Monitoring Strategy for Enhanced Adaptability in QKD Networks

Blanca Lopez [1,2,*], Angela Diaz-Bricio [1,2], Ivan Vidal [2], Francisco Valera [2], Diego R. Lopez [3]
[1]IMDEA Networks Institute, Avda. del Mar Mediterráneo, 22, 28918 Leganés, Madrid, Spain
[2]Telematic Engineering Department, Universidad Carlos III de Madrid, Avda. Universidad, 30, 28911 Leganés, Madrid, Spain
[3]Telefónica, Ronda de la Comunicación, Fuencarral-El Pardo, 28050 Madrid, Spain
[*]Correspondence: blanca.lopez@imdea.org (B.L.)

ABSTRACT

Monitoring mechanisms are essential in modern telecommunications networks, ensuring reliability, performance, and security by continuously tracking network health and status. These mechanisms enable real-time anomaly detection and response, allowing for fast intervention and dynamic optimization of network resources. To advance Quantum Key Distribution (QKD) networks toward becoming a mainstream service within current telecommunication infrastructures, comprehensive monitoring mechanisms must be available. This paper explores the integration of such a monitoring approach for QKD. Leveraging a recently proposed model that utilizes virtualization to create an abstraction layer over the highly heterogeneous array of quantum and classical hardware, this monitoring mechanism offers significant advantages. These include the continuous provisioning of real-time data and traffic statistics on network segments, as well as the detection of potential attacks or failures on critical network elements, enabling dynamic, data-driven routing, operations, and management. The proposed monitoring strategy, which incorporates both passive and active measurements, enables proactive and immediate actions when any network irregularity is detected. Beyond detailing the monitoring mechanism, we demonstrate its feasibility through a virtual environment test, validating its effectiveness in a controlled setting.

1 INTRODUCTION

Quantum Key Distribution (QKD) currently stands as the only known strategy for guaranteeing the absolute security of information exchanges within telecommunication networks. Quantum Mechanics principles ensure that the keys exchanged using QKD are intrinsically secure, safeguarding communication even against adversaries equipped with quantum computing capabilities. However, designing, deploying, and maintaining a QKD network that offers these cryptographic services is not a trivial task. Much like the early days of communication services, quantum network deployments remain heavily reliant on hardware and its physical features. This implies a serious lack of flexibility and scalability, significantly limiting their potential for widespread adoption, keeping them mostly in the realm of experimental technology or partial setups rather than being a viable solution for full-scale deployment.

In order to make quantum cryptography a real service included in the infrastructure of our networks, advancements in quantum technology are essential. Equally important, however, is the need to design QKD

networks based on the same principles that guide modern communication networks, ensuring they are interoperable, scalable, and practical for real-world deployment.

Along these lines, several documents and standards have been developed advocating for the integration of well-established technologies like Software-Defined Networking (SDN) into the architecture of QKD networks [1–3], as well as implementations in network infrastructures [4, 5]. Softwarization allows the abstraction of certain network functions from the hardware that underlies the network. Through softwarization, network resources and services can be configured, managed, and easily updated, without requiring extensive hardware modifications [6]. SDN, often considered the main implementation of the softwarization concept, offers enhanced flexibility and reconfigurability, which brings QKD one step closer to becoming a real service integrated into modern telecommunication network infrastructures.

Another major trend in modern telecommunications networks is virtualization. This strategy allows the creation of virtual instances of hardware devices, making it possible, for example, for different systems or applications to run independently on the same hardware. A key technology in this trend is Network Function Virtualization (NFV), which specifically focuses on decoupling network functions, such as firewalls or routers, from dedicated hardware and running them as virtualized services [6]. This improves scalability and flexibility, as virtual machines or virtualization containers can be created and removed on demand, enabling dynamic strategies. Additionally, NFV optimizes resource utilization across the infrastructure and enhances network resilience by allowing easy creation of backup copies to recover from partial hardware failures.

Recently, we proposed a model that virtualizes key management functions of QKD networks [7]. This approach moves away from the monolithic design of current QKD hardware by introducing an abstraction layer on top of quantum devices, allowing the integration of functionalities already well-established in classical networks to its quantum counterparts. The model paves the way for transforming QKD networks to support more flexible and highly scalable architectures. This shift towards a more softwarized conception of QKD networks brings multiple benefits, including the possibility to perform data-driven control and management, customized access and admission mechanisms, or real-time performance monitoring.

