Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 51-6 (2018) 462–467

# On Feasibility of Post-Quantum Cryptography on Small Devices

**Lukas Malina Lucie Popelova Petr Dzurenda Jan Hajny Zdenek Martinasek**

*Brno University of Technology, Technicka 12, Brno, Czech Republic (e-mail: malina@feec.vutbr.cz, xpopel19@stud.feec.vutbr.cz, dzurenda@feec.vutbr.cz, hajny@feec.vutbr.cz, martinasek@feec.vutbr.cz)*

**Abstract:** In this work, we investigate the feasibility of post-quantum cryptography in small and constrained devices such as those used for mobile and Internet-of-Things networks. We describe our experimental post-quantum cryptography implementations on small devices with different platforms. Then, we present and compare the performance results of chosen post-quantum key exchange schemes and their message sizes on selected ARM devices. In addition, we discuss the perspective types of post-quantum cryptographic schemes for various IoT systems with different requirements.



Keywords: Applied Cryptography, Constrained Devices, Efficient Evaluation, Embedded Systems, IoT Systems, Post-Quantum Cryptography, Security

## 1. INTRODUCTION

Common public key cryptosystems are usually based on the hardness of the Integer Factoring Problem (IFP), the Discrete Logarithm Problem (DLP), or its elliptic curve variant, the ECDLP. Nevertheless, all these problems could be effectively solved with a quantum computer by the Shor's algorithm, and common key exchange and digital signature schemes risk of being broken. Therefore, Post-Quantum Cryptography (PQC) schemes arise in order to be resist to the quantum computers. There are many designed PQC schemes that have been already implemented and used in ICT. For example, the New Hope key exchange scheme has been tested in the Google Chrome Canary web browser. Further, the quantum-resistant cryptographic library called liboqs has been integrated into the openssl library. Finally, National Institute of Standards and Technology (NIST) has released a call for proposals in order to solicit, evaluate, and standardize one or more post-quantum cryptosystems. The successful submissions of quantum-resistant public key encryption algorithms, key agreement mechanisms, and digital signature schemes could potentially replace the currently used cryptosystems such as RSA. The deadline for the submission of PQC schemes from various organizations and individuals was November 2017 and round-1 submissions were outlined. NIST estimates that a draft standard with finally chosen schemes will be available between 2023 - 2025. Five submitted schemes such as NTRU, New Hope, McBits, SIDH and Frodo are also investigated in this paper.

In general, asymmetric PQC schemes and also some Public Key Cryptographic (PKC) schemes are considered as not very appropriate for ICT systems that employ constrained devices due to many cycles during their basic operations (i.e. sign, verify, key exchange, encrypt/decrypt) and sizable parameters, ciphertexts or signatures leading to high memory usage and cryptographic overhead. In addition, some asymmetric PQC schemes are more memory or computational expensive than common PKC schemes such as ECDSA and RSA. In this paper, we investigate the feasibility of some perspective PQC schemes for small and mobile devices and also compare these schemes with common PKC schemes. We believe that this paper could be useful for security engineers that consider employing PQC schemes into systems with constrained devices such as IoT, sensor networks, mobile networks, smart grid, smart homes and others.

### *1.1 Post-Quantum Public-Key Cryptosystems*

PQC cryptosystems have been introduced in Bernstein (2009). In general, post-quantum public key cryptosystems can be divided into five main categories: hash-based cryptography, code-based cryptography, multivariate cryptography, lattice-based cryptography and isogeny-based cryptography. In the following text, we provide brief introduction and examples for these categories:

- Hash-based cryptography - these schemes are based on the security of hash functions (as one-way function) and require less security assumptions than number-theoretic signature schemes (e.g. RSA, DSA). Ralph Merkle in 1979 introduced Merkle Signature Scheme (MSS) that is based on a one-time signatures (e.g. the Lamport signature scheme) and uses a binary hash tree (Merkle tree). MSS is resistant against quantum computer algorithms. More details can be found in this survey on hash-based schemes Butin (2017).
- Code-based cryptography - these cryptosystems are based on error correcting codes to construct a one-way function. The security is based on the hardness of decoding a message which contains random errors and recovering the code structure. The McEliece public key encryption scheme is based on binary Goppa


10.1016/j.ifacol.2018.07.104

codes with high error correction capability and works with matrices. A receiver secretly chooses a private key that is a binary Goppa code. The corresponding public key is generator matrix $G$ that describes a scrambled and randomly permuted variant of the Goppa code. A sender first encodes the plain text using $G$ and adds $t$ random errors during the encryption. Then, the receiver who knows the private key (the hidden algebraic structure of the Goppa code) is able to correct the errors and recover the message. The McEliece scheme is still considered as secure for 40 years. The Niederreiter cryptosystem (a McEliece variant) provides both encryption and signature schemes. Nevertheless, many McEliece variants require large public keys. Sendrier (2017) presents the introduction of code-based cryptography and its perspectives.

