Performance Evaluation of Quantum-resistant Open Fronthaul Communications in 5G

Ricardo Harrilal-Parchment*, Isabela Fernandez Pujol*, Kemal Akkaya*
*Department of Electrical and Computer Engineering, Florida International University, Miami, USA, 33174
Email: {rharr119, ifern093, kakkaya}@fiu.edu

***Abstract*—As 5G is offering new services that include improved security for its core functions, there is an effort to secure all domains in 5G, including control, data, and synchronization. This has turned the attention to 5G fronthaul communication security, which has not been considered crucial for past generations of cellular technologies. With overall information security efforts increasing preparation for the deployment of post-quantum cryptographic algorithms, there is also a need to assess the feasibility and overhead when such algorithms are considered for 5G Open Fronthaul communications between the radio heads in base stations and distributed units within the network. This is crucial for protocols such as eCPRI which has certain real-time requirements to meet. To this end, this paper first proposes an integrated security solution that combines IEEE 802.11AE (MACsec) along with a post-quantum-based EAP-TLS authentication within a typical ethernet-based fronthaul topology. We then implement a proof of concept to integrate all these components in a virtualized setting for the first time and evaluate the associated transmission delay with the eCPRI messages under various settings. The results demonstrate that MACsec can be a viable option that can satisfy the real-time requirements when used with post-quantum-based EAP-TLS1.3 that offers perfect forward secrecy.**

***Index Terms*—Open Fronthaul; Open RAN; CPRI; MACsec; 5G; post-quantum cryptography; EAP-TLS; transport security**

I. Introduction

As 5G comes with a new service model that changes the way core functions are managed, there is an ongoing effort to comprehensively secure all components, from the User Equipment (UE) to base stations (gNB) and stand-alone core network [1]. In addition to upgrading certain security services that were prone to attacks in 4G/LTE, there are also new security requirements that are being enforced by the 3rd Generation Partnership Project (3GPP) [2], which is the umbrella term used to indicate several standardization organizations that produce specifications for mobile telecommunications.

In line with these efforts, one of the emerging initiatives is the Open-RAN Alliance [3] which aims to standardize and create inter-operable products within the Radio Access Network (RAN). It aims to disaggregate the traditional RAN functionality and use open interface specifications between elements. As a result, this introduced new terms for the traditional links between the base stations and the core network, creating 5G Fronthaul, Midhaul, and Backhaul [4]. Our focus in this paper is *5G Open Fronthaul* which covers the portion of communication between the radio units (RUs) at the base stations and a corresponding distributed unit (DU). Data flowing in the 5G fronthaul is transported via the Enhanced Common Public Radio Interface (eCPRI) protocol defined by the CPRI Cooperation [5].

5G allows a new wave of real-time applications to be deployed, from autonomous vehicles to AR/VR applications, which require that sensitive and timely data be sent throughout the network [6]. Since all the data for these applications will pass through this fronthaul network, standard threats still apply here that can compromise the security of data communications coming from the users. In addition, the same network will be used to exchange critical control messages that are part of the O-RAN standards.

Securing the 5G fronthaul network comes with its challenges, in fact, including falling within strict timing parameters set by the communication protocol (i.e. eCPRI) and choosing a solution that will last through a post-quantum (PQ) world where existing cryptographic standards are insufficient to protect communications. Key exchange and the added security headers to every frame create additional overhead on the channel both in terms of throughput and delay, and while there are certain suggestions for securing 5G fronthaul, such as MACsec and IPsec standards, there is no comprehensive solution that integrates and tests these under an actual environment with PQ and IPv6 conditions while enabling interoperability with other existing Internet systems.

This paper offers a quantum-resistant 5G Open Fronthaul security solution with minimal overhead, thus meeting eCPRI delay requirements. This is an integration of secure data communication using MACsec at the data link layer and authentication and key agreement using Extensible Authentication Protocol (EAP)-TLS [7]. The motive behind using MACsec is to minimize the ethernet packet size to meet the CPRI deadlines while the purpose of choosing EAP-TLS1.3 is twofold: 1) Enabling the use of any type of certificates including PQ, and 2) Supporting perfect forward secrecy by re-generating a new session key upon reauthentication.

