Experimental Demonstration of Optical Encryption Using Quantum Keys: Two Scenarios

Morteza Ahmadian[1], Rafael J. Vicente[2], Juan P. Brito[2], Álvaro López-García[2], Antonio Pastor[3], Jose M. Rivas[3], Jaume Comellas[1], Marc Ruiz[1], Vicente Martin[2], and Luis Velasco[1*]

[1] Universitat Politècnica de Catalunya (UPC), Barcelona, Spain; [2] Universidad Politécnica de Madrid (UPM), Madrid, Spain; [3] Telefonica I+D, Madrid, Spain; e-mail: luis.velasco@upc.edu

Abstract: Optical encryption using Quantum keys retrieved from real QKD and QRNG systems will be demonstrated. Retrieved keys are expanded to the required bitrate and then used to encrypt the input bit stream at line speed.

1. Overview

The high capacity and low latency of optical connections are supporting 5G and beyond communication. Although some of services are already secured at the packet layer using standard stream ciphers, like Advanced Encryption Standard and ChaCha, secure transmission at the optical layer is still not massively implemented, mainly due to the added delay and the limitations of those cryptographic methods to work at line speeds of 400Gb/s or higher. In our paper [1], we proposed a secure cryptographic solution for optical signals (e.g., 16 QAM) named *Light Path SECurity* (LPsec), that involves fast bit stream encryption using stream ciphers and key exchange by implementing the Diffie-Hellman protocol through the optical channel. One of the main limitations of LPSec is the way symmetric keys are generated, which reduces the security level.

In this demonstration, we will showcase, two scenarios where keys are retrieved from: *A*) a Quantum Key Distribution (QKD) network, assuming that the optical transponders (Tp) are inside the security perimeter of the QKD network; and *B*) a Quantum Random Number Generator (QRNG) in the case that only one of the Tps is inside that security perimeter. In both cases, the retrieved keys leverage the random properties of quantum physics. Such keys are then expanded to the desired line rate using a secure Pseudo-Random Number Generator (PRNG). An optical connection between two Tps will be set up using a software-defined networking (SDN). Both Tps will connect to its local Key Manager (KM) in Demonstration A, to retrieve keys exchanged through a QKD system connecting the premises of Telefónica in Madrid, Spain, whereas Demonstration B, the Tp inside the security perimeter retrieves keys from a local key server in a QRNG and exchanges them through the classical optical channel in a secure way. The workflows have been designed in the Horizon Europe ALLEGRO project [2].

2. Innovation

QRNGs devices produce high quality entropy, hence an ideal resource for cryptographic purposes. Although QRNGs can be used on its own for classical processes, they are usually part of QKD networks that allow for the distribution of symmetric keys with bounded security between parties, e.g., the Tps in our case. Hence, a QKD network can be seen as a distributed QRNG, where the random keys appear at two ends of the network.

In this demonstration, we will showcase encryption of an optical signal, e.g., a 16 QAM signal, using the keys retrieved from a QKD network. Retrieved keys are used in LPSec as input to a key expansion mechanism based on a secure PRNG to encrypt the input bit stream at line rate. Note that the perfect security provided by the QKD network is reduced by the high expansion rate needed to expand retrieved key to the line rate, as QKD systems provide throughputs from hundreds of kb/s to few Mb/s, depending on the system. To achieve the highest security level, both Tps need to be within the security perimeter of the QKD network, which is not always possible, e.g., when the optical connection involves the access network. For this very reason, in this demonstration, we will additionally show LPSec using a single QRNG as the entropy source, where keys are retrieved by the local Tp, used to encrypt the optical signal, and exchanged to the remote Tp within the optical signal. Therefore, LPSec is used to extend farther the reach of the security perimeter. Typically, QRNGs have very high throughput (about 1 Gb/s), so moderate expansion to reach the desired line rate is needed in this case.

3. OFC Relevance

LPSec has shown its ability to provide bit stream encryption at line speed, while introducing noticeable low processing delay. That solution allows to cover a gap in standard optical communications, where data is generally transmitted as plain text. The combination of LPSec and quantum security represents another turn of the screw to classical optical

The research leading to these results has received funding from the European Union's Horizon Europe research and innovation programme under G.A. No. 101092766 (ALLEGRO) from the MICINN IBON (PID2020-114135RB-I00) projects and from the ICREA Institution.