- Multivariate cryptography - these schemes are based on systems of multivariate polynomial equations over a finite field $F$.There are several variants of multivariate cryptography schemes based on Hidden Field Equations (HFE) trapdoor functions such as The Unbalanced Oil and Vinegar Cryptosystems (UOV). UOV are used for signatures. Other examples of multivariate cryptography are Rainbow, TTS or MPKC schemes. More about current state of the multivariate cryptography schemes can be found in the paper of Ding and Petzoldt (2017).
- Lattice-based cryptography - these schemes are based on lattice-based computational problems, e.g. the Shortest Vector Problem (SVP) and the Ring Learning With Errors (RLWE) problem. A lattice $L \subset R^n$ is defined as the set of all integer linear combinations of basis vectors. Lattice-based public key schemes are used for public key encryption, key exchange, signature and hash functions. Well known cryptosystems are Frodo, LASH and Ring-Learning with Errors (Ring-LWE) schemes such as NTRU, New Hope or Kyber. Nejatollahi et al. (2017) analyze lattice-based schemes in more details, investigate their properties and survey existed implementations.
- Isogeny-based Cryptography - these schemes are based on supersingular elliptic curve isogenies that are secure against quantum adversaries. These schemes are secured under the problem of constructing an isogeny between two supersingular curves with the same number of points. Isogeny-based schemes may serve as digital signatures or key exchange such as Supersingular Isogeny Diffie-Hellman (SIDH) scheme. More about schemes based on supersingular isogeny problems can be found in Galbraith et al. (2017).

## 2. RELATED WORK

There are several works that investigate the performance and memory parameters of post-quantum cryptographic schemes on different platforms including small devices such as microcontrollers, mobiles, single boards or smart cards.

Chang et al. (2014) modify the PolarSSL library by adding post-quantum lattice-based key exchange schemes and TTS and Rainbow schemes as MPKC signature schemes in order to extend current TLS suites. Their post-quantum SSL/TLS solution could be used on embedded systems but the performance is tested only in X86_64 platform with Intel CPU Xeon quad core E3-1245v3 3.4 GHz. On this machine, only 4.5 TLS handshakes per second are performed with the LATTICE - TTS (128b) cipher suite.

Guillen et al. (2017) presents various implementations of the NTRU encryption scheme in ARM Cortex M0 based microcontrollers. Authors provide the detailed performance and memory analysis of various versions of NTRU implementations and compare their practical results with other works that implemented NTRU on various microcontrollers. Their results show that NTRU could be practical on small devices. Further, Liu et al. (2018) survey the recent implementations of lattice-based cryptosystems for 8 and 32-bit microcontrollers, compare found results and discuss their impact. Yuan et al. (2016) present an interesting work with the JavaScript implementation of six lattice-based Ring-LWE encryption schemes.Their JavaScript implementation could be widely spread in IoT systems due to run via a web browser on multiple platforms. The performance of the implementations is tested on platforms such as PC, Android and small embedded devices like Tessel. For example, 128-bit NTRU encryption takes about 37 ms on Nexus 7 (quad-core 1.3 GHz CPU), and on ARM Cortex M3 microcontroller (Tessel with 180 MHz CPU) the same operation takes about 29 seconds. Moreover, Yuan et al. (2017) show the first implementation of a lattice-based encryption scheme on standard Java Card. The ring-LWE based schemes running time takes about 100 seconds in decryption for 128-bit security by combining the use of iterative fast Fourier transform and improved Montgomery modular multiplication. However, the results indicate that a smart card environment is still too constrained for lattice/based schemes. In this work, we focus on 32-bit ARM devices and mobiles that should be reasonable platforms for PQC.

