Chapter 7
Secure Communication in the Quantum Era: (Group) Key Establishment

Christian Colombo, María Isabel González Vasco, Rainer Steinwandt, and Pavol Zajac

Abstract This paper gives a brief overview of NATO SPS project G5448. This project, which involves partners from four countries, focuses on the task of establishing a cryptographic key among a group of users over a (classic) public communication network. It is assumed that the adversary may eventually obtain access to quantum computers, i. e., standard cryptographic hardness assumptions like factoring integers being asymptotically hard may be invalidated. Next to laying out the basic project structure, we describe some of the work that has already been done since the beginning of the project, including in particular some of the work done by students.

Keywords Quantum computing · Cryptography · Runtime verification · Group key establishment

7.1 Introduction

At this point in time, quantum computing is widely considered more than a mere distant possibility. Substantial resources areds currently invested into bringing this technology to the level where it can be commercially used. Unsurprisingly,

C. Colombo
University of Malta, Msida, Malta
e-mail: christian.colombo@um.edu.mt

M. I. González Vasco
Universidad Rey Juan Carlos, Móstoles, Spain
e-mail: mariaisabel.vasco@urjc.es

R. Steinwandt (✉)
Florida Atlantic University, Boca Raton, FL, USA
e-mail: rsteinwa@fau.edu

P. Zajac
Slovak University of Technology in Bratislava, Bratislava, Slovakia
e-mail: pavol.zajac@stuba.sk

C. Palestini (ed.), *Advanced Technologies for Security Applications*, NATO Science for Peace and Security Series B: Physics and Biophysics,
https://doi.org/10.1007/978-94-024-2021-0_7

the implications of large-scale quantum computers on the security of today's communication networks receive a lot of attention at this point: From Shor's seminal work [20], we know that a large-scale quantum computer has the ability to invalidate assumptions that underlie widely deployed cryptographic solutions and to render these protocols insecure. Several popular cryptographic primitives for digital signatures and key establishment can no longer be considered secure in such a quantum era. With a large-scale quantum computer, secret keys in these protocols can be easily recovered from public data alone.

A particularly prominent effort to address this uncomfortable situation is NIST's ongoing effort to standardize some fundamental post-quantum cryptographic schemes. The process is remarkably complex in that there are no obvious drop-in replacements for currently deployed schemes. Even in the current (Round 2) stage of this effort, there is still quite some movement in the details of the proposed schemes. Our NATO SPS project G5448 already helped to establish a "PQC Wiki," which gives easy access to the current status of the considered candidates [16].

Overarching goal of this NATO SPS project is to provide a robust solution for (group) key establishment in the quantum era. In other words, we tackle the task that two or more parties want to establish a high-entropy secret over a public, insecure, communication network, and the adversary must be expected to invoke a quantum computer to undermine the security of the protocol. The project's scope includes the design of an authenticated group key establishment protocol and its theoretical analysis, but extends beyond this: to improve resistance against implementation-level attacks, we want to leverage runtime verification. In this way, we can add protections at execution time of a protocol and go beyond what is covered in a standard cryptographic protocol analysis.

7.2 Project Structure and Activities

To provide all necessary expertise, four different partners are involved in the research.

7.2.1 Project Partners

7.2.1.1 Florida Atlantic University (US)

Here, Rainer Steinwandt's team offers the needed expertise in using quantum computers for cryptanalysis. A member of the this team (Edoardo Persichetti) is a co-author of multiple active proposals in NIST's current post-quantum standardization effort, so that it can be ensured that state-of-the-art cryptographic tools are used.

7.2.1.2 Slovak University of Technology in Bratislava (SK)

Otokar Grošek serves as NPD for this project, and his team brings many years of experience in the secure implementation of cryptographic schemes to the project. The Slovak partner can in particular help with the detection (and prevention) of side-channel attacks, e.g., based on timing behavior or power consumption of an implementation.

7.2.1.3 University of Malta (MT)

Christian Colombo and his team bring extensive expertise in runtime verification to the project. Once the cryptographic protocol details have been determined, the Malta partner will take the lead in specifying and deriving a software implementation that offers robust security guarantees against implementation-level attacks at runtime.

7.2.1.4 Universidad Rey Juan Carlos (ES)

Under the leadership of María Isabel González Vasco, this project partner takes the lead in the protocol design phase. Owing to a strong track record in group key establishment, it is ensured that state-of-the-art security models and solutions are leveraged. The Spanish partner is also communicating with an end user, helping to ensure that practical needs are met.

