Distributed Key Relay: OSPF for Effective QKD

Youssouf Drif*, Intidhar Bedhief*, Symeon Chatzinotas*
*Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg
Kirchberg, Luxembourg
email: firstname.lastname@uni.lu

***Abstract*—With the advent of powerful quantum computers, an emerging quantum-resilient security mechanism within existing networks is crucial. Thus, Quantum Key Distribution (QKD), which generates symmetric secret keys with Information-Theoretical Security (ITS) for secure point-to-point communication, is present as a solution. Despite QKD's potential, its deployment is challenging due to significant investment and infrastructure requirements. Using state-of-the-art technology, the maximum distance between nodes is limited to hundreds of kilometers, necessitating multiple nodes to relay keys. For now, QKD Network (QKDN) relies on software-defined networking (SDN) and a central orchestrator, but this centralized approach has limitations. Hence, the main challenge is to select an efficient path between key relays to avoid communication failure.**

To address these challenges, a more distributed approach could be envisioned. In this paper, we propose to leverage one of the distributed routing protocols, OSPF, to redefine and improve the current architecture. We propose a novel QKDN-based OSPF (Open Shortest Path First) architecture for efficient key relay routing, offering a completely different dimension to the overall network management: we modify QKDN architecture to exploit the OSPF networks and present a QKD-enabled OSPF version capable of computing the best path for key relay, offering a cost-effective and scalable solution for widespread QKD adoption.

***Index Terms*—Quantum-Key Distribution, Key Relay, OSPF**

I. Introduction

With more potent than ever quantum computers and fast-growing capabilities, there is a better time to deploy quantum resilient security mechanisms within existing networks. Two candidates are under heavy investigation: Post-Quantum Cryptography (PQC), providing quantum-resistant asymmetric encryption, and Quantum Key Distribution (QKD), enabling Information-Theoretical Security (ITS) symmetric encryption [1]. Both technologies can also be used complementarily, as it is achieved today with asymmetric/symmetric encryption. However, deploying QKD technologies is a big challenge for operators, requiring tremendous investments and considerable upgrades to the current infrastructure, thus limiting and making it a significant limitation for large-scale deployment.

QKD is essential for the security of upcoming networks, as the cryptographic community is moving towards technologies merging PQC, QKD, and classical encryption as the basis of symmetric encryption. Therefore, deploying QKD and PQC incrementally within current networks is essential, gradually becoming the cornerstone of security technologies. QKD hardware is already available in the market, and a few actors have already built the first QKD commercial appliances to be deployed in networks and built QKD Networks (QKDN) on top of existing networks.

Considering that Standardization Organizations (SDOs) such as ITU [2] and ETSI [3] are working on standard solutions to deploy QKD. They defined QKD Networks (QKDNs) as standalone networks independent from existing ones in terms of their architecture, components, and protocols. QKDN orchestration primarily relies on SDN and a central orchestrator to relay keys between nodes across the network [4]. Indeed, given the current technology constraints and limitations, relaying keys between nodes is a sensitive task requiring careful planning, as all the QKDN depends on global key availability. The centralized approach has a few limitations: first, it requires quantum-safe communications between the SDN controller and each QKD node, which entails additional QKD links, thus increasing costs. Second, the non-standard SDN solution does not guarantee interoperability between different QKDN domains. Finally, it creates a single physical point of failure in the network for key relaying procedures, as replicating the controller implies replicating all its QKD links, which is not feasible in practice. These limitations could be mitigated by studying and adopting either a fully distributed or hybrid approach.

In this article, we take a network-centric approach and leverage the existing network infrastructure to deploy QKDN. Various Interior Gateway Protocols (IGPs) can constitute the basic brick of distributed QKDNs such as Intermediate System to Intermediate System (IS-IS), and Open Shortest Path First (OSPF). By extending this approach, other protocols could also be studied to interconnect various QKDNs and extend the capabilities of the QKD infrastructure. In this paper, we focus on the integration of IGPs within a single QKD domain, by leveraging OSPF [5] as the reference protocol for the QKDN integration. The work presented within this paper could then be refined and adapted to other IGPs, which, could also be used by QKDNs.

Our contribution is organized as follows. In the first part of the article, we describe in detail the cryptographic-oriented QKDN architecture and evaluate the various key relay methods to exchange keys on distant nodes. This step is critical to understanding the impact of the relay mechanisms on the network metrics. In the second part of the article, we focus on the integration of QKD in OSPF; we describe the modifications to bring to the QKDN architecture to exploit the OSPF networks and define a QKD-enabled OSPF version capable of computing the best path to relay keys between network nodes. QKD key relaying is a Quality of Service (QoS) routing problem with dynamic link costs. As such, we carefully modify the path calculation and internal OSPF tables

