Applying QKD to improve next-generation network infrastructures

Victor Lopez, Antonio Pastor, Diego Lopez
Telefonica Investigacion y Desarrollo/gCTIO
Ronda de la Comunicacion s/n 28050
Madrid. Spain

Alejandro Aguado, Vicente Martin
Center for Computational Simulation & DLSIIS, ETSIinf
Universidad Politécnica de Madrid
Campus de Montegancedo, Boadilla del Monte, 28660
Madrid, Spain

***Abstract*— There is a great attention to quantum technologies in the ICT environment. In particular, when dealing with security matters, the most prominent quantum technology is Quantum Key Distribution (QKD). QKD allows the sharing of symmetric keys with information theoretic security (ITS, i.e. independently of the computational power of the attacker) between two remote network nodes. QKD is the only known method to share a key able to reach ITS. During the last two decades, there has been a tremendous technological progress in QKD research, that has led to the availability of QKD network demonstrators.**

Software Defined Networks (SDN) enables the automation of service provisioning within network operator infrastructures. With the advent of web-scale services and dynamic network requirements, operators can not anymore deploy their services based on manual intervention or using proprietary vendor solutions. Programmability is key in the next-generation network infrastructure and any new technology must be integrated with this paradigm. Let us highlight that this requirement is even more important with virtual environments, where a Virtual Network Function (VNF) can be deployed in any point-of-presence of the operator. The new 5G deployments will enable operators to have edge computing services supporting different capabilities, thus increasing even more the complexity to deliver services without any automation.

In spite of the high potential of QKD, this technology has not yet found its path to wide adoption, commercialization and deployment. QKD is a physical technology that requires the existence of a quantum channel, a physical connection able to transmit quantum bits without perturbation, making hard its integration in networks. The aim of this document is to present how QKD can be deployed in next generation infrastructures, based on realistic scenarios. To do so, this paper describes the technological components of the solutions, as well as the use cases that motivate such effort. These use cases are described from a telecommunication provider's point of view, as they are the actors in charge of deploying QKD systems in their networks.

Keywords — Security; optical communications; quantum key distribution

I. INTRODUCTION

Network security has become a relevant topic for network infrastructure, not just in wireless, but also in wired communications. Operator infrastructures must cope with general security goals like authentication, information integrity and confidentiality. Secure communications rely on the strength of the encryption algorithms, either symmetric or asymmetric. When symmetric encryption is used, the messages are ciphered and deciphered using the same key. Consequently, the key must be known only by the parties in the communication. This requires the keys to be distributed across the parties in order to have a different key between each two communication entities. On the other hand, asymmetric encryption systems are based on handling a secret and a public key. Each end point publishes its public key and associates it with its identity. In order to send a message to this end point, the source ciphers the message using the public key, and this encrypted message, due to the encryption algorithm, can only be deciphered using the secret key. Essentially, asymmetric encryption systems are considered secure because of algorithmic complexity, which makes very hard computationally to obtain the secret key from the public key in a reasonable time. However, with the advent of new quantum computers, this approach could become obsolete, as these cryptographic schemes can be broken with quantum computers.

Quantum Key Distribution (QKD) appears as a solution to the distribution of keys that applies the laws of quantum physics. QKD enables the key exchange between two end points by means of a quantum channel [1-2]. From a high-level perspective, if two end points (Alice and Bob) are exchanging keys to later cipher their messages, it is not possible for Eve to access the key information, because it would change the state of the information sent between Alice and Bob. Thanks to Quantum Cryptography protocols, the increase in the error rate that the eavesdropper necessarily produces, can be recognized and a key with bounded information loss to the eavesdropper can be created. The bound can be as low, i.e. secret, as the users requires. A detailed explanation of QKD technologies can be found in [1-2].

