Title*: **QSC WI#1 – Quantum-Safe Primitives**

from **Source***: CESG
Contact: Michael Groves

input for **ISG***: QSC

Contribution **For***:

Decision	
Discussion	x
Information	

Submission date*: TBC

Meeting & Allocation: **QSC#4**

1. Introduction

This document aims to give an overview of the current understanding and best practice in academia and industry about quantum-safe cryptography (QSC). Specifically, we focus on identifying and assessing cryptographic primitives that have been proposed or are being developed for efficient key establishment, integrity and authentication applications which may be potentially suitable for standardization by ETSI and subsequently used by industry to develop quantum-safe solutions for real-world applications.

QSC is a rapidly growing area of research: there are already academic conference series such as PQC, annual workshops have been established by ETSI/IQC [1] and NIST, and the European Commission has recently granted funding two QSC projects under the Horizon 2020 framework, the SAFEcrypto [2] and PQCrypto [3], [4] projects. This document draws on all these research efforts.

As agreed at the first meeting of QSC ISG, the scope of this document will cover three main areas. Section 2 discusses an assessment framework; Section 3 lists some representative cryptographic primitives, taken from recent conferences; and Section 4 gives a preliminary discussion on key sizes and how to assess these for quantum as well as classical security.

2. Assessment framework

We propose to assess candidate cryptographic primitives for standardization against the criteria below, organised under the headings of security, efficiency and deployment considerations. The following sections of this document will then go into more detail on the security considerations, and the efficiency and deployment criteria will be covered in much more detail under other Work Items.

2.1. Security

Relevant criteria under this heading would include considerations such as:

- The amount of public scrutiny and level of acceptance by the academic community.
- Confidence in the associated security proofs or reductions to hard problems by the academic community.
- Which attacks have been proposed against the primitive or underlying hard problem?
- Is the primitive suitable for use in a forward secure key establishment protocol?
- Does the primitive provide or enable multiple security features?
- How easy is it to quantify the claimed classical and quantum security levels?

- How certain are the recommended key sizes for a given level of security (e.g. 80-, 112-, 128- or 256-bits)?

We do not propose to cover side-channel attacks or any other implementation-specific considerations in this overview.

2.2. Efficiency

Relevant criteria under this heading would include considerations such as:

- How large are the recommended parameter sizes for a given level of security?
- The speed / time / number of basic arithmetic operations / round trip times to establish a key or sign/verify on a representative set of platforms or processors.
- Time to generate keys or to re-key.
- Any other practical considerations e.g. failure rate for decryption/key-establishment or maximum number of signatures.

These criteria will be covered in more detail under QSC ISG Work Item #2.

2.3. Implementation and deployment issues

Relevant criteria under this heading would include considerations such as:

- Ease of implementation (by non-experts).
- Size of Implementation (particularly relevant to FPGA and embedded devices).
- Are the key and signature sizes practical for transmission and storage (particularly on resource-limited devices)?
- Ease of integration into existing protocols or systems (e.g. is this drop-in replacement?).
- Minimising costs of changing or upgrading.
- Memory requirements during execution (especially on resource-limited devices).
- Re-use of code base (e.g. to provide authentication as well as key establishment).
- Interoperability considerations (e.g. flexibility in choice of hash function in Merkle tree schemes).
- IPR concerns or any other restrictions on usage.

These criteria will be covered in more detail under QSC ISG Work Items #2 and #4.

2.4. Application-specific or restricted-use cases

In addition to recommending one or more general-purpose primitives for key establishment, data integrity and authentication applications, we might seek to identify and recommend primitives that are particularly well suited for application-specific use cases e.g. for constrained environments, or to provide very short signatures, etc. There will also be systems which are not affected by constraints on packet or handshake sizes that are imposed by some current communications protocols or interoperability requirements.

These considerations will be covered in more detail under QSG ISG Work Item #3.

3. Primitives for consideration

Here we suggest a draft list for consideration of (mostly) public key primitives which have been proposed for use in key establishment or authentication schemes. We have not tried to be comprehensive and list every QSC primitive that has ever been proposed, but instead we will be looking to identify modern primitives that have credibility in academia, are supported by currently-active research teams, look practical for real-world applications and hence are suitable candidates for consideration by ETSI for standardization.

The references given are intended to provide QSC ISG members with useful starting points for the best and most current configurations, security analyses and recommended parameter sizes. We will of course have to verify the claims and recommendations independently ourselves further down the line, perhaps when we have narrowed the list down to a more manageable size.

We now give a brief assessment of the security properties and maturity of design for a representative set of current proposals in the academic literature.

3.1. Lattices and polynomials

A good introduction to lattice-based cryptography is [5]. With reference to the assessment criteria in Section 2.1, we briefly note that this is a very active research field with several active research groups, dedicated conferences and more than 100 papers on the IACR ePrint server over the past two years. Many schemes are based on Ring-LWE or Ring-SIS, which both have worst-case to average-case reductions from approximate short vector problems in ideal lattices. The best generic attacks against key establishment are based around lattice reduction algorithms, such as LLL and BKZ, or lattice sieving, which are also active areas of research and supported by challenge problems such as [6]. Lattice-based signatures also need to be wary of private information leakage leading to attacks such as the “parallelepiped” [7] and “zonotope” [8] attacks against NTRUSign. Resistance against quantum attacks has been specifically addressed in [9].

3.1.1. Key establishment

- NTRUEncrypt [10] – This is a well established scheme with very good efficiency properties. It lacks the formal security reductions of more modern Ring-LWE schemes but it has received many years of public scrutiny and is well regarded. Reference [11] suggests various parameter sets for the 128-, 192- and 256-bit security levels. A recent paper [12] giving the first subexponential algorithm for recovering NTRUEncrypt private keys may affect the security estimates for these parameters. Fluhrer [13] also proposed new quantum attacks on NTRUEncrypt which led to the parameters being updated in [14].

- Peikert [15] – This is a state-of-the-art proposal for a general purpose key establishment based on Ring-LWE. It offers a security reduction, actively secure and authenticated modes, and forward security. The implementation in [16] gives suggested parameters for 128-bits of classical security and 80-bits of quantum security while references [17] and [18] include parameters for a range of classical and quantum security levels. A recent paper by Alkim et al [19] describes further optimisations for the unauthenticated key exchange and gives associated parameters for 128-bits of quantum security.

- Zhang et al [20] and Ghosh-Kate [21] – These are two promising ideas for efficient two-way and one-way authenticated key agreements with security reductions from Ring-LWE. The proposals are not as thoroughly worked out as Peikert or NTRUEncrypt and offer only weak forward security. The parameters given for Zhang et al’s two-pass AKE are at the 75/80-bit and 210/230-bit security levels. The hybrid one-way AKE by Ghosh and Kate gives a parameter set that they claim offers "high security".

- HIMMO [22] – This is an interesting new key pre-distribution scheme with good efficiency properties. Although there is not a formal reduction, the main attacks against HIMMO appear to lead to two related lattice sub-problems. The underlying security mechanisms are new and would benefit from more academic scrutiny. Reference [22] includes provisional parameters, which are subject to revision by a crypto challenge [23], and reference [24] explicitly considers quantum security.

3.1.2. Authentication

- Lyubashevsky [25] [26] – The Fiat-Shamir style signature in [25] was the first to introduce the now widespread technique of rejection sampling to remove information leakage from the signatures. This was updated in [26] to produce a more efficient signature with a security reduction from Ring-LWE rather than Ring-SIS. Dagdelen et al [27] also show that, with minor modifications, the signature is secure in the quantum random-oracle model. The EUROCRYPT paper [26] contains parameter sets for a fixed, but unspecified, security level and could be used to construct others.

- NTRU-MLS [28] – NTRU-MLS is the most recent signature proposal from the designers of NTRUSign. It incorporates a form of rejection sampling and has a proof that the signatures do not leak information, but there is no formal security reduction and it has not had any published independent analysis. Reference [28] suggests parameters for the 112-, 128-, 192- and 256-bit security levels.

- Aguilar et al [29] – This is a proposal for a hash-and-sign signature which applies Lyubashevsky's rejection sampling directly to NTRUSign. It has a security reduction from Ring-SIS and provides suggested parameters for 100-, 128- and 160-bits of security.

- Güneysu-Lyubashevsky-Pöppelmann [30] – This is a version of Lyubashevky's Fiat-Shamir signature [26] with bounded uniform distributions. This simplifies the implementation but increases signature length and means that the security reduction is from a non-standard version of the decision Ring-LWE problem. The paper [30] includes parameters which they claim provide 100- and 256-bits of security, but Ducas et al [31] reduce the estimate of the smaller parameters to around 80-bits.

- BLISS [31] – BLISS is a Fiat-Shamir signature which uses bimodal Gaussian distributions and a modified rejection sampling process to reduce the signature size. The unusual construction needs more analysis and the security reduction is from the NTRU problem rather than standard Ring-SIS. Nevertheless, BLISS is the most widely cited of the recent lattice-based signature proposals. Reference [31] suggests 128-, 160- and 192-bit secure parameters. BLISS-B [32] is a variant of BLISS which uses the same parameters, but has improved key generation and signing times.

- Ducas-Lyubashevsky-Prest [33] – This hash-and-sign signature is presented as the key extraction step for an identity-based encryption scheme. It is a variant of NTRUSign which adjusts the distributions used during the signing process to avoid information leakage rather than using straightforward rejection sampling. It achieves very short signatures, but has had almost no independent analysis and does not come with a security reduction. They propose parameters for 80- and 192-bits of security.

- HIMMO [22] – HIMMO also provides authentication. Once a pairwise key has been established, the peers can run a simple challenge-response mutual authentication handshake to verify that their computed key is equal, which authenticates their identities at the same time. Further HIMMO-based extensions to this handshake exist to enable credential verification and source authentication.

