National Cyber Security Centre
Ministry of Justice and Security

Guidelines for quantum-safe transport-layer encryption

These guidelines are written for an audience of architects responsible for specifying cryptographic requirements. They can also be used in R&D and prototyping as well as for contract negotiations. For a more general introduction, see NLNCSA's brochure and our own factsheet. For further details, follow NIST, ETSI, IETF, and ISO standardisation efforts and read publications by ENISA and TNO.

Our recommendations target the early adopters who follow our advice to apply quantum-safe cryptography to ensure long-term confidentiality against store-and-decrypt attacks. Signatures are not part of these guidelines as they are not vulnerable to such attacks. The guidelines recommend hybrid key exchange to mitigate potential vulnerabilities in novel post-quantum algorithms and implementations. Besides a list of algorithms and recommended parameters, this document also contains some questions to ask when choosing implementations.

Combine traditional algorithms with quantum-safe key encapsulation

Key agreement should rely on multiple algorithms. For other purposes, apply established methods. You should use algorithms that have stood the test of time and that are future-proof. However, post-quantum cryptography is a new and fast-moving field. As such, ensure that you can quickly replace any algorithms and implementations that you rely on – so-called cryptographic agility.

Use all of the following standard cryptographic algorithms[*]:

— AES-256-GCM or ChaCha20-Poly1305 (for bulk encryption)
— SHA-256 or SHA3-256 (for hashing, viz. key derivation)
— ECDSA-secp256r1 or Ed25519 (for certificate verification)
— ECDH-secp256r1 or ECDH-X25519 (for key exchange)

Combine these with at least one of the following quantum-safe key encapsulation mechanisms:

— FrodoKEM at level 3+ (frodokem976 or higher)
— Classic McEliece at level 3+ (mceliece460896 or higher)
— CRYSTALS-Kyber at level 5 (kyber1024)

Apply one of the following key derivation mechanisms to get a hybrid construction:

— Concatenation of shared secrets (as specified by NIST in SP 800-56C Rev. 2) using HKDF-256
— Cascade of shared secrets (as specified by ETSI in TS 103 744) using HKDF-256

Alternatively, protocol stacking is another possible approach, where at least one of the protocols supports the standard cryptographic algorithms given above and where one or more protocols provide a quantum-safe key encapsulation mechanism. In a situation where TLS is used as the protocol that implements standard cryptographic algorithms, note our guidelines for TLS.

---

[*] Longer hash functions and elliptic curves of the same type can be used, e.g. of 384 or 512 bits. Note that other AEAD modes which use a synthetic IV are less brittle, but also less performant.

Only choose a KEM with shorter keys following a risk assessment

For FrodoKEM and Classic McEliece, a minimum key length is specified based on previous NLNCSA, BSI, and ANSSI recommendations; level 5 parameters can also be used. Note that using Kyber or another structured-lattice algorithm is riskier. As such, longer keys are recommended until enough confidence has been gained in them. However, this does not exclude novel attacks that may be discovered on structured lattices. When shorter keys are used – as with kyber768 and sntrup761 – a risk assessment should be carried out with a decision taken by the asset owner, which should be recorded, tracked, and regularly revisited based on the level of uncertainty.[†]

As noted in our factsheet on migration planning, a clear view of information assets and data flows helps to feed risk assessment decisions. Additionally, the involvement of professionals in the area of applied cryptography may be valuable, especially when looking into the use of riskier parameters and when faced with specific constraints. Either way, consider applying mitigating measures such as over-provisioning systems (so that they support stronger but heavier cryptography) and testing that implementations can be replaced when necessary. This will also be useful in case any future standard ends up deviating from the most recent specifications.[‡]

Use production-grade implementations that have been suitably vetted

Due to the critical role that cryptography plays in securing information, implementations should be mature and assured commensurate with the sensitivity of the data involved. This applies both to traditional cryptography as well as to quantum-safe cryptography – although the latter is relatively immature, especially when it comes to implementations. Given this immaturity, it is vital that a system's architecture enables agility: easily replaceable implementations are a prudent safety net. Even so, using production-grade and vetted implementations is an important principle to aim for.

Besides implementation quality, there are various other aspects that require attention. Contextual factors include stakeholder acceptance of performance and latency overheads, their risk appetite, available budget, switching costs, and interoperability concerns. Such considerations influence how trade-offs are made. By addressing these issues proactively during the life-cycle of existing and future deployments, it will be easier to achieve flexible systems that remain aligned to standards.

Colophon

The following organisations and individuals have contributed to these guidelines: I&W, NLNCSA, Andreas Hülsing (TU/e), Anthony Hu (wolfSSL), Bas Westerbaan (Cloudflare), Marc Stevens (CWI).

Version 1.0, May 2022. This information is not legally binding. It is to be updated yearly.

National Cyber Security Centre (NCSC)
Turfmarkt 147, 2511 DP, The Hague

info@ncsc.nl
070 751 5555

[†] The same applies when choosing symmetric key lengths, e.g. AES-128, AES-192, or AES-256.
[‡] Note that if such changes are made, experts should be given time to study these modifications.