Recently, Suomalainen et al. (2018) investigate the feasibility and usability of three quantum immune key exchange algorithms (Secrecy coding scheme, NewHope, Frodo) for common IoT hardware platforms such as Raspberry Pi2 and Raspberry Pi3. Their results show that both lattice-based key exchange schemes NewHope and Frodo take tens seconds. Nevertheless, they provide only simulation results measured by the ABSOLUT model library with virtual ARM devices because they do not port the Open Quantum Safe library to 32-bit ARM devices.

In our work, we deal with more types of post-quantum key exchange schemes, measure the schemes on real 32-bit ARM devices and provide discussion about the feasibility of PQC in various IoT systems.

### *2.1 Our Contribution*

The contribution of this work is threefold:

- We describe the setup and implementation issues of PQC cryptosystems on constrained platforms used in IoT systems.
- We outline our practical performance results on two ARM devices with different platforms (32-bit ARM mobile devices with the Android platform, 32-bit ARM single board devices with Linux OS).

- We discuss the feasibility of PQC schemes in various use cases with performance and memory restricted embedded and IoT systems.

## 3. IMPLEMENTATION OF PQC ON CONSTRAINED AND MOBILE DEVICES

This section describes the setup and software implementation details and issues of PQC cryptosystems on 32-bit CPU single-board devices and mobile devices. PQC schemes can be implemented from scratch by using the descriptions in research papers in various programming language such as C/C++, JAVA, Python and others. Nevertheless, developers have to implement and prepare many math, cryptographic and arithmetic modules, e.g. Gaussian sampler and matrix/polynomial multiplication and so on. This from-scratch implementations of the schemes may cause many security flaws and performance issues. However, several open source projects and libraries that implement PQC schemes and math functions already exist, e.g. Codecrypt - the post-quantum cryptography tool, Java Lattice Based Cryptography Library (jLBC), libPQP - Python post-quantum library, the LatticeCrypto Library Longa and Naehrig (2016) and liboqs library (the Open Quantum Safe project Stebila and Mosca (2016)). Therefore, employing the libraries with PQC primitives could be more efficient and stable. In this work, we mainly use the liboqs library (the Open Quantum Safe project Stebila and Mosca (2016)). In the following text, we describe the Android platform and its suitability for PQC libraries and schemes, and the setup steps on a single board 32-bit ARM device with Linux OS.

### *3.1 Android Setup and Implementation*

Many post-quantum libraries such as liboqs are implemented in C/C++. In order to wrap PQC C-written libraries or schemes in the Android platform, the developers can use Android Native Development Kit (NDK) to write code in C/C+ and Java Native Interface (JNI) to call native functions from the C libraries or schemes. Then, PQC implementations written in C could be modified and adapted to be accessible by JAVA via JNI. There are also few PQC implementations already written in JAVA but these implementations are usually inefficient, see our experimental results in the next section.

As an alternative option, the developers and security engineers may use the ARM big.LITTLE heterogeneous computing architecture to embedded C/C++ software development on ARM processors. The PQC implementations written directly in assembly can be faster than C-written implementations, see results Jalali et al. (2017).

### *3.2 32-bit CPU Setup and Implementation*

The liboqs library can be easily installed and built on 64-bit platforms with various Linux OS. Nevertheless, we use a single-board device that has only 32-bit ARM CPU. Hence, some steps have to be made prior to run, benchmark and modify liboqs with PQC schemes on the 32-bit platform. Firstly, header files and methods have to define ARM as the platform. Further, it is necessary to retype

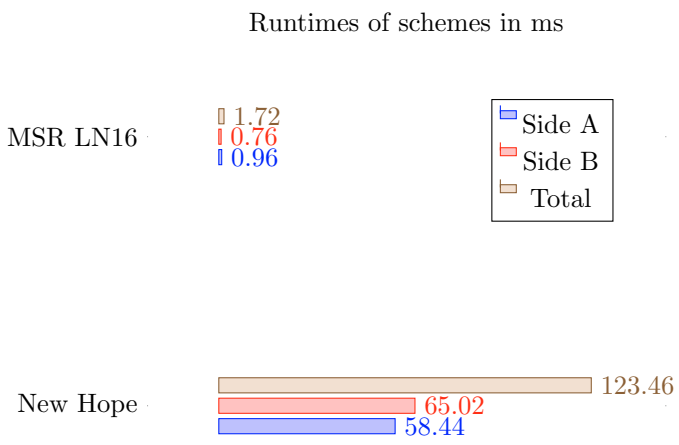


Fig. 1. The performance of PQC key exchange schemes on mobile devices

all 64-bit types that 32-bit CPU cannot handle. The implementation must be adapted to the 32-bit architecture.