3.2. Multivariate quadratics

A good reference for MQ-based cryptography is [34]. With reference to the assessment criteria in Section 2.1, we briefly note that this is an active research field with multiple active research groups and dozens of conference presentations and IACR ePrints over the past two years. Solving random systems of multivariate

quadratic equations over finite fields is NP-complete, but almost no MQ-based schemes have a full security reduction from this problem. At best, there are reductions from the problem of solving the specific type of trapdoor system used in the public key [35] or under restricted attack models [36]. Consequently, the security of most MQ-based schemes is dependent on estimates of the computational difficulty of solving the public systems using the best known generic attacks such as MinRank [37] or Gröbner bases [38] [39]. This is still an active area of research and is supported by challenge problems [34].

There seems to have been a decline in confidence in the ability of MQ to provide a secure key establishment after a series of earlier proposed schemes were broken [40] [37] [41] [42] [43]. However, there is slightly more acceptance of MQ-based signatures despite the devastating attack against the NESSIE-selected signature SFLASH [44]. Resistance against quantum attacks does not appear to have been specifically addressed in the literature. The schemes below do not appear to be suitable for forward secure key establishment protocols.

3.2.1. Key establishment

- SimpleMatrix [45] – This is a recent proposal for a multivariate encryption scheme where the public system is constructed from products of square polynomial matrices. Analysis in [46] outlined a structural attack against the scheme and highlighted an issue with the decryption failure rate. The updated rectangular version [47] avoids both of these problems. Reference [47] includes suggested parameters for 80-, 90- and 100-bits of security.

- ZHFE [48] – This is a new idea for an encryption scheme which uses a pair of high-degree HFE polynomials in a way that still allows for efficient decryption. They argue that the scheme is secure against MinRank and direct algebraic attacks, but there has been no independent analysis. Parameters are only proposed for the 80-bit security level.

3.2.2. Authentication

- Quartz [49] – This is a multivariate signature scheme based on HFEv- which has exceptionally short signatures, but long signature generation times. There is a security reduction [35] for HFEv- signatures from the problem of inverting the public system. The Quartz proposal included 80-bit secure parameters and, although initial analysis appeared to reduce this estimate [50], recent work [39] has confirmed their security.

- Gui [51] [52] – This is a new proposal for an HFEv- signature which improves the efficiency of signature generation by lowering the rank of the hidden polynomial. The security of the scheme is assessed against MinRank and direct algebraic attacks, but more analysis is required to ensure that the lower rank polynomial does not allow other attacks. Parameters are provided for 80- and 128-bits of security.

- UOV [53] – This is a multivariate signature scheme constructed using a step-wise triangular system. There is a security reduction [35] for UOV signatures from the problem of inverting the public system and this behaves like a random quadratic system under a certain class of direct algebraic attacks [36]. However, improved approaches to solving underdetermined quadratic systems [54] [55] have lowered the security estimate of the original parameters. Newer parameters with 80-, 100-, 128-, 192- and 256-bits of security can be found in [56].

- Rainbow [57] – Rainbow is a layered version of the UOV signature for faster signature generation and has a cyclic variant [58] for smaller public key sizes. There is no formal security reduction and a series of attacks [59] [60] [61] have exploited the additional structure provided by the layers. In particular, [61] appears to lower the security estimates for the larger-field parameters in [62] and may also apply to some of the more recent parameters suggested in [56].

3.3. Codes

A good reference for code-based cryptography is the introduction to [63]. With reference to the assessment criteria in Section 2.1, we briefly note that this is an active research field supported by active research groups and a small but steady stream of presentations at conferences and on the IACR ePrint server. The decoding problem for random linear codes is NP-complete, but security reductions for code-based schemes also require indistinguishability results for the actual codes involved which may not always hold [64]. The original McEliece and Niederreiter schemes based on binary Goppa codes still look very secure, however many proposals for reducing the size of the public keys by using different codes or structured public keys have been broken [65] [66] [67] [68]. The best attacks are based on information set decoding algorithms [69]. Resistance against quantum attacks has been specifically addressed in [70]. Most of the key establishment primitives below are not suitable for use in forward secure key protocols.

3.3.1. Key establishment

- McEliece [71] – The original McEliece [71] and Niederreiter [72] encryption schemes using binary Goppa codes are well established and trusted. They need to be transformed into semantically secure encryption schemes [73], but parameters have only been adjusted to account for gradual improvements to the information set decoding attack [69] [74] [75] [76]. Reference [77] suggests parameters for up to 109-bits of security and [78] includes parameters at higher security levels.

- Wild McEliece [79] [80] – This was an initially promising proposal to reduce public key sizes by using wild Goppa codes. However, there is no security reduction and recent work [81] [82] has begun to exploit the hidden algebraic structure. A range of parameters is suggested for the 128-bit security level.

- MDPC McEliece [63] – Medium-density parity check codes are probably the best current proposal for reducing the size of the public keys used in McEliece while also removing algebraic structure. There is a security reduction from the decoding problem since distinguishing medium-density parity check codes from random requires the attacker to determine the existence of codewords of a given weight. The quasi-cyclic version re-introduces some structure to further reduce the key size. There is some evidence [83] that this also reduces the security of the scheme so the quasi-cyclic version could benefit from some additional academic scrutiny. Reference [63] suggests parameters for 80-, 128- and 256-bits of security.

- LRPC McEliece [84] – This is another interesting alternative to standard McEliece which is based on low-rank parity check codes. The rank-metric decoding problem is NP-complete and the authors claim that the security reduction for medium-density parity check codes can be adapted to low-rank parity check codes. However, the quasi-cyclic version suffers from a much more signification loss of security than quasi-cyclic medium density parity check codes [85]. In general, low-rank parity check codes require more academic scrutiny to understand how the attacks work under this new metric. Reference [86] provides updated parameters with 80-, 100- and 128-bit security.

3.3.2. Authentication

- CFS [87] – This is a hash-and-sign signature based on the Niederreiter encryption scheme using binary Goppa codes. It achieves very small signature lengths, but the signature generation process is slow and it requires high-rate codes. This means that the public keys are large and the security reduction from the syndrome decoding problem given in [88] is invalidated by the high-rate Goppa code distinguisher from [89]. Parallel-CFS [90] is an updated version of the signature which blocks the decoding-one-out-of-many attacks [91] against the original. Reference [92] includes 80- and 120-bit secure parameters for Parallel-CFS.

- Cayrel et al [93] – This is an improvement of Stern's original code-based identification scheme [94] which can be converted into a signature scheme using a generalised Fiat-Shamir transform [95]. The scheme uses random q-ary codes to reduce the number of rounds required to achieve a given security level. It has a security reduction from the syndrome decoding problem and there is a quasi-cyclic version to reduce the size of the public keys. Signature generation is much more efficient than CFS, although the signatures are significantly longer. Reference [96] includes parameters for 80- and 143-bit secure signatures.

- RankSign [97] – This is a hash-and-sign signature which is similar to CFS, but uses augmented low-rank parity check codes to improve both signature generation times and public key sizes. The security reduction is from a non-standard approximate syndrome decoding problem which is assumed to be hard. There is also a cyclic version of RankSign which further decreases the size of the public keys. More analysis is required to gain confidence in the security of schemes based on low-rank parity check codes. Reference [97] contains 90-, 120- and 130-bit secure parameters for cyclic-RankSign.

3.4. Hash trees

Although it does not cover the most recent proposals, a good reference for hash-based signatures is [98]. With reference to the assessment criteria in Section 2.1, we note that there has been a resurgence of interest in the field with a few active groups, several presentations at recent conferences and some early standardisation work in the IETF [99] [100]. The original Merkle signatures [101] [102] are well understood and considered very secure, but potentially impractical due to the need to maintain state between signatures. Recent work has been focused on efficiency improvements and the issue of statefulness. Quantum attacks are limited to using Grover's algorithm to speed up the search for hash-function pre-images. The efficiency of using quantum computers to find hash-function collisions has been analysed in [103].

3.4.1. Authentication

- Merkle [102] – The original Merkle signature scheme was proposed for standardisation in the internet draft [104] which has since expired. It has a security reduction, [105] for the one-time signature together with [106] for the tree, from the collision resistance of the underlying hash function. However, the draft RFC has recently been updated [99] to use a variant of the Merkle signature scheme by Leighton and Micali which has better provable security properties [107] and shorter signatures. The draft RFC includes parameters for 128- and 256-bits of classical security.

- XMSS [108] [109] – This is a more efficient hash-based signature which uses tree chaining to increase the total number of signatures available. It has a tight security reduction from the second-preimage resistance of the hash function family and Song [110] shows that this still holds against quantum adversaries [110]. Unfortunately, the version of XMSS initially proposed for standardisation [111] was vulnerable to multi-target preimage attacks and although the revised version [100] blocks these it does not come with a security reduction. The current internet draft [100] includes parameters for 256- and 512-bits of classical security.

- SPHINCS [112] – This is a hash-based signature which avoids the need to retain state at the cost of significantly increasing the signature length. It includes a security reduction and an analysis of its security against quantum adversaries, but as it is built from the version of XMSS [111] that has been withdrawn, more analysis is needed. Reference [112] includes parameters for 128-bits of quantum security.

3.5. Isogenies

A good general reference for isogenies and elliptic curves is [113]. With reference to the assessment criteria in Section 2.1, we briefly note that this is a new research field with relatively few active research groups or publications. The security reductions are to unusual versions of the Diffie-Hellman problem and the primitives are not supported by public challenge problems, but the classical [114] [115] and quantum [116] [117] complexity of recovering unknown isogenies between elliptic curves has been well studied. This is an interesting new primitive with good properties such as small key size and forward security that deserves more academic scrutiny to establish a consensus on its security properties.

3.5.1. Key establishment

- Jao-De Feo [118] – This is a novel Diffie-Hellman style key-exchange using isogenies of supersingular curves. The extended paper [119] includes a more detailed security analysis and a reduction from a variant of the decision Diffie-Hellman problem for supersingular isogenies. The code referred to in [118] contains suggested parameters for 128-, 192- and 256-bits of classical security with various choices of isogeny degrees.