7.2.2 Initial Work

In the initial project phase, we arranged for several in-person meetings among all project teams and of subsets of the project teams – the implementation specialists from Malta and Slovakia met, and the partners from Spain and the US met to work on protocol design and the security model. Both on the theoretical/conceptual side (with the partners in Spain and in the US taking the lead) and on the implementation side (with the partners in Malta and Slovakia taking the lead), the project has been active.

7.2.2.1 Theoretical/Conceptual Thrust

Much effort went into laying foundations – clarifying the security model and understanding classical and quantum resources for attacking relevant cryptographic primitives. A presentation on a Grover-based key search for AES at *Quantum Resource Estimation 2019* gives an example of our work in this line [13]. For the cryptographic tools to be used in our protocol solution, our focus is on code-

and lattice-based approaches, and this is reflected in the papers published in the project so far [14]. While a signature-based authentication would conceptually be the simplest approach to ensure authentication in a group-key establishment protocol, especially in a post-quantum scenario, such an approach comes with some (performance) challenges. Consequently, we looked in the initial project phase also at a different authentication mechanism, using passwords. It is our aim to use, whenever possible, non-expensive cryptographic primitives (e.g., message authentication codes and key encapsulation mechanisms), focusing on those whose resilience to quantum-attacks is better understood.

7.2.2.2 Implementation Thrust

From the more practical side, an initial design of a runtime verification setup for security protocol implementation has been drawn up and experiments are underway to validate the concept. A position paper is being drafted in this direction with the aim of disseminating the idea and getting feedback. We also involved students in our research, which has the beneficial side effect of increasing awareness of the NATO SPS Programme. We start the discussion of our practical efforts by highlighting some results of relevant student projects.

Post-quantum Digital Signatures on Android Phones

The transfer to post-quantum cryptography is also an engineering challenge that requires to adapt the used cryptographic infrastructure to new parameters and constraints enforced by post-quantum algorithms. The MS thesis [22] adapts currently available APIs and libraries to implement an application that can be used to sign (PDF) documents on Android devices with the post-quantum algorithm SPHINCS [4].

The main challenge was the key management, as the standard Android APIs only accommodate classical algorithms such as RSA. We have tested an experimental application on a Samsung Galaxy S10+ phone, and compared the signature and verification times with standard RSA-3076 (with equivalent pre-quantum security level as SPHINCS-256, which we used). While verification times are similar (in milliseconds), the SPHINCS-256 signature algorithm is relatively slow (0.5 s, compared to 5 ms of an RSA signature). However, from the user perspective, 0.5 s is still below noticeable delay. The main problem is SPHINCS-256's signature size of 41 kB (compared to 0.4 kB of RSA). This is partially solved in some proposed post-quantum candidates in NIST's standardization effort, but unfortunately most of the candidates still lack implementations in Android libraries.

Authenticating Ephemeral Post-quantum Public Keys with a Merkle Tree

Client-server communication is commonly initiated using public-key cryptography, and *forward secrecy* is a popular security requirement: old sessions are not to be compromised when long term secrets are revealed. This can be accomplished by using ephemeral (single-use) key pairs. Ephemeral keys are also useful to prevent some types of statistical attacks on post-quantum cryptosystems such as [7].

When client-server communication is initiated, typically at least server-side public keys need to be authenticated to prevent man-in-the-middle attacks. Unlike standard RSA/DSA/ECDSA signature schemes, post-quantum schemes have often large public keys and/or large signatures. Adapting specific post-quantum signatures can thus significantly slow down the protocol execution and consume resources on constrained devices.

In the MS thesis [15] a candidate for a quantum-safe key authentication proposed by one of the authors [23] was implemented and tested. In this system, a set of ephemeral key pairs[1] is generated in advance in a secure device on the server. A hash tree is constructed (see Fig. 7.1), and the root of the hash tree serves as a (short) public key for verifying authenticity of each key in the tree structure. A key is signed by providing a valid signature path from the leaf (containing the