We implemented and tested this security solution in a virtualized environment using Linux containers, and performed a simulation of eCPRI traffic over this fronthaul network. The transmission delays for MACsec packets are recorded and compared to the suggested minimum delay proposed by [8] under different ethernet interface data rates. The evaluation results indicated that the EAP-TLS-based authentication and key agreement brings negligible overhead and thus does not

impact the transmission delay times for the MACsec packets even if the keys are regenerated at reauthentication time. In addition, we showed that MACsec can meet the eCPRI deadlines though slightly increasing the delay compared to the case where there is no security at all.

The rest of this paper is structured as follows: In the next section, we summarize the related work. Section III provides background on certain topics such as MACsec and eCPRI. In Section IV, we describe our integrated solution and implementation. Section V presents our experiments and analysis of the results. Finally, Section IV concludes the paper.

II. Related Work

In this section, we summarize the previous related works conducted on 5G Fronthaul security.

The work in [4] analyses security threats to Open Fronthaul communications, and discusses how mitigating these threats can bring overhead to CPRI protocol at different planes (i.e., control, user, synchronization, and management). Eventually, it analyses specific overhead that will come with MACSec in terms of packet size. There is no implementation or simulation for the performance evaluation of MACSec.

One of the initial works which quantitatively analyzed Open Fronthaul security was [9]. In this work, the authors studied the data encapsulation overhead between MACsec, IPsec, and WireGuard, and concluded that while overhead alone may not be the best indicator for choosing a security protocol, MACsec has the least overhead of all three protocols analyzed. However, their implementation does not exactly follow MACSec specifications, as they just measured the latency of encrypted traffic on an optical link. The work does not consider PQ, IPv6, and authentication as part of the analysis.

In [10], the same authors explored MACsec under a quantum security threat model. They argue that MACSec key agreement with EAP-TLS will not be efficient for meeting the strict timing requirements, and propose a peer-to-peer key agreement between nodes that support PQ-based signatures. They then tested this approach by establishing a MACsec connection between two FPGAs using Linux and performed two key exchanges, both on the application running in the host machine. Their data showed that the average latency for 64-byte packets was 34 µs under 2.3Gbps, and this increased to approximately 115 µs for a packet of 1420 bytes under 9Gbps. However, their setup requires FPGAs to be able to cope with AES-256 encryption. In addition, they did not measure EAP-TLS overhead to compare with their approach. Contrary to this work, we argue that EAP-TLS1.3 with PQ certificates can be used without violating the timing requirements of eCPRI. In addition, we also consider IPv6 and performed testing under commercial Ethernet data rates.

III. Background

A. Open Fronthaul and its Security

Based on the Open-RAN (O-RAN) standard, Open Fronthaul is referred to as the communication infrastructure between radio units (RUs) on the base stations and distributed units (DUs) as shown in Fig. 1. It delivers data on the user (IQ sample data), control (real-time control), and management planes (non-real-time management of network operations).

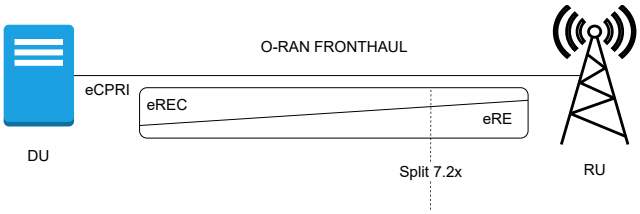

Fig. 1. O-RAN Fronthaul Architecture

Inside the control plane, RUs and DUs coordinate and schedule messages. Although no user data is sent through this channel, control messages must be secured to avoid tampering. Countless attacks can originate between an RU and DU based on the man-in-the-middle (MiTM) threat model. These include injecting, changing, and stopping messages at both the control and user planes, as well as eavesdropping. Securing the channel can prevent these illegitimate control messages from being exchanged between DUs and RUs, thus protecting system integrity [4]. User data is sent through the user plane inside the fronthaul, which also faces constant security threats. Several attacks can be carried out inside the user plane, including wiretapping and driving traffic out of a network into a rogue base station posing as legitimate equipment [4]. Preventative measures must be in place to ensure the authenticity of both DUs and RUs, ensuring that each device exchanges messages with a legitimate peer. Per-packet integrity checks are also essential to ensure that each message arrives at its destination unaltered.