3.5.2. Authentication

- Jao-Soukharev [120] – This is an undeniable signature based on the supersingular isogeny key agreement from [118]. The signature length compares favourably to lattice-based signatures, but as it is an interactive protocol it will not be suitable for all applications. More security analysis is also required. Reference [120] includes parameters for 80-, 112- and 128-bits of quantum security.

- Sun-Tian-Wang [121] – This is a strong designated verifier signature based on the supersingular isogeny key agreement from [118]. It is a non-interactive protocol, but only verifiers chosen in advance by the signer can verify the signature so again it will not be suitable for all applications.

3.6. Other mechanisms

Although this document has mainly focussed on public key solutions, we briefly note here that it is possible to achieve quantum resistance via a range of other mechanisms. For example authentication schemes such as MACs [122], Wegman-Carter [123], and Kerberos [124] can be built from block ciphers or hash functions and the stream cipher QUAD [125] is based on multivariate quadratic equations. Quantum key distribution (QKD) has also been proposed for generating symmetric keys or one time pad [1]. By far the most impressive existing quantum-safe system is the global 3GPP mobile network which achieves authentication via pre-shared keys embedded in SIMs at manufacture, and key agreement and message integrity via a complex set of symmetric protocols, see e.g. [126]. There are many other large systems that could be configured to rely on pre-shared keys or with key distribution centres, for example TLS [127] and the ZigBee wireless mesh network for IoT applications [128], as well as protocols HIMMO and Kerberos, mentioned earlier in this document.

There will be more discussion of these under QSC ISG Work Item #3.

4. Key lengths and security

4.1. Key length

We will eventually want to recommend parameter sizes that will provide a required level of security (e.g. 80-, 112-, 128- and 256-bits) that is appropriate to the intended real-world use case. This is not entirely straightforward given the current level of understanding and confidence in the various methods proposed in academic papers.

Some primitives are well established but have large key sizes (e.g. McEliece); newer primitives are much more efficient but less well analysed (e.g. ring-LWE). Some primitives have formal security reductions to known hard problems (e.g. ring-LWE and multivariate quadratics); other primitives rely on the practical difficulty of key recovery or forgery attacks (e.g. NTRU and isogenies). Very few primitives have a reasonable rule of thumb for assessing their quantum security (see Section 4.2). We do not attempt to solve these problems here but note them for future consideration, once we have narrowed down the list of primitives.

In this section we give an illustration of the key sizes for a subset of the primitives considered in Section 3. These are based on published parameter sets and are described in terms of their *classical* (n.b. not quantum) security level. Appendix A contains a comparison of the suggested key sizes for all of the algorithms at the 128-bit classical security level wherever possible.

4.1.1. Key establishment

Figures 1 and 2 give the size of the public key and the length of the message for ten of the key establishment schemes. The message will either be the encrypted symmetric key for encryption schemes or the public key plus any additional fields for key agreements. In both figures the lengths are plotted on a *logarithmic* scale for clarity.

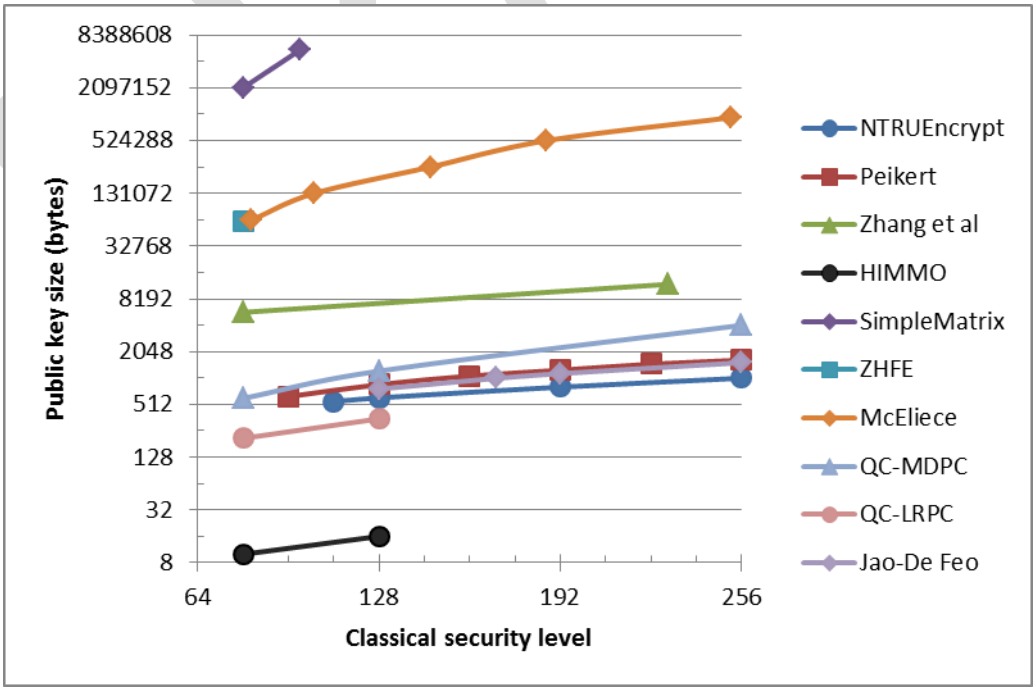

Figure 1: Public key sizes for key establishment

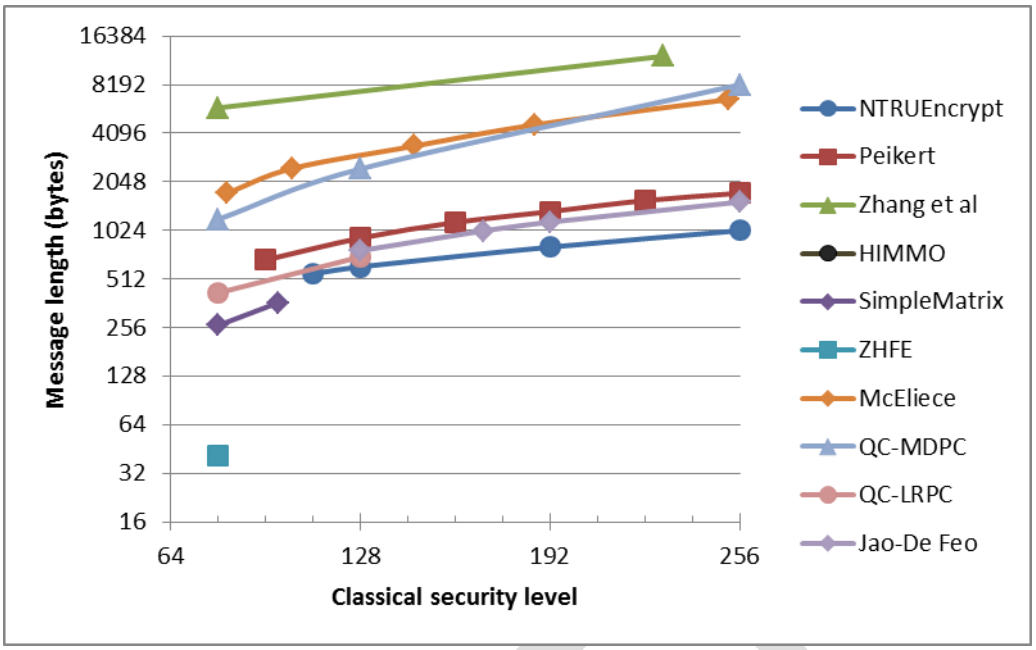

Figure 2: Message lengths for key establishment

We omit the Ghosh-Kate lattice-based key agreement from Section 3.1.1 since there is no security estimate for its parameters. For the code-based schemes we focus on the original binary Goppa code McEliece and on the more efficient quasi-cyclic versions of MDPC McEliece and LRPC McEliece from Section 3.3.1. We omit Wild McEliece as it does not achieve the same reduction in key size.

Note that the quoted "public key" size for HIMMO corresponds to the length of the user's identifier and that no additional communication is required between users in order to establish a shared secret key.

4.1.2. Authentication

Figures 3 and 4 give the public key and signature sizes for ten of the authentication schemes. Again, in both figures the sizes are plotted on a logarithmic scale for clarity.