In order to provide a QKD commercial-ready infrastructure, it is key to have a solution that can be seamlessly integrated with current network solutions. Most of the QKD products work based on proprietary management solutions without any standard interface. There is a need to provide interfaces that enable service creation following the Software Defined Networking (SDN) paradigm [3]. The remainder of this work is organized as follows: Section II presents the Software Defined QKD Node (SDQKDN) based on the work done within the European Telecommunications Standards Institute (ETSI), The European Telecommunications Standards Institute, and also introduces the architectural design for a commercial-ready QKD solution. Section III presents, from an operator point of view, those use cases that are more demanding in realistic scenarios. Finally, Section IV concludes this article.

II. Abstract Node Model for Software-Defined QKD

This section introduces the concept of a QKD system and presents a node model for QKD-enabled networks.

A. QKD system

Thanks to the QKD technology, a physical security layer can be implemented on an optical network. To do so, a quantum channel is required, implemented with a dedicated fibre as a transport media for the quantum signals (qubits) or a wavelength that coexist with traditional WDM channels. Using this quantum channel, the legitimate users can create a secret key shared only by them. Consequently, QKD extends the security perimeter from the computing and communications devices to the optical fibre used to connect them. A public and authenticated classical channel is also needed for QKD systems, in addition to the quantum channel. The public channel is used for post-processing operations, such as key distillation and privacy amplification.

QKD is a technology that is limited in terms of distance as the qubits have a finite probability to interact with the transport medium. Quantum information, being transmitted by individual quanta, is critically sensitive to the exponential attenuation that suffers any signal transmitted in a medium. Moreover, QKD system consider that any interference can be the action of a spy, so any error in the communication requires to discard the transmitted information, thus penalizing the secret key throughput. Using state of the art technology, a QKD system tolerates a maximum loss of about 30 dB, which is around 150 Km considering only the attenuation losses (0.2 dB per Km when using the telecommunications C-band at 1550 nm). At the same time, the longer the distance, the worse the key throughput. As an example, current throughput is about 1Mbit/s of final secret key at 40 Km distance in direct links (i.e. with ~8 dB losses), without incurring in any other loss.

B. A QKD Network layered view

Traditionally, networks are defined based on three main planes:

- **Data Plane** (or user plane): is used for the transmission of information packets among the network customers.
- **Management plane**: allows the access to the device from an entity dedicated to administrate the network, and deals with global operations, including accounting, security evaluation, monitoring reports, etc.
- **Control plane**: is in charge of decentralized operational issues such as the exchange of routing information, monitoring of link state and the set up and tear down of connections.

As soon as we integrate the concept of QKD, it is reasonable to extend the architecture into two more planes:

- **QKD plane**: This layer is the physical instance of QKD systems. It can be based on single QKD systems, with their own components (photon emitter, detector...) or by several QKD systems that are abstracted into a global QKD system.
- **Key plane**: is in charge of the management and generation of keys for the applications. It requires the specification of the required keys, as well as the adaptation of the QKD plane to operate in a mode suitable to obtain the required key throughput and enable the synchronization between the QKD systems.

C. Archictecture of a Software Defined QKD Node

Software-Defined Networking appeared (SDN) [3] as an approached to separate the data plane from the management and control planes, making them programmable. This implies to allocate the intelligence in a logically centralized entity, which is called the SDN controller. The role of an SDN controller is to program the network elements with the rules required. to provide the intended services. However, the communication of such central entity with the network elements requires the development of standard protocols to enable the interoperability of any device with the SDN controller, in our case SDQKDN controller. The paradigm of SDN, which started in packet networks, has been defined for the application layer as well as for optical networks. The reason is that SDN facilitates dramatically the integration of new devices and technologies in the network. We believe that applying such approach to QKD systems will help to speed up their deployment in realistic network environments.