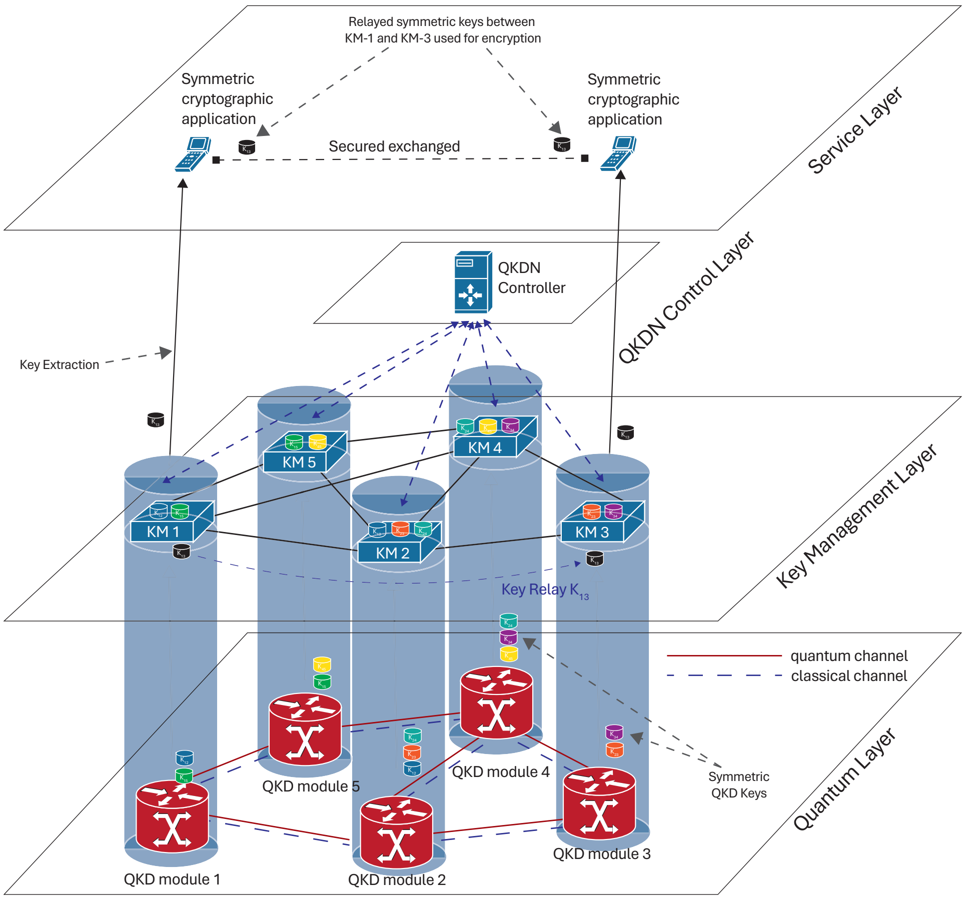


Fig. 1. QKD Networks Layer Split and Architecture [2], [3]

to solve the key relay problem within the QKDN domain. From our literature review, only Dianati et al. [6] took a similar approach, which proposed a full-stack QKDN with OSPF as a basis for routing keys in a "best-effort" mode. Significant modifications are brought to the OSPF protocol to compute multiple paths for load-balancing between links. Compared to their approach, where a single routing table is maintained for each router's interfaces, ours only requires maintaining a single routing table with the most suitable path for key routing at the router level, making it less complex and more memory efficient.

II. Quantum-Key Distribution Networks

A. QKD Technology

The main goal of QKD is to exchange symmetric data and ensure communication safety while exchanging these keys by relying on quantum physics. If someone eavesdrops on the communication, it introduces measurable anomalies that will be used to compute a security threshold for that communication by peers. Peers are connected point-to-point using dedicated optical fibers and execute QKD protocols [7] to exchange symmetric keys. The protocols require two distinct channels: the quantum channel to execute photon-related operations and a classical channel to exchange traditional data and compute the security threshold. The quantum channel requires optical connectivity, without any active component between peers, to avoid altering the photon states and increasing the Quantum Bit Error Rate (QBER) security threshold measurement. Unlike the quantum channel, the classical one does not require specific technology; thus, it can use any existing network if end-peers authenticate messages to synchronize their exchanges and compute the threshold to agree on keys (e.g., CHAP). This communication model needs to be replicated at a large scale to build a QKD network. Each node willing to exchange QKD keys should implement the protocols and deploy the associated infrastructure.

B. How to build QKD Networks?

Numerous QKD nodes connected using direct fibers must be deployed to extend the QKD approach and build large-scale QKD networks. Indeed, the current commercial QKD technologies are at most 200 km using single-modal fibers between two QKD nodes. Due to this strong constraint, various nodes within the network will have to act as Trusted Nodes (TNs) to relay quantum keys between distant nodes and create virtual QKD links between them. As stated previously, quantum communications are peer-to-peer based, exchanging local keys only accessible to direct peers. This results in the impossibility of TNs directly relaying quantum information between nodes. This task is then handled by another layer containing all the necessary tools to relay keys securely. The key relay procedure should retain the same level of security as the one achieved for the original keys. Each key pair in the overall network should, at the end, be uniquely identified. Each key pair will then be extracted and used for end-to-end encryption by applications on traditional IP networks. To achieve ITS within the network, the key's length must equal the size of the data to be sent. In practice, the XOR primitive can be used for highly sensitive and small amounts of data encryption. Still, for more significant data streams, QKD keys are often used directly as seeds for symmetric encryption algorithms (e.g., AES-GCM or ChaCha20) or as inputs for Key Derivation Functions (KDFs) to generate subsidiary keys for encryption algorithms. When used for encryption other than XOR, keys must be frequently rotated to retain the security properties of each encryption algorithm, which induces a constant Secure Key Consumption Rate (SKCR).

Multiple SDOs, such as ITU and ETSI, have worked on defining QKD network architecture and specifications; the global resulting architecture is depicted in Fig. 1. Overall, QKDN can be decomposed into four distinct layers: the quantum layer, the key management layer, the control layer, and the service layer.

- **Quantum Layer:** embeds QKD devices that implement the QKD protocols and physically connect QKD nodes using optical fibers for key exchange.
- **Key Management Layer:** embeds Key Managers (KM) responsible for secure key storage, key lifecycle management, executing key relay, and key-exposing service to applications. Virtual QKD links can be established using key relays, with distant nodes sharing and storing locally symmetric keys (e.g., link B-D). The KM should be physically co-located with the QKD device to communicate securely and safely. The underlying network infrastructure might differ from the one at the quantum layer. KM should be securely connected to form the network's Key Management System (KMS).
- **Control Layer:** embeds the network controller responsible for end-to-end network management and orchestration of the QKDN; it is connected to each QKD node and manages the overall network key distribution procedure.
- **Service Layer:** embeds the applications that request keys and encrypt user traffic end-to-end.