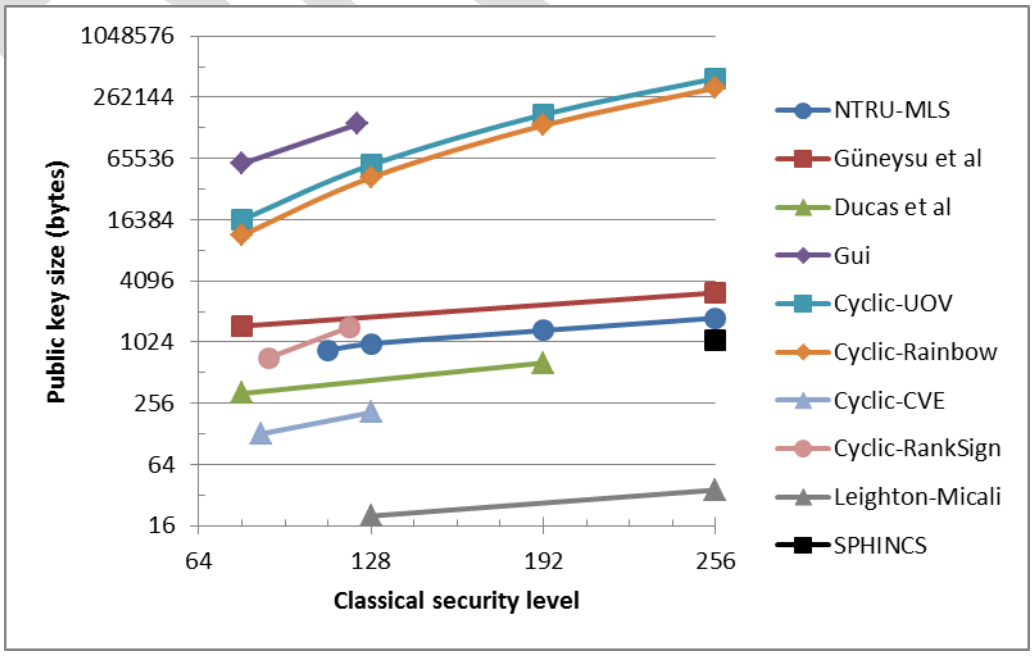

Figure 3: Public key sizes for authentication

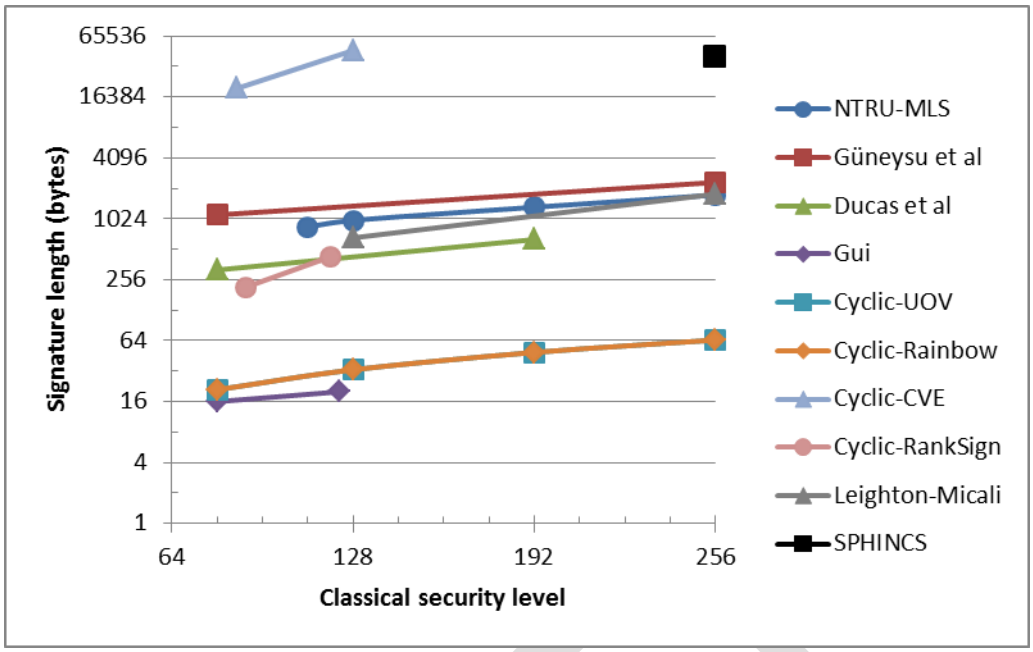

Figure 4: Signature lengths for authentication

We omit the lattice-based Lyubashevsky signature as there is no specific security estimate for the parameters; the signature by Aguiler et al since it will have similar public key sizes to, but larger signatures than, NTRU-MLS; BLISS since it is not as efficient as Ducas et al; and HIMMO as it cannot be used as a general purpose signature. For the MQ-based schemes we focus on Gui and the cyclic versions of UOV and Rainbow from Section 3.2.2. We omit QUARTZ as it is similar to the 80-bit parameters for Gui. Similarly, for code-based signatures we focus on the cyclic versions of Cayrel et al and RankSign, but omit CFS signatures because of their significantly larger key sizes. For hash-based schemes, we include Leighton-Micali over the original version of Merkle because of the smaller signature sizes and omit XMSS as it only has parameters for the higher security levels. Finally, we omit both isogeny-based schemes since they cannot be used as general purpose signatures.

Note that the Leighton-Micali parameters are limited to 2^{20} signatures and the SPHINCS parameters are limited to around 2^{60} signatures.

4.2. Quantum Security

It is interesting to note that most of the recommended key sizes in the referenced proposals are actually aimed against classical rather than quantum adversaries. The recommended key sizes against classical attackers are often based on assessing the best known attacks via various practical “crypto-challenges”, see e.g. [6], [23], [34].

It is not possible to give definitive work estimates for quantum attacks at this point in time, since it is not yet clear what a large scale future quantum computer will look like, which technologies it will be based on, or how to quantify the auxiliary resource requirements for things like fault tolerance, error correction and memory look-ups. However for some primitives it does seem possible give some reasonable “rules of thumb” based on the properties of the algorithms that will be run on a quantum computer.

Grover’s algorithm provides a “square-root speed up” over a classical adversary when searching unstructured data sets. In the context of cryptography this means that for a generic block cipher to maintain a given level of security (e.g. 128 bits) against a quantum adversary running Grover’s algorithm to recover a key, the rule of thumb should be to “double the key size” (e.g. to 256 bits). However it is important to note that that each individual iteration of Grover’s algorithm corresponds to performing one encryption of the block cipher. This is typically equivalent to several thousand basic operations (AND, XOR, etc.) on a classical computer and it is not

yet understood what the equivalent measured on a quantum computer should be. So we should not view the rule of thumb as providing a precise cost analysis for running Grover's algorithm but rather as a very conservative estimate of the resources required for the attack.

In terms of the primitives discussed in Section 3:

- The best quantum attacks for generic cryptographic hash algorithms also come from versions of Grover's search algorithm. For signatures such as Leighton-Micali, XMSS or SPHINCS whose security is based on the pre-image resistance of the hash function the rule of thumb is that we should double the hash output length to retain a given level of security. This doubles the public key size and quadruples the length of signatures. The security of the original Merkle signature scheme, such as [104], is based on collision resistance and in this case the rule of thumb is different. Early estimates claimed that Grover would provide a cube root speed up for collision finding [129] however this was disputed in [103] where it was argued that the quantum attack had the same complexity as the classical attacks. We can interpret this as saying that it is not necessary to increase the hash output length in the original Merkle signature scheme to defend against quantum attacks. As a consequence, it is not obvious whether the additional complexity of signatures such as XMSS offer any advantage over simpler Merkle signatures when considering quantum security.

- The first connection between quantum computation and a lattice problem – the $O(n^{5/2})$-unique short vector problem – was described in [130] and the first quantum attack on a lattice-based cryptographic primitive was described in [131]. However, these both addressed somewhat special cases. The best reference for quantum approaches to solving general short vector problems appears in [9] where the authors describe several algorithms which combine lattice sieving with Grover's algorithm. In general, they were able to reduce the log-complexity of the lattice sieves by up to a quarter. The usual rule of thumb, at least for Ring-LWE distinguishing attacks (see for example [132]), is that parameter sizes scale linearly in the security parameter. This means that to retain a given level of security in a Ring-LWE scheme we would likely need to increase the size of the public key by a third. NTRU-based schemes are slightly more complicated as it depends on whether Grover's algorithm speeds up the meet-in-the-middle portion of a hybrid lattice attack. If we only consider quantum improvements to the lattice reduction then a similar rule of thumb applies. Quantum speed-ups against HIMMO were explicitly considered in [24].

- Quantum improvements to information set decoding attacks against code-based systems were addressed in [70]. The rule of thumb is that quantum algorithms halve the security level so to compensate you need to double the dimension of the code. For unstructured codes this corresponds to doubling the length of the ciphertext and quadrupling the size of the public key. On the other hand, for quasi-cyclic MDPC codes this should only mean doubling the length of the ciphertext and public key although [70] does not specifically consider attacks against structured codes. There is relatively little literature on attacking rank-metric systems [85] and we have already noted above that these primitives would benefit from more academic assessment.

- There does not seem to be a good rule of thumb for MQ-based schemes. The general analysis in [133] on NP-complete problems implies that Grover's algorithm is essentially optimal for solving random quadratic systems. This means that primitives that use small systems, such as QUARTZ and Gui, will need to double the number of polynomials and variables. This doubles the signature length, but increases the size of the public keys by a factor of 8 [52]. We are not aware of any other significant research into quantum attacks against MQ-based primitives so at the moment it appears that for the remaining primitives the key size may not need to be increased.

- Biasse et al [108] describe a quantum algorithm for recovering an isogeny between two supersingular curves defined over a quadratic extension field. For the Jao-De Feo key agreement this is beaten by a quantum claw algorithm [134] which turns a fourth-root classical attack into a sixth-root quantum

attack. To maintain the security level the rule of thumb should therefore be to increase the size of the public keys by a half.

Appendix B contains rough estimates of the potential key sizes for the algorithms at the 128-bit quantum security level.

4.3. Provable, forward and active security

4.3.1. Provable security

Security reductions can give confidence that the security of a scheme is based on a problem that is known to be hard. However, it is important to understand the precise security guarantees given by the reduction. In many cases it is unwise to derive parameter sizes directly from the provable security level [135]. In some cases it may be better to ignore the reduction if it allows a more efficient scheme, provided that we can be sure that the practical security is unaffected. The question of how to extend classical security reduction techniques to quantum adversaries is a new topic of research [110].

[Next version of the document will include the security contribution from Niels.]

4.3.2. Forward security

A key establishment protocol is forward secure if the compromise of a long-term private key does not affect the security of previously established symmetric keys. It is usually considered important for modern security protocols to provide forward security [136], particularly for widespread or general purpose applications. For example, the draft specification for TLS 1.3 [137] only supports forward secure cipher suites. Consequently, it will be necessary to identify quantum-safe key establishment primitives that are suitable for use in such protocols. As forward security generally involves ephemeral keys, this means that the quantum-safe primitives must have efficient key generation and small public keys.

- Lattice-based primitives have fast key generation and, when used with ideal lattices, small public keys. In particular, NTRUEncrypt and Peikert's key encapsulation mechanism could both be used in forward secure protocols. However, the authenticated key exchanges by Zhang et al and Ghosh-Kate are two-pass protocols so can only provide forward security against passive attackers [138]. Further, HIMMO operates as a static-static key exchange so would not be able to provide forward security by itself.

- The public keys for MQ-based primitives are large and key generation can be slow. This means they would not be suitable for use in a forward secure protocol.