Monitoring, in particular, plays a crucial role in modern telecommunications networks, and its inclusion in this model represents a significant step forward. Effective monitoring systems are essential for ensuring network reliability, performance, and security. By continuously tracking the health and status of the network, such mechanisms enable real-time detection of anomalies or inefficiencies, allowing for fast intervention and dynamic optimization. Combined with virtualization, these monitoring strategies greatly improve adaptability and responsiveness.

In this paper, we rely on the properties of the proposed model, presenting a novel general monitoring mechanism suitable to be incorporated into the model and able to take advantage of the softwarization

and virtualization approach it follows. Due to the inherent flexibility of the model, this monitoring mechanism does not need to be fixed or permanent. Instead, it can be updated and reprogrammed to meet specific needs at any given moment.

This paper continues as follows. Section 2 introduces the general model, presenting the different layers and components, and analyzes its convergence with current standards. Section 3 delves into the proposed monitoring strategy, while Section 4 introduces a proof of concept of the mechanisms described previously. Lastly, Section 5 gives an overview of the conclusions of the work and presents some future research lines.

2 GENERAL MODEL

If quantum cryptography is to be a service embedded in our telecommunication infrastructure, QKD networks have to be adaptable to changing demands, flexible, and easy to update, to facilitate the integration of its services into the current telecommunication networks. Building on the experience gained from the evolution of classical networks, adopting SDN and NFV strategies appears to be the logical next step forward.

The main idea behind the general model considered in this paper is the softwarization and virtualization of the key management functions that are usually enclosed in QKD devices. The indivisible design of these devices leads to severe limitations in building functional and scalable QKD networks. By separating the key management functions from most of their physical constraints, while preserving common protocols to communicate with QKD devices, QKD networks gain the agility and flexibility to adapt to future developments and standards, without the need for a replacement or overhaul of the hardware that supports them.

In particular, this shift towards a more structured architecture enables the possibility of developing a more complex control plane, which allows the addition of some innovative elements, like access and admission control components, or a monitoring element [7].

2.1 Model components

The model, as can be seen in Fig. 1, is divided into three planes. The underlying element consists of the physical infrastructure that sustains the network. This includes not only the QKD devices but also the optical fibers, switches, and other essential hardware components, such as server computers. Each QKD module must have a component that implements a standardized protocol, as those described in the next subsection, for receiving and responding to requests for cryptographic material in what appears in the Figure as KMA, which stands for Key Manager Agent.

Above this hardware layer, supported by network servers, we find the key management plane. This plane is responsible for all the key management functionalities, that are originally bound to the QKD devices provided by manufacturers. Effective key management includes distribution, potential storage of crypto-

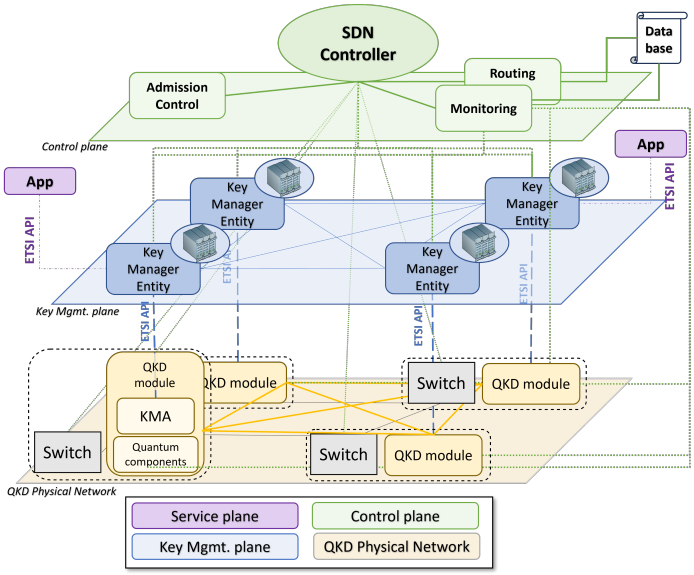

Figure 1: Outline of the general model design.

graphic keys, and managing key expiration and renewal processes, among others. The softwarization of these functions creates a very suitable environment for customization and innovation at this plane, which has been highly hardware-dependent until now. The Key Manager Entities (KMEs) in this plane handle the requests that come from the applications of the network users, acting as intermediates between them and the quantum hardware layer. Communication regarding key requests can be carried out following either standardized or proprietary protocols, thanks to the flexibility provided by this solution.