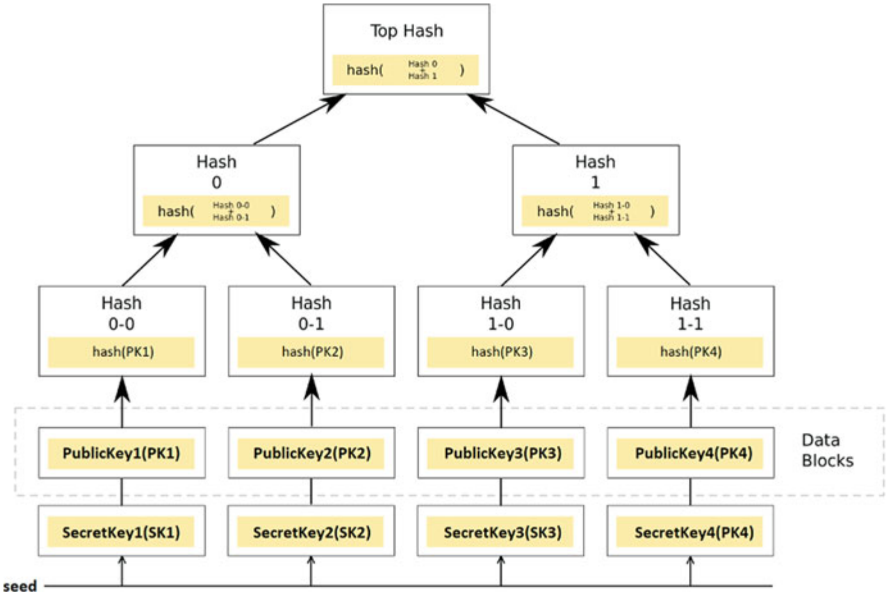

Fig. 7.1 Tree authenticated public keys

[1] For the tests, BIKE [1] was used.

authenticated ephemeral key) to the root of the tree. Experiments show that both signing and verification of ephemeral keys is fast (microseconds per keypair), and the signature size can be relatively short (up to 1 kB, depending on the number of pre-authenticated keys). In our stress test, a standard notebook was able to handle a data structure with 32 million key pairs. This suffices for the communication needs of a standard intra-net application. However, a more complex solution is required to cover the needs of large internet sites.

Systems Based on QC-MDPC and QC-LDPC Codes

Low and medium density parity check (LDPC/MDPC) codes are codes based on sparse parity-check matrices that allow to repair a large number of transmission errors. Structured quasi-cyclic (QC) versions of these codes have a very compact representation that makes them attractive for use in various post-quantum cryptosystems [2]. Classical decoding for LDPC and MDPC codes is based on bit-flipping. The main disadvantage for cryptographic use is its probabilistic nature, as the algorithm does not have an assured success. There are various attacks on QC-MDPC and QC-LDPC code-based systems that exploit the correlation between decoding failure rate and a structure of the secret key [6–8].

The MS thesis [12] successfully tested all proposed attacks on QC-MDPC code based systems in practice. Furthermore, unlike some theoretical predictions, these attacks were confirmed to work against QC-LDPC systems as well. On the other hand, the MS thesis [3] focused on defending these types of cryptosystems. It was experimentally verified that the Miladinovic–Fossorier decoding algorithm can be parametrized in such a way that the decoding failure rate is not correlated with the secret parameters. These algorithms and selected others were added to our BitPunch library [11].

Deeper Understanding of Post-quantum Tools and Resources

While it is not unusual that post-quantum cryptographic proposals lack a formal security evaluation within the theoretical framework known as *provable security*, it is indeed helpful to understand formally the reasons behind successful cryptanalysis of such constructions. This, in particular, helps identify the critical points to fend off in a new design.

In [10] we report on the Walnut Digital Signature Scheme (*WalnutDSA*). This paper is the first contribution of our Phd. student Jose I. Escribano; it details the original proposal and describes the main attacks that have been presented against this construction. Furthermore we discuss several modifications of the proposal, currently under debate, towards a secure implementation of this signature scheme.

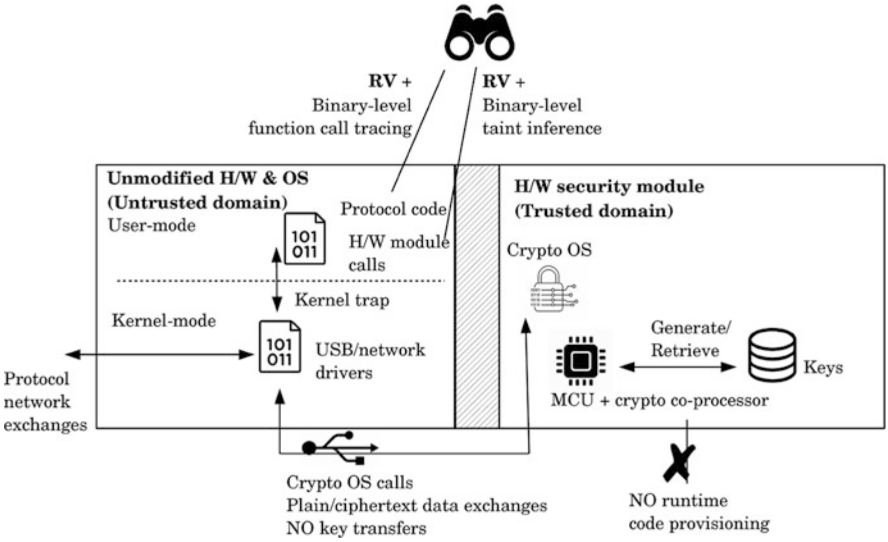

Fig. 7.2 RV-centric comprehensive security for cryptographic protocol implementations (USB stick example)