In the next section, we outline the experimental results from both platforms described above.

## 4. PERFORMANCE ASSESSMENT OF PQC KEY EXCHANGE SCHEMES

This section presents our experimental results of post-quantum key exchange cryptographic schemes that are implemented on small ARM devices.

### *4.1 PQC key exchange schemes on Android mobile devices*

In this experimental measurement, we use a mobile device with 32-bit CPU Qualcomm Snapdragon 801, 4 cores, 2.5 GHz, 2 GB RAM with Android 6. We test two PQC key establishment schemes with the configuration defined as follows:

- New Hope - lattice-based scheme with 206-bit post quantum security level. The implementation is written in JAVA. The details can be found in Alkim et al. (2016).
- MSR LN16 - lattice-based scheme with 128-bit post quantum security level. The implementation is written in C and uses JNI. The details can be found in Longa and Naehrig (2016).

Figure 1 depicts the performance results of two PQC key exchange schemes measured on the mobile ARM devices. The depicted results are the runtimes of the schemes on side A, side B and the total runtimes on both sides. The values are averaged from 30 measurements. The results indicate that the C/JNI application of the lattice-based key exchange scheme (MSR LN16) is more efficient than the JAVA application of the lattice-based key exchange scheme scheme (New Hope). However, New Hope provides higher security level (206-bit) than MSR LN16 with 128-bit.

### *4.2 PQC key exchange schemes on single-board devices*

In our second experimental measurement, we use a single-board ARM device with 32-bit CPU ARMv7l 1.2 GHz,

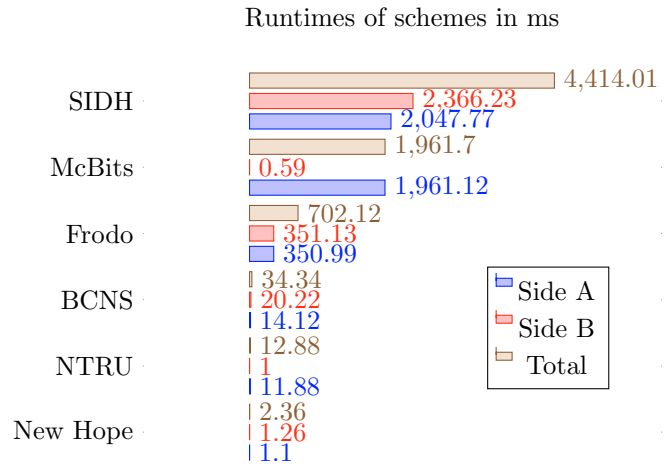


Fig. 2. The performance of PQC key exchange schemes on single-board devices

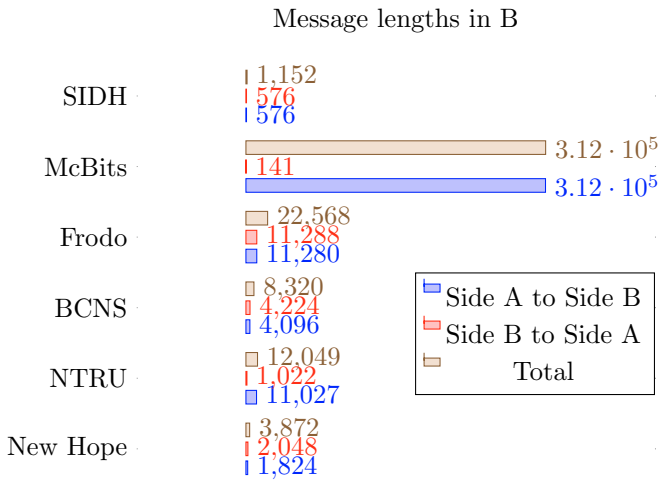


Fig. 3. The total message lengths of the PQC key exchange schemes

1 GB RAM with Linux OS (Raspbian Stretch Lite). In this measurement, we employ six PQC key establishment schemes with the configuration defined as follows:

- New Hope - lattice-based scheme with 206-bit post quantum security level. The full specification can be found in Alkim et al. (2016).
- NTRU - lattice-based scheme with 128-bit post quantum security level. The full specification can be found in Hoffstein et al. (1998).
- BCNS - lattice-based scheme with 78-bit post quantum security level. The full specification can be found in Bos et al. (2015).
- Frodo - lattice-based scheme with 130-bit post quantum security level. The full specification can be found in Bos et al. (2016).
- McBits - code-based scheme with 120-bit post quantum security level. The full specification can be found in Bernstein et al. (2013).
- SIDH - isogeny-based Supersingular Isogeny Diffie-Hellman (SIDH) scheme with 128-bit post quantum security level. The full specification can be found in Costello et al. (2016).