The uppermost layer is the control plane, where network administrators and operators can implement a wide range of components to ensure the correct operation of the network and its services. Fig. 1 illustrates four key components as examples. The central element, the SDN controller, is responsible for overseeing general configuration, signaling, and dynamic network management, ensuring that the network adapts to real-time demands. Additionally, the admission control component manages resource allocation based on current capacity and predefined network policies, optimizing the distribution of network resources. The original model also identifies a monitoring module, but only sketches its functionality. The complete design of such a monitoring functionality is the main goal of this paper. This component plays a crucial role by continuously tracking the performance of the network in real time. The monitoring element provides critical insights into network status, allowing for the efficient utilization of resources and infrastructure. It gathers information on the operational status of the network and its individual links, storing this data in a database that is accessible to other control plane elements, such as the dedicated routing module that can be seen in the figure, as well as to network administrators. The details of this monitoring mechanism are discussed in later sections.

2.2 Convergence with standards

The rapid advancement of quantum technologies, coupled with confidence in their integration into everyday life, has intensified the need for the development of standards. These standards are crucial to facilitate the early adoption and seamless integration of quantum technologies across industries, ensuring compatibility, security, and scalability on a large scale. Globally recognized organizations, such as ITU-T, ETSI, and the IETF have been engaged for some time in the creation and dissemination of documents and standards that are beginning to define the path forward. These documents address a wide range of topics related to quantum networks and to QKD networks in particular, from functional requirements to machine learning enhancement, including architecture models and application interfaces.

Both ETSI and ITU-T documents follow a broadly similar model for QKD networks, though they differ in the names of certain components and some specific functionalities. From ETSI, the work that has the greatest implications for our research is the standards for the Application Programming Interfaces (APIs) between the various network components. In particular, two standards have already been published that define how applications should communicate with QKD devices [8, 9]. ETSI 014 can be considered a simplified version of the more general ETSI 004. A document describing the communication between key manager systems, especially for devices at the border of two different domains, is also under construction.

In these documents, ETSI adopts a model in which QKD networks are organized into sites. Each site contains applications that require cryptographic material and QKD modules that provide it. When multiple QKD modules are present within a single site, an additional element, known as the QKD Key Server, may be introduced to manage the keys from all QKD modules within the site, as can be seen in Fig. 2.

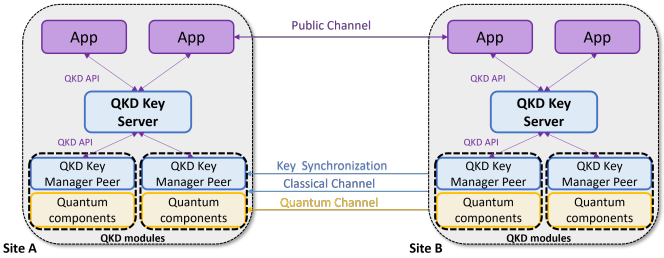

Figure 2: Two-site QKD network model from ETSI [8].

Additionally, ETSI has published several standards addressing the integration of SDN technologies for certain network control functions [1, 2]. In this context, an SDN agent is introduced to each site, capable of communicating with the network controller and the QKD modules. This agent reports status and configures the modules based on instructions from the network controller

ITU-T, in contrast, outlines a comprehensive functional architecture for QKD networks, divided into six distinct layers: the service layer, control layer, management layer, user network management layer, key management layer, and quantum layer, each serving a specific function within the network [10]. Additionally,

ITU-T introduces two architectural modelsone with a centralized network controller and another with a distributed controller, offering flexibility in network management based on specific deployment needs. An example of the centralized model architecture can be found in Fig. 3.