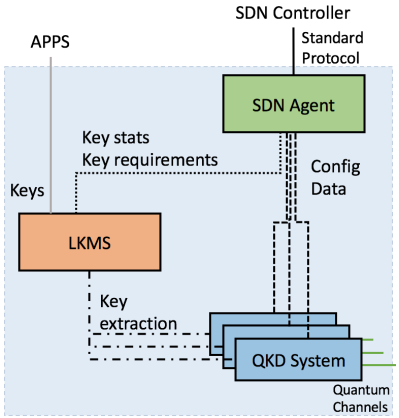

Fig. 1. *Archictecture of a Software Defined QKD Node*

Figure 1 shows our architectural proposal for a Software Defined QKD Node (SDQKDN) that is under definition in the ETSI ISG on QKD [4]. This architecture is defined based on three main components: a QKD system, a Local Key Management System (LKMS) and an SDN agent. The QKD system is an abstraction defined as an aggregation of one or multiple QKD systems. This approach simplifies the node view from the outside. The LKMS is responsible for maintaining and distributing the generated keys that are pushed (or extracted) to a local key store. Moreover, any request from applications is registered at the LKMS, including their identifiers, QoS, the key demands of each of them. Finally, the SDN agent is responsible for the communication with a central SDN controller. The SDN agent holds enough information about the QKD system to obtain the best performance of the devices. It is important to remark that the interface between SDN agent and controller allows to configure the behaviour of the QKD systems to create, remove or update key associations and to retrieve information from the QKD domain.

III. Use Cases

This section explains five use cases that presents how QKD technologies can be used in realistic scenarios, especially applicable to next-generation networks.

A. TLS integration with SDQKDN Controller

TLS v1.3 has been recently released. This new version includes several improvements in the protocol, including better cipher suites, with mandatory properties such as PFS (Perfect Forward Secrecy) and ephemeral keys. The integration of quantum generated keys in the implementation can increase the confidence in this protocol and their massive use. Also, some concerns arise in the case of users with high security requirements such as banks and corporations that need to ensure security by traffic inspection and still need static keys.

This use case promotes network providers SDQKDN-based controllers for a QKD infrastructure can offer a good balance between security and management of the TLS based encrypted communication.

TLS1.3 [5] released in August 2018 has some properties relevant to the QKD technology:

- Defines the use of Pre-shared Keys (PSK) as a key agreement process, jointly with Ephemeral DH.
- Removes static key management and promote PFS.

The second point is part of the assumption that pervasive surveillance can happen at large scale today, so guaranteeing PFS through ephemeral keys is mandatory to assure privacy. This assumption is correct in multiple situations involving TLS, such as browsing activities, e-commerce, etc. Therefore, this approach has seen a general adoption by browsers stakeholders (chrome, Edge, Firefox, etc.) and opensource projects (openSSL, wolfSSL, etc). Nonetheless, alternative scenarios are foreseen, such as devices with limited resources and/or only M2M (Machine to Machine) communications. In this case, PSK has been presented as an alternative, and indeed with this mode it is possible to avoid the use of certificates and PKI. One relevant example is the 5G Core network, where REST API services is being adopted with TLS as part of the SBA (Service Based Architecture), with M2M control plane communications using PSK.

To address this scenario, we propose a SDQKDN QKD controller, able to manage: a) the demand of key generation, or b) the secure transport of static keys via different APIs. For the first case, PSK keys can be generated in 2 sites over the SDQKDN controller domain and distributed to the TLS client and server. Both of them negotiate a PSK as key agreement (RFC4279 in TLS1.2 [6] and RFC8446 in TLS 1.3 [5]). This process removes the need of certificates and PKI implementation within the same domain. Optionally, it could be possible to use SDNQKD to provide symmetric keys as an alternative to any cipher suite negotiated. In this situation, the key generation rate must accomplish the requirements of TLS session refresh.

In the second case, presented in Fig. 2, a centralized management deals with the keys for eTLS (ETSI TS 103 523-3) [7] where the SDQKDN controller will leverage the presence of QKD nodes to transport securely over an encrypted channel (optical encryption, IPSec, another TLS) the static DH keys to be used by the middleboxes and TLS server deployed in different sites.

