BIS Innovation Hub

Fully Scalable Settlement Engine (FuSSE)

A strategic framework for quantum-cryptography in FuSSE

April 2024

Executive Summary

The Fully Scalable Settlement Engine Project (FuSSE) seeks to develop a highly scalable and modular approach to providing future-proof settlement functionality for the modernisation of Financial Market Infrastructures (FMIs). This multifaceted project addresses numerous challenges, with cybersecurity as a key focus.

Within the scope of Project FuSSE, several security-enhancing features are proposed, with a focus on crucial cryptographic functionalities including signing, signature validation, encryption, and decryption. These functionalities, part of a broader suite of tools, provide security, confidentiality, integrity and non-repudiation capabilities.

Recent advancements in quantum computing present profound risks to existing cryptographic systems, which underpin the security of financial applications worldwide. As these quantum technologies evolve, they threaten to compromise the cryptographic safeguards that protect digital transactions, thereby posing significant risks to the integrity and security of FMIs.

In order to achieve its goal of providing future-proof components, Project FuSSE included within its scope the development of quantum resistant cryptographic components for signing, signature validation and data encryption/decryption. As research progresses on the subject of quantum resistance, the landscape of available solutions is constantly changing, thus the FuSSE system adopts a crypto agile approach that allows it to easily integrate new solutions as they become available.

This document specifically focuses on the cryptographic modules within Project FuSSE. These modules represent a partial deliverable of the complete system but are designed to be immediately beneficial. Due to the modular design of FuSSE, central banks and other stakeholders can leverage these quantum-resistant cryptographic capabilities today, by requesting access to the project’s source code, without the need for a full deployment of the entire FuSSE framework.

Project FuSSE's approach not only aligns with the urgent need for quantum-resistant cryptographic solutions but also provides scalable and adaptable technologies to ensure that the financial sector remains secure and trustworthy in an increasingly digital and quantum-enabled future.

Contents

Introduction

Introduction

In the rapidly evolving landscape of global finance, the modernisation of Financial Market Infrastructures (FMIs) is paramount. The Fully Scalable Settlement Engine Project (FuSSE) aims to provide tools to contribute to the evolution of these infrastructures through a modular and scalable architecture designed to enhance settlement functionalities. This framework is poised to support the growing complexity and increasing demands of financial systems, by ensuring their resilience and efficiency in the digital age.

A critical aspect of modernizing Financial Market Infrastructures (FMIs) is enhancing their cybersecurity. Project FuSSE addresses this challenge by integrating advanced cryptographic functions essential for securing the integrity, confidentiality, and non-repudiation of digital transactions. These functions are a crucial component of Project FuSSE's extensive suite of security tools, which play a pivotal role in protecting financial transactions from the myriad cyber threats existing today and in the future.

The rapid progression of quantum computing technologies poses a significant threat to the cryptographic systems currently underpinning the security of financial applications worldwide. Accordingly, building systems that can continue to guarantee the integrity and confidentiality of financial transactions is essential.

In response to these challenges, the development of quantum-resistant cryptographic capabilities are considered to be within the scope. These capabilities are focused on signing, validation, and data encryption/decryption protocols. As the landscape of quantum resistance evolves, the definitive solutions for these cryptographic challenges have not yet been finalised by bodies such as the National Institute of Standards and Technology (NIST). Therefore, Project FuSSE adopts a crypto-agile approach, ensuring that its cryptographic modules can adapt to new solutions as they emerge and are validated. (Chen, L., et al (2016))

FuSSE complements other BIS Innovation Hub projects designed to address the threat of quantum-enabled cryptographic attacks. Project Leap experimented with securing communication channels through advanced virtual private network (VPN) encryption. Project FuSSE seeks to extend the scope by working on three aspects: (i) fortifying the digital signatures used to guarantee integrity and ownership of financial data;(ii) authenticating these signatures efficiently to allow for large scale volumes of validations; and (iii) providing quantum safe mechanisms to exchange keys used for the encryption of financial data. (Bank for International Settlements. (2023))

This report delves into the implementations of quantum-resistant cryptographic functionalities within Project FuSSE, aimed at securing (FMIs) against the emerging threats posed by quantum computing. The document explores the selection of cryptographic algorithms, describes the importance of quantum agility in adapting to new cryptographic standards, and details the future roadmap for integrating and enhancing other cryptographic functionalities. This report aims to inform stakeholders about the ongoing efforts to fortify FMIs with quantum-resistant technologies, ensuring their resilience and trustworthiness in a rapidly evolving digital landscape.

Due to its highly modular and flexible architecture, reminiscent of Lego blocks, Project FuSSE allows for the cryptographic modules to be deployed either as integral parts of the full system or as standalone solutions. This modularity enables central banks and other stakeholders to immediately leverage these advanced, quantum-resistant cryptographic functionalities independently – securing both current systems and laying a robust foundation for futureproofing against quantum threats[1].