Figure 2 depicts our performance results of six chosen PQC key exchange schemes measured on the single-board ARM devices with a single core. The runtimes on A side, B side and in total are averaged from 10 measurements. The lattice-based New Hope scheme is the most efficient scheme from 6 measured schemes. New Hope, NTRU, BCNS schemes take less than 35 ms. On the other hand, SIDH scheme takes about 4414 ms.

Figure 3 shows the message lengths of chosen PQC schemes. The scheme with the least number of exchanged bytes during the key exchange protocol is SIDH with 1152 B. To be noted, that this scheme is less efficient than 6 measured schemes. The New Hope and NTRU schemes exchange less than 4kB during the protocol and due to their efficiency these schemes offer tradeoff between efficiency and message size parameters.

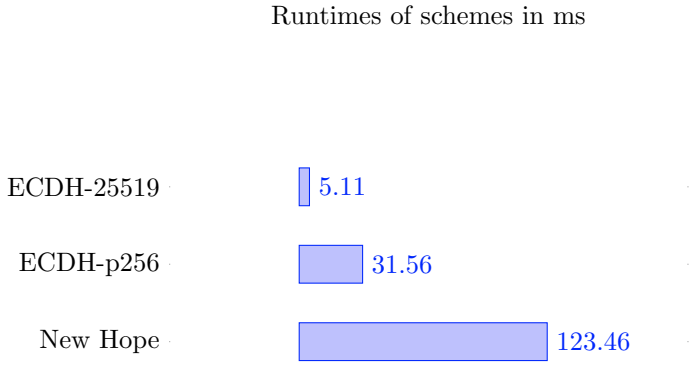


Fig. 4. Performance of PQC New Hope and ECDH on the mobile devices

### *4.3 Comparison of New Hope and ECDH schemes*

In this part, we compare classic ECDH key establishment scheme with a PQC key exchange scheme, namely, New Hope.

Figure 4 compares New Hope and ECDH key exchange schemes on the mobile device. The ECDH-25519 takes only 5.11 ms, ECDH-p256b takes 31.56 ms and the post-quantum New Hope scheme takes 123.46 ms.

Figure 5 compares New Hope and ECDH key exchange schemes on the single-board device. New Hope takes 2.36 ms and is slightly less efficient than ECDH-p256 that takes 2.11 ms. Both schemes are written in C. Nevertheless, the ECDH scheme written in JAVA takes more time than in C (68.69 ms for ECDH-p256, 29.15 ms for ECDH-25519) on the single-board devices. Our results prove that post-quantum cryptographic schemes such as lattice-based key exchange schemes could be competitive to classic asymmetric cryptosystems for key establishment.

Table 1. The feasibility of PQC schemes in IoT (☆ - not suitable; ☆ ☆ - conditionally suitable; ☆ ☆ ☆ - suitable)

| PQC categories | Systems with performance restrictions | Systems with memory restrictions | Systems with restricted communication |
|---|---|---|---|
| Hash-based | ☆ ☆ | ☆ ☆ | ☆ |
| Code-based | ☆ ☆ | ☆ | ☆ |
| Lattice-based | ☆ ☆ ☆ | ☆ ☆ | ☆ ☆ |
| Multivariate-based | ☆ ☆ | ☆ | ☆ ☆ |
| Isogeny-based | ☆ | ☆ ☆ ☆ | ☆ ☆ ☆ |