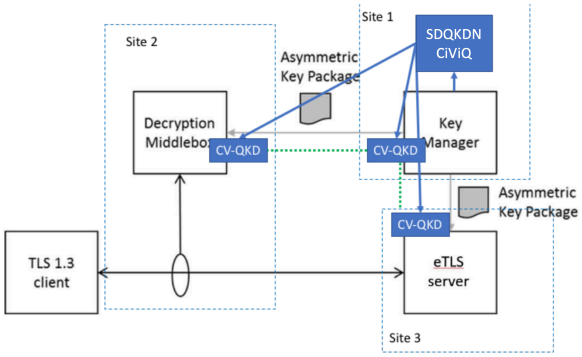

Fig. 2. *CIVIQ SDQKDN interacts with eTLS Centralized static key management*

B. Network management secured with QKD

Novel network paradigms can play a very important role for the integration of QKD in operator networks. Within the network, QKD is a technology to be deployed only in secure areas or PoPs, where the rest of the network elements (NEs) are also present.

This situation allows such NEs, points-of-presence and data centers to make use of QKD-derived keys to secure its own communications towards network management systems or SDN controllers. Therefore, we can simultaneously control the QKD elements, while securing any control plane channel among PoPs and data centers.

With this goal, it is required to analyze the most deployed security protocols and their key exchange protocols and algorithms to make them compatible with QKD. To achieve this, some steps have to be taken:

- Perform an initial requirements analysis (such as integrability by extending the key agreement techniques and further extensions to be pushed in standardisation forums, etc.). Also describe the most used management protocols and how this analysis fits to any of them.
- Define the workflows for such integration. The QKD network should be transparent to applications but, if not defined properly, it might lead to multiple implementations not compatible with one another.
- Create a final implementation for the different protocols and cases, either using a virtual QKD network, or finally integrating in a demonstrator.

Some examples of protocols that may integrate this security mechanisms are the secure shell (SSH) protocol, SSL/TLS, SFTP, etc. A first integration for securing such channels was implemented in [8].

This security mechanisms are meant to be implemented in the network management plane, to securely handle any centralized operation, including the communications channels between NFV platforms (e.g. OpenSource MANO [9] managing a virtual infrastructure manager (VIM) based on OpenStack), the communication between a SDN controller and a network device via OpenFlow or NETCONF protocols, the exchange of traffic engineering information between a node and a path computation element, etc. Figure 3 illustrates the control plane protocols and interfaces for NFV scenarios.

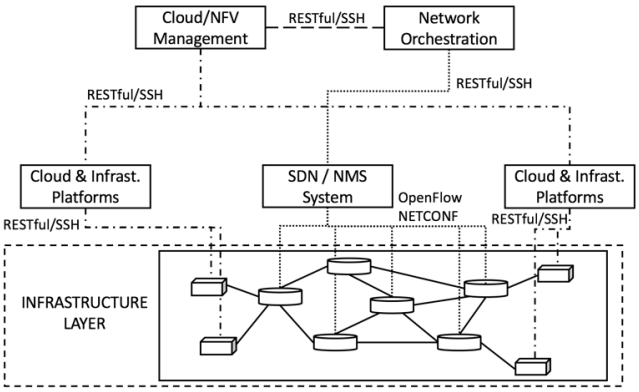

Fig. 3. *Control plane protocols and interfaces within a network.*

C. *Quantum cryptography for secure ordered proof-of-transit*

Network architectures are on their way to evolve towards virtualized systems and network elements that will soon replace traditional hardware appliances for software running in virtual environments at homogeneous data centers. This situation supports a faster innovation pace, a reduced time-to-market of new solutions, and a reduction in time and costs for deploying new elements to cope with the service demands. Nonetheless, this situation brings a new non-desired uncertainty when creating virtualized end-to-end services, commonly realized by means of Service Function Chaining (SFC). Applying QKD it would be possible to enhance existing algorithms to provide a solution based on ordered proof-of-transit (OPoT), so any traffic flow traversing an SFC can be validated and it is possible to verify that every packet has been handled by the required nodes in the required order (Figure 4).