B. CPRI/eCPRI

The Fronthaul communication between remote radio heads (RRHs) and a baseband unit (BBU) is made possible by the CPRI protocol, which carries messages for both control and data planes, as well as means for synchronization. CPRI is able to translate radio signals into computing functions and carry these from RRHs to a BBU. Because of the increase in traffic in 5G networks, the CPRI specification was enhanced into eCPRI as a way to split the functions inside the BBU for load balancing. Within one eNB/gNB, eCPRI further divides the functional components into two notes: eREC (Radio Equipment Control) and eRE (Radio Equipment). The most common split is the 7.2x: Low PHY/High PHY split.2x. This split is created between the PHY component inside the eNB/gNB, and is advantageous to some extent, though it requires a fronthaul network with higher capacity and lower latency. Components of eCPRI require very accurate timing, to ensure that data throughput at the UE can be maintained when switching between RUs [5]. In the intra-PHY split, user data requires $\sim$20 Gbps of bandwidth and 3/1.5 Gbps of throughput in [5]. The 4-byte eCPRI header contains 5 fields: eCPRI protocol revision, reserved section, C bit, eCPRI message type, and eCPRI payload size. The 4-bit eCPRI protocol revision field indicates the current eCPRI revision and the next three bits are reserved, being vendor specific [5]. The C bit details whether

the current message is the last inside the PDU, and the 8-bit eCPRI message type field specifies which type of service carried the message. Lastly, the 16-bit eCPRI payload size specifies the payload size, which has a maximum size of $2^{(16)} - 1$ bits.

C. MACsec

MACsec (aka IEEE 802.11AE) offers security at the link layer which is established either through manual configuration or dynamically with verification and key exchange between both devices. A MACsec frame is based on the standard ethernet format plus an 8-16 byte MACsec Security Tag (SecTag) and an 8-16 byte Integrity Check Value (ICV) as shown in Fig. 2. When using the GCM-AES-256 cipher suite, both of these fields are at their maximum length.

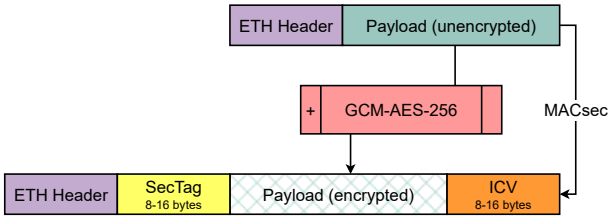

Fig. 2. MACsec Ethernet frame

There is a companion protocol, MACsec Key Agreement (MKA) which can facilitate dynamically establishing keys for MACsec and allows the dynamic setup of a MACsec channel to accompany certain network access control methods within a segment. MKA works assuming that peers on the same LAN are part of the same Connectivity Association (CA), within which the Connectivity Association Key (CAK) is the long-term key used as the base for every interaction within MACsec. Different encryption keys can be derived from the CAK, such as the Secure Association Key (SAK), for securing traffic within the data plane. The details of this protocol and process were originally defined in the IEEE 802.1X-2010 standard, which has since been superseded by [7]. MACsec can be set up using Static CAK Mode or Dynamic CAK Mode. In Static CAK mode, MACsec is enabled through a pre-shared key. In Dynamic CAK mode, the Master Session Key (MSK) is enabled through distributed authentication protocols such as Extensible Authentication Protocol (EAP)-TLS [11] which is widely used today for securing network connections. In this setup, there is a separate authentication server called RADIUS which verifies clients' identities and can generate key material through various methods.

IV. PQ-based Open Fronthaul Security

This section presents the problem, motivation, and proposed solution.

