# Experimental Validation of an End-to-End QKD Encryption Service in MPLS environments

A. Aguado[1], J. Martinez-Mateo[1], V. Lopez[2], D. Lopez[2], M.Peev[3], V. Martin[1]
[1] CCS - Universidad Politécnica de Madrid. Campus de Montegancedo. Boadilla del Monte, 28660 Madrid. Spain
[2]Telefónica GCTO, Ronda de la Comunicación s/n 28050 Madrid. Spain
[3]Huawei Technologies Duesseldorf GmbH, Riesstrasse 25, 80992 Munchen. Germany
(a.aguadom@fi.upm.es, vicente@fi.upm.es)

***Abstract*—Current network architectures are rapidly evolving towards more dynamic solutions, due to the ever-growing demand of resources from highly-heterogeneous new services. This forces network management systems and protocols to quickly adapt to support new capabilities and security is one of the main concerns. In this work, we define and demonstrate for the first time extensions for multi-protocol label switching (MPLS) networks using quantum key distribution (QKD) keys to secure end-to-end (E2E) network services. The proposed solution allows to synchronize key IDs between remote endpoints, as well as to transmit other parameters required for the encryption. Results show how these new services could be integrated in existing operators control plane architectures.**

NETWORK services are continuously evolving, moving from a traditional approach, based on proprietary devices performing a fixed function, to more flexible and open solutions, implemented in general purpose hardware and using open standards. Traditional services usually require several days (or even weeks) to be established, while new applications and services change their requirements much faster. This evolution, aiming to cope with this dynamicity, is possible thanks to novel network paradigms, such as software defined networking (SDN) [1]. SDN allows to decouple the control plane (the management of the network itself, traditionally running in the core of a network device) from the data plane (the actual transmission of the data). It manages network services from a logically centralized entity (network controller) that can be physically distributed. Security is an increasing concern in communications networks, as critical information travels across an entire infrastructure. However, network security has not been the result of a systematic effort, but more as a series of ad-hoc solutions. Today, networks are more complex and, especially with SDN, much more configurable. SDN networks, by boosting the interoperability character, are more sensitive to security breaches that can extend faster and reach more nodes. The security risks are correspondingly larger and, therefore, security in network infrastructures must be enhanced.

QKD technology [2], [3] can be regarded as two sources of synchronized random numbers that are separated in space. It also allows to upper bound the maximum information that is leaked out of these two sources, hence they can be used as secret keys to cypher the communicatios. However, it is a delicate technology and dealing with the preparation and measurement of single quantum signals impose very stringent requirements on the physical implementation, specially in the maximum range achievable and in its tolerance to external noise (i.e. sharing the same media with classical communications signals). It is also intrinsically point to point. Two paradigms have been proposed to mitigate this restriction: one is the switched network, where uninterrupted and unamplified, point to point optical paths supporting a quantum channel are created in an optical passive network; the second is a trusted node network, where the end to end paths pass through intermediate nodes. Both paradigms have been implemented in practice [4]–[6]. However the difficulties, if properly implemented, the security of the symmetric keys produced by systems built around QKD can be very high. QKD is, by principle, immune to any algorithmic cryptanalysis and provides forward and backward security. In consequence, QKD can be seen as a new opportunity for operators and infrastructure providers as it can enable the provision of new, high security encryption in end-to-end (E2E) services. But bringing quantum encryption awareness and the capability of providing inline encryption into a logically centralized control plane will require a modification of the existing protocols and to develop some necessary extensions. This work describes the proposed solution for integrating QKD into network services via MPLS control plane (as its generalized version, GMPLS), initially presented in [7].

In order to provide such services in current communication networks, we must first define network nodes capable of providing this quantum encryption -QE- (see Fig. 1). This type of node must:

- Generate keys synchronized with remote peers to perform symmetric encryption (in our case, via QKD).
- Encrypt the outcoming traffic utilizing those keys.
- Be able of performing switching or routing (optional).
- Communicate (northbound intarface) with a network controller.