- Key generation for code-based key establishment primitives can be fast, but the public keys are large when used with unstructured codes. This means that only the quasi-cyclic versions of MDPC McEliece or LRPC McEliece could be used in a forward secure protocol.

- Finally, the isogeny-based key agreement has reasonable key generation and small public keys so could be used in a forward secure protocol.

4.3.3. Active security

It is also important to consider security against active adversaries since quantum-safe primitives that are only passively secure could be weak when used in certain protocols [139]. This should not be a concern for most

authentication primitives as the standard security notion for signatures is existential unforgeability against adaptively-chosen message attacks. However, there is a much wider range of security notions for key establishment primitives so it is necessary to understand their security against adaptively-chosen plaintext and ciphertext attacks. For example, malleable primitives can reveal information about the shared symmetric key (eg. Bleichenbacher's padding attack against RSA [140]) and key establishment failures can reveal information about the static private key (eg. invalid point attacks against ECDH [141]).

- Lattice-based key establishment primitives can be vulnerable when used with static key pairs. Decryption failures in NTRUEncrypt lead to a key-recovery attack [142] so it needs to be used with a plaintext-aware padding scheme that blocks these [143]. Similarly, key exchange failures in Peikert's key encapsulation mechanism mean that for active security it needs to incorporate a key validation step [15]. The Ghosh-Kate and Zhang et al AKEs are shown to be secure against certain types of active attacker, but they may not block all active attacks. For example, the generic attacks against two-pass protocols should still apply [138]. As HIMMO operates as a static-static key exchange there should be very little scope for active attacks.

- Neither of the MQ-based key establishment primitives have been analysed in terms of their active security. Further, the Simple Matrix encryption scheme has a relatively high probability of decryption failure so more study is required to understand if this reveals any information about the private key.

- All of the versions of McEliece encryption considered in Section 3.3.1 are vulnerable to a range of active attacks so must be used with a semantically secure transformation such as [73]. In [84] it is noted that decryption failures in LRPC McEliece can be prevented from leaking information by applying the Fujisaki-Okamoto transformation [144].

- The security of the isogeny-based key agreement against active attackers has not been studied, but it is likely that some form of key validation will be necessary.

It may be possible to relax some of the requirements for fully rigorous security proofs, forward security or active security for some niche or restricted use-cases.

5. Conclusions

We are fortunate that there already exists a good range of primitives to consider against the assessment framework. Some are more established than others, some are more novel but have good efficiency properties; we will need to find the right balance between these for real-world deployment.

Some preliminary conclusions at this stage of the process are:

- There are a small set of lattice- and code-based primitives that should be considered in more detail for key establishment. Other primitives seem less certain:
  - There has been a loss of confidence in MQ schemes and a move towards Ring LWE.
  - Isogeny schemes appear to have good practical properties but more research is needed for a consensus to be established around their security.

- There is much more choice for authentication schemes, where hash trees, lattices, MQ and coding schemes all look likely to provide secure signature schemes.
  - Hash trees look secure but they require re-seeding and are not suitable for all applications.
  - Lattices, MQ and coding schemes all look possible for general purpose applications and there are many different proposals to choose from.

- There are several promising new primitives (including isogenies, rank metric coding schemes and HIMMO) that would benefit from more independent academic assessment to build confidence and achieve a consensus on their security features and recommended key lengths. *Could we ask academia to help here?*

- Formal security reductions, forward security and active security are important for general purpose, widely-used protocols. However it may be possible to relax these requirements for some more restricted or application-specific use-cases.

- More work will be required on key sizes. Many of the "recommendations" in the citations are more like suggestions for further study than concrete proposals for standardization.

We recommend that after this initial assessment phase is complete the ETSI ISG QSC ISG reduce the list of primitives given above down to a more manageable size. We could then give a more detailed analysis and recommendations for a small number of proposals.

References

[1] ETSI, "Quantum safe cryptography and security," ETSI White Paper No. 8, 2015.

[2] M. O'Niell, "SAFEcrypto Project," in *NIST PQC workshop*, 2015.

[3] T. Lange, "PQCrypto project," in *NIST Workshop on Cybersecurity in a Post-Quantum World*, 2015.

[4] PQCrypto, "Initial recommendations of long-term secure post-quantum systems," pqcrypto.eu.org, 2015.

[5] C. Peikert, "A decade of lattice cryptography," IACR ePrint Archive 2015/939, 2015.

[6] TU Darmstadt, "Lattice challenge," 2015. [Online]. Available: www.latticechallenge.org.

[7] P. Q. Nguyen and O. Regev, "Learning a parallelepiped: Cryptanalysis of GGH and NTRU sigantures," in *EUROCRYPT*, 2006.

[8] L. Ducas and P. Q. Nguyen, "Learning a zonotope and more: Cryptanalysis of NTRUSign countermeasures," in *ASIACRYPT*, 2012.

[9] T. Laarhoven, M. Mosca and J. van de Pol, "Finding shortest lattice vectors faster using quantum search," in *Designs, Codes and Cryptography*, 2014.

[10] J. Hoffstein, J. Pipher and J. H. Silverman, "NTRU: A ring-based public key cryptosystem," in *ANTS III*, 1998.

[11] J. Hoffstein, J. Pipher, J. M. Schanck, J. H. Silverman, W. Whyte and Z. Zhang, "Choosing parameters for NTRUEncrypt," IACR ePrint Archive 2015/708, 2015.

[12] P. -A. Fouque and P. Kirchner, "An improved BKW algorithm for LWE with applications to cryptography and lattices," in *CRYPTO*, 2015.

[13] S. Fluhrer, "Quantum cryptanalysis of NTRU," IACR ePrint Archive 2015/676, 2015.

[14] W. Whyte, "EEES#1: Implementation aspects of NTRUEncrypt, Version 3.1," 2015. [Online]. Available: https://github.com/NTRUOpenSourceProject/ntru-crypto/blob/master/doc/EESS1-v3.1.pdf.

[15] C. Peikert, "Lattice cryptography for the internet," in *PQC*, 2014.

[16] J. W. Bos, C. Costello, M. Naehrig and D. Stebila, "Post-quantum key exchange for the TLS protocol from the ring learning with errors problem," IACR ePrint Archive 2014/599, 2014.

[17] V. Singh, "A practical key exchange for the internet using lattice cryptography," IACR ePrint Archive 138/2015, 2015.

[18] V. Singh and A. Chopra, "Even more practical key exchanges for the internet using lattice cryptography," IACR ePrint Archive 2015/1120, 2015.

[19] E. Alkim, L. Ducas, T. Pöppelmann and P. Schwabe, "Post-quantum key exchange - a new hope," IACR ePrint Archive 2015/1092, 2015.

[20] J. Zhang, Z. Zhang, J. Ding, M. Snook and O. Dagdelen, "Authenticated key exchange from ideal lattices," in *EUROCRYPT*, 2015.

[21] S. Ghosh and A. Kate, "Post-quantum forward secure onion routing (future anonymity in today's budget)," in *IACR*

ePrint Archive 2015/008, 2015.

[22] O. Garcia-Morchon, R. Rietman, L. Tolhuizen, D. Gomez and J. Gutierrez, "HIMMO - A lightweight, fully collusion resistant key pre-distribution scheme," IACR ePrint Archive 2014/698, 2014.

[23] Philips, "HIMMO challenge," 2015. [Online]. Available: www.himmo-scheme.com.

[24] O. Garcia-Morchon, R. Rietman , I. Shparlinski and L. Tolhuizen, "Results on polynomial interpolation with mixed modular operations and unknown moduli," IACR ePrint Archive 2015/1003, 2015.

[25] V. Lyubashevsky, "Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures," in *ASIACRYPT*, 2009.

[26] V. Lyubashevsky, "Lattice signatures without trapdoors," in *EUROCRYPT*, 2012.

[27] Ö. Dagdelen, M. Fischlin and T. Gagliardoni, "The Fiat-Shamir transformation in a quantum world," in *ASIACRYPT*, 2013.

[28] J. Hoffstein, J. Pipher, J. M. Schanck, J. H. Silverman and W. Whyte, "Transcript secure signatures based on modular lattices," in *PQC*, 2014.

[29] C. M. Aguilar, X. Boyen, J. C. Deneuville and P. Gaborit, "Sealing the leak on classical NTRU signatures," in *PQC*, 2014.

[30] T. Güneysu, V. Lyubashevsky and T. Pöppelmann, "Practical lattice-based cryptography: A signature scheme for embedded systems," in *CHES*, 2012.

[31] L. Ducas, A. Durmus, T. Lepoint and V. Lyubashevsky, "Lattice signatures and bimodal Gaussians," in *CRYPTO*, 2013.

[32] L. Ducas, "Accelerating BLISS: The geometry of ternary polynomials," IACR ePrint Archive 2014/874, 2014.

[33] L. Ducas, V. Lyubashevsky and T. Prest, "Efficient identity-based encryption over NTRU lattices," in *ASIACRYPT*, 2014.

[34] T. Yasuda, X. Dahan, Y.-J. Huang, T. Takagi and K. Sakurai, "MQ challenge: Hardness evaluation of solving multivariate quadratic problems," in *IACR ePrint Archive 2015/275*, 2015.

[35] K. Sakumoto, T. Shirai and H. Hiwatari, "On provable security of UOV and HFE signature schemes against chosen-message attack," in *PQC*, 2011.

[36] S. Bulygin, A. Petzoldt and J. Buchmann, "Towards provable security of the Unbalanced Oil and Vinegar signature scheme under direct attacks," in *INDOCRYPT*, 2010.

[37] A. Kipnis and A. Shamir, "Cryptanalysis of the HFE public key cryptosystem by relinearization," in *CRYPTO*, 1999.

[38] V. Dubois and N. Gama, "The degree of regularity of HFE systems," in *ASIACRYPT*, 2010.

[39] J. Ding and B. Y. Yang, "Degree of regularity for HFEv and HFEv-," in *PQC*, 2013.