A. Motivation and Overview

The 5G Fronthaul network supports the RAN architecture by connecting RRHs to a centralized BBU. This network is logically separated by eCPRI which effectively balances the load between radio equipment (RE) and radio equipment control (REC). However, as previously mentioned, this communication needs to be secured against various attacks. As such, the communication will need to support encryption, integrity, and authentication at the least. In addition, given the upcoming post-quantum era, these security services should be quantum-resistant. Finally, it is desirable that these services can be integrated with existing standards on the web to be interoperable and support the latest security requirements.

On top of these security needs, there are also application-specific restrictions. For instance, for the fronthaul to communicate messages effectively, it must meet the requirements imposed by the architecture of eCPRI. The deadline in eCPRI is imposed by the Hybrid Automatic Repeat reQuest (HARQ) loop/protocol in the vendor-specific eREC, which works at the link layer and is in charge of forward error detection and correction [12]. From eCPRI v2.0, the ACK/NACK for one sub-frame must arrive within the next three sub-frames, where one sub-frame is equivalent to one Transmission Time Interval (TTI). Looking back at 4G LTE, the TTI was 1ms, whereas, in 5G-New Radio (NR), this number is scalable between 62.5μs to 1ms [13]. To calculate the minimum roundtrip delay left for messages, one needs to consider the BBU processing time. According to [14], the reported processing time for a BBU is $\sim$2.75μs, leaving 246μs for transmission/response time. Cutting this in half for one-way transmission leaves a maximum delay of 123μs each way to meet the deadline, and the IEEE P802.1CM standard for local and metropolitan Fronthaul networking [8] restricts this even further, to 100μs.

Therefore, in this paper, we offer a PQ-resistant Open Fronthaul security solution that comes with minimal overhead and meets the eCPRI delay requirements. Our solution utilizes MACSec at the data link layer and integrates it with EAP-TLS1.3 to be able to generate per-session symmetric keys while enabling the use of PQ certificates for authentication purposes. EAP requires an authentication server, which although is not typically a core component of 5G fronthaul networks, is fairly lightweight and can easily be added to a fronthaul topology that based on an Ethernet network (Fig. 3).

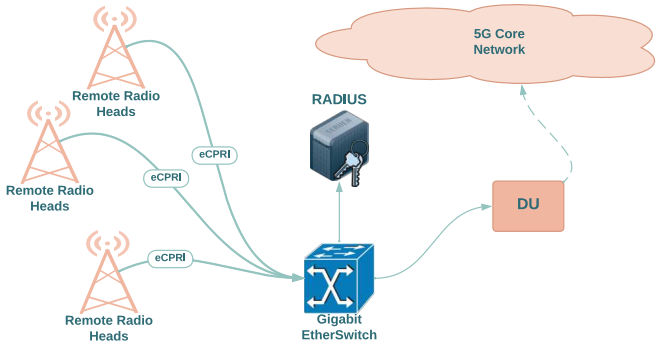

Fig. 3. 5G Fronthaul Network with a RADIUS Authentication Server.

B. PQ-based Authentication

The 5G core network was envisioned as cloud-native functions that communicate with each other using software-defined

networking (SDN) technology. In SDN-based networks, virtualized functions interact with each other through APIs. Given this change, the 5G core network became a public key infrastructure (PKI)-based authentication and authorization ecosystem. In this security system, the virtual functions each use a public/private key pair, as well as a certificate for authentication and authorization [15]. Given the nature of this system, it is important that it remains functional in a post-quantum world. In a post-quantum world, Shor's algorithm would render the security of PKI which relies on standards like RSA useless [16]. This is because it enables the factoring of large products of prime numbers extremely efficiently, which is the basis of the relationship between public and private keys in RSA. Therefore, we propose using IEEE 802.1X EAP-TLS framework that can accommodate PQ certificates. The authentication is based on a certificate authority that can generate any PQ certificates such as Falcon and Dilithium [17] for the involved DUs or RUs, as well as the RADIUS server to authenticate connecting nodes.