This set of capabilities will additionally bring certain requirements from the NBI and management perspective, as each of this nodes will need to be operated from the centralized controller. The main operations to be adapted will be: capabilities dissemination and device configuration (together with the key synchronization process).

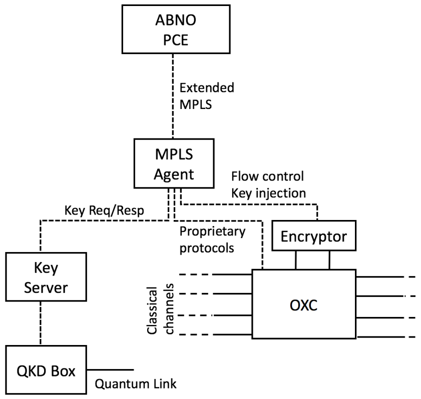


Fig. 1. Example of an SDN-enabled network node, providing encryption via QKD-generated keys.

Our proposed workflow, including those operations and the main messages you be exchanged in a MPLS-enabled network, is shown in Fig. 2. Each step is as indicated below:

- Initially, a QE node should expose its capabilities to the centralized controller. In our case, it will consist on a path computation element -PCE- or more complex architectures (such as the applications-based network operations -ABNO- architecture).

- The request for an E2E QE service could come from the controller's NBI or from the device itself. In our figure, the node 1 sends a path computation request, sending the encryption requirements encapsulated in metrics.
- The path is returned to the source node, which will detect the QE requirements. Upon receipt, this node will extract the key and key ID pair from a QKD system (or key manager).
- The key ID will be forwarded via RSVP path message to the destination node, which will extract the required key from the peer QKD system (or key manager).
- This process will finalise when a RSVP Resv message arrives to the source node, acknowledging the configuration.

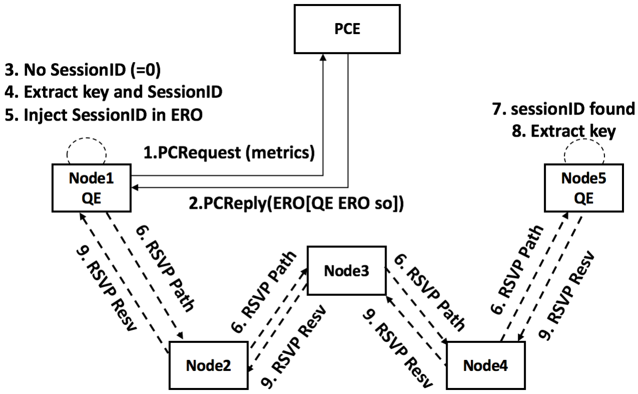


Fig. 2. MPLS workflow for setting up a E2E QE service.

### A. Capabilities Dissemination

Centralized control plane entities (controller/PCE) are required to gather basic information from the devices (statistics, system information) to build a graph of the existing network and compute and optimize network services to better use the available resources. Encryption services (if centrally managed by a controller/PCE) require at least a minimum information to be exposed from the network devices that can perform such process.

The proposed extension to disseminate QE capabilities is based on the RFC7770 [], which defines OSPF extensions for optional router capabilities. This extension, implemented for OSPFv2, uses the router information (RI) opaque link state advertisement (LSA) within an OSPF update message. A single bit is exposed within an informational capabilities TLV, to let the PCE know that it can provide such encryption services to end users.

### B. Device and Service Configuration

When a PCReply or a PCInitiate message arrives to a PCC of a network node, this node is in charge to start the signaling process by extracting the ERO from the PCEP message and transmitting it via an RSVP Path message down to the destination node. In our case, it is mandatory to synchronize the QKD-generated keys in both sides. This process can be easily done by exchanging IDs to be extracted in both endpoints. While doing this process through a non-standard channel could be done, RSVP is the best candidate to automate the key synchronization process using a standard protocol. It is capable of forwarding the encryption requirements across the path and return a confirmation (Resv message) if the resources (keys) have been reserved, while the signaling process traverses the network.