This report and partial project deliverable has been made in partnership with

BANK OF CANADA
BANQUE DU CANADA

IDB

evolution

[1] For example, the modules created in this deliverable can be used to sign official communications using quantum secure algorithms, and to provide a service to validate such signatures by exposing these functionalities to users that need to verify signatures.

Project FuSSE and its cryptographic components

About Project FuSSE

Project FuSSE is aimed at supporting the modernisation of FMIs through the development of a set of scalable and modular components that jointly provide settlement functionalities. The core objective of FuSSE is to enhance the resilience, efficiency, and security of financial transactions in the face of rapidly changing market demands and advancing technologies. The security and resilience of the system and its data are fundamental to achieving this goal, and thus, the project aims to secure confidentiality, integrity, and availability to the fullest extent possible.

FuSSE follows a series of design principles that influence the way it addresses functionality in general and cryptography in particular. A key design principle is modularity. FuSSE's modular approach is based on an architecture in which programs that provide a simple independent function (microservices) are connected through lists of tasks (publisher/subscriber infrastructure). The previous microservice (publisher) writes the tasks, and the following subsequent microservice (subscriber) reads its next chore All of this is coordinated through instructions contained in each individual message, that indicate to which outbound task list the result of a microservice function should be added. The cryptographic functions of the system follow this architectural design, allowing them to be reusable for other projects and to be easily replaced when necessary.

A high-level overview of the cryptographic use cases in FuSSE

Cryptographic techniques are essential for securing various facets of digital communication and data management. To understand this deliverable's scope, it is relevant to understand the different use cases for cryptography.

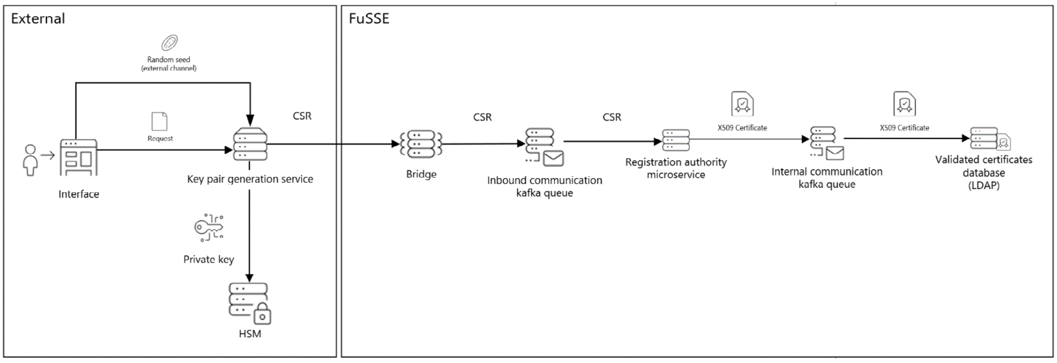
1. **Digital signatures** -these hold significant importance in securing financial transactions by verifying the authenticity of digital messages or documents. As the financial industry increasingly moves towards digital platforms, the integrity and security of these transactions become vital. They ensure that messages have not been altered in transit and validate the identity of the sender, playing a critical role in the integrity and security of financial data.

2. **Data encryption in transit** - this refers to the protection of data while they are being transmitted across networks. As digital service usage expands, securing information as it travels between devices, servers, or between clients and servers becomes crucial to safeguarding data privacy and security. This type of encryption is vital in preventing unauthorised interception, eavesdropping, and manipulation by malicious actors during data transit. This ensures that even if data packets are captured during transmission, they cannot be read or altered without the appropriate encryption keys, thereby maintaining the confidentiality and integrity of the data throughout their journey.

3. **Data encryption at rest** -this use case focuses on protecting data stored on any physical or virtual storage system. This security measure is essential in an era in which data breaches and unauthorised data access is prevalent. Encryption at rest ensures that data such as files, databases, and logs stored on hard drives, SSDs, or cloud storage are encrypted using cryptographic algorithms to prevent data visibility in the event of system compromise or theft.

The implementation of these functionalities within the system is done through three microservices, the first produces digital signatures on the communications that the system needs to generate (signing service), the second validates the digital signatures on the data elements that the system receives from the participants or from other components of the system (signature verification service), and the third generates the elements necessary to build the cryptographic key pairs needed for the creation and validation of such signatures (key pair generation service). To function together properly, these elements also require an additional element that keeps track of which of the key pairs generated by the key pair generation service are valid for instructing the system. The following graphs show a high-level overview of the interaction between those services within the FuSSE environment.