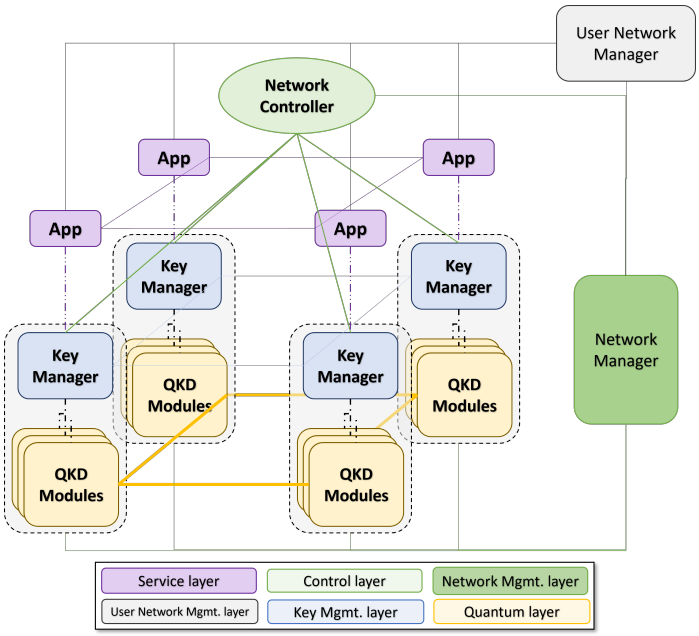

Figure 3: Centralized model functional architecture from ITU-T [10].

Building upon this layered architecture, further standards have been developed to incorporate SDN strategies, which enhance the control layer by streamlining processes and optimizing the management of network resources.

Finally, the IRTF, the research branch of the IETF, has created a dedicated group to explore challenges related to the development of the Quantum Internet. Within this group, there are already several documents published and in preparation that discuss, among other things, an architecture for the Quantum Internet divided into several planes [11], based on lessons learned from QKD deployments. Unlike the more detailed standards previously discussed, this proposal takes a broader approach, focusing on the overall functionalities of the Quantum Internet planes rather than delving into the specifics of individual components or processes. In the context of QKD networks, the document considers an architecture comprising three essential planes: the quantum forwarding plane, responsible for the transmission of quantum information, the services plane, which provides the necessary applications and security services, and the control plane, which manages the overall network operation and resource allocation.

As can be seen by comparing the figures, which follow a consistent color code for similar planes, there is a clear alignment between the proposed model and the architectures defined by ETSI and ITU-T. Additionally, it is easy to identify parallels with the architecture put forward by the IRTF. The proposed model adheres closely to these standards, yet extends the boundaries of current frameworks by leveraging NFV and SDN strategies and offering new possibilities.

3 MONITORING MECHANISM

3.1 Passive and active monitoring

Monitoring is an essential component of modern telecommunications networks, playing a critical role in maintaining optimal performance, reliability, and security. A network monitoring system tracks the health and functionality of the network, detecting potential issues such as performance bottlenecks, disruptions, or security vulnerabilities. When anomalies or faults are identified, the system issues the corresponding alerts, informing network administrators, and probably triggering the action of automated control mechanisms to address the anomaly. This way, proper monitoring translates into fast and accurate intervention, before the detected issues escalate into larger problems.

Network monitoring mechanisms can be broadly categorized into two groups: passive and active methods. Each of them serves different network monitoring objectives [12].

Passive monitoring methods analyze network traffic exclusively through observation. This means that they capture data as it flows through the network, without introducing any additional traffic or network disturbance. The major advantage of passive measurements is that they avoid generating additional network overhead, providing information about natural network performance without interfering with actual user traffic. Using these methods, network administrators obtain insights about the patterns, bandwidth usage, latency, and error rates based solely on the clients usage of the network. However, these strategies themselves do not guarantee that all the information needed to ensure optimal and secure network operation will be obtained. At times (or links) in which the traffic exchanged on the network is lower, passive measures may be insufficient to observe potential network failures, making it virtually impossible for administrators to anticipate and fix a potential network defect before traffic increases.

On the other hand, active measures introduce additional traffic probes into the network, to monitor and asses network performance. Analyzing the behavior of these packets, active monitoring mechanisms provide precise information into metrics such as latency, jitter, packet loss, and overall network reliability. The paradigmatic example of this type of measurement is the Internet Control Message Protocol (ICMP), applied to monitor the network layer. The two most popular tools that utilize ICMP are ping and traceroute, which send echo requests to a target measuring the time it takes for a reply to return. Traceroute, in addition, tracks each hop on the path a packet follows to reach its destination. These two methods provide information on the responsiveness of the network.

For application and cloud services, synthetic transactions are a common active monitoring method. These are emulated user interactions that monitor response times and service quality. Nonetheless, active measurements, if done massively and in large quantities, could affect the performance of the network.

When choosing a monitoring strategy for a communication network, in our case quantum, the most effective approach is usually to combine both passive and active mechanisms. Our proposal is a hybrid

measurement system that monitors both the QKD network hardware layer and the virtualized key management layer.