In addition to authenticating the parties, EAP-TLS can provide keying material for securing data communication, which can be used for confidentiality and integrity purposes. Using this keying material, symmetric keys can be generated for standard encryption such as AES, which to be quantum-secure, should use keys of at least 256 bits, and will add more information to the data packets. These keys can be derived pairwise between RU and DU, enabling the establishment of a secure data channel for data message communications as will be described in the following subsection. To provide perfect forward secrecy, we propose using the latest TLS1.3 standard which can facilitate update of the keys for each session by ensuring the availability of fresh key material [18]. EAP-TLS 1.2 would not provide this forward secrecy. To the best of our knowledge, this is the first comprehensive setup for enabling PQ-based authentication and key agreement for Open Fronthaul communication security in 5G.

C. MACsec-based Data Channels

Among the existing alternative standards, we chose MACsec for securing our communications since it has the minimal overhead of the security protocols being considered. However, in a 5G Fronthaul use case, the static configuration is burdensome to scale and administer. We therefore leverage EAP-TLS with MKA to implement MACsec dynamically in this environment. To enable this, authenticators (i.e., our ethernet switch) and supplicants (i.e., our BUs or DUs) are also set with a MACsec policy, enforcing the use of MKA after successful authentication.

From a MACsec-specific perspective, the root key is the CAK, from which the confidentiality and integrity keys are derived. A CAK can be obtained in several ways, one of which is through mutual authentication and key derivation via the IEEE 802.1X EAP [7]. EAP-TLS 1.2 accomplishes this with the use of the *EAP-SessionID* parameter, which is linked to the *EAP-Key-Name RADIUS* attribute. However, in EAP-TLS 1.3, this parameter is deprecated in favor of ephemeral "key_share" extensions in the TLS $ClientHello$ and $ServerHello$ messages. Once the authentication procedure has been completed, both sides of the connection will have derived the appropriate symmetric AES keys. Note that these MACsec keys are derived among a DU/BU and Ethernet switch for each individual link separately.

D. Implementation and Integration

To be able to realistically evaluate the performance of the proposed secure communication Fronthaul, we implemented a virtual prototype. Before we show the conducted experiments, we present the implementation details as this is important to understand the behavior of the system.

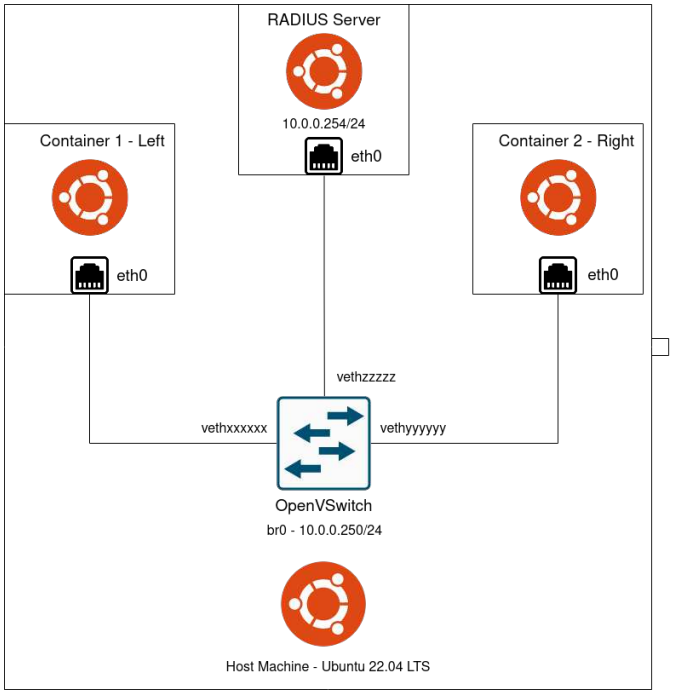

Fig. 4. Proof of Concept Fronthaul Virtual Network Topology.

We used a virtualization-based environment as seen in Fig. 4. The left Linux container is used as the source node (i.e. an RU), while the right node is the sink node (i.e. a DU). We introduced a virtual switch (OpenVSwitch) which not only connects the RUs and DUs but also the RADIUS Server for authentication purposes.