Key pair generation and registration: the key pair generation microservice uses a random seed and proceeds to generate a key pair using the selected cryptographic algorithm.[2] It then communicates a certificate signing request (CSR) - that includes the public part of the pair, to the authority to request registration. The authority signs the CSR and registers the certificate as valid. Once the certificate is generated, it is registered in the system as a valid certificate for its use in FuSSE.

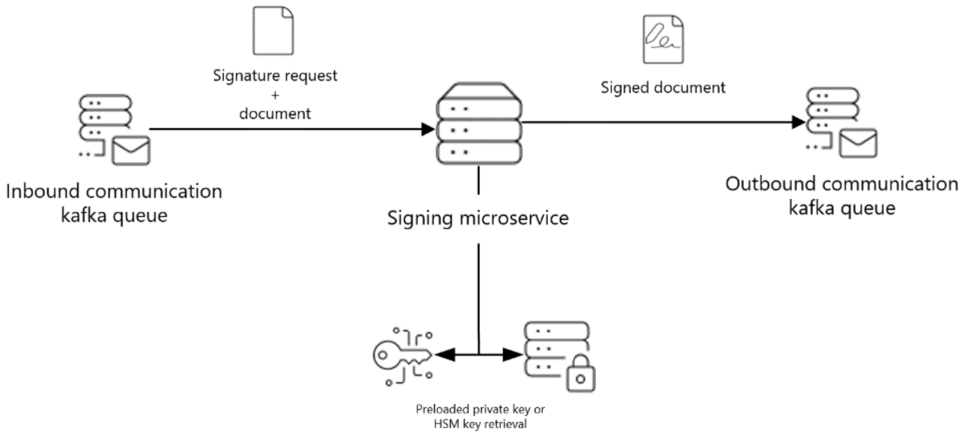


Graph 1 Key pair generation process.

Signing: the signing microservice has been preloaded with a specific private key (generated using the previous flow), and it receives requests over its inbound message queue that includes the data element to be signed. Using its private key and the selected cryptographic algorithm it generates a signature byte array that is attached to the original message converting it into a signed message. The resulting signed

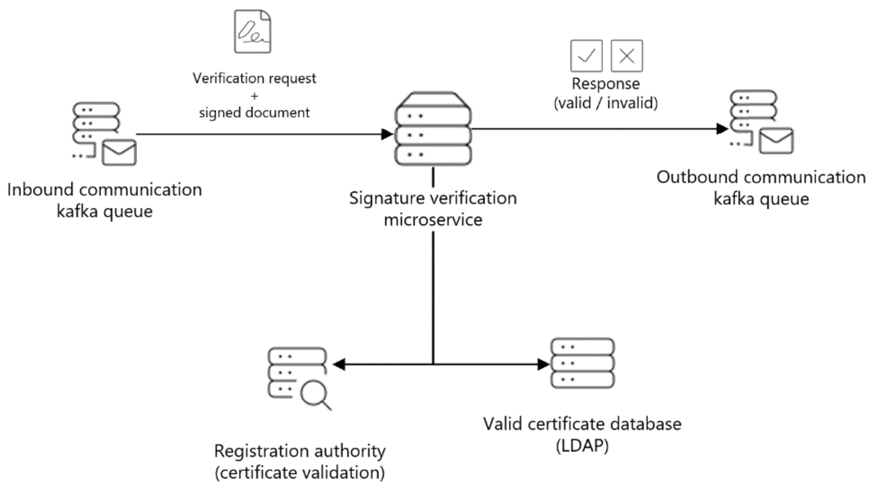
[2] Note that this keypair generation service may live within the system for those keys necessary for the operation of the system and can be provided externally to generate the keypairs of the users so that their private keys are secure.

message is in turn submitted to the relevant outbound queue according to the instructions in the message.



Graph 2 Signing process.

Signature validation: the signature verification microservice receives a data element and the correspondent signature on its inbound message queue, together with the information to identify the certificate of the keypair with which it was signed. The verification service requests to the registration authority and also requests validation of the associated certificate by the valid certificate database. A certificate is considered valid if it is known by FuSSE and has not been revoked or expired, and if so, to send it. The microservice then uses a cryptographic process to review the original data element and the information from the certificate to validate the signature. The end result of the validation is then in turn submitted to the next microservice as specified by the instructions contained in the message.



Graph 3 Signature validation process.

The cryptographic functionality described in graph 3 applies to current cryptographic algorithms as well as newer quantum resistant ones. It is not currently known when sufficiently powerful quantum computers will become available and therefore some jurisdictions may decide to wait. Nevertheless, it is useful to describe the effects of such a breakthrough in computing.