3.2 Mechanism design

For the QKD network hardware layer, a passive measurement strategy is proposed to enable continuous, unobtrusive monitoring. The monitoring component located in the control plane should be connected to the QKD modules at every time, allowing it to gather operational status information in real time. This would include key metrics such as the Quantum Bit Error Rate (QBER), calculated for each key exchange on a given link, the photon loss rate between pairs of QKD modules, and the rate of keys produced between two nodes in the network. Some commercial QKD modules already support these features, with some offering Simple Network Management Protocol (SNMP) compatibility to access collected data. However, any REST API could serve this purpose, allowing the monitoring component to access this data seamlessly without introducing additional traffic to the QKD modules. ETSI has acknowledged the need for a standard on passive monitoring and is currently working on a data model to be used for collecting these measurements [13].

Furthermore, the monitoring component may extend this type of passive oversight to other classical network elements, including routers, switches, and server computers. This complete monitoring approach supports proactive maintenance of both quantum and classical network components, improving the overall robustness and reliability of the network infrastructure.

By leveraging the virtualized layer of key managers introduced in the model, the monitoring component can also conduct active measurements, such as synthetic transactions, across the QKD network. In these scenarios, the virtualized key managers would function as network clients, providing realistic measurement results that closely mimic actual operating conditions and reporting these results to the monitoring component in the control plane. During periods of low network utilization, administrators can execute various tests on the physical infrastructure, gathering critical insights into performance metrics such as latency in key exchanges between specific QKD modules. Additionally, these tests allow administrators to assess the resilience of the network by evaluating diverse scenarios, such as overload conditions or link outages, enabling the evaluation and optimization of network adaptability under varied operational stresses.

This hybrid monitoring strategy, which incorporates passive and active measurements, can deliver precise, real-time data that enables a comprehensive and current view of network status. First of all, it guarantees the study and analysis of the network operational functioning, allowing administrators to anticipate, identify, and address potential failures that could escalate into critical issues.

In addition, the routing element, collocated in the control plane along with the monitoring component, would benefit from the data collected by the latter, computing and organizing key exchange routes based

on active link utilization and the latest available metrics. Furthermore, this approach significantly strengthens the defense capability of the network by identifying suspicious activity that could indicate a compromised link by an eavesdropper or attacker. This proactive detection capability allows the avoidance of potentially compromised links, dynamically reconfiguring existing routes through a coordinated response between the central controller, the monitoring element, and the routing component. If an attack or link failure is detected, the controller would act on all lower-layer elements, allowing routes to be reconfigured and processes to be redistributed as needed.

The proposed system enables a proactive management approach that not only maintains optimal performance but also strengthens the adaptability of the network to changing demands and conditions.

4 PROOF OF CONCEPT

To demonstrate the feasibility of the proposal and illustrate a use case, we have designed a proof of concept involving the three layers present in the model using a synthetic environment. The experiment has been carried out using Quditto, an open-source tool that allows the deployment of configurable digital twins in virtualization containers or virtual machines [14]. Quditto is organized into three different Python packages. The first package corresponds to the orchestrator that manages network deployment and configuration. The second package supports the network nodes, enabling the emulation of the quantum properties of the links and providing an implementation of a standardized ETSI API. Lastly, the third package is designed for the users of the network, allowing them to develop applications that interact directly with the standardized API to access cryptographic material generated by the emulated QKD modules.

For our experiment, we built the scenario that can be seen in Fig. 4. Each element in that figure corresponds to a virtual machine deployed in the 5G Telefonica Open Network Innovation Centre (5TONIC). Note that each QKD and KMA module pair forms a unit, a QKD node emulated with Quditto, and both elements are therefore hosted in the same virtual machine.

We deployed a six-node QKD network using Quditto, with the topology that can be seen in the lower layer of Fig. 4. Upon the emulated QKD network, we built a virtualized key management network to handle application requests (Fig. 4 upper layer). Additionally, we implemented on the same machine that acts as the Quditto orchestrator, a controller for the key management layer, handling route configuration and signaling as needed. We also added an emulated attestation component that would serve as a simplified monitoring element in this proof of concept. On a real network deployment, this component would apply integrity verification on all the running software elements, detecting security compromises [15]. Lastly, we integrated two applications functioning as network clients, each requesting a shared key between two distinct ends of the network. Application 1 initiates a request from KME 1, while application 2 requests from KME 6. Due to the current absence of quantum repeater technology, this key exchange is based on a trusted-relay scheme, which guarantees secure transmission over the network despite the

lack of a direct quantum link between the two sides.