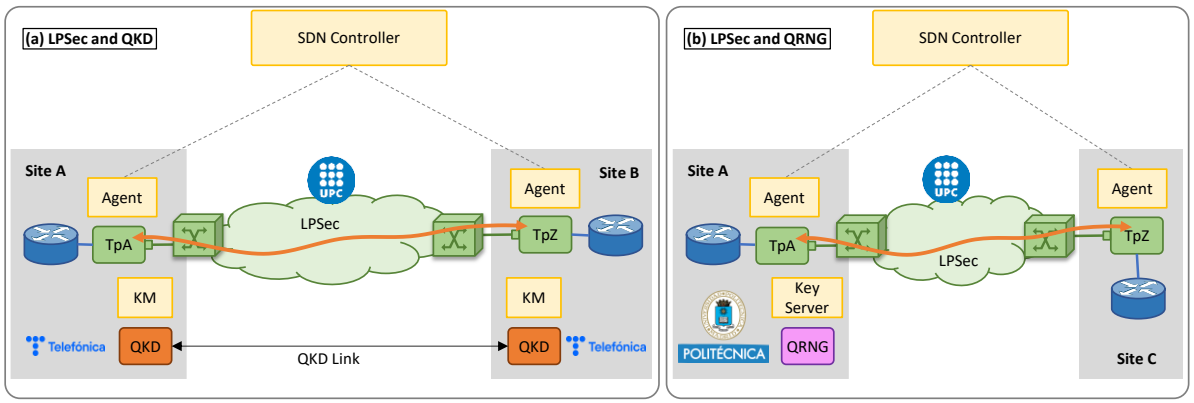

Fig. 1: Demonstration scenarios: Lightpath end points are inside QKD security perimeters (a) and only one end point is in a security perimeter.

communications, as it allows upgrading technologies currently deployed in operators network with quantum security, thus extending their lifespan, which is of the interest of vendors and operators. In addition, the possibility to extend the security perimeter of the QKD network will be received with interest form the audience in general.

4. Demo content & implementation

LPsec extends the standard coherent transponder with optical encryption and decryption blocks, as well as with some key management functionalities; note that cryptographic blocks operate at line speeds, so optical encryption is based on simple operations performed on the input bit stream. The encryption is based on two nested ciphers: *i*) a substitution cipher for scrambling symbols, where a Lookup Table (LUT) is used to create a ciphered gray map constellation through LUT permutations of incoming bits; and *ii*) a stream cipher that encrypts data chunks of predefined size based on a cryptographically secure PRNG to generate a sequence of stream keys from an input random key *k*. However, the sequence of stream keys generated by the PRNG from a given input key cannot be infinite as this would reduce the security level, and the LUT permutation needs to be periodically regenerated to minimize vulnerabilities. In consequence, we limit the lifetime of keys *k*, e.g., to 1 sec., which entails new keys being periodically made available at the two ends of the lightpath. To facilitate communication between the two Tps, a special frame named Key exchange Frame (KxF) is used. A KxF is generated by the Tx and sent to the Rx periodically. The KxF includes a header of a fixed size that allows the Rx to detect its arrival.

Two demonstrations will be showcased. In the first demonstration (*Demo A*), the two TPs of the lightpath are in sites covered by the QKD network, so keys *k* can be retrieved from the local KMs using a standard interface, we rely on ETSI GS QKD 004 [3], and used as input of the PRNG. In this case, the KxF header is used for synchronization purposes between the two Tps. Fig. 1a presents the scenario of this demo, where two QKD systems with their respective KMs are deployed in the premises of Telefónica, in Madrid, Spain and connected to create a QKD link. Optical transmission and LPSec, including encryption and key exchange is implemented in a simulator running in UPC premises in Barcelona, Spain. An SDN controller is used for lightpath set up and LPSec configuration. Finally, two TP agents are in charge of configuring the local Tps and communicate with the SDN controller. Please, note that details of the workflows of both demonstrations have been intentionally omitted due to the space constraints; please refer to [1] for details, especially those related to LPSec, e.g., the use of public/private keys, KxF, LUT, etc.

Demo A extends case 1 “undefined KSID in a single link scenario” defined in [3], to be used for LPSec connection set-up. Note that case 5 defined in [3] could be also implemented in the case that QKD systems in sites A and B are in the QKD network but not directly connected through a single QKD link.

The workflow starts after the lightpath has been established over the optical network (labeled 0 in Fig. 2). LPSec requires an initial public key exchange to be carried out during connection set-up through the SDN controller. In this case, the keys are used to generate the particular KxF header pattern that will be used on the optical channel for synchronization between the two Tps. The SDN collects the public key and the ID of TpZ (1) and sends them to the agent of TpA together with the details of the local KM (2). The agent in TpA connects to the local KM using the OPEN_CONNECT function and indicating the ID of TpA as source and that of TpZ as destination, and receives the *Key stream ID* (KSID) to be used for retrieving keys for optical encryption (3). KM in site A coordinates internally with the KM in site B to create the association KSID in the two sites (4). The KSID needs to be used in site B, so TpA agent sends it together with the public key, the ID of the local Tp and the KxF header pattern encrypted with the public key of TpZ to the SDN controller (5), which, in turn, sends them to TpZ agent (6). The agent in site B uses the KSID together with the IDs of the local Tp to connect to the local KM (7) and reports to the SDN controller. At this time, both ends are ready to use the optical connection with encryption.