Understanding the quantum threat

Quantum computing represents a transformative advance in computational capabilities, posing significant challenges to current cryptographic methods that underpin global financial systems. The validity of traditional cryptographic algorithms like RSA and elliptic curve cryptography (ECC) stems from the computational difficulty of factoring large prime numbers and solving discrete logarithm problems. However, these tasks can be efficiently solved by a sufficiently powerful quantum computer using a mathematical approach known as Shor's algorithm. It is considered that: "Most of the public key cryptography that is used on the internet today is based on algorithms that are vulnerable to quantum attacks. These include public key algorithms such as RSA, ECC, Diffie-Hellman and DSA. All these examples are easily broken by Shor’s algorithms and are deemed to be insecure as quantum computing matures.”.(Campagna, M et al (2015)).

The implications of quantum threats on financial systems are profound. As quantum technology matures, it will affect the security mechanisms of digital signatures, which are crucial for securing tamper evidence and non-repudiation in digital transactions. Digital signatures are created through asymmetric cryptographic schemes that depend on the algorithms described above (RSA and ECC), which are susceptible to quantum attacks. [3] Thus, sufficiently powerful quantum computers can generate capabilities through which bad actors could create fraudulent instructions that would be indistinguishable from licit ones, violating the integrity and authenticity of financial transactions and, making financial resources vulnerable to unauthorised access. In other words, a successful quantum attack on digital signatures would compromise their integrity and their ability to serve as a reliable proof of origin. This would effectively nullify their role in legal validations, financial agreements and other critical applications in which undeniable proof of participation is required.

The cryptographic elements threatened by quantum computing do not only secure transactions, but also play an important role in maintaining the integrity and confidentiality of the data that flow across financial networks[4]. A breach in cryptographic defences could expose digital communications - leading to the

[3] ”The reason Shor’s algorithms break these public key cryptosystems is that they are based on two specific computational problems - namely, Integer factorization and discrete logarithm. These problems are believed to be hard for a classical computer to solve, but are known to be easily solved by a quantum computer. In order to sustain the security of the Internet and other technologies reliant on cryptography it is necessary to identify new mathematical techniques upon which cryptography can be built that are resilient against quantum attacks.” (Campagna, M et al, 2015).

[4] Many secure communication channels use asymmetric key encryption to send a shared secret to the recipient of the information, which is then used to encrypt and decrypt relevant information through efficient symmetric processes. If this initial information exchange is compromised, as it can be because asymmetric encryption is vulnerable to quantum attacks, then the full encryption of the channel is at risk.

exposure of financial data. Any of these issues could have a tremendous impact on the trust necessary for the financial system to operate.

With respect to the information stored within the system (at rest), its vulnerability to quantum threats depends on the type of encryption used to protect it. The same challenges apply. If asymmetric cryptography was used. However, given that at rest data are usually encrypted using symmetric keys, the challenge is not fully clear.

Therefore, adapting digital signature schemes and fortifying communication processes to be resistant to quantum attacks through the use of post-quantum cryptographic algorithms becomes imperative to preserve their essential roles in maintaining the legitimacy and security of digital transactions in a post-quantum world (Campagna, M et al 2015).

Implementation of quantum-resistant cryptography in Project FuSSE

Cybersecurity is paramount for systems like FMIs that handle sensitive financial data and transactions. Traditional cryptographic systems, foundational to the security protocols in FMIs, are becoming vulnerable due to the advent of quantum computing technologies. Thus, any solution proposed today need to consider strategies to cope with this future situation.

Project FuSSE is designed to address these concerns within its security framework, ensuring robust protection against emerging threats, by integrating quantum-resistant cryptographic technologies. These include:

1. **Cryptographic assurance for data integrity and non-repudiation**-maintaining both data integrity and non-repudiation is essential to ensure the authenticity and verifiability of information under classical and emerging quantum threats. Lattice-based digital signature schemes, such as Crystals-Dilithium and Falcon, effectively address these needs by ensuring that any unauthorised data alterations are detectable and that once a document is signed, the involvement of the signatory is indisputable. This is vital for applications in legal, financial, and security contexts in which the reliability of transaction authenticity is paramount.

2. **Maintaining data confidentiality** - there are two dimensions of data confidentiality, namely in transit and at rest. On one hand, data confidentiality in transit usually employs a two-stage process for securing the communication channels. In the first stage a common secret is shared among sender and receiver so that in a second stage, the sender will use that secret to encrypt the information to be sent. The problem with this is that the original secret sharing is done, in many cases, through asymmetric encryption, which is vulnerable to attack. To eliminate this problem FuSSE will use an algorithm called Kyber as a quantum safe key exchange mechanism (KEM). On the other hand, as described before, to provide encrypted communication channels while securing data at rest can safely be done by using standard encryption

techniques. While specific quantum-resistant encryption algorithms have been developed, traditional symmetric key encryption algorithms are still considered safe, with a few caveats (Chen, L. et al (2016))

Quantum-resistant technology is still in the early stages of research, and we expect new algorithms appear. Meanwhile existent algorithms may show critical vulnerabilities. This implies that any cryptographic solution implemented by FuSSE should be materialised by adopting a quantum-agile approach. FuSSE should allow for the easy integration and replacement of cryptographic solutions as they become available, thus addressing the requirement to future- proof the project requirement.