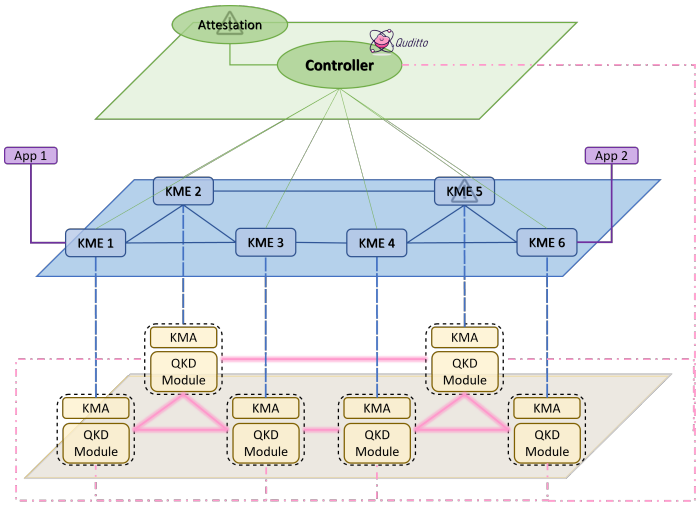

Figure 4: Deployed scenario for the proof of concept.

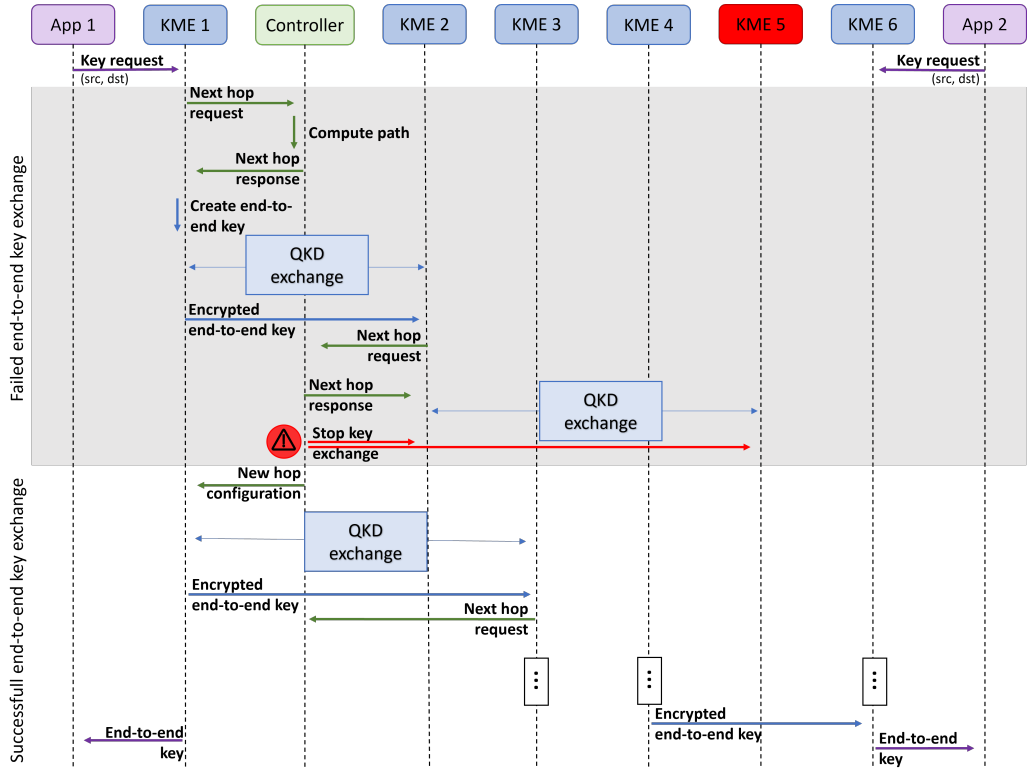

Figure 5: Flow diagram of the proposed proof of concept.

As can be seen in Fig. 4, there are different routes to get from KME 1 to KME 6. For simplicity, we defined two main paths: the upper path, involving KME 2 and KME 5, and the lower path, involving KME 3 and KME 4. Initially, these two routes are similar. However, once the key exchange begins, the attestation element alerts the controller of a compromised node, in particular, the KME 5. The controller must then immediately stop the key exchange on the upper path to prevent the end-to-end key from reaching the compromised node, which would pose a critical security risk. After terminating the exchange on the com-

promised path, the controller reconfigures an alternate path that avoids the compromised node to ensure secure end-to-end key exchange. This reconfiguration directs the key exchange along the lower path, maintaining the integrity and confidentiality of the key distribution process.