To enable integration with DUs and RUs, we did many configurations at the RADIUS Server as follows:

- The CA was configured & established at the site of the RADIUS server.
- The server certificate was generated for the RADIUS server.
- The clients' certificates were generated and transferred to the clients, along with a copy of the CA certificate for verification.
- On the RADIUS server, *radiusd* is configured for EAP-TLS authentication, with the version set to TLS 1.3.
- On the network authenticator (i.e., OpenVSwitch), *hostapd* is configured to use EAP-TLS with the RA-

DIUS server as a backend and to follow up successful authentication requests with MKA.
- On the supplicant (i.e., Left and Right containers), *wpasupplicant* is configured for EAP-TLS authentication, and to enforce MACsec for successfully authenticated links.

The MACsec channel was configured manually using a pair of shell scripts between the Left and Right containers, using static IPv4 and IPv6 addresses and the GCM-AES-256 cipher suite for quantum-resistant security.

V. Experimental Evaluation

In this section, we present the conducted experiments to test speed and latency requirements.

A. Setup and Metrics

We used the setup in Fig. 4 for conducting our experiments. Average request/response time was initially measured for standard Ethernet interface speeds from 1 Gbps to 100 Gbps. However, we had strong indicators that our hardware could not perform at 100 Gbps, and our results were wildly inconsistent when set to this speed, so we reduced this upper limit to 40 Gpbs.

As the performance metric, we measured the total transmission time for a packet to verify if it satisfied the deadline set by the eCPRI protocol. We tested both a cleartext channel and a MACsec channel, each with IPv4 and IPv6.

Other details of the setup before collecting the results are given below:

1) All three processes, *radiusd*, *hostapd*, and *wpasupplicant* were started sequentially and the authentication process was observed.
2) At the supplicant site, Wireshark was used to monitor the authentication process.
3) Using a pair of Python scripts, a stream of 1,000,000 IP (initially IPv4) packets were generated and sent between nodes, each with a payload of 1420 bytes.
4) Traffic was transmitted using UDP as the upper layer transport protocol, as CPRI/eCPRI uses UDP transport.
5) The channel was monitored using Wireshark on the host machine, as this enforced eliminated any potential clock skew.
6) The time delta between the first packet leaving the source node and the last packet arriving at the sink node was recorded.
7) An average per-packet delay was obtained by dividing the time delta by the number of packets sent.
8) Steps 3-7 were repeated for various interface speeds.
9) Steps 3-8 were repeated for a MACsec channel.
10) Steps 3-9 were repeated for IPv6.

B. Results Analysis

1) MACsec data latency performance: We conducted experiments to measure the latency values for MACsec data packets and compared them under different gigabit interface speeds and IP versions. The results are shown in Fig. 5 and 6 for IPv4 and IPv6 respectively.

As can be seen, while MACSec delays are fairly constant, cleartext results reduce until reaching a certain bandwidth, after which the curve flattens. Based on the values collected, we observed that there is a logarithmic relationship between bandwidth and delay in this scenario, with the delay approaching some minimum/floor value around 4-5 μs for cleartext, versus 17-19μs with MACsec enabled. From our results, the difference in delay between MACsec and cleartext maxed out at around 13-15μs at higher bandwidths, with negligible differences based on the IP protocol version.