Current quantum-resistant cryptographic microservices

This report concentrates primarily on the quantum resistant digital signature processes and is an integral part of the deliverables alongside the source code generated to provide the functionality [5]. Data encryption at rest and in motion will be discussed in subsequent deliverables. To address the use of signatures, three microservices where developed:

1. **Signature execution**: Implementation of secure mechanisms for signing digital content.

2. **Signature validation**: Implementation of a process to verify the authenticity and integrity of digital signatures.

3. **Key pair generation**: This comprises the generation of secure and reliable public and private cryptographic keys, their inclusion alongside profile data in a CSR and the final encapsulation of these data in a digital certificate following the X.509 v3 Standard.

Algorithm selection

The National Institute of Standards and Technology (NIST) plays a crucial role in the development and standardisation of cryptographic technology, especially in the emerging field of post quantum cryptography (PQC) (CSRC - NIST Computer Security Resource Center (2016)). NIST selected three algorithms as finalists for digital signature standardisation:

1. **CRYSTALS-Dilithium** - its signing, and verification processes are computationally less demanding, making it a strong candidate for applications requiring fast and secure digital signatures. Overall, results in a simpler implementation than its competitor Falcon. (Alagic, G, et al (2020)).

[5] Central banks and other national authorities interested in access to the code please contact toronto.centre@bisih.org.

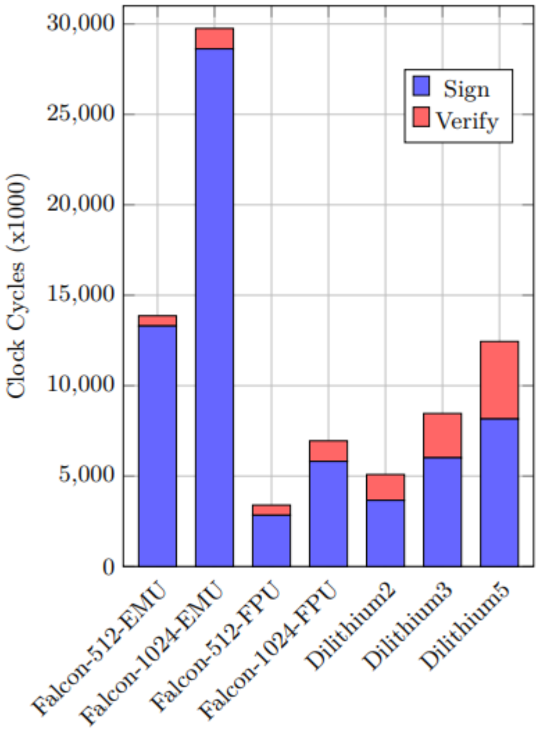
2. **FALCON** - uses lattices-based cryptography and is, noted for its small signature sizes but involves more complex computations (Fouque et al (2020)).

3. **SPHINCS+**- a stateless hash-based signature scheme, offering high security but at the cost of larger signatures and slower computations (Bernstein et al. (2019)). Within Project FuSSE, SPHINCS+ was not pursued for further experimentation primarily due to its larger signature sizes and significantly slower performance compared with CRYSTALS-Dilithium and FALCON (NIST(2022)). These characteristics are less favourable in high-volume transaction environments, where speed and data minimisation are essential for maintaining system performance and scalability. As a result, the focus shifted towards CRYSTALS-Dilithium and FALCON, both of which offer better performance metrics and smaller key sizes, without compromising on security.

Both CRYSTALS-Dilithium and FALCON are cutting-edge algorithms that aim to offer high security against quantum and classical attacks. Specifically, both algorithms target NIST security level 5, which is the most secure level (CSRC - NIST Computer Security Resource Center (2024a)). This level is analogous to the security provided by AES-256. Despite their equivalence in targeted security levels, there are fundamental differences in design and implementation which influence the adaptability of each to financial systems.

In selecting the most efficient and secure algorithm for Project FuSSE, the comparative graph below tells a compelling story about speed and security.

The graph is a representation of how quickly each security system can be activated (locked) and deactivated (unlocked). The time taken for each action is measured in



Graph 4 Signature benchmarks of CRYSTALS-Dilithium and Falcon (tree) on ARM Cortex M4 processor (Howe, J and Westerbaan, B (2023)). Reprinted courtesy of the National Institute of Standards and Technology, U.S. Department of Commerce. Not copyrightable in the United States.

"clock cycles", which is akin to counting the number of ticks on a clock, fewer ticks mean the action is completed faster.