The next step is to synchronize the two Tps so they can start using the right keys to encrypt the bit stream. Then, the SDN controller requests both ends to start with that tasks (8, 8'). In the case of the agent of TpA, it acknowledges the request and asks the local Tp to get the first key (10), uses it to encrypt a known bit stream and inserts the KxF header periodically including in the header the index of the retrieved key (12). On site B, the agent sends the KSID and the ID of TpA to the local Tp, which retrieves two keys (10', 11), the first one will be used during the synchronization period and the second once that period finished and the real bit stream is received. When TpZ is able to successfully decrypt the known bit stream and finds the KxF header with the same index as the one got from the local KM, it reports to the SDN controller. The SDN controller requests TpA agent to start (13), which in turns notifies the local Tp (14). At this time both ends are ready to start with the real data encryption (15). Then, TpA gets the next key from the local KM (16), sends the index inside the KxF header (17) and starts encrypting the incoming bit stream with the new key. Note that TpZ already had the key to use for decrypt the data, so it uses that immediately, and gets the next key from the local KM (18). Steps 16-18 repeat at every time interval until the lightpath is torn-down, when both agents use the KSID to terminate the association with the KMs.

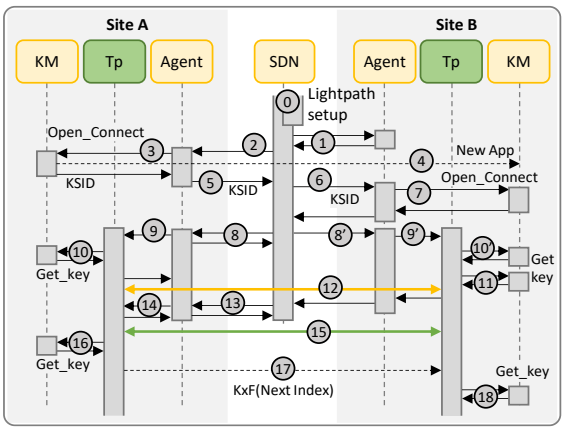

Fig. 2: Workflow of Demo A

In the second demonstration (*Demo B*), only one of the TPs is in a site covered by the QKD network. In this case, keys *k* can be retrieved from a local QRNG that provides a vendor-proprietary interface for key retrieval. Here, the KxF header is used for key exchange between the two TPs, a part of for synchronization purposes as in the previous demo. Fig. 1b presents the scenario, where the QRNG system is used.

As in Demo A, the workflow starts after the lightpath has been established over the optical network (labeled 0 in Fig. 3). The initial public key exchange is carried out and used to generate the particular KxF header pattern that will be used on the optical channel for synchronization between the two Tps (1-3). In addition, TpA retrieves a key from the local key server that will be used for data encryption (4). The retrieved key is sent to the SDN controller encrypted using the public key of TpB, together with the generated KxF header pattern, which are sent to TpB via its agent (6-7). Once TpB is able to get synchronized with TpA through the optical channel, it replies the agent, which in turn, replies the SDN controller. When TpA receives the confirmation that the encryption of the incoming bit stream can start (8,9), it requests a new key from the local key server (11), encrypts it using the previous retrieved key and sends it in the next KxF header (12). Steps 11-12 repeat at every time interval until the lightpath is torn-down.

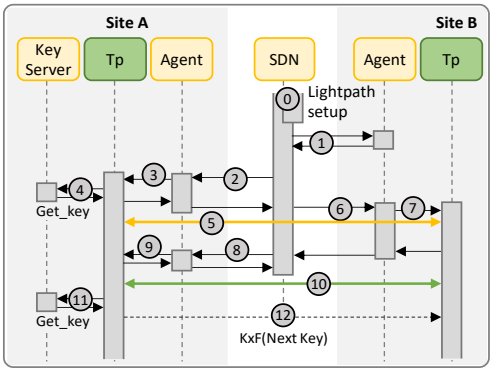

Fig. 3: Workflow of Demo B

For the demos, real QKD and QRNG systems will be used. Specifically, Demo A will rely on a QKD link using experimental HWDU continuous-variable QKD devices [4]. The link connects two Telefonica's facilities; it spans 15 km and it has 10.2 dB losses to provide secret key rate of 8.4 Kb/s through an ETSI GS QKD 004 interface. Regarding Demo B, a QuSIDE QRNG system will be used as quantum entropy source for the encryption key. The QRNG implements a proprietary phase-diffusion quantum random number generation technology and has embedded randomness metrology capabilities [5], [6] to produce very high quality random bits at 4 Gb/s, thus noticeably reducing key expansion. The system exposes a proprietary REST API interface to deliver the random bits.

A video streaming service will be used to demonstrate the connectivity and an eavesdropper will show data encryption. Iteration of the attendees with the system will be facilitated with a Web interface, so they can define the scenario to be demonstrated and change parameters.

References

[1] M. Iqbal *et al*., "LPsec: A Fast and Secure Cryptographic System for Optical Connections," JOCN, 2022.
[2] HORIZON-CL4-2022 "Agile ultra-low energy secure networks" (ALLEGRO) [On-line] https://www.allegro-he.eu/
[3] "Quantum Key Distribution (QKD); Application Interface," ETSI GS QKD 004 v.2.1.1, 2020.
[4] H. Brunner *et al*., "Demonstration of a switched CV-QKD network," EPJ Quantum Technology, 2023.
[5] A. Mitchell *et al*., "Strong experimental guarantees in ultrafast quantum random number generation," Physical Review, 2015.
[6] C. Abellán *et al*., "Generation of fresh and pure random numbers for loophole-free Bell tests," Physical review letters, 2015.