Runtime Verification (RV)

Figure 7.2 shows a proposed RV-centric solution of protocol implementation security. This setup requires no special hardware or OS modifications, mitigates threats related to hardware issues, including side channel attacks on ciphers, while keeping runtime overheads to a minimum. The primary components of this design are two RV monitors executing within the untrusted domain and a hardware security module (HSM) providing the trusted domain of the trusted execution environment (TEE). The chosen example HSM is a USB stick, comprising a micro-controller (MCU), a crypto co-processor providing h/w cipher acceleration and true random number generation (TRNG), as well as flash memory to store long term keys. In this manner, the cryptographic primitive and key management code are kept out of reach of malware that can potentially infect the OS and applications inside the untrusted domain. The co-processor in turn can be chosen to be one that has undergone extensive side-channel security analysis, thus mitigating the remaining low-level hardware-related threats (e.g., [5]). The *Crypto OS* is executed by the MCU, exposing communication and access control interfaces to be utilized for HSM session negotiation by the protocol executing inside the untrusted domain, after which a cryptographic service interface becomes available (e.g., PKCS#11). In a typical TEE fashion, cryptographic keys never leave the HSM. The proposed setup forgoes dealing with the verification of runtime provisioned code since the cryptographic services offered by the HSM are expected to remain fixed for long periods.

The RV monitors verify correct implementation of protocol steps and inspect all interactions with the hardware module, both of which happen through the network and external bus OS drivers respectively. Verifying protocol correctness leverages the high-level flavors of RV, checking that the network exchanges follow the protocol-defined sequence and that the correct decisions are taken following protocol verification steps (e.g., digital certificate verifications). Inspecting interactions with the HSM, on the other hand, requires a low-level approach similar to the Frama-C/SecureFlow plug-in. In both cases the monitors are proposed to operate at the binary (compiled code) level. The binary level provides opportunities to secure third-party protocol implementations, as well as optimized instrumentation applied directly at the machine instructions level. Overall, binary instrumentation is a widely-adopted technique in the domain of software security, including the availability of widely used frameworks (e.g., Frida [17]) that simplify tool development. The higher-level RV monitor is tasked with monitoring protocol steps and as such instrumentation based on library function hooking suffices. This kind of instrumentation is possible to deploy with minimal overheads.

In contrast the lower-level RV monitor has to rely on monitoring information flows, specifically, untrusted flows [18]. The main limitation is presented by impractical overheads [9]. Our proposed solution concerns inferring, rather than tracking, taint [19] and which takes a black-box approach to taint flow tracking, trading off between accuracy and efficiency. This method only tracks data flows at sources/sinks and then applies approximate matching in order to decide whether tainted data has propagated all the way in-between. With slowdowns averaging only 0.035× for fully-fledged web applications, this approach seems promising. In fact we propose that this approach requires the same library function hooking type of instrumentation as with the higher-level RV monitor. Crypto OS calls may be considered both taint sources and sinks. In the case of data flowing into Crypto OS call arguments originating from suspicious sites, e.g., network input, interprocess communication or dynamically generated code, the Crypto OS calls present the sinks. All these scenarios are candidates of malicious interactions with the HSM. In the reverse direction, whenever data flows resulting from Crypto OS call execution and that end up at the same previously suspicious sites, the calls present the tainted sources while the suspicious sites present the sinks. In this case these are scenarios of malicious interactions targeting leaks of cryptographic keys/secrets, timing information or outright plaintext data leaks. Whichever the direction of the tainted flows, the same approximate matching operators can be applied between the arguments/return values of the sources/sinks.

A nonce-based remote-code attestation, e.g., [21], can optionally close the loop of trust, executed by the Crypto OS, with the HSM performing the tasks intended to be executed by a trusted platform module in such protocols.

7.3 Outlook and Opportunities for Cross-Project Collaboration

The decision on a target protocol is expected to be made as planned. Then the transition from theoretical protocol analysis to a software implementation and implementation-specific security aspects is to be tackled next, and we expect the project to proceed as planned.