These features align more closely with the operational demands and security requirements of financial systems. The subsequent comparison between CRYSTALS-Dilithium and FALCON was centred on factors such as ease of integration, computational efficiency, and robustness against quantum threats, which are vital for ensuring the long-term security of financial infrastructure against emerging cryptographic challenges.

Falcon´s performance gains are primarily realised when leveraging a floating-point unit (FPU), which can significantly enhance processing speed for cryptographic operations. However, dependence on a FPU introduces certain complexities (Howe and Westerbaan (2023)). Not all operational environments support FPU, particularly in settings with limited resources or those that do not have advanced technological infrastructures. This dependency may restrict Falcon's applicability and complicate its deployment. Conversely, CRYSTALS-Dilithium showcases a robust performance across a diverse range of hardware settings without the need for such specialised components (Howe and Westerbaan (2023)). It has been implemented and tested across a range of hardware, from high-performance servers to low-power microcontrollers, demonstrating its efficiency in different technological settings (Banegas et al, (2021)). This attribute makes CRYSTALS-Dilithium a highly adaptable choice while ensuring reliable performance and security without necessitating complex hardware upgrades.

Upon reflection, while speed is crucial, consistent, and dependable performance in verification is critical for FMI's. CRYSTALS-Dilithium's unwavering efficiency across various levels of security and its robust performance in verification tasks make it the superior choice. It is better suited to our purposes, for the following reasons:

1. **Security**: CRYSTALS-Dilithium has a stronger resilience against timing attacks compared with Falcon. This is crucial for a payment system where ensuing the security of transaction signatures against potential attacks is paramount (Howe, J and Westerbaan, B (2022)).

2. **Stability and predictability:** CRYSTALS-Dilithium's performance is more consistent across different computing environments because it offers predictable performance without depending on specific hardware features. This contrasts with FALCON, which requires a hardware FPU for optimal performance (Howe, J and Westerbaan, B (2022)). This characteristic allows Project FuSSE the flexibility to be deployed across various hardware platforms.

3. **Simplicity and robustness:** the simplicity of CRYSTALS-Dilithium leads to easier implementation and maintenance, reducing the chance of security flaws arising from implementation errors (Howe and Westerbaan (2022)).

4. **Performance:** although FALCON can offer more efficient signature verification times and smaller key and signature sizes, CRYSTALS-Dilithium is recommended by NIST for most digital signature applications due to its

strong security profile and excellent performance across a range of scenarios National Institute of Standards and Technology (NIST) (2022).

Quantum agility

Quantum agility refers to the capability of cryptographic systems to adapt and seamlessly transition to quantum-resistant algorithms in response to the emerging threats posed by quantum computing. This concept is critical in cybersecurity, particularly as advances in quantum computing threaten to undermine traditional cryptographic protocols, which are deemed vulnerable to quantum attacks (Chen, L., et al., 2016).

Quantum agility allows organisations to stay adaptable in a rapidly changing technological landscape. By preparing for the integration of new cryptographic standards and technologies, organisations can protect their information systems against emerging threats without extensive overhauls.

High level strategy for quantum agility

For an application or system to have quantum agility, several key strategies must be followed and implemented:

1. **Crypto-agile architecture -** creating flexible cryptographic architectures that can support multiple algorithms and seamlessly integrate new, quantum-resistant algorithms as they become available. This design approach ensures that systems can be updated without comprehensive overhauls (Campagna, M et al (2015)). This strategy is in line with the Project FuSSE's security and architectural principles that require FuSSE components to be modular to ease system maintenance and security principles that require periodic evaluation of cryptographic components.

2. **Algorithm agility -** enabling easy switching between cryptographic algorithms to respond to changes in the threat landscape and technological advancements. This often involves developing standardise interfaces and protocols for algorithm updates (Bernstein et al., 2019). This strategy is in line with Project FuSSE's architectural guidelines require that the consumer of a service remain agnostic about how the service is implemented within its provider.

3. **Future-proof policies and operations** - developing policies and operational practices that prepare for algorithm transitions, including regular security assessments, technical team training, and staying informed about quantum computing developments. This strategy implies that current architectural and implementation decisions should be reviewed periodically over time in order to determine if the supported algorithms are still deemed safe or if new algorithms are worth including within FuSSE.

Specific requirements supporting the strategy

To ensure that the cryptographic components discussed in the **modularity and cryptographic modules** section (signature execution service, signature validation service, key pair generation, and X.509 v3 certificate generation) are quantum agile, the following requirements should be met:

1. **Flexible cryptographic framework -** each component must be designed within a framework that supports the integration and replacement of cryptographic algorithms without impacting other system functionalities.