To better illustrate the actions we expect to see reflected in our implementation, Fig. 5 shows a flow chart with the entire process described previously.

First, the two applications send their request, including the source (KME 1) and the destination (KME 6) for the relay process. As source, KME 1 starts the process by asking the next hop to the controller, which chooses a path (firstly, the upper path) and responds with the address of KME 2. KME 1 creates the end-to-end key to be relayed, and the QKD exchange between the QKD modules associated with KME 1 and KME 2 starts. Once they both have a shared quantum key, KME 1 encrypts the end-to-end key using a one-time pad scheme, and sends it to KME 2.

In KME 2 this process is repeated. First, a request for the next hop is sent to the controller, since it is the first time a request is made and there is no record in its forwarding tables of any route, and the QKD exchange begins between the modules associated with KME 2 and KME 5. However, the controller receives a warning alert indicating that KME5 has been compromised, and immediately sends a stop instruction to end the key exchange process, preventing the end-to-end key from reaching the compromised node. Once the key exchange process in the upper route is terminated, the controller sends the address of the new next hop to the KME 1, so it can start again the relay process.

This time, the relay scheme is performed without any disruption and once the end-to-end key reaches KME 6, both applications receive their shared key.

4.1 Results

Within the outlined experimental setup, we take time measurements for each event in the process, from the initial key request by the applications to the final key response by KME 1 and KME 6. Fig. 6 shows the results of these measurements.

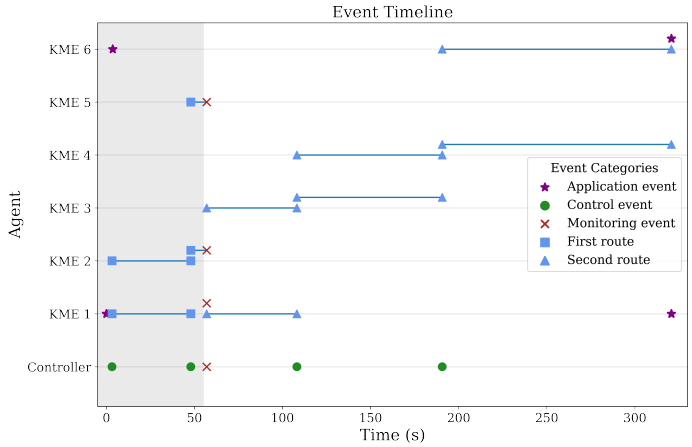

Figure 6: Time measurements for the different events in the designed proof of concept.

As can be seen in the figure, the implementation precisely follows the design steps. Once the application initiates a key request, the key relaying scheme starts on the upper path, moving sequentially from KME 1 to KME 2, then to KME 5. However, upon receiving a warning, the controller alerts the three KMEs involved quickly, and the process stops. Immediately after, the key retransmission scheme is rerouted by the controller to the lower path, going from KME 1 to KME 3, then to KME 4, and finally reaching KME 6. In the end, both final KMEs successfully deliver a shared key for their respective applications.

It should be noted that the timing displayed may not accurately represent the actual duration of the QKD exchange, as Quditto does not ensure a realistic time scale in the emulation environment at this time.

5 CONCLUSIONS AND FUTURE WORK

The QKD network monitoring strategy presented in this paper brings quantum cryptography one step closer to its inclusion in the service portfolio of modern telecommunications networks. By using a hybrid approach that combines passive and active monitoring measures, the proposed mechanism provides real-time insights into network health, optimizing reliability, security, and adaptability. Built upon a more general model based on a virtualized key management layer, the monitoring mechanism demonstrates notable benefits, including dynamic route selection informed by up-to-date traffic statistics, continuous monitoring of sensitive network elements, and rapid detection of potential threats or failures.

To demonstrate the feasibility of the model, we conducted a proof-of-concept demonstration aimed at testing route reconfiguration upon detecting a compromised node in a controlled synthetic environment. This experiment allowed us to evaluate the advantages of integrating monitoring strategies that enable automated and dynamic responses by network management components. The results highlight the capacity of the model for swift reconfiguration, reinforcing network resilience and adaptability to potential security threats.