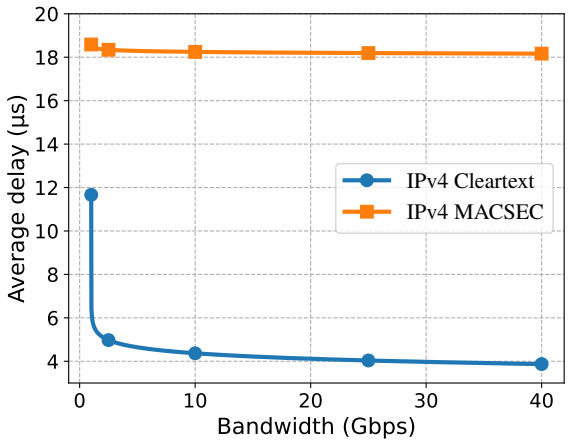

Fig. 5. IPv4 Delay

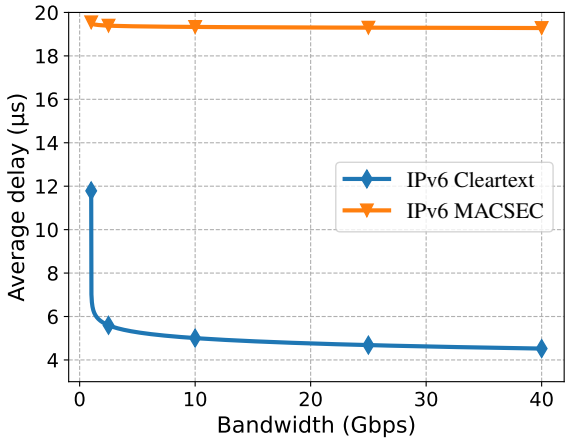

Fig. 6. IPv6 Delay

Comparing the latency between a cleartext channel and a MACsec channel, we can conclude that adding encryption and integrity protection adds some overhead. However, the results suggest that for higher bandwidths, the additional delay incurred maxes out at around 13-15µs even for IPv6. As our scenario was virtualized, the additional delay provided by the physical communication medium was not present, however, there are a number of articles that discuss this, such as [19], which reports the additional delay imposed by optical fiber at approximately 5μs/km. Using 100μs as the maximum delay allowable, the maximum distance between RU and DU would

be around 20km, taking into account no other considerations. Given that there is a base delay of around $\sim 4\mu$s for cleartext communications, this maximum is closer to 19km. With the delay per km of single-mode fiber being 5μs/km, adding this extra 15μs of delay overhead due to MACsec would reduce the maximum possible distance between DU and RU by around 3km in order to meet the deadline (i.e. 16km). Thus, the burden incurred by this security protocol to 5G infrastructure will be a reduction in the maximum transmissible distance between DU and RU of around 3 km, and the associated infrastructure and equipment costs required to compensate for this.

2) Impact of Authentication on the MACsec: In this experiment, we assessed whether the process of authentication and use of EAP-TLS1.3 with periodic re-keying would have any impact on the MACsec data latency as compared to having no re-keying (i.e. TLS1.2). Interestingly, we observed that the time taken for this authentication is negligible. This is because, during the authentication, a MACSEC channel has either not yet been established, or is already established and communicating. Based on how the MKA exchange works, the keys used to secure the channel are generated by some local cryptographic calculations on the device that happens simultaneously with the data transmission. Thus, regardless of the re-keying frequency or the types of certificates (i.e., RSA, Dilithium, or Falcon), there is no significant impact on the data latency performance. This is in part thanks to the existence of a separate RADIUS server and an authenticator sitting on the ethernet switch, to which authentication and key generation are offloaded, so that DUs/RUs keep using the existing AES keys at the link layer. As opposed to the approach in [10], these results suggest that MACsec can be a viable option for a PQ-based solution in 5G fronthaul.

VI. Conclusion

In this paper, through our proposed security setup for 5G Open Fronthaul, we analyzed the effect MACsec has on the latency for eCPRI packets sent in the fronthaul network, as well as explored the dynamic establishment of MACsec channels with EAP-TLS. Our results show that EAP-TLS1.3 authentication and re-keying will not significantly disrupt ongoing communications, meaning TLS with post-quantum cryptography using NIST-approved algorithms [20] such as CRYSTALS-Dilithium, Falcon, and so on can be supported without any concern. Forward secrecy would be achieved using TLS 1.3 in the underlying EAP session, as this facilitates periodic regeneration of key material for the MACsec channel [18]. Testing a range of interface speeds and IP versions, we conclude that MACsec's additional delay on packets maxes out at around 13-15 µs, which is $\sim$15% of the allowable delay according to IEEE P802.1CM standards.

These results are important as we look to the future when IPv6 and PQ become more standardized, ubiquitous, and robust. We plan to conduct more large-scale testing with more complex topologies.

Acknowledgement

This work is supported by US National Science Foundation (NSF) under the grant #2147196.

