105 (2016)
8. Cho, J., Sergeev, A.: Post-quantum MACsec key agreement for ethernet networks. In: Proceedings of the 15th International Conference on Availability, Reliability and Security, ARES (2020)
9. Cooper, D., Apon, D., Dang, Q., Davidson, M., Dworkin, M., Miller, C.: Recommendation for stateful hash-based signature schemes. Draft NIST Special Publication 800–208, December 2019. `NIST.SP.800-208-draft.pdf`
10. D'Anvers, J., Karmakar, A., Roy, S., Vercauteren, F.: SABER: Mod-LWR based KEM (2019). https://www.esat.kuleuven.be/cosic/pqcrypto/saber/index.html
11. Ding, J., Chen, M., Petzoldt, A., Schmidt, D., Yang., B.: Rainbow (2019)
12. DPDK: Data plane development kit. https://www.dpdk.org
13. Günther, F., Thomson, M., Wood, C.A.: Usage limits on AEAD algorithms, August 2020. https://www.ietf.org/id/draft-irtf-cfrg-aead-limits-00.txt
14. Huelsing, A., Butin, D., Gazdag, S., Rijneveld, J., Mohaisen, A.: XMSS: Extended Hash-Based Signatures. Internet-Draft draft-irtf-cfrg-xmss-hash-based-signatures-12, Internet Engineering Task Force, January 2018. Work in Progress
15. IEEE: IEEE standard for local and metropolitan area network-bridges and bridged networks. IEEE Std 802.1Q-2018 (Revision of IEEE Std 802.1Q-2014), pp. 1–1993, July 2018
16. IEEE. Local and metropolitan area networks-port-based network access control. IEEE Std 802.1X-2010 (Revision of IE EE Std 802.1X-2004), pp. 1–205, February 2010
17. IEEE: Local and metropolitan area networks-media access control (MAC) security. 802.1AE: MAC Security (MACsec). https://1.ieee802.org/security/802-1ae/
18. IEEE: Media access control (MAC) security amendment 1: Galois counter mode-advanced encryption standard- 256 (GCM-AES-256) cipher suite. 802.1AEbn-2011. https://1.ieee802.org/security/802-1aebn/
19. IEEE: Media access control (MAC) security amendment 2: Extended packet numbering. 802.1AEBW-2013. https://1.ieee802.org/security/802-1aebw/
20. KernelNewbies: 802.1AE MAC-level encryption (MACsec), Linux 4.6, May 2016
21. Liu, Y., Li, W.: VXLAN Security Option, May 2015. https://tools.ietf.org/html/draft-liu-nvo3-vxlan-security-option-01
22. Luykx, A., Paterson, K.: Limits on authenticated encryption use in TLS. www.isg.rhul.ac.uk/~kp/TLS-AEbounds.pdf
23. Lyubashevsky, V., et al.: Crystals-dilithium (2019). https://pq-crystals.org/dilithium/index.shtml
24. McGrew, D., Curcio, M., Fluhrer, S.: Leighton-Micali Hash-Based Signatures. RFC 8554, April 2019. https://rfc-editor.org/rfc/rfc8554.txt
25. National Security Agency: Ethernet security specification, version 0.5, October 2011
26. Prest, T., et al.: Falcon: Fast-Fourier lattice-based compact signatures over NTRU (2019). https://falcon-sign.info/

27. Rescorla, E.: The transport layer security (TLS) protocol version 1.3, March 2016. Internet-Draft draft-ietf-tls-tls13-12
28. Schwabe, P., et al.: Crystals-kyber (2019). https://pq-crystals.org/kyber/index.shtml
29. Steblia, D., Fluhrer, S., Gueron, S.: Hybrid key exchange in TLS 1.3, February 2020