[40] J. Patarin, "Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt '88," in *CRYPTO*, 1995.

[41] L. Goubin and N. T. Courtois, "Cryptanalysis of the TTM cryptosystem," in *ASIACRYPT*, 2000.

[42] P.-A. Fouque, L. Granboulan and J. Stern, "Differential cryptanalysis for multivariate schemes," in *EUROCRYPT*, 2005.

[43] V. Dubois, L. Granboulan and J. Stern, "Cryptanalysis of HFE with internal perturbation," in *PKC*, 2007.

[44] V. Dubois, P.-A. Fouque, A. Shamir and J. Stern, "Practical cryptanalysis of SFLASH," in *CRYPTO*, 2007.

[45] C. Tao, A. Diene, S. Tang and J. Ding, "Simple matrix scheme for encryption," in *PQC*, 2013.

[46] D. Moody, R. Perlner and D. Smith-Tone, "An optimal structural attack on the ABC multivariate encryption scheme," in *PQC*, 2014.

[47] C. Tao, H. Xiang, A. Petzoldt and J. Ding, "Simple Matrix - A multivariate public key cryptosystem (MPKC) for encryption," *Finite Fields and their Applications,* vol. 35, pp. 352-368, 2015.

[48] J. Porras, J. Baena and J. Ding, "ZHFE, a new multivariate public key encryption scheme," in *PQC*, 2014.

[49] J. Patarin, N. Courtois and L. Goubin, "Quartz, 128-bit long digital signatures," in *2001*, CT-RSA.

[50] N. T. Courtois, M. Daum and P. Felke, "On the security of HFE, HFEv- and Quartz," in *PKC*, 2003.

[51] J. Ding, "Gui: Revisiting multivariate digital signature schemes based on HFEv- (draft)," in *NIST Workshop on Cybersecurity in a Post-Quantum World*, 2015.

[52] A. Petzoldt, M. S. Chen, B. Y. Yang, C. Tao and J. Ding, "Design principles for HFEv- based multivariate signature schemes," in *ASIACRYPT*, 2015.

[53] A. Kipnis, J. Patarin and L. Goubin, "Unbalanced Oil and Vinegar signature schemes," in *EUROCRYPT*, 1999.

[54] L. Bettale, J.-C. Faugere and L. Perret, "Hybrid approach for solving multivariate systems over finite fields," *Journal of Mathematical Cryptology,* vol. 3, no. 3, pp. 177-197, 2009.

[55] E. Thomae and C. Wolf, "Solving underdetermined systems of multivariate quadratic equations revisited," in *PKC*,

2012.

[56] A. Petzoldt, “Selecting and reducing key sizes for multivariate cryptography,” TU Darmstadt, 2013.

[57] J. Ding and D. Schmidt, “Rainbow, a new multivariable polynomial signature scheme,” in *Applied Cryptography and Network Security*, 2005.

[58] A. Petzoldt, S. Bulygin and J. Buchmann, “CyclicRainbow – A multivariate signature scheme with a partially cyclic public key,” in *INDOCRYPT*, 2010.

[59] O. Billet and H. Gilbert, “Cryptanalysis of Rainbow,” in *Security and Cryptography for Networks*, 2006.

[60] J. Ding, B.-Y. Yang, C. H. O. Chen, M. S. Chen and C. M. Cheng, “New differential-algebraic attacks and reparametrization of Rainbow,” in *Applied Cryptography and Network Security*, 2008.

[61] E. Thomae, “A generalisation of the Rainbow band separation attack and its applications to multivariate systems,” IACR ePrint Archive 2012/223, 2012.

[62] A. Petzoldt, S. Bulygin and J. Buchmann, “Selecting parameters for the Rainbow signature scheme,” in *PQC*, 2010.

[63] R. Misoczki, J.-P. Tillich, N. Sendrier and P. S. Barreto, “MDPC-McEliece: New McEliece variants from moderate parity-check codes,” in *ISIT*, 2013.

[64] N. Sendrier, “The tightness of security reductions in code-based cryptography,” in *ITW*, 2011.

[65] A. Otmani, J.-P. Tillich and L. Dallot, “Cryptanalysis of McEliece cryptosystem based on quasi-cyclic LDPC codes,” in *Symbolic Computation and Cryptography*, 2008.

[66] C. Wieschebrink, “Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes,” in *PQC*, 2010.

[67] J.-C. Faugere, A. Otmani, L. Perret and J.-P. Tillich, “Algebraic cryptanalysis of McEliece variants with compact keys,” in *EUROCRYPT*, 2010.

[68] J.-C. Faugere, A. Otmani, L. Perret, F. De Portzamparc and J.-P. Tillich, “Structural cryptanalysis of McEliece schemes with compact keys,” in *Designs, Codes and Cryptography*, 2014.

[69] J. Stern, “A method for finding codeworkds of small weight,” in *Coding Theory and Applications*, 1988.

[70] D. J. Bernstein, “Grover vs McEliece,” in *PQC*, 2010.

[71] R. McEliece, “A public key cryptosystem based on algebraic coding theory,” DSN progress report 42.44, 1978.

[72] H. Niederreiter, “Knapsack-type cryptosystems and algebraic coding theory,” *Problems of Control and Information Theory,* vol. 15, no. 2, pp. 159-166, 1986.

[73] K. Kobara and H. Imai, “Semantically secure McEliece public-key cryptosystem - Conversions for McEliece PKC,” in *PKC*, 2001.

[74] D. J. Bernstein, T. Lange and C. Peters, “Attacking and defending the McEliece cryptosystem,” in *PQC*, 2008.

[75] A. May, A. Meurer and E. Thomae, “Decoding random linear codes in O(2^0.054n),” in *ASIACRYPT*, 2011.

[76] A. Becker, A. Joux, A. May and A. Meurer, “Decoding random binary linear codes in 2^n/20: How 1+1=0 improves information set decoding,” in *EUROCRYPT*, 2012.

[77] R. Niebuhr, M. Meziani, S. Bulygin and J. Buchmann, “Selecting parameter sizes for secure McEliece-based cryptosystems,” *International Journal of Information Security,* vol. 11, no. 3, pp. 137-147, 2012.

[78] D. J. Bernstein, T. Chou and P. Schwabe, “McBits: Fast constant-time code-based cryptography,” in *CHES*, 2013.

[79] D. J. Bernstein, T. Lange and C. Peters, “Wild McEliece,” in *SAC*, 2011.

[80] D. J. Bernstein, T. Lange and C. Peters, “Wild McEliece incognito,” in *PQC*, 2011.

[81] A. Couvreur, A. Otmani and J.-P. Tillich, “Polynomial time attack on wild McEliece over quadratic extensions,” in *EUROCRYPT*, 2014.

[82] J.-C. Faugere, L. Perret and F. De Portzamparc, “Algebraic attack against variants of McEliece with Goppa polynomial of a special form,” in *ASIACRYPT*, 2014.

[83] P. Loidreau, “On cellular code and their cryptographic applications,” in *ACCT*, 2014.

[84] P. Gaborit, G. Murat, O. Ruatta and G. Zémor, “Low rank parity check codes and their applications to cryptography,” in *WCC*, 2013.

[85] A. Hauteville and J.-P. Tillich, “New algorithms for decoding in the rank metric and an attack on the LRPC cryptosystem,” ArXiv Preprint 1504.05431, 2015.

[86] P. Gaborit, O. Ruatta, J. Schrek and G. Zémor, “New results for rank-based cryptography,” in *AFRICACRYPT*, 2014.

[87] N. T. Courtois, M. Finiasz and N. Sendrier, “How to achieve a McEliece-based digital signature scheme,” in *ASIACRYPT*, 2001.

[88] L. Dallot, “Towards a concrete security proof of Courtois, Finiasz and Sendrier signature scheme,” in *Research in Cryptology*, 2008.

[89] J.-C. Faugere, V. Gauthier, A. Otmani, L. Perret and J.-P. Tillich, “A distinguisher for high rate McEliece cryptosystems,” in *ITW*, 2011.

[90] M. Finiasz, “Parallel-CFS,” in *SAC*, 2011.

[91] N. Sendrier, “Decoding one out of many,” in *PQC*, 2011.

[92] G. Landais and N. Sendrier, “Implementing CFS,” in *INDOCRYPT*, 2012.

[93] P.-L. Cayrel, P. Véron and S. M. El Yousfi Alaoui, “A zero-knowledge identification scheme based on the q-ary syndrome decoding problem,” in *SAC*, 2010.

[94] J. Stern, “A new identification scheme based on syndrome decoding,” in *CRYPTO*, 1993.

[95] S. M. El Yousfi Alaoui, Ö. Dagdalen, P. Véron, D. Galindo and P.-L. Cayrel, “Extended security arguments for signature schemes,” in *AFRICACRYPT*, 2012.

[96] P.-L. Cayrel, S. M. El Yousfi Alaoui, F. Günther, G. Hoffmann and H. Rother, “Efficient implementation of code-based identification schemes,” in *Western European Workshop on Research in Cryptology*, 2011.

[97] P. Gaborit, O. Ruatta, J. Schrek and G. Zémor, “RankSign: an efficient signature algorithm based on the rank metric,” in *PQC*, 2014.

[98] J. Buchmann, E. Dahmen and M. Szydlo, “Hash-based digital signature schemes,” in *PQC*, 2009.

[99] D. McGrew and M. Curcio, “Hash-based signatures (Draft RFC),” draft-mcgrew-hash-sigs-03, 19 October 2015.

[100] A. Hülsing, D. Butin, S. Gazdag and A. Mohaisen, “XMSS: Extended hash-based signatures (Draft RFC),” draft-irtf-cfrg-xmss-hash-based-signatures-01, July 3 2015.

[101] R. C. Merkle, “Secrecy, authentication and public-key systems,” Stanford University, 1979.

[102] R. C. Merkle, “A certified digital signature,” in *CRYPTO*, 1989.