References

[1] I. Ahmad, S. Shahabuddin, T. Kumar, J. Okwuibe, A. Gurtov, and M. Ylianttila, "Security for 5g and beyond," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3682–3722, 2019.

[2] "3rd generation partnership project," Dec. 2022. [Online]. Available: https://www.3gpp.org

[3] "Open ran alliance specifications," Dec. 2022. [Online]. Available: https://www.o-ran.org/specifications

[4] D. Dik and M. S. Berger, "Transport security considerations for the open-ran fronthaul," in *2021 IEEE 4th 5G World Forum (5GWF)*, 2021, pp. 253–258.

[5] "eCPRI Specification V2.0," http://www.cpri.info, Standard, May 2019.

[6] O. O. Erunkulu, A. M. Zungeru, C. K. Lebekwe, M. Mosalaosi, and J. M. Chuma, "5g mobile communication applications: A survey and comparison of use cases," *IEEE Access*, vol. 9, pp. 97 251–97 295, 2021.

[7] "Ieee standard for local and metropolitan area networks–port-based network access control," *IEEE Std 802.1X-2020 (Revision of IEEE Std 802.1X-2010 Incorporating IEEE Std 802.1Xbx-2014 and IEEE Std 802.1Xck-2018)*, pp. 1–289, 2020.

[8] "Ieee standard for local and metropolitan area networks – time-sensitive networking for fronthaul," *IEEE Std 802.1CM-2018*, pp. 1–62, 2018.

[9] J. Y. Cho, A. Sergeev, and J. Zou, "Securing ethernet-based optical fronthaul for 5g network," in *Proceedings of the 14th International Conference on Availability, Reliability and Security*, ser. ARES '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3339252.3341484

[10] J. Y. Cho and A. Sergeev, "Post-quantum macsec in ethernet networks," *Journal of Cyber Security and Mobility*, pp. 161–176, 2021.

[11] B. Weis, "Overview of ieee 802.1 x-rev dynamic session key agreement," *Teaches the IEEE standard of the MACsec Key Agreement*, 2006.

[12] S. Khalili and O. Simeone, "Uplink harq for distributed and cloud ran via separation of control and data planes," 2015. [Online]. Available: https://arxiv.org/abs/1508.06570

[13] A. Padmanabhan, "5g nr transmission time interval," Available at https://devopedia.org/5g-nr-transmission-time-interval (2022/2/15), 2022.

[14] H. J. Son and S. Shin, "Fronthaul size: Calculation of maximum distance between rrh and bbu," Available at https://www.netmanias.com/en/post/blog/6276/c-ranfronthaul-lte/fronthaul-size-calculation-of-maximum-distance-between-rrhand-bbu. (2014/04/1).

[15] T. C. Clancy, R. W. McGwier, and L. Chen, "Tutorial: Post-quantum cryptography and 5g security," in *WiSec'19: ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2019.

[16] F. de Lima Marquezino, R. Portugal, and C. Lavor, "Shor's algorithm for integer factorization," in *A primer on quantum computing*. Springer, 2019, pp. 57–77.

[17] National Institute of Standards & Technology, "Post-quantum cryptography," Jul. 22, 2022. [Online]. Available: https://csrc.nist.gov/Projects/post-quantum-cryptography

[18] J. P. M. . M. Sethi, "Eap-tls 1.3: Using the extensible authentication protocol with tls 1.3," Internet Requests for Comments, RFC Editor, RFC 9190, 02 2022. [Online]. Available: https://www.rfc-editor.org/rfc/rfc9190.txt

[19] J. Coffey, "Latency in optical fiber systems," Commscope, Tech. Rep. [Online]. Available: https://www.commscope.com/globalassets/digizuite/2799-latency-in-optical-fiber-systems-wp-111432-en.pdf?r=1

[20] National Institute of Standards & Technology, "Nist announces first four quantum-resistant cryptographic algorithms," Jul. 5, 2022. [Online]. Available: https://www.nist.gov/news-events/news/2022/07/nist-announces-first-four-quantum-resistant-cryptographic-algorithms