Some of the research in this project may well be of interest when combining two-party solutions for *quantum key distribution* to larger networks: While the focus in our project is on classical communication networks, it is common to use a two-party solution as a primitive, and one could potentially explore to what extent non-classical (quantum) solutions could be used here as well, possibly leading to a hybrid protocol for group key establishment. Moreover, it is common to use a classical authenticated channel in quantum key distribution, and when transitioning to group communication, the solutions considered here might be of interest to establish such authenticated channels.

Acknowledgments This work has been supported through NATO SPS Project G5448.

References

1. Aragon N, Barreto P, Bettaieb S, Bidoux L, Blazy O, Deneuville J-C, Gaborit P, Gueron S, Guneysu T, Melchor CA et al (2017) BIKE: Bit Flipping Key Encapsulation. Submission to the NIST post quantum standardization process
2. Baldi M (2014) QC-LDPC code-based cryptography. Springer Science & Business, Cham
3. Baraniak T (2019) Decoding of QC-MDPC codes. Master's thesis, Slovak University of Technology in Bratislava
4. Bernstein DJ, Hopwood D, Hülsing A, Lange T, Niederhagen R, Papachristodoulou L, Schneider M, Schwabe P, Wilcox-O'Hearn Z (2015) SPHINCS: practical stateless hash-based signatures. In: Annual international conference on the theory and applications of cryptographic techniques, pp 368–397
5. Bollo M, Carelli A, Di Carlo S, Prinetto P (2017) Side-channel analysis of SEcube^{TM} platform. In: 2017 IEEE east-west design & test symposium (EWDTS), pp 1–5
6. Eaton E, Lequesne M, Parent A, Sendrier N (2018) QC-MDPC: a timing attack and a CCA2 KEM. In: International conference on post-quantum cryptography, pp 47–76
7. Fabšič T, Hromada V, Stankovski P, Zajac P, Guo Q, Johansson T (2017) A reaction attack on the QC-LDPC McEliece cryptosystem. In: International workshop on post-quantum cryptography, pp 51–68
8. Guo Q, Johansson T, Stankovski P (2016) A key recovery attack on MDPC with CCA security using decoding errors. In: International conference on the theory and application of cryptology and information security, pp 789–815
9. Jee K, Portokalidis G, Kemerlis VP, Ghosh S, August DI, Keromytis AD (2012) A general approach for efficiently accelerating software-based dynamic data flow tracking on commodity hardware. In: NDSS
10. Pablos JIE, María Isabel González Vasco MEM, del Pozo ALP (2019) The cracking of WalnutDSA: a survey. Symmetry 11:9

11. Klein M (2016) Side channels in SW implementation of the McEliece PKC. Infocommun J 8:10–16
12. Kováč J (2019) Reaction attack on the QC-MDPC McEliece cryptosystem. Master's thesis, Slovak University of Technology in Bratislava
13. Langenberg B, Pham H, Steinwandt R (2019) Reducing the cost of implementing AES as a quantum circuit. Presentation at Quantum Resource Estimation QRE 2019, slides available at https://www.quantumresource.org/pdfs/pham.pdf
14. Marak P (2019) Secure communication in the quantum era. http://re-search.info/node/27. Web site for NATO SPS Project G5448. Section on *publications and project dissemination*
15. Novotný M (2019) Implementation of the experimental post-quantum protocol. Master's thesis, Slovak University of Technology in Bratislava
16. Persichetti E, Bai S, Karabina K, Ngo T, Steinwandt R, Catalano Gonzaga di Cirella M (2019) PQC WIKI. A platform for NIST post-quantum cryptography standardization. https://pqc-wiki.fau.edu
17. Ravnås OAV (2019) FЯIDA. Dynamic instrumentation toolkit for developers, reverse-engineers, and security researchers. https://frida.re/
18. Schwartz EJ, Avgerinos T, Brumley D (2010) All you ever wanted to know about dynamic taint analysis and forward symbolic execution (but might have been afraid to ask). In: 2010 IEEE symposium on security and privacy (SP), pp 317–331
19. Sekar R (2009) An efficient black-box technique for defeating web application attacks. In: NDSS
20. Shor P (1997) Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J Comput 26(5):1484–1509
21. Stumpf F, Tafreschi O, Röder P, Eckert C et al (2006) A robust integrity reporting protocol for remote attestation. In: Second workshop on advances in trusted computing (WATC'06 Fall), pp 25–36
22. Pernický Ľ (2019) Post-quantum cryptography on Android. Master's thesis, Slovak University of Technology in Bratislava
23. Zajac P (2019) Tree authenticated ephemeral keys, Cryptology ePrint Archive, Report 2019/921. https://eprint.iacr.org/2019/921