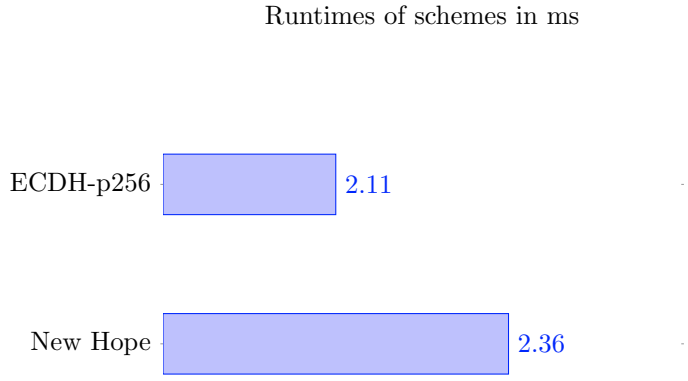


Fig. 5. Performance of PQC New Hope and ECDH on the single board devices

## 5. DISCUSSION

In this section, we discuss the feasibility of PQC schemes that can suite in IoT systems employing computational constrained nodes, memory constrained nodes and message sized restricted communication protocols. Table 1 summarizes the categories of PQC cryptosystems and their suitability for IoT systems with various restrictions. The following text presents some perspective PQC schemes for IoT based on our experimental results and general knowledge.

- **PQC for systems with performance restrictions** - these systems employ nodes with several MHz. The cryptographic and PQC schemes and their operations/phases should have minimal cycles. The key exchange protocol PQC schemes that take up to several tens of ms in total can be suitable. As the most efficient PQC schemes are considered lattice-based schemes. Also our experimental results indicate that schemes such as New Hope can be quite fast and could be as efficient as a not-quantum resistant ECDH key exchange scheme.
- **PQC for systems with memory restrictions** - these systems employ memory constrained nodes, e.g. microcontrollers with several KB, smart cards as secure elements and other. All parameters and keys used in a PQC scheme should have moderate sizes and take reasonable-portion of memory (RAM, EEPROM) in nodes. Isogeny-based schemes such as isogeny-based Supersingular Isogeny Diffie-Hellman (SIDH) that employ supersingular elliptic curves offer short lengths of parameters and keys. Employing compression algorithms and techniques Costello et al. (2017) offers SIDH with 330-bytes public keys at a 128-bit quantum security level. On other hand, Multivariate PKCs schemes such as Rainbow use secret and public keys having several tens of kB. The long keys are also used in code-based cryptography (e.g. the McEliece scheme).
- **PQC for systems with restricted communication protocols** - several IoT and wireless communication protocols and technologies have restricted lengths of messages such as LORAWAN, SigFox, Application Protocol Data Unit (APDU) used by smart cards and other protocols. This restriction requires that exchanged messages must keep minimal sizes during the key establishment. In our measurement, SIDH with 1152 bytes seems as the most promising scheme. The low performance of SIDH could be solved by utilizing a high performance cryptographic co-processor such as in Seuschek et al. (2015) or by accelerating this algorithm on FPGA cards, similarly like AES in Smekal et al. (2016). Code-based schemes that send hundreds kB length messages are not suitable in this scenario.

## 6. CONCLUSION

The post-quantum public key cryptography offers several types of constructions and many key exchange schemes which are analyzed in this work. The lattice-based schemes such as New Hope and NTRU seem to be promising for various scenarios in IoT due their efficiency and relatively reasonable lengths of keys and parameters. These schemes could be efficient and comparable with classic schemes such as ECDH and can be runnable on various small devices with 32-bit architecture and mobile devices with Android OS.

Our future work will aim at the feasibility of post-quantum signature schemes in IoT systems and will explore more schemes from the NIST Post-Quantum Cryptography Standardization competition.

## ACKNOWLEDGEMENTS

Research described in this paper was financed by the National Sustainability Program under grant LO1401 and Ministry of Interior under grant VI20162018003. For the research, infrastructure of the SIX Center was used.

## REFERENCES

Erdem Alkim, Léo Ducas, Thomas Pöppelmann, and Peter Schwabe. Post-quantum key exchange-a new hope. In *USENIX Security Symposium*, volume 2016, 2016.

Daniel J Bernstein. Introduction to post-quantum cryptography. In *Post-quantum cryptography*, pages 1–14. Springer, 2009.

Daniel J Bernstein, Tung Chou, and Peter Schwabe. Mcbits: fast constant-time code-based cryptography. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 250–272. Springer, 2013.

Joppe Bos, Craig Costello, Léo Ducas, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Ananth Raghunathan, and Douglas Stebila. Frodo: Take off the ring! practical, quantum-secure key exchange from lwe. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 1006–1018. ACM, 2016.

Joppe W Bos, Craig Costello, Michael Naehrig, and Douglas Stebila. Post-quantum key exchange for the tls protocol from the ring learning with errors problem. In *Security and Privacy (SP), 2015 IEEE Symposium on*, pages 553–570. IEEE, 2015.