To encapsulate the key ID together with other important information we have created an explicit route object -ERO- subobject -SO-, as shown in Fig. 3. This structure contains encryption information, such as key lenght, encryption algorithm, key refresh values (if necessary) and layer of encryption. This structure is transmitted and modified on-the-fly between PCE and the network devices to synchronized both ends of the encrypted path.

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|L|    Type     |     Length    |      Session ID  (64 bytes)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            ...  (64 bytes)                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Session ID  (64 bytes)   |             KEY LEN           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Enc_layer   |    Ref Type   |    Refresh value  (2 bytes)   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Enc_Algo (2 bytes)      |                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fig. 3. ERO Subobject used to exchanged QE information between the two enpoints of the path and the PCE.

### C. Experimental Results

The full set of the messages transmitted across the GMPLS control plane network is shown in Fig. 4. The first message is an OSPF update from the fifth node (others are omitted), which contains the router information opaque LSA, with the traffic engineering capable and que QE capable bits set to 1 within the informational capabilities TLV. The second and third messages are the PCRequest and PCReply messages, including the new metrics and the new QE ERO SO (Fig. 3). The remaining messages are the RSVP path and resv messages, used to forward the configuration from the ingress (source) to the engress (destination) nodes of the path.

Finally, Fig. 5 shows how the QE ERO SO is modified by the source node, by including the valid key ID to be used by the encryption path. The interface used to retrived the keys from the QKD systems has been designed using the API specification described in [8]. Upon receipt, the destination node gets the valid key ID to retrieve a key from the corresponding QKD system for the encrypted channel.

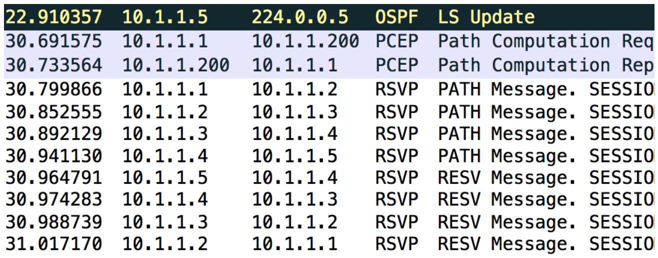
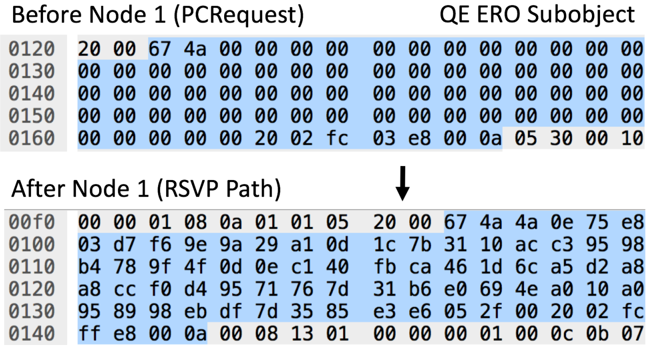

```
22.910357  10.1.1.5     224.0.0.5    OSPF  LS Update
30.691575  10.1.1.1     10.1.1.200   PCEP  Path Computation Req
30.733564  10.1.1.200   10.1.1.1     PCEP  Path Computation Rep
30.799866  10.1.1.1     10.1.1.2     RSVP  PATH Message. SESSIO
30.852555  10.1.1.2     10.1.1.3     RSVP  PATH Message. SESSIO
30.892129  10.1.1.3     10.1.1.4     RSVP  PATH Message. SESSIO
30.941130  10.1.1.4     10.1.1.5     RSVP  PATH Message. SESSIO
30.964791  10.1.1.5     10.1.1.4     RSVP  RESV Message. SESSIO
30.974283  10.1.1.4     10.1.1.3     RSVP  RESV Message. SESSIO
30.988739  10.1.1.3     10.1.1.2     RSVP  RESV Message. SESSIO
31.017170  10.1.1.2     10.1.1.1     RSVP  RESV Message. SESSIO
```

Fig. 4. Capture of the set of messages exchanged during the MPLS workflow.