Although the QKDN architecture described above makes sense and is wholly aligned with their cryptographic goal, it requires significant investments, is complex to deploy, and is hard to maintain side-by-side with existing networks. Relying on Software-Defined Networks (SDN) and Networks Functions Virtualization (NFV) can lighten QKDN operational costs, but deploying and maintaining them remains a significant hurdle. In that sense, the existing network architecture and, more precisely, OSPF can accelerate QKDN deployment, solve the fundamental QKD relay problem, and reduce investment costs, which can remove the barriers to entry.

C. QKD Key Relay

The key relay procedure is fundamental to avoid key depletion on QKD links, maintain healthy behavior of QKDNs, and ensure secure communications between all nodes within the network. Without proper relaying, a QKDN can deplete its keys in minutes and become entirely off the grid until it regenerates after sufficient key exchanges and relays. Relaying key implies that at least one path in the quantum layer must exist between the distant peers and that each link on the path has stored sufficient keys for a secure relay. One distant node initiates the relay process and generates a random key using a True Random Number Generator (TRNG) (a TRNG retains ITS). It should securely send this key to the destination one. In that sense, multiple key relay methods [8] can be deployed within a QKDN, as shown in Fig 2. to relay keys between distant peers 1 and 2:

- **Hop-by-hop relay:** the key is relayed through nodes from the source to the destination peer and travels through all the nodes on the path (i.e., KM 1,2, and 3). Using successive XOR, the final node can retrieve the final key. This relay method can apply to any network topology, but every node in the path can retrieve the final key to be relayed. If one TN is compromised, the end-to-end communication will be compromised;
- **Central relay:** one node in the network is responsible for first collecting all the keys on the path (KM 4), XORing all the keys together, and sending the result to the destination node (KM 3), which can retrieve the final key. In that case, the destination node will be the only node capable of retrieving the final key. The topology is the drawback, which requires a full-mesh connectivity between nodes;
- **Direct relay:** the direct method is identical to the central one except that the final node receives all the keys and can retrieve the final key. It should be faster than the central one but requires full-mesh connectivity and additional links.

In addition to these methods, multi-path relaying can enhance the final key's security. Instead of relaying a single key, each path should relay a different key, and the final key will be the XOR combination of all the keys. This is extremely useful to mitigate the security risk of the hop-by-hop relay. If TRNGs are unavailable, it is also possible to use one of the keys already exchanged by the source node with its neighbor instead of generating a random key.

We conducted a detailed simulation of various key relay methods to evaluate their performance within a large-scale Quantum Key Distribution Network (QKDN) comprising 5,000 nodes and a central controller responsible for orchestrating the relay process. The key metrics assessed in this simulation—namely, the relay time (r) and the number of keys consumed to relay a single key (η)—are presented in Fig. 3.

As we foreseen, the direct and central relays have constant relay times compared to hop-by-hop, which depends on the path length. Direct and central relays are preferred to relay keys but require a secure full-mesh topology. The hop-by-hop method could be used in various cases, such as interconnection with external QKDNs. This also shows the delay in relaying a single key through the network for all three methods. Optimization could be applied, and multiple keys could be relayed in a single communication, which will still delay

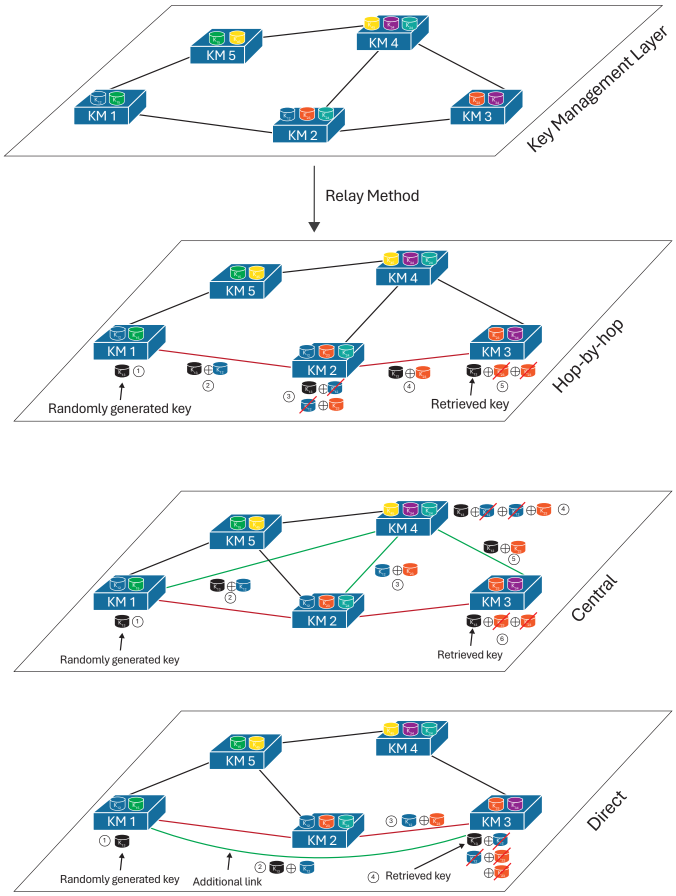


Fig. 2. Hop-by-hop, central and direct QKD Key Relay

the end-to-end encryption. It highlights the importance of proactively relaying and caching keys between nodes to serve distant users' requests quickly.