[103] D. J. Bernstein, “Cost analysis of hash collisions: Will quantum computers make SHARCS obsolete?,” in *SHARCS*, 2009.

[104] D. McGrew and M. Curcio, “Hash-based signatures (Draft RFC),” draft-mcgrew-hash-sigs-02, 4 July 2014.

[105] C. Dods, N. P. Smart and M. Stam, “Hash based digital signature schemes,” in *Cryptography and Coding*, 2005.

[106] L. C. C. Garcia, “On the security and efficiency of the Merkle signature scheme,” IACR ePrint Archive 2005/192, 2005.

[107] J. Katz, “Analysis of a proposed hash-based signature standard,” 2015. [Online]. Available: https://www.cs.umd.edu/~jkatz/papers/HashBasedSigs.pdf.

[108] J. Buchmann, E. Dahmen and A. Hülsing, “XMSS - A practical forward secure signature scheme based on minimal security assumptions,” in *PQC*, 2011.

[109] A. Hülsing, L. Raush and J. Buchmann, “Optimal parameters for XMSS^MT,” in *Security Engineering and Intelligence Informatics*, 2013.

[110] F. Song, “A note on quantum security for post-quantum cryptographty,” in *PQC*, 2014.

[111] A. Hülsing, D. Butin, S. Gazdag and A. Mohaisen, “XMSS: Extended hash-based signatures (Draft RFC),” draft-irtf-cfrg-xmss-hash-based-signatures-00, 8 April 2015.

[112] D. J. Bernstein, D. Hopwood, A. Hülsing, T. Lange, R. Niederhagen, L. Papachristodoulou, M. Schneider, P. Schwabe and Z. Wilcox-O'Hearn, “SPHINCS: practical stateless hash-based signatures,” in *EUROCRYPT*, 2015.

[113] L. Washington, Elliptic curves: number theory and cryptography, CRC Press, 2008.

[114] S. Galbraith and A. Stolbunov, “Improved algorithm for the isogeny problem for ordinary elliptic curves,” *Applicable Algebra in Engineering, Communication and Computing,* vol. 24, no. 2, pp. 107-131, 2013.

[115] C. Delfs and S. D. Galbraith, “Computing isogenies between supersingular elliptic curves over Fp,” in *Designs, Codes and Cryptography*, 2014.

[116] A. Childs, D. Jao and V. Soukharev, “Constructing elliptic curve isogenies in quantum subexponential time,” *Journal of Mathematical Cryptology,* vol. 8, no. 1, pp. 1-29, 2014.

[117] J.-F. Biasse, D. Jao and A. Sankar, “A quantum algorithm for computing isogenies between supersingular elliptic curves,” in *INDOCRYPT*, 2014.

[118] D. Jao and L. De Feo, “Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies,” in *PQC*, 2011.

[119] L. De Feo, D. Jao and J. Plût, “Towards quantum-resistant cryptosystems from supersingular elliptic curve

isogenies," *Journal of Mathematical Cryptography,* vol. 8, no. 3, pp. 209-247, 2014.

[120] D. Jao and V. Soukharev, "Isogeny-based quantum-resistant undeniable signatures," in *PQC*, 2014.

[121] X. Sun, H. Tian and Y. Wang, "Toward quantum-resistant strong designated verifier signature from isogenies," in *Intelligent Networking and Collaborative Systems*, 2012.

[122] B. Schneier, Applied cryptography, John Wiley and Sons, 1996.

[123] J. Carter and M. Wegman, "Universal classes of hash functions," in *ACM Symposium on Theory of Computing*, 1977.

[124] C. Neuman, T. Yu, S. Hartman and K. Raeburn, "The Kerberos network authentication service (V5)," RFC 4120, 2005.

[125] C. Berbain, H. Gilbert and J. Patarin, "QUAD: A practical stream cipher with provable security," in *EUROCRYPT*, 2006.

[126] C. Blanchard, "Security for the third generation (3G) mobile system," *Information Security Technical Report,* vol. 5, no. 3, pp. 55-65, 2000.

[127] P. Eronen and H. Tschofenig, "Pre-Shared Key Ciphersuites for TLS," RFC 4279, 2005.

[128] Zigbee, "Zigbee alliance website," www.zigbee.org, 2015.

[129] G. Brassard, P. Hoyer and A. Tapp, "Quantum cryptanalysis of hash and claw-free functions.," *LATIN,* 1998.

[130] O. Regev, "Quantum Computation and Lattice Problems," arXiv:cs/0304005, 2003.

[131] P. Campbell, M. Groves and D. Shepherd, "Soliloquy: A cautonary tale," in *2nd ETSI Quantum Safe Workshop*, 2014.

[132] M. R. Albrecht, R. Player and S. Scott, "On the concrete hardness of learning with errors," IACR ePrint Archive 2015/046, 2015.

[133] C. H. Bennett, E. Bernstein, G. Brassard and U. Vazirani, "Strengths and weaknesses of quantum computing," *SIAM Journal on Computing,* vol. 26, no. 5, pp. 1519-1523, 1997.

[134] S. Tani, "Claw finding algorithms using quantum walk," *Theoretical Computer Science,* vol. 510, no. 50, pp. 5285-5297, 2009.

[135] C. Bader, T. Jager, Y. Li and S. Schäge, "On the impossibility of tight cryptographic reductions," IACR ePrint Archive 2015/374, 2015.

[136] M. Bellare and B. Yee, "Forward-security in private-key cryptography," in *CT-RSA*, 2003.

[137] E. Resorla, "The Transport Layer Security (TLS) protocol version 1.3 (Draft RFC)," draft-ietf-tls-tls13-09, 5 October 2015.

[138] H. Krawczyk, "HMQV: A high performance secure Diffie-Hellman protocol," in *CRYPTO*, 2005.

[139] M. Motley, "Failure is not an option: standardization issues for post-quantum key agreement," in *NIST Workshop on Cybersecurity in a Post-Quantum World*, 2015.

[140] D. Bleichenbacher, "Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS#1," in *CRYPTO*, 1998.

[141] I. Biehl, B. Meyer and V. Müller, "Differential fault attacks on elliptic curve cryptosystems," in *CRYPTO*, 2000.

[142] N. Howgrave-Graham, P. Q. Nguyen, D. Pointcheval, J. Proos, J. H. Silverman, A. Singer and W. Whyte, "The impact of decryption failures on the security of NTRU encryption," in *CRYPTO*, 2003.

[143] N. Howgrave-Graham, J. H. Silverman, A. Singer and W. Whyte, "NAEP: Provable security in the presence of decryption failures," IACR ePrint Archive 2003/172, 2003.

[144] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes," in *CRYPTO*, 1999.

6. Appendix A – Classical key size comparison

The tables given here compare the suggested key sizes for the algorithms listed in Section 3, based wherever possible on the author's suggestions for the 128-bit *classical* (i.e. not quantum) security level. For the key establishment schemes we list the size of the public key and the length of the message. The message will either be the encrypted symmetric key for encryption schemes or the public key plus any additional fields for key agreements. For the authentication schemes we list the size of the public key and the length of the signature. In both cases we omit the private keys as they can often be compressed by deriving them deterministically from a smaller seed.

6.1. Key establishment

Note: These figures are intended as informal guidance only. They have not been independently validated by QSC ISG and are not endorsed by ETSI.

Type	Scheme	Security	Public key	Message	Comments
Lattice	NTRUEncrypt	128 bits	610 bytes	610 bytes	[14]
	Peikert	128 bits	864 bytes	918 bytes	[18], Note 1
	Zhang et al	140 bits	4,096 bytes	4,224 bytes	[20], Note 2
	Ghosh-Kate	---	1,344 bytes	1,440 bytes	[21], Note 3
	HIMMO	128 bits	16 bytes	---	[22], Note 4
MQ	SimpleMatrix	100 bits	5,669,88 bytes	364 bytes	[47]
	ZHFE	80 bits	63,566 bytes	42 bytes	[48]
Code	McEliece	129 bits	221,646 bytes	512 bytes	[78]
	Wild McEliece	128 bits	89,988 bytes	535 bytes	[80], Note 5
	MDPC McEliece	128 bits	12,142,592 bytes	2,464 bytes	[63]
	QC-MDPC McEliece	128 bits	1,232 bytes	2,464 bytes	[63]
	LRPC McEliece	128 bits	18,610 bytes	703 bytes	[86]
	DC-LRPC McEliece	128 bits	352 bytes	703 bytes	[86]
Isogeny	Jao-De Feo	128 bits	768 bytes	768 bytes	[119]

Notes:

1. The message size is for the passively secure key agreement from [15]. The additional field used for key validation in the actively secure agreement would likely lead to a 934 byte message.
2. These parameters are only for the one-pass authenticated key establishment. There are no suggested 128-bit secure parameters for the two-pass protocol, but the 210-bit secure parameters have a 12,800 byte public key and a 13,056 byte message.
3. Ghosh and Kate state that their parameters offer "high security" but do not give a specific security estimate. The listed public key and message sizes are only for the Ring-LWE component of their hybrid key establishment.
4. These are 128-bit secure parameters where the identifiers are not hashed. The quoted size of the "public key" is the length of the user's identifier. No additional communication is required to establish a shared symmetric key between a pair of users.
5. These are the 128-bit secure parameters with q=31.

6.2. Authentication

Note: These figures are intended as informal guidance only. They have not been independently validated by QSC ISG and are not endorsed by ETSI.