## Conclusions.

The combination of novel network paradigms and QKD technologies is just starting. Network infrastructures must quickly adapt to enhance security for their underlying services. In this work we describe and implement extensions to integrate QKD resources into E2E services, all automated across the MPLS control plane. These protocol extensions will be an enabler for operators to offer and capitalize new encrypted network services powered by QKD technologies. QKD as we know it today is just the starting point, but novel models and techniques, such SDN, will allow for the evolution and smooth adaptation these new capabilities and devices.



```
0120  20 00 67 4a 00 00 00 00  00 00 00 00 00 00 00 00
0130  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0140  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0150  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00
0160  00 00 00 00 00 20 02 fc  03 e8 00 0a 05 30 00 10
```

```
00f0  00 00 01 08 0a 01 01 05  20 00 67 4a 4a 0e 75 e8
0100  03 d7 f6 9e 9a 29 a1 0d  1c 7b 31 10 ac c3 95 98
0110  b4 78 9f 4f 0d 0e c1 40  fb ca 46 1d 6c a5 d2 a8
0120  a8 cc f0 d4 95 71 76 7d  31 b6 e0 69 4e a0 10 a0
0130  95 89 98 eb df 7d 35 85  e3 e6 05 2f 00 20 02 fc
0140  ff e8 00 0a 00 08 13 01  00 00 00 01 00 0c 0b 07
```

Fig. 5. QE ERO SO changed as it traverses the source node. It finally includes the key ID to be used by the destination node.

## Brief Glossary.

IGP : Interior Gateway Protocol.
BGP : Border Gateway Protocol.
MPLS : Multiprotocol Label Switching. Encapsulation technique for fast data routing avoiding routing tables.
NVF : Network Function Virtualization.
OSPF-TE: Open Shortest Path First routing protocol for Traffic Engineering. Used to describe the topology of a network.
SDN : Software defined network.
VNF : Virtualized Network Function. Describes an instance of a software image performing a network function in the NVF paradigm.

## Acknowledgements.

This work has been partially supported by the project CVQuCo, TEC2015-70406-R, funded by the Spanish Ministry of Economy and Competitiveness and QUITEMAD+, S2013-IC2801, funded by Comunidad Autónoma de Madrid and EPSRC EP/M013472/1: UK Quantum Technology Hub for Quantum Communications Technologies.

## References

[1] 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, March 2008.

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

[3] V. Martin, J. Martinez-Mateo, and M. Peev, “Quantum key distribution, introduction,” in *Wiley Encyclopedia of Electrical and Electronics Engineering*, 2017.

[4] D. Lancho, J. Martinez, D. Elkouss, M. Soto, and V. Martin, “QKD in standard optical telecommunications networks,” in *Quantum Communication and Quantum Networking*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer Berlin Heidelberg, 2010, vol. 36, pp. 142–149.

[5] M. Peev *et al.* , “The SECOQC quantum key distribution network in Vienna,” *New J. Phys.*, vol. 11, no. 7, p. 075001, 2009. [Online]. Available: http://stacks.iop.org/1367-2630/11/i=7/a=075001

[6] M. Sasaki *et al.*, “Field test of quantum key distribution in the Tokyo QKD Network,” *Opt. Express*, vol. 19, no. 11, pp. 10 387–10 409, 2011.

[7] A. Aguado, V. Lopez, J. Martinez-Mateo, M. Peev, D. Lopez, and V. Martin, “Gmpls network control plane enabling quantum encryption in end-to-end services,” in *International Conference on Optical Network Design and Modelling (Best Paper Award)*, 2017.

[8] “Quantum key distribution (qkd); application interface,” in *ETSI GS QKD 004 V1.1.1*, 2010-12.