The first and most important step of these relay methods is to find a path between distant nodes with sufficient keys to execute the relay procedures. This is where OSPF can be efficiently integrated: it can provide the network topology and the best relay path and handle the key relay procedure.

III. How can QKD leverage OSPF?

Current routing inside Autonomous Systems (ASs) is handled by IGP such as IS-IS and OSPF. These protocols are designed to discover the network and find an efficient path between all network routers. OSPF is a link-state protocol in which each router sends updates regarding its links to other network routers. Every router in the network will receive these updates and will be able to build the complete network topology and compute the best path between itself and all routers. Multiple mechanisms are in place to ensure reliable exchanges between routers. The OSPF architecture is widespread and constitutes the cornerstone technology for the worldwide internet. It is reliable, efficient, secure, and, most importantly, scalable, suitable for large-scale networks. It also provides mechanisms to re-distribute routes from external networks towards the OSPF domains, which form the internet.

A. How is OSPF working?

The OSPF protocol is a distributed and link-state-based routing protocol where all routers flood the network with the state of their links to help other routers compute the best path with them and build routing tables. Any change in router links will trigger an update that will reach all other routers in the network. Routers exchange updates using Link-State Advertisements (LSAs) and are grouped by areas, as depicted in Fig. 4. LSA can be of different types depending on the update sent by routers. Some of the interesting LSA for QKD are listed below:

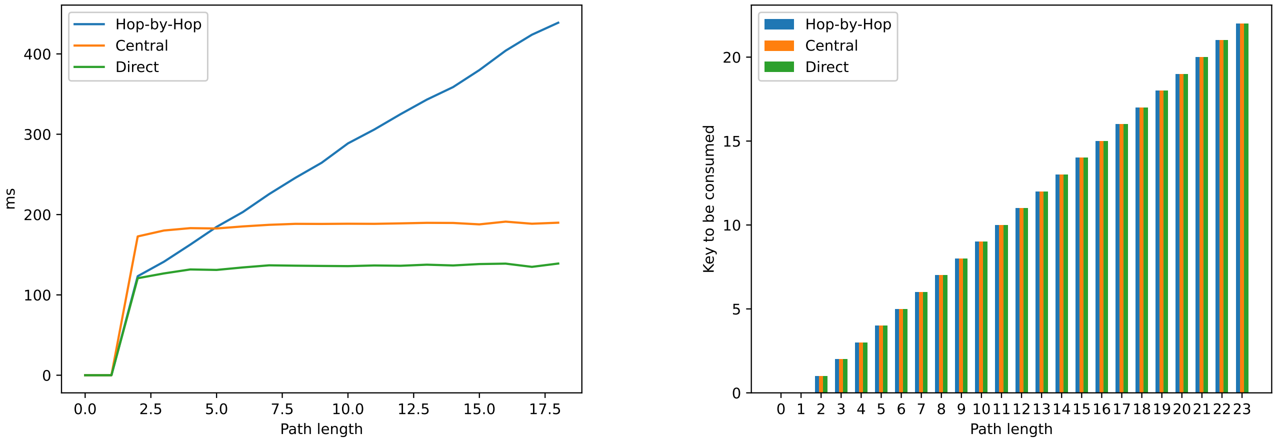
- **Type-1 LSA (Router) and Type-2 LSA (Network)** are used between routers of the same area to exchange link updates. Type-1 is used between peer-to-peer routers, while Type-2 is used for routers interconnected using multi-access networks.
- **Type-3 LSA (Summary)** is used by Area Border Router (ABR) to summarize all the routes within a single area and flood them into the next one (e.g., area 10 to area 20). This mechanism reduces the size of areas and enables more efficient routing.
- **Type-4 LSA (Summary ASBR) and Type-5 LSA (Autonomous System External)** are used to redistribute routes from external networks towards the OSPF network and enable global interconnection.
- **Type-9 LSA (OSPF Link Scope Opaque), Type-10 LSA (OSPF Area Scope Opaque), and Type-11 LSA (OSPF AS) [9]** are used to exchange non-standard proprietary information within the OSPF domain with different scopes. Type-9 exchanges proprietary information at the link level, Type-10 between all routers within an area, and Type-11 between all routers within the OSPF domain.

Each router stores LSAs in its own Link-State Database (LSDB) and uses this database to build the complete network topology and compute the best path between the router and all others. Each LSA update related to a link includes a cost used by the router Dijkstra algorithm to calculate the end-to-end shortest route. The cost is configured as the ratio between the reference bandwidth (default to 100 Mbits) and the interface bandwidth, which favors the links with higher bandwidth.

In summary, OSPF is a routing protocol widely used in today's network, which efficiently builds network topologies and enables end-to-end communications between routers. OSPF also presents other benefits, but we only presented the features that we feel are related and can be leveraged by QKDNs.

B. QKD-based OSPF Architecture

As we discussed in Section 2. QKD key depletion must be avoided at all costs, making the key relay a fundamental aspect of QKDN. Key relay is a routing problem where each QKD node must reach all TNs to relay keys. In that sense, OSPF can be used to build the QKDN topology, help discover all nodes, and provide the key relay feature. To integrate OSPF in the QKD Architecture, it is necessary to define the network layer, as shown in Fig. 4. We propose then to integrate OSPF within QKDN and leverage all the OSPF capabilities.



(a) Relay time r

(b) Keys to be consumed η

Fig. 3. QKD Key Relay Method Comparison

In QKD-based OSPF architecture, the OSPF network layer embeds all the quantum layer-related infrastructure with quantum and classical channels interconnecting QKD-enabled OSPF routers. These routers are then able to exchange QKD keys. The key management layer can then leverage the functionalities offered by this new network layer to delegate the key relay procedure and proactively relay keys. The key managers then can retrieve the keys, store them securely, and manage their lifecycle. At the same time, using OSPF also removes the need for a QKDN controller, which centralizes the network's control plane, thus offering a completely different dimension to the overall network management. The key management layer can directly offer services to the user in the service layer, which can request keys to implement quantum-resilient cryptographic communications.