For future work, we plan to develop an implementation of the complete monitoring mechanism, for use on real devices and field experiments, allowing us to assess its performance and practicality in real-world conditions. Additionally, we will further analyze the interactions between the monitoring component and other potential elements within the control plane to fully explore all the benefits it can offer. Following the idea of leveraging the experience gained in QKD networks, we also intend to explore the application of these principles to general entanglement distribution quantum networks.

ACKNOWLEDGMENT

This work has been partially funded by the project 6GINSPIRE PID2022-137329OB-C42, funded by MCIN/AEI/10.13039/501100011033/. This publication is part of the I+D+I project MADQuantum-CM, financed by the European Union NextGeneration-EU, Madrid Government and by the PRTR. This work has also been partially supported by the EU Horizon Europe project Quantum Security Networks Partnership (QSNP), under grant 101114043.

REFERENCES

[1] ETSI, *Quantum Key Distribution (QKD); Control Interface for Software Defined Networks*; ETSI GS QKD 015 V2.1.1 (2022-04); ETSI: Sophia Antipolis, France, 2022.

[2] ETSI, *Quantum Key Distribution (QKD); Orchestration Interface for Software Defined Networks*; ETSI GS QKD 018 V1.1.1 (2022-04); ETSI: Sophia Antipolis, France, 2022.

[3] ITU-T, *Quantum key distribution networks Software-defined networking control*; ITU-T Y.3805 (2021) Amd. 1 (11/2023); ITU: Geneva, Switzerland, 2023.

[4] Martin, V., Brito, J.P., Ortíz, L. et al. *MadQCI: a heterogeneous and scalable SDN-QKD network deployed in production facilities.* npj Quantum Inf 10, 80 (2024). https://doi.org/10.1038/s41534-024-00873-2

[5] Sim, D.-H., Shin, J., Kim, M. H. (2023). *Software-Defined Networking Orchestration for Interoperable Key Management of Quantum Key Distribution Networks.* Entropy, 25(6), 943. https://doi.org/10.3390/e25060943

[6] M. Condoluci, T. Mahmoodi, *Softwarization and virtualization in 5G mobile networks: Benefits, trends and challenges.* Computer Networks, Volume 146, 2018, Pages 65-84, ISSN 1389-1286

[7] B.Lopez, I. Vidal, F.Valera, D. Lopez, A. Pastor, *Unleashing Flexibility and Interoperability in QKD Networks: The Power of Softwarized Architectures.* In Proceedings of the 2024 International Conference on Quantum Communications, Networking, and Computing (QCNC) pp. 216–220.

[8] ETSI, *Quantum Key Distribution (QKD); Application Interface*; ETSI GS QKD 004 V2.1.1 (2020-08); ETSI: Sophia Antipolis, France, 2020.

[9] ETSI, *Quantum Key Distribution (QKD); Protocol and Data Format of REST-Based Key Delivery API*; ETSI: Sophia Antipolis, France, 2019.

[10] ITU-T, *Quantum key distribution networks Functional architecture*; ITU-T Y.3802 (2020) Amd. 1 (11/2023); ITU: Geneva, Switzerland, 2023.

[11] Diego Lopez, Vicente Martin, Blanca Lopez, and Luis Miguel Contreras, *A Multiplane Architecture Proposal for the Quantum Internet*. Internet Engineering Task Force, 2023. Work in Progress. Available online: https://datatracker.ietf.org/doc/draft-lopez-qirg-qi-multiplane-arch/

[12] N. L. M. van Adrichem, C. Doerr and F. A. Kuipers, *OpenNetMon: Network monitoring in OpenFlow Software-Defined Networks* 2014 IEEE Network Operations and Management Symposium (NOMS), Krakow, Poland 2014, pp. 1-8, doi: 10.1109/NOMS.2014.6838228.

[13] ETSI, *Quantum Key Distribution (QKD); Monitoring Interface and Data Model*; ETSI DGS QKD 023 0.0.3 Draft (2024-11); ETSI: Sophia Antipolis, France, 2024.

[14] Raul Martin, Blanca Lopez, Ivan Vidal, Francisco Valera, and Borja Nogales *Service for Deploying Digital Twins of QKD Networks.* Appl. Sci. 2024, 14, 1018. https://doi.org/10.3390/app14031018

[15] L. Jacquin, A. Lioy, D. R. Lopez, A. L. Shaw, and T. Su *The trust problem in modern network infrastruc-*

tures. In Revised Selected Papers of 4th Cyber Security and Privacy Innovation Forum, CSP Innovation Forum 2015, Brussels, Belgium (April 28-29, 2015), pp. 116-127.