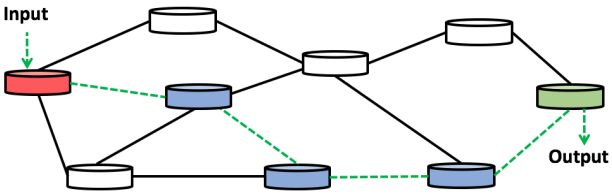

Fig. 4. *Nodes within a OPoT scheme*

Some participants of the SFC working group in the IETF created a first version of an internet draft defining a technique for proof-of-transit [10]. The initial version of the document, apart from having some security implications and vulnerabilities associated, defined an additional solution for bringing order which did not fit with the initial proposal and also had some computational implications. A later proposal incorporated an incremental approach over the original algorithmic solution (based on Shamir Secret Sharing) and to enhance its security while providing order to the scheme. This solution requires symmetric key algorithms (which are not computationally costly) and a well-provisioned parallel source of symmetric keys, storing enough keys available for the scheme.

While intra-data center communications can be assumed as safe (a trusted zone from the QKD perspective), QKD can provide such source of symmetric keys, avoiding the problems of the original PoT scheme while providing the required order to the solution.

D. *Quantum security embedded in network elements and hardware appliances*

Network vendors are evolving their hardware solutions in order to integrate encryption cards or hardware security modules (HSM) to secure the data transmitted in network services at different layers. Such solutions still rely on traditional schemes to produce symmetric keys (e.g. Diffie-Hellman) that will be used to encrypt a big amount of traffic. In addition, an everlasting security solution might be required, as the traffic might contain critical data not to be disclosed at any time in the future. QKD is positioned as an applicable source of keys for such services, while they coexist with the quantum channels to maximise/optimise spectrum utilisation. There are few of such integrations described in the literature, being all of them deployed for very specific and ad-hoc scenarios.

To achieve this goal, it is necessary to accomplish a few incremental steps:

- First, list any scenario that can be relevant for customers and, therefore, capitalized by the network operation.
- For such scenarios, evaluate all the network services and security techniques being underutilisation.

- Investigate the data plane requirements for integrating the quantum keys and use them as the input for the definition of a QKD key delivery API.
- Investigate common control plane protocols used for deploying the services listed in the second point above, proposing any required extension for synchronising the quantum-encrypted channel.

New hardware appliances could implement any of the extensions resulting from this study, integrating with a QKD-aware controller to automate such new services. They could apply to business-to-business services, business-to-customers or inter-data centre connectivity, among others. Figure 5 illustrates an example where the nodes in the edge could encrypt end-user traffic using for example OTN cards.

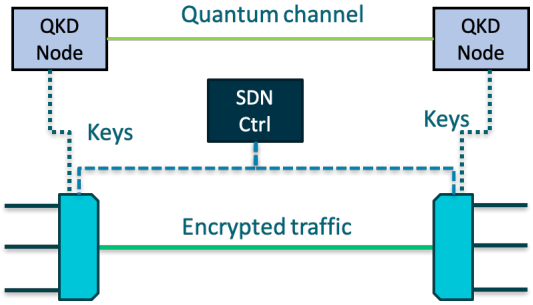

Fig. 5. *Set of nodes involved in the E2E encryption service use case*

E. Quantum cryptography for IPSec via SDN

This use case presents how it is possible to provide IPsec-based flow protection by means of an SDN Controller with QKD integration (Figure 6). The solution would use SDN controller capacities to integrate the management and generation of keys (based on a QKD infrastructure) used by IPsec security associations (SA) as an alternative to IKE protocol configuration.