1) Secure Key Rate Flooding: As mentioned in 3.A., OSPF instances exchange LSAs for advertising any modification on network links, and these updates could also propagate the number of keys stored on each link regularly. This information could be flooded through QKD-link State Advertisements (Q-LSAs), which are Type 9/10/11 LSAs while leaving Type 1/2/3 LSAs flooding only topology changes. Indeed, not all links within an OSPF domain can be "QKD-enabled." Only specific routers would perhaps implement this technology (e.g., cost limitations, use cases), which would leave the OSPF behavior unchanged. The QKD topology to build should also be carefully planned, as the LSA propagation will be affected by network topology and area types, which may result in discontinuities between QKD nodes (e.g., Non-Broadcast Multi-Access with no QKD link, point-to-point link with no QKD support, ABR without QKD links). All routers within the area/domain will receive these QKD-link State Advertisements (Q-LSA), update their neighbor table to indicate the OSPF router support QKD exchanges, and they will keep the related QKD-link states in memory next to the cost of each physical link. The rate at which these Q-LSAs will be flooded could be negotiated between routers when exchanging Hello packets. The rate could be a QKD-interval timer similar to the dead-interval, which should be the same to establish the full neighbor relationship. The timer value should be set according to the SKR, with lower timer values required for high SKRs.

2) Key Relay Routing: After receiving all initial LSAs from other routers, each node of the OSPF network can compute the best route to other nodes using the Dijkstra algorithm and populate its routing table accordingly. A similar process must occur to calculate the best path for QKD key relays. To successfully execute this process, we must take into account:

- The updated SKR of the link: The SKR depends on the quantum and classical channels, which vary greatly in an unstable network. This affects the total number of QKD keys stored on the link. The SKCR also varies depending on the applications requesting keys and will impact the final number of keys stored.
- Not all OSPF routers might have QKD-enabled links, which will impact the topology for path calculation (it will most likely reduce the number of nodes in the topology).

Given these constraints, each router must derive the QKD topology from the current OSPF topology, compute the best relay path, and update its tables with new QKD paths. Given the nature of information exchanged between routers, determining the QKD topology is immediate. However, calculating the path is more complex than for the "basic" routing of OSPF. The shortest path algorithm used by default could also be used in the QKD context as long as the "QKD-convergence" time reflects the actual changes in the network and the cost is well-defined. The cost must be minimized to favor a link, which could result in the ratio between a reference value and the number of keys stored on each link. However, it will not reflect the link's actual usage and depletion speed, which is a crucial metric in the network. It is imperative to reflect the depletion rate in the path calculation procedure; we, therefore, propose to include a three-color scheme complement to the decision process as depicted in Fig. 5:

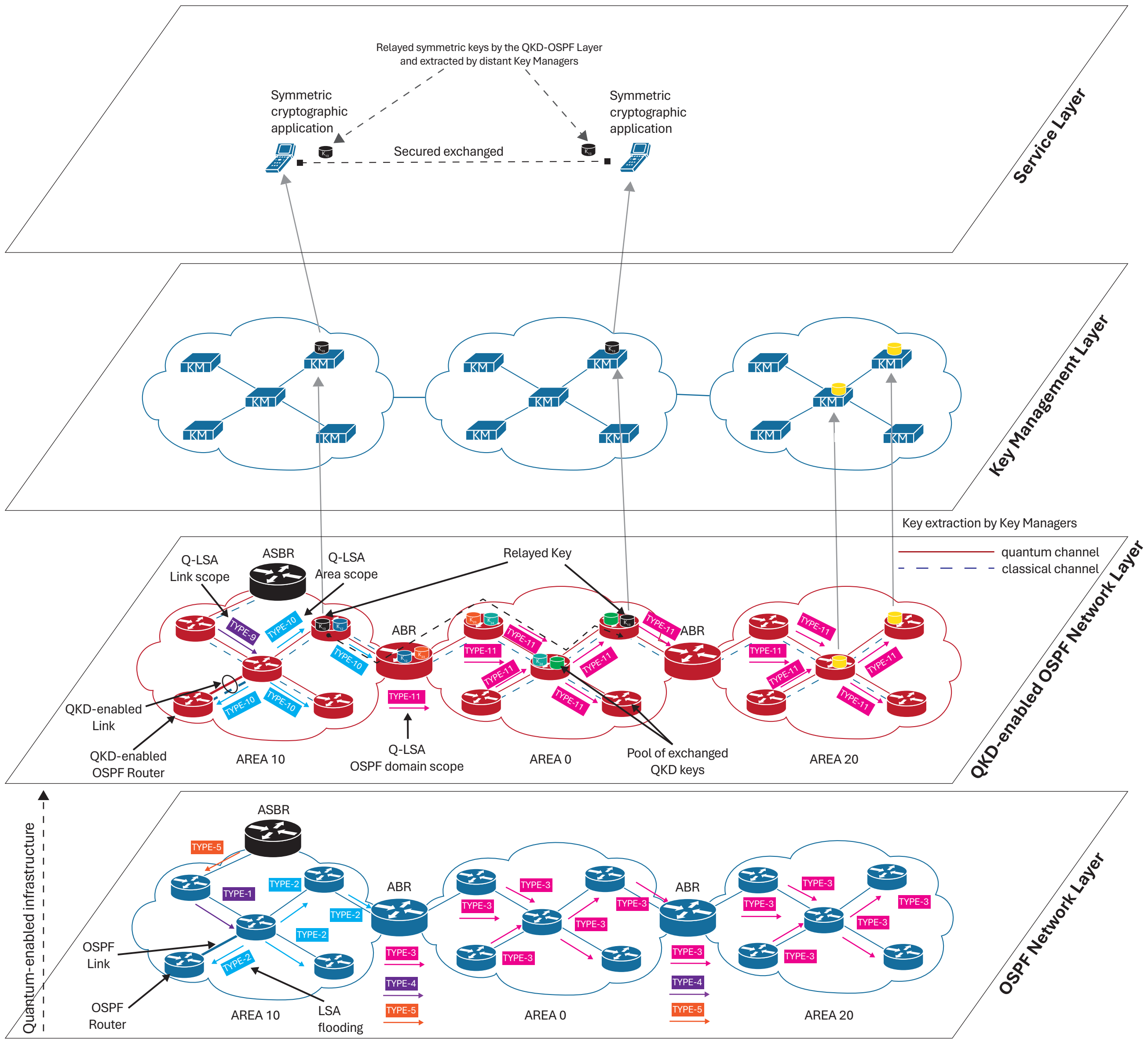


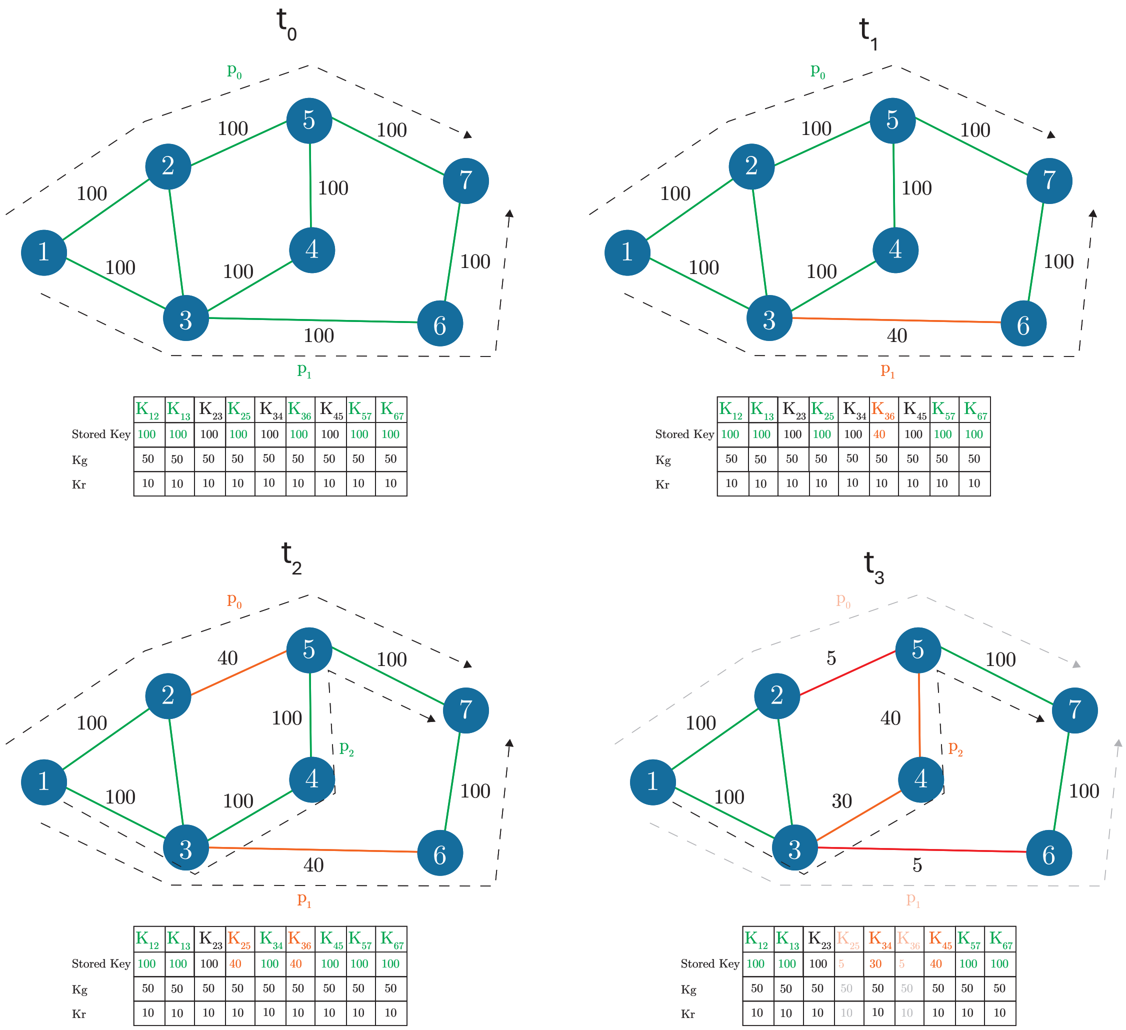
Fig. 4. QKD-enabled OSPF Network Layer

- **Green:** if the number of keys stored is above the "green" threshold K_g, the link is kept in the topology and could be used for the path calculation. In the example depicted in Fig 5., at t_0, p_0 and p_1 are the shortest paths with the most available number of keys to relay keys from node 1 to node 7;
- **Orange:** if the number of keys stored goes below the threshold K_g and the depletion rate increases, the link switches to the "orange" state. A backup path is computed to bypass the link. The link might still be used for path calculation if no backup path is available. The link will go back to the "green" state when the number of keys stored will go higher than K_g. This corresponds to the t_1 state where the number of stored K_{36} keys dropped below $K_g = 50$ making p_0 the most favorable path available;
- **Red:** If the number of keys stored goes below the critical threshold K_r, the link switches to the "red" state; the link is removed from the QKD topology and cannot be used for path calculation until it recovers sufficient keys stored to go beyond K_r. If there are no alternative paths, the QKD relay procedure will be blocked until the "red" link regenerates. In Fig. 5, this corresponds to the t_2 state where p_0 and p_1 switch to the red state, leaving only p_2 available for key relay.