   Implementation in FuSSE - currently all FuSSE microservices including the ones in the scope of this report support a modular architecture that loosely couples components in such a way that changes in one component have minimal impact on the remaining components in the ecosystem. Additionally, each microservice is required to implement an internal interface that effectively shields service consumers undue implementation dependencies with the specific algorithms.

2. **Standardised interfaces - d**evelop and utilise standardised interfaces for cryptographic processes to facilitate the easy swapping of cryptographic algorithms as newer, more secure options become available.

   Implementation in FuSSE – currently, all FuSSE microservices, including those in scope of this report, offer standardised interfaces, thus service consumers only have to deal with creating service request messages according to the corresponding FuSSE message schema. Thus, any future algorithm change could be easily done inside the microservice or by deploying an independent microservice supporting the new algorithm while supporting the same message schema.

3. **Configurable algorithm parameters -** allow for the configuration of algorithm parameters at runtime to enable quick responses to cryptographic threats without needing system downtime for upgrades.

   Implementation in FuSSE - algorithm parameters for the microservices in scope for this project are currently fed through the inbound message schema which currently specifies the payload to work on, the associated certificate serial number (to identify the keys to use) and the requested algorithm name if applicable, thus there is no need for additional parameter configuration. If the need appears in the future, non-message-based parameters can be configured directly through the application configuration file.

4. **Modular Design** - design each component as a modular entity, which can be updated independently to adopt new cryptographic standards without extensive reengineering of the entire system.

   Implementation in FuSSE -as per architectural guidelines mentioned above, all microservices in FuSSE are constructed as independent stand-alone modules, thus they can be independently maintained and modified with non impact or minor impacts on the entire system. The same philosophy applies within every microservice in which internal interfaces support abstraction between the service and concrete implementation of cryptographic standards.

5. **Continuous security assessment -** implement continuous security assessment protocols to evaluate the efficacy of current cryptographic measures and the necessity for transitions to quantum-resistant algorithms.

   Implementation in FuSSE: This particular requirement is a project-oriented requirement, in order to address it, regular assessments should be performed on the cryptographic standards used to determine if they are still considered safe. Equal consideration should be given to integrating new standards as they become available and considered both mature and relevant for implementation in FuSSE.

Roadmap and next steps

Roadmap for quantum resistant cryptographic microservices

As Project FuSSE advances, subsequent iterations will enhance its cryptographic framework, focusing on establishing a more complete and robust infrastructure including in the following key areas:

Internalisation of the certificate authority functionality

- **Objective**: to bring in-house the current authority or public key infrastructure (PKI) services that are currently implemented through EJBCA, integrating them within the FuSSE profile services domain. This shift aims to centralise and streamline certificate management processes, enhancing control and security. However, this should be implemented in a way that support available open standards or APIs if possible, so as to maintain freedom of implementation of the PKI functionality.

- **Benefits**: internalising PKI functionality will reduce dependence on external systems, allowing for more rapid deployment and customisation of security policies. It will also facilitate better integration with existing and future cryptographic services within Project FuSSE, providing a unified approach to security.

Development of quantum-resistant transport layer security for secure communications

- **Objective**: to implement Transport Layer Security (TLS) protocols that are resistant to quantum computing attacks. This involves the integration of post-quantum cryptographic algorithms into the TLS protocol, ensuring secure data transmission even in the presence of quantum threats. This particular requirement will address the data encryption in motion requirement.

- **Benefits**: quantum-resistant TLS will protect against eavesdropping and tampering with data in transit - critical requirements for maintaining the confidentiality and integrity of financial transactions. This development ensures that communications remain secure against both current and future cryptographic challenges.

Development of quantum-resistant key exchange mechanisms (KEM)

- **Objective**: to develop and deploy KEMs that are secure against quantum attacks. This includes researching and implementing advanced cryptographic algorithms that can operate effectively under the threat of quantum decryption techniques.

- **Benefits**: a quantum-resistant KEM will ensure that the initial exchange of cryptographic keys—used for securing subsequent communications—remains confidential and secure from interception or compromise by quantum computers.

Development of secure key management services

- **Objective**: to enhance key management services by incorporating quantum-resistant features, ensuring that all aspects of key generation, storage, distribution, and deletion are safeguarded against quantum threats.

- **Benefits**: secure key management is foundational to cryptographic security, as it ensures that cryptographic keys are handled in a secure manner throughout their lifecycle. Enhancing these services with quantum resistance adds an additional layer of security- protecting against potential vulnerabilities introduced by quantum computing.

Annexes

Glossary

C

CMP CSR: The Certificate Management Protocol (CMP) is an Internet protocol used for obtaining X.509 digital certificates in a public key infrastructure (PKI). It is described in RFC4210 and is one of two protocols to use the Certificate Request Message Format (CRMF), described in RFC4211.