Type	Scheme	Security	Public key	Signature	Comments
Lattice	Lyubashevsky	---	1,664 bytes	2,560 bytes	[26], Note 1
	NTRU-MLS	128 bits	988 bytes	988 bytes	[28]
	Aguilar et al	128 bits	1,082 bytes	1,894 bytes	[29], Note 2
	Güneysu et al	80 bits	1,472 bytes	1,120 bytes	[30], Note 3
	BLISS	128 bits	896 bytes	640 bytes	[31], Note 4
	Ducas et al	80 bits	320 bytes	320 bytes	[33]
	HIMMO	128 bits	32 bytes	---	[22], Note 5
MQ	Quartz	80 bits	72,237 bytes	16 bytes	[49]
	Ding	123 bits	142,576 bytes	21 bytes	[52]
	UOV	128 bits	413,145 bytes	135 bytes	[56], Note 6
	Cyclic-UOV	128 bits	60,840 bytes	135 bytes	[56], Note 6
	Rainbow	128 bits	139,363 bytes	79 bytes	[56], Note 6
	Cyclic-Rainbow	128 bits	48,411 bytes	79 bytes	[56], Note 6
Code	Parallel-CFS	120 bits	503,316,480 bytes	108 bytes	[92]
	Cayrel et al	128 bits	10,920 bytes	47,248 bytes	[93], Note 7
	Cyclic-Cayrel et al	128 bits	208 bytes	47,248 bytes	[93], Note 7
	RankSign	130 bits	7,200 bytes	1,080 bytes	[97]
	Cyclic-RankSign	130 bits	3,538 bytes	1,080 bytes	[97]
Hash	Merkle	128 bits	32 bytes	1,731 bytes	[104], Note 8
	Leighton-Micali	128 bits	20 bytes	668 bytes	[99], Note 9
	XMSS	256 bits	64 bytes	8,392 bytes	[100], Note 10
	SPHINCS	256 bits	1,056 bytes	41,000 bytes	[112]
Isogeny	Jao-Soukharev	128 bits	768 bytes	1,280 bytes	[120], Note 11
	Sun-Tian-Wang	128 bits	768 bytes	16 bytes	[121], Note 12

Notes:

1. These are the parameters that are compatible with the Ring-LWE version of the signature. No specific security estimates are given in [26].
2. These are the 128-bit secure size-optimised parameters.
3. These are the smaller parameters from [30], but the estimate of their security was reduced to 80 bits in [31].
4. These are the 128-bit secure size-optimised parameters and the quoted signature length includes the use of additional compression techniques.
5. These are 128-bit secure parameters where the “public key” is a 256-bit hash of the user’s credentials. No additional communication is required to implicitly authenticate a pair of users.
6. The 128-bit secure parameters for UOV, Cyclic-UOV, Rainbow and Cyclic-Rainbow are all over GF(256).
7. The signature length has been estimated by scaling the communication costs for the identification scheme to give a forgery cost of 128 bits.
8. These are the smallest 128-bit secure parameters with binary trees that allow up to 2^{20} signatures.
9. These are the smallest 128-bit secure parameters that allow up to 2^{20} signatures.
10. These are the smallest 256-bit secure parameters that allow up to 2^{60} signatures.
11. The quoted “signature length” is the length of the commitment sent by the signer during the confirmation and disavowal protocols.
12. This assumes that the signature uses the 128-bit secure parameters for the underlying isogeny-based key agreement [119] together with a 128-bit hash function.

7. Appendix B – Quantum key size comparison

The tables given here provide estimated key sizes for the algorithms listed in Section 3 for the 128-bit *quantum* security level. As in Appendix A, for the key establishment schemes we list the size of the public key and the length of the message and for the authentication schemes we list the size of the public key and the length of the signature. When possible we take the author's suggested parameters for 128-bits of quantum security. For primitives where these are not available we roughly estimate the key sizes by taking published parameters at appropriate classical security levels and adjusting them using the rules of thumb from Section 4.3.

7.1. Key establishment

Note: These figures are intended as informal guidance only. They have not been independently validated by QSC ISG and are not endorsed by ETSI.

Type	Scheme	Security	Public key	Message	Comments
Lattice	NTRUEncrypt	128 bits	610 bytes	610 bytes	[14], Note 1
	Peikert	128 bits	1,080 bytes	1,148 bytes	[18], Note 2
	Zhang et al	120 bits	3,840 bytes	3,968 bytes	[20], Note 3
	Ghosh-Kate	---	---	---	Note 4
	HIMMO	128 bits	48 bytes	---	[24], Note 5
MQ	SimpleMatrix	100 bits	5,669,88 bytes	364 bytes	[47], Note 6
	ZHFE	80 bits	63,566 bytes	42 bytes	[48], Note 6
Code	McEliece	131 bits	1,046,739 bytes	870 bytes	[78], Note 7
	Wild McEliece	128 bits	424,899 bytes	1,070 bytes	[80], Note 8
	MDPC McEliece	128 bits	134,242,305 bytes	8,193 bytes	[63], Note 9
	QC-MDPC McEliece	128 bits	4,097 bytes	8,193 bytes	[63], Note 9
	LRPC McEliece	---	---	---	Note 10
	DC-LRPC McEliece	---	---	---	Note 10
Isogeny	Jao-De Feo	128 bits	1,152 bytes	1,152 bytes	[118]

Notes:

1. These are the 128-bit classically secure parameters which [14] suggests also provide 128-bits of quantum security.
2. These are the 160-bit classically secure parameters which [18] suggests provide 128-bits of quantum security.
3. These are the 160-bit classically secure one-pass parameters which the rule of thumb suggests have 120-bits of quantum security. Although they are smaller than the parameters in Section 6.1, the probability of a key agreement failure has increased.
4. The parameters suggested for the Ghosh-Kate AKE do not come with a security estimate.
5. These are the suggested parameters with 256-bit symmetric keys and 384-bit hashed identifiers.
6. Quantum algorithms do not appear to affect the security of the MQ-based key establishment primitives.
7. These are the 263-bit classically secure parameters which should provide 131-bits of quantum security. They are also the parameters in the initial recommendations from the PQCrypto project [4].
8. These are the 128-bit classically secure parameters for q=31 with the length of the code doubled.
9. These are the 256-bit classically secure parameters from [63] which should provide 128-bits of quantum security. The dimension of the code has more than tripled rather than doubling as suggested by the rule of thumb.
10. It is not clear how quantum algorithms will affect the security of rank-based McEliece.

7.2. Authentication

Note: These figures are intended as informal guidance only. They have not been independently validated by QSC ISG and are not endorsed by ETSI.

Type	Scheme	Security	Public key	Signature	Comments
Lattice	Lyubashevsky	----	----	---	Note 1
	NTRU-MLS	128 bits	1,138 bytes	1,138 bytes	[28], Note 2
	Aguilar et al	120 bits	1,238 bytes	2,100 bytes	[29], Note 3
	Güneysu et al	128 bits	2,300 bytes	1,800 bytes	[30], Note 4
	BLISS	120 bits	896 bytes	768 bytes	[31], Note 5
	Ducas et al	128 bits	580 bytes	580 bytes	[33], Note 6
	HIMMO	128 bits	48 bytes	---	[24], Note 7
MQ	Quartz	80 bits	577,896 bytes	32 bytes	[49], Note 8
	Ding	123 bits	1,140,608 bytes	42 bytes	[52], Note 8
	UOV	128 bits	413,145 bytes	135 bytes	[56], Note 9
	Cyclic-UOV	128 bits	60,840 bytes	135 bytes	[56], Note 9
	Rainbow	128 bits	139,363 bytes	79 bytes	[56], Note 9
	Cyclic-Rainbow	128 bits	48,411 bytes	79 bytes	[56], Note 9
Code	Parallel-CFS	120 bits	2,013,265,920 bytes	216 bytes	[92], Note 10
	Cayrel et al	128 bits	43,680 bytes	94,496 bytes	[93], Note 10
	Cyclic-Cayrel et al	128 bits	416 bytes	94,496 bytes	[93], Note 10
	RankSign	---	---	---	Note 11
	Cyclic-RankSign	---	---	---	Note 11
Hash	Merkle	128 bits	32 bytes	1,731 bytes	[104], Note 12
	Leighton-Micali	128 bits	34 bytes	1,740 bytes	[99], Note 13
	XMSS	128 bits	64 bytes	8,392 bytes	[100], Note 14
	SPHINCS	128 bits	1,056 bytes	41,000 bytes	[112]
Isogeny	Jao-Soukharev	128 bits	1,152 bytes	1,152 bytes	[120]
	Sun-Tian-Wang	128 bits	1,152 bytes	32 bytes	[121], Note 15

Notes:

1. The parameters for Lyubashevksy's signature given in [26] do not come with security estimates.
2. These are the 128-bit classically secure parameters scaled up by a third according to the rule of thumb for lattice-based primitives.
3. These are the 160-bit classically secure size-optimised parameters which the lattice-based rule of thumb suggests provide 120-bits of quantum security.
4. These are extrapolated from the 80- and 256-bit classically secure parameters which the lattice-based rule of thumb suggests should provide 60- and 192-bits of quantum security.
5. These are the 160-bit classically secure size-optimised parameters which the lattice-based rule of thumb suggests provide 120-bits of quantum security.
6. These are extrapolated from the 80- and 192-bit classically secure parameters which the lattice-based rule of thumb suggests should provide 60- and 144-bits of quantum security.
7. These are the suggested parameters with 256-bit symmetric keys and 384-bit hashed credentials.
8. The rule of thumb for small MQ systems suggests that to maintain the level of security for Quartz and Gui the signature length should double and the public key should increase by a factor of 8.
9. The rule of thumb for large MQ systems suggests that no changes are needed for the UOV and Rainbow signatures.
10. The rule of thumb for code-based primitives suggests that the signature lengths for Parallel-CFS and unstructured Cayrel et al should double and the public key sizes should quadruple whereas for cyclic Cayrel et al both should double.
11. It is not clear how quantum algorithms will affect the security of RankSign.

12. The rule of thumb is that no changes are needed for hash-based signatures whose security is based on the collision resistance of the hash function.
13. These are the smallest parameters with 128-bits of quantum security that allow up to 2^{20} signatures.
14. These are the smallest parameters with 128-bits of quantum security that allow up to 2^{60} signatures.
15. This assumes that the signature uses the 128-bit quantum secure parameters for the underlying isogeny-based key agreement [119] together with a 256-bit hash function.