This color scheme and path computation brings various benefits in the network:

- The traffic will be load-balanced between available links to minimize the risk that one link becomes unavailable.
- By tweaking the K_g and K_r values, it is possible to adjust the behavior of the network depending on the number of keys stored over time.
- The color scheme could be derived, and an intermediate color state can be added to support QoS. Another threshold K_q will be the value under which only traffic with

This article has been accepted for publication in IEEE Communications Standards Magazine. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MCOMSTD.2025.3572642



t_0	K_{12}	K_{13}	K_{23}	K_{25}	K_{34}	K_{36}	K_{45}	K_{57}	K_{67}
Stored Key	100	100	100	100	100	100	100	100	100
Kg	50	50	50	50	50	50	50	50	50
Kr	10	10	10	10	10	10	10	10	10

t_1	K_{12}	K_{13}	K_{23}	K_{25}	K_{34}	K_{36}	K_{45}	K_{57}	K_{67}
Stored Key	100	100	100	100	100	40	100	100	100
Kg	50	50	50	50	50	50	50	50	50
Kr	10	10	10	10	10	10	10	10	10

t_2	K_{12}	K_{13}	K_{23}	K_{25}	K_{34}	K_{36}	K_{45}	K_{57}	K_{67}
Stored Key	100	100	100	40	100	40	100	100	100
Kg	50	50	50	50	50	50	50	50	50
Kr	10	10	10	10	10	10	10	10	10

t_3	K_{12}	K_{13}	K_{23}	K_{25}	K_{34}	K_{36}	K_{45}	K_{57}	K_{67}
Stored Key	100	100	100	5	30	5	40	100	100
Kg	50	50	50	50	50	50	50	50	50
Kr	10	10	10	10	10	10	10	10	10

Fig. 5. Color Scheme Key Relay Routing with OSPF

reserve SKR can consume keys before reaching K_r.

- Although the computation is integrated into OSPF, it can also be deployed on a centralized version of the network with SDN if the controller is able to monitor the network, retrieve the status of each link, and program each switch.

The color scheme implementation also requires additional memory on routers to store the state of the network links at $(t-1)$, as it is done with the incremental SPF (iSPF) [10] optimization. Following path computation, routing tables could be updated with additional information related to QKD routes for the key relay procedure. It is worth noting that in the central relay method, all OSPF routers (classical or QKD-enabled) could be used as central nodes even if they don't have QKD links, which might be helpful for resource optimization.

C. Mutual Benefits

Integrating QKD and OSPF can provide additional advantages. Indeed, they can benefit from each other's features:

- **Failover mechanisms:** To ensure more reliable communication, fast rerouting could be deployed for OSPF, such as Loop-Free Alternative (LFA) or Remote-LFA, to reroute traffic in case of a node failure without recomputing the path. QKD-enabled nodes can benefit from this feature in case of complete QKD link failure, as they can "re-route" keys without re-computing the path.
- **Peer security:** OSPF embeds security mechanisms to secure peer-to-peer connectivity using simple password authentication or encryption (IPSec [11]). QKD keys can also be fed to the OSPF process to ensure all OSPF exchanges.
- **Network stability and convergence:** OSPF optimizations such as Fast Hello, LSA-throttling, LSA arrival, SPF throttling, and iSPF, when used together, ensure better network stability and considerably improve the network convergence time. All these optimizations can also be used in the context of QKD, where the SKR could vary considerably and create instabilities for QKD relaying.
- **Inter-domain routing:** external routes could be exchanged between other routing protocols that constitute the global internet, such as BGP [12] and OSPF. Assuming the feasibility studies and adaption of these protocols

to support QKD exchanges, QKD inter-domain routing will be enabled worldwide with very limited changes to bring to the current infrastructure.

IV. Conclusion

QKD networks are still under definition, and SDOs such as ETSI and ITU-T are actively working on their specification. Dedicated architecture has been defined as having numerous functional elements responsible for the overall QKDN functioning. These architectures come to solve the numerous QKDN challenges and, thus, are cryptographic-oriented. As QKD requires specific hardware, its large-scale deployment already requires a significant investment at the infrastructure level to enable QKD peer-to-peer exchanges. The existing network infrastructure and routing protocols can help implement the first version of QKDNs.

In this work, we have evaluated the impact of the relay procedure in the network, highlighting the critical need to reliably and efficiently implement these procedures. We propose to leverage the OSPF protocol, adapting it and the infrastructure with minimal modifications to act as a delegate for QKDN and provide the infrastructure with key relay service. We solve the routing problem for QKD keys, allowing the upper cryptographic layers to be supported by a robust infrastructure, ensuring minimal cost for operators to deploy QKD, which can provide QKD services and considerably enhances the security of their internal network. With the according level of implication, our work could also be replicated using other IGPs such as IS-IS, as the behavior of the protocol is similar to OSPF with differences in path calculation and control messages.

Yet, this approach still has its limitations. First, the relay path computations rely on the traditional Dijkstra algorithm coupled with the three-color scheme topology. While this computation is straightforward, it could be significantly improved (e.g., ML algorithm) at the cost of increased complexity on each router. Second, the regular flooding of Q-LSAs might conflict with LSAs and impact the performance of the traditional OSPF routing process. Dedicated timers could be defined to reduce the overlap between these LSAs as much as possible. Finally, it does not reduce the investment cost of deploying QKD hardware within each router or the cost of securely storing exchanged keys.