CRL: A certificate revocation list (CRL) is a list of digital certificates that have been revoked by the issuing certificate authority (CA).

D

DSA: (Digital Signature Algorithm) is a federal Information Processing Standard for digital signatures. It was developed by the U.S. National Security Agency and published by the National Institute of Standards and Technology as a U.S. Federal Information Processing Standard in 1991.

E

EJBCA: EJBCA is one of the longest running CA software projects, providing time-proven robustness and reliability. EJBCA is platform independent and can easily be scaled out.

EJBCA VA: The EJBCA Validation Authority (VA) software component enables certificate validation using OCSP or CRLs.

KEM: Key Exchange Mechanism.

L

LDAP: Lightweight Directory Access Protocol is a software protocol that enables anyone to locate organizations, individuals, and other resources such as files and devices in a network, whether on the public Internet or on a corporate intranet. Traditionally used in cyber security to support authentication and authorization of users.

N

NIST: National Institute of Standards and Technology.

O

OCSP: Online Certificate Status Protocol) is used by PKI-clients to verify the validity of certificates in real-time.

P

PKI: Public Key Infrastructure.

R

RSA: Rivest-Shamir-Adleman, is an asymmetric cryptographic algorithm used for secure data exchange, commonly used in cryptography today.

S

SSD: Solit State Drive, a type of storage device used in computers for non-volatile, persistent data.

T

Timing attacks: Timing attacks enable an attacker to extract secrets maintained in a security system by observing the time it takes the system to respond to various queries.

TLS: Transport Layer Security.

V

VPN: Virtual Private Network.

References

Alagic, G, et al (2020): *Status report on the second round of the NIST post-quantum cryptography standardization process.* Retrieved from: https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8309.pdf

Banegas, G, et al (2021): *Quantum-resistant security software updates on low-power networked embedded devices.* Retrieved from: https://arxiv.org/abs/2106.05577/

Bank for International Settlements. (2023). *Project Leap: Quantum-proofing the financial system* Retrieved from https://www.bis.org/publ/othp67.pdf

Bernstein, D. J., Hülsing, A., Kölbl, S., Niederhagen, R., Rijneveld, J., & Schwabe, P. (2019). *The SPHINCS+ signature framework. In CCS 2019 - Proceedings of the 2019 ACM SIGSAC conference on computer and communications security* (pp. 2129-2146). Association for Computing Machinery, Inc., https://doi.org/10.1145/3319535.3363229

Campagna, M et al (2015): *Quantum safe cryptography and security: an introduction, benefits, enablers and challenges*, European Telecommunications Standards Institute White Papers, no 8, June, https://www.etsi.org/images/files/ETSIWhitePapers/QuantumSafeWhitepaper.pdf.

Chen, L. , Jordan, S. , Liu, Y. , Moody, D. , Peralta, R. , Perlner, R. and Smith-Tone, D. (2016), *Report on post-quantum cryptography*, NIST Interagency/Internal Report (NISTIR), National Institute of Standards and Technology , https://doi.org/10.6028/NIST.IR.8105.

CSRC - NIST Computer Security Resource Center (2024). *CRYSTALS-Dilithium Update*. Retrieved from: https://csrc.nist.gov/Presentations/2022/CRYSTALS-Dilithium-update

——— (2016). *Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process.* Retrieved from https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf

——— (2024a). *Security (Evaluation Criteria).* Retrieved from: https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/evaluation-criteria/security-(evaluation-criteria)

ENS - L'École Normale Supérieure PSL (2020). "*Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU. Retrieved*" from: https://Falcon-sign.info/Falcon.pdf

Fouque P-A et al (2020): *"Falcon: fast-fourier Lattice-based compact signatures over NTRU"*, L'École Normale Supérieure PSL, mimeo, falcon-sign.info/falcon.pdf.

Howe, J and Westerbaan, B (2022). "*Benchmarking and Analysing NIST PQC Lattice-Based Signature Scheme Standards on the ARM Cortex M7*". Retrieved from: https://csrc.nist.gov/csrc/media/Events/2022/fourth-pqc-standardization-conference/documents/papers/benchmarking-and-analysiing-nist-pqc-lattice-based-pqc2022.pdf

Howe, J and Westerbaan, B (2023). "*Benchmarking and Analysing the NIST PQC Lattice-Based Signature Schemes Standards on the ARM Cortex M7*". Retrieved from: https://eprint.iacr.org/2022/405

National Institute of Standards and Technology (NIST) (2022). "*NIST Announces First Four Quantum-Resistant Cryptographic Algorithms*". Retrieved from: https://www.nist.gov/news-events/news/2022/07/nist-announces-first-four-quantum-resistant-cryptographic-algorithms

Bank for International Settlements (BIS)

ISSN XXXX-XXXX
ISBN XXX-XX-XXXX-XXX-X