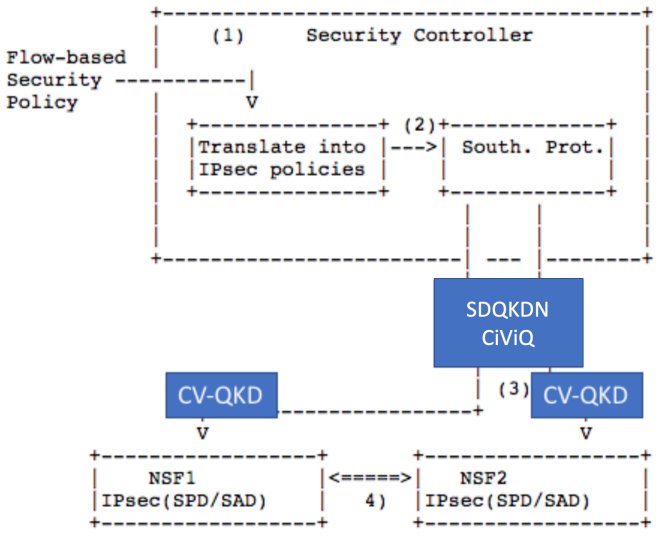

Fig. 6. *CIVIQ manages the key delivery service to IPsec Nodes*

Current IPSec management process is manually done by administrators, using IKE protocol configuration. Multiple sites require multiple configurations. Weak password selection, such as those vulnerable to dictionary attacks, or weak cipher suites, could create vulnerable VPNs services. To alleviate this, we can apply a combination of 3 elements:

- IPSec devices. Typically, routers or VNFs using the IPSec tunnel mode (or transport mode)
- QKD devices and their associated quantum links, all controlled by a SDNQKD architecture.
- The SDN security controller, with capacity to manage IPsec and QKD devices. An original approach for automating such services was presented in [11].

We foresee the integration of an SDN and NFV controlled environment for QKD with the IPSec model defined by IETF I2NSF [12]. In this model, an I2NSF security controller in charge of managing IKE and the SAD/SPD will interact or will be a component of the SDNQKD node and will perform key delivery on demand to the IPSec devices (represented by an Network Security Function (NSF) in the figure below).

IV. CONCLUSIONS

This paper presents an architectural design of SDNQKD nodes and their key elements to support secure network operations in realistic scenarios, providing a seamless integration with operator systems. Besides, the article describes several use cases where QKD can be applied to improve next-generation network infrastructures. They cover from the TLS 1 integration with SDNQKD controller to the integration of QKD with IPSec.

ACKNOWLEDGMENT

The work described in this paper was carried out with the support of the FET Flagship on Quantum Technologies, European Union's Horizon 2020 research and innovation programme under grant agreement No 820466: Continuous Variable Quantum Communications (CiViQ).

REFERENCES

[1] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, "Quantum cryptography," Rev. Mod. Phys., vol. 74, no. 1, pp. 145–195, Mar. 2002.

[2] V. Martin, J. Martinez-Mateo, and M. Peev, "Quantum key distribution, introduction," in Wiley Encyclopedia of Electrical and Electronics Engineering, Wiley, 2017, pp. 1–17.

[3] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: enabling innovation in campus networks," SIGCOMM Comput. Commun. Rev., vol. 38, no. 2, pp. 69–74, Mar. 2008.

[4] [Online], "ETSI QKD ISG" Available: https://www.etsi.org/technologies/quantum-key-distribution (Accessed February 20, 2019).

[5] RFC8446 "The Transport Layer Security (TLS) Protocol Version 1.3", https://tools.ietf.org/html/rfc8446.

[6] RFC4279 "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", https://tools.ietf.org/html/rfc4279.

[7] “CYBER; Middlebox Security Protocol; Part 3: Profile for enterprise network and data centre access control” ETSI TS 103 523-3 V1.1.1 (2018-10)

[8] A. Aguado, V. Lopez, J. Martinez-Mateo, T. Szyrkowiec, A. Autenrieth, M. Peev, D. Lopez, and V. Martin, "Hybrid Conventional and Quantum Security for Software Defined and Virtualized Networks," J. Opt. Commun. Netw. 9, 819-825 (2017).

[9] [Online] “ETSI OSM Website” https://osm.etsi.org/ (Accessed February 20, 2019)

[10] IETF Internet Draft: “Proof of Transit”, draft-ietf-sfc-proof-of-transit-01.

[11] A. Aguado, V. Lopez, J. Martinez-Mateo, M. Peev, D. Lopez, and V. Martin, "Virtual Network Function Deployment and Service Automation to Provide End-to-End Quantum Encryption," J. Opt. Commun. Netw. 10, 421-430 (2018).

[12] IETF Internet Draft: “Software-Defined Networking (SDN)-based IPsec Flow Protection”, draft-ietf-i2nsf-sdn-ipsec-flow-protection-03