As part of our future research directions, we are investigating the performance of SDN compared to the OSPF approach within the context of QKDN. We have started to explore this by benchmarking both methods, subjecting them to various cyberattacks to assess their robustness in ensuring service continuity, data security, and integrity. Our QKD-OSPF implementation builds on the FRRouting project, while our SDN implementation leverages Ryu and Open vSwitch.

Acknowledgment

The results of this research paper were supported by the LUQCIA and LuxQCI infrastructures, as well as the XTRUST-6G project. We gratefully acknowledge the financial support from the European Union's Recovery and Resilience Facility under the NextGenerationEU initiative for LUQCIA, and the Digital Europe Programme under Grant Agreement number 101091508 for Lux4QCI. XTRUST-6G is co-funded by the European Union.

References

[1] L. Garms, T. K. Paraïso, N. Hanley, A. Khalid, C. Rafferty, J. Grant, J. Newman, A. J. Shields, C. Cid, and M. O'Neill, "Experimental Integration of Quantum Key Distribution and Post-Quantum Cryptography in a Hybrid Quantum-Safe Cryptosystem," *Advanced Quantum Technologies*, vol. 7, no. 4, p. 2300304, 2024.
[2] ITU-T, "Quantum key distribution networks – Functional architecture - Corrigendum 1," International Telecommunication Union, Tech. Rep. Recommendation ITU-T Y.3802 - Corrigendum 1, Apr. 2021.
[3] ETSI, "Quantum Key Distribution (QKD); Components and Internal Interfaces," European Telecommunications Standards Institute, Group Report ETSI GR QKD 003, Mar. 2018.
[4] H. Wang, Y. Zhao, and A. Nag, "Quantum-Key-Distribution (QKD) Networks Enabled by Software-Defined Networks (SDN)," *Applied Sciences*, vol. 9, no. 10, p. 2081, Jan. 2019.
[5] J. Moy, "OSPF Version 2," Internet Engineering Task Force, Request for Comments RFC 2328, Apr. 1998.
[6] M. Dianati and R. Alleaume, "Architecture of the Secoqc Quantum Key Distribution network," in *2007 First International Conference on Quantum, Nano, and Micro Technologies (ICQNM'07)*. Guadeloupe, French Caribbean: IEEE, Jan. 2007, pp. 13–13.
[7] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theoretical Computer Science*, vol. 560, pp. 7–11, Dec. 2014.
[8] ITU-T, "Quantum key distribution networks – Key management," International Telecommunication Union, Tech. Rep. Recommendation ITU-T Y.3803, Dec. 2020.
[9] R. Coltun, A. D. Zinin, I. Bryskin, and L. Berger, "The OSPF Opaque LSA Option," Internet Engineering Task Force, Request for Comments RFC 5250, Jul. 2008.
[10] Cisco Systems, "OSPF Incremental SPF," Tech. Rep., 2024.
[11] D. Ferguson, A. Lindem, and J. Moy, "OSPF for IPv6," Internet Engineering Task Force, Request for Comments RFC 5340, Jul. 2008.
[12] Y. Rekhter, S. Hares, and T. Li, "A Border Gateway Protocol 4 (BGP-4)," Internet Engineering Task Force, Request for Comments RFC 4271, Jan. 2006.

V. Biography

Dr. Youssouf Drif (PhD) received his PhD degree from the Federal University of Toulouse Midi-Pyrénées (France) in 2022. His research interests are SDN and NFV for Satcom and, more precisely, in using these technologies for satellite integration into mobile networks. Youssouf joined the Signal Processing & Satellite Communications research group, SIGCOM, headed by Prof. Symeon Chatzinotas. He has since improved his knowledge in networking and extended his range of activities to 6G-NTN and Quantum Technologies. He has been involved in many projects related to mobile and quantum networks and built the HybridNet Lab to experiment with the orchestration of heterogeneous networks. He is actively engaged in the LUQCIA and LUX4QCI projects with the main objective of deploying QKD in Luxembourg.

Dr. Intidhar Bedhief (PhD) received her PhD degree from the University of Tunis El Manar, Tunisia, in 2023. Her research interests are IoT, SDN, Network Virtualization, and Network Orchestration. More precisely, applying the network softwarization technique to set up various approaches to improve the management and control of IoT networks. Intidhar joined the Signal Processing & Satellite Communications research group, SIGCOM, headed by Prof. Symeon Chatzinotas.

Prof. Symeon Chatzinotas (MEng, MSc, PhD, FIEEE) received the M.Eng. degree in telecommunications from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2003, and the M.Sc. and Ph.D. degrees in electronic engineering from the University of Surrey, Surrey, U.K., in 2006 and 2009, respectively. He is currently a Full Professor, the Chief Scientist I, and the Co-Head of the SIGCOM Research Group, SnT, University of Luxembourg. In the past, he has been a Visiting Professor with the University of Parma, Italy. He was involved in numerous research and development projects of the National Center for Scientific Research Demokritos, the Center of Research and Technology Hellas, and the Center of Communication Systems Research, University of Surrey. He has (co-)authored more than 400 technical articles in refereed international journals, conferences, and scientific books. He was a co-recipient of the 2014 IEEE Distinguished Contributions to Satellite Communications Award, the CROWNCOM 2015 Best Paper Award, and the 2018 EURASIC JWCN Best Paper Award. He is also on the editorial board of the IEEE Open Journal of Vehicular Technology and the International Journal of Satellite Communications and Networking.