Denis Butin. Hash-based signatures: State of play. *IEEE Security & Privacy*, 15(4):37–43, 2017.

Yun-An Chang, Ming-Shing Chen, Jong-Shian Wu, and Bo-Yin Yang. Postquantum ssl/tls for embedded systems. In *Service-Oriented Computing and Applications (SOCA), 2014 IEEE 7th International Conference on*, pages 266–270. IEEE, 2014.

Craig Costello, Patrick Longa, and Michael Naehrig. Efficient algorithms for supersingular isogeny diffie-hellman. In *Annual Cryptology Conference*, pages 572–601. Springer, 2016.

Craig Costello, David Jao, Patrick Longa, Michael Naehrig, Joost Renes, and David Urbanik. Efficient compression of sidh public keys. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 679–706. Springer, 2017.

Jintai Ding and Albrecht Petzoldt. Current state of multivariate cryptography. *IEEE Security & Privacy*, 15(4):28–36, 2017.

Steven D Galbraith, Christophe Petit, and Javier Silva. Identification protocols and signature schemes based on supersingular isogeny problems. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 3–33. Springer, 2017.

Oscar M. Guillen, Thomas Pöppelmann, Jose M. Bermudo Mera, Elena Fuentes Bongenaar, Georg Sigl, and Johanna Sepulveda. Towards post-quantum security for iot endpoints with ntru. In *Proceedings of the Conference on Design, Automation & Test in Europe*, DATE '17, pages 698–703, 3001 Leuven, Belgium, Belgium, 2017. European Design and Automation Association. URL `http://dl.acm.org/citation.cfm?id=3130379.3130548`.

Jeffrey Hoffstein, Jill Pipher, and Joseph H Silverman. Ntru: A ring-based public key cryptosystem. In *International Algorithmic Number Theory Symposium*, pages 267–288. Springer, 1998.

Amir Jalali, Reza Azarderakhsh, and Mehran Mozaffari-Kermani. Efficient post-quantum undeniable signature on 64-bit arm. In *International Conference on Selected Areas in Cryptography*, pages 281–298. Springer, 2017.

Zhe Liu, Kim-Kwang Raymond Choo, and Johann Grossschadl. Securing edge devices in the post-quantum internet of things using lattice-based cryptography. *IEEE Communications Magazine*, 56(2):158–162, 2018.

Patrick Longa and Michael Naehrig. Speeding up the number theoretic transform for faster ideal lattice-based cryptography. In *International Conference on Cryptology and Network Security*, pages 124–139. Springer, 2016.

Hamid Nejatollahi, Nikil Dutt, Sandip Ray, Francesco Regazzoni, Indranil Banerjee, and Rosario Cammarota. Software and hardware implementation of lattice-cased cryptography schemes. 2017.

Nicolas Sendrier. Code-based cryptography: State of the art and perspectives. *IEEE Security & Privacy*, 15(4): 44–50, 2017.

Hermann Seuschek, Piyush Khurana, and Georg Sigl. Hipechigh performance cryptographic service for heterogeneous network-on-chip systems. *IFAC-PapersOnLine*, 48(4):31–36, 2015.

David Smekal, Jakub Frolka, and Jan Hajny. Acceleration of aes encryption algorithm using field programmable gate arrays. *IFAC-PapersOnLine*, 49(25):384–389, 2016.

Douglas Stebila and Michele Mosca. Post-quantum key exchange for the internet and the open quantum safe project. In *International Conference on Selected Areas in Cryptography*, pages 14–37. Springer, 2016.

Jani Suomalainen, Adrian Kotelba, Jari Kreku, and Sami Lehtonen. Evaluating the efficiency of physical and cryptographic security solutions for quantum immune iot. *Cryptography*, 2(1):5, 2018.

Ye Yuan, Chen-Mou Cheng, Shinsaku Kiyomoto, Yutaka Miyake, and Tsuyoshi Takagi. Portable implementation of lattice-based cryptography using javascript. *International journal of networking and computing*, 6(2):309–327, 2016.

Ye Yuan, Kazuhide Fukushima, Shinsaku Kiyomoto, and Tsuyoshi Takagi. Memory-constrained implementation of lattice-based encryption scheme on standard java card. In *Hardware Oriented Security and Trust (HOST), 2017 IEEE International Symposium on*, pages 47–50. IEEE, 2017.