الهيئة الوطنية للأمن الـسيبـراني
National Cybersecurity Authority

National Cryptographic Standards

(NCS – 1 : 2020)

Sharing Notice: White
Document Classification: Open

In the Name of Allah,
The Most Gracious,
The Most Merciful

Traffic Light Protocol (TLP):

This marking protocol is widely used around the world. It has four colors (traffic lights):

🔴 **Red – Personal, Confidential and for Intended Recipient Only**

The recipient has no rights to share information classified in red with any person outside the defined range of recipients either inside or outside the organization.

🟠 **Amber – Restricted Sharing**

The recipient may share information classified in amber only with intended recipients inside the organization and with recipients who are required to take action related to the shared information.

🟢 **Green – Sharing within the Same Community**

The recipient may share information classified in green with other recipients inside the organization or outside it within the same sector or related to the organization. However, it is not allowed to exchange or publish this information on public channels.

⚪ **White - No Restrictions**

Updates to Document

Version	Date	Changes
1.0	July 2020	Initial version

Table of Contents:

Executive Summary

As per the mandate of the National Cybersecurity Authority (NCA) that was issued by the Royal Order number 6801, dated October 31, 2017, the NCA is mandated to draft the national cryptographic policies and standards, to ensure compliance with these standards and policies, and to review and update them periodically.

From this perspective, the NCA has developed the National Cryptographic Standards (NCS –1:2020) to prescribe the minimum acceptable cryptographic requirements for civilian and commercial purposes to protect national data, systems and networks. This document highlights the details of the national cryptographic standards, which are composed of two strength levels: MODERATE and ADVANCED.

This standards document prescribes the accepted symmetric and asymmetric primitives, symmetric and asymmetric schemes, some of the accepted common application protocols related to cryptography, Public Key Infrastructure (PKI) and Key Lifecycle Management (KLM). It further presents appendices that include some considerations regarding Pseudo Random Number Generation (PRNG), Post-Quantum Cryptography and Side-Channel Attacks.

1 Introduction

1.1 Scope

The National Cryptographic Standards (NCS - 1: 2020) defines the minimum cryptography requirements to be met by national entities when using cryptography to protect data (in use, at rest and in transit), systems and networks for civilian or commercial purposes.

This cryptographic standards document is built with consideration of current and foreseeable advancements in the affordable computational power and assuming the absence of quantum computing capabilities. The document covers cryptography primitives, schemes, main protocols, Public Key Infrastructure (PKI) and Key Lifecycle Management (KLM).

While this document prescribes the minimum acceptable cryptographic requirements, it is extremely crucial to implement these requirements carefully to avoid implementation-based vulnerabilities. The NCS will be updated on an as-needed basis. The latest released version of the NCS overrides previous versions.

1.2 Levels of Cryptographic Standards

The NCS defines two strength levels for cryptographic standards: the MODERATE level and the ADVANCED level. Having two strength levels provides more flexibility to choose the appropriate level of protection for different classes of data, systems and networks. Each national entity is required to choose and implement the appropriate cryptographic standard level based on the nature and sensitivity of the data, systems and networks to be protected. Furthermore, other cybersecurity regulations, issued by the NCA, may mandate the use of a particular cryptographic standard level to protect data, systems and networks. The MODERATE and ADVANCED strength levels are designed to target 128-bit and 256-bit security levels, respectively. Specific requirements for each strength level are specified throughout this document. Any requirement not specifically associated with one of these two strength levels applies equally to both.

1.3 Structure of the Document

The rest of this document is organized as follows. Section 2 lists the accepted symmetric and asymmetric primitives with their key, block and initialization vector sizes. Section 3 provides the accepted symmetric and asymmetric schemes: block cipher modes of operation, MAC, AEAD, key wrap functions, key derivation functions, key agreement, key transport, hybrid encryption and public key signatures. Section 4 provides requirements for the most prevalent application protocols such as DNS Security (DNSSEC), IP Security (IPsec), Bluetooth, SSH, TLS, UMTS/LTE/5G, WPA and Kerberos. Section 5 provides a list of the accepted algorithms for certificates and the validity of the certificates. Section 6 provides the requirements for the different steps of the key lifecycle to ensure that keys are managed properly from the moment they are created until their destruction, and their usage is standardized across processes. Finally, Section 7 provides appendices that present some information about Pseudo Random Number Generation (PRNG), Post-Quantum Cryptography, Side-Channel Attacks, Definitions and Acronyms.

2 Cryptographic Primitives

2.1 Symmetric Algorithms

2.1.1 Stream Cipher Algorithms

Accepted stream cipher algorithms:

- SNOW 2.0 (ISO/IEC 18033-4)
  - 128-bit key length for MODERATE.
  - 256-bit key length for ADVANCED.
- SOSEMANUK[1] (from eSTREAM project)
  - 128-bit and 256-bit key lengths for MODERATE.
  - Not accepted for ADVANCED.

Common notes:

- Initialization Vector (IV) must be at least 128 bits.
- Stream ciphers must be used with a different IV for each key.
- A key must be used only once.
- Valid decryption must never be relied on for authenticity.

2.1.2 Block Ciphers Algorithms

Accepted block cipher algorithms:

- Advanced Encryption Standards (AES) as in FIPS-197
  - 128-bit and 192-bit key lengths for MODERATE.
  - 256-bit key length for ADVANCED.
- Camellia (ISO/IEC 18033-3)
  - 128-bit and 192-bit key lengths for MODERATE.
  - 256-bit key length for ADVANCED.
- Serpent[2]
  - 128-bit and 192-bit key lengths for MODERATE.
  - 256-bit key length for ADVANCED.

[1] C. Berbain *et al.* “Sosemanuk, a Fast Software-Oriented Stream Cipher.” In: Robshaw M., Billet O. (eds.) New Stream Cipher Designs. LNCS 4986. *Springer*, 2008.

[2] E. Biham, R. Anderson, and L. Knudsen. SERPENT: A new block cipher proposal. In Fast Software Encryption - FSE’98, volume 1372 of Lecture Notes in Computer Science, pages 222–238. *Springer-Verlag*, 1998.

2.2 Asymmetric Algorithms

Accepted asymmetric algorithms:

- RSA
  - . At least 3072 bits for key length and "e" value > 65537 for MODERATE.
  - . Not accepted for ADVANCED.
  - . Strong primes must be used.
- Diffie-Hellman and Finite Field
  - . At least 3072 bits for key length and "q" value (subgroup size) of 256 for MODERATE.
  - . Not accepted for ADVANCED.
- Elliptic Curve Discrete Logarithm Problem based Algorithms (ECDLP)
  - . NIST P-256, NIST P-384, BrainpoolP256r1, BrainpoolP384r1 and Curve25519 for MODERATE.
  - . NIST P-521, Curve448[3] and BrainpoolP512r1 for ADVANCED.

2.3 Hash Functions

Accepted hash functions:

- Secure Hash Algorithm 2 (SHA-2)
  - . SHA2-384 and SHA2-512/256 for MODERATE.
  - . Not accepted for ADVANCED.
- Secure Hash Algorithm 3 (SHA-3)
  - . SHA3-256, SHA3-384, SHAKE128 and SHAKE256 for MODERATE.
  - . SHA3-512 for ADVANCED.

Common notes:

- Hash functions must be inversion-resistant, pre-image resistant and collision resistant.
- For SHAKE128, the output size "d" value must be ≥256 bits.
- For SHAKE256, the output size "d" value must be ≥512 bits.
- SHA2-384 and SHA2-512/256 are sometimes referred to as SHA-384 and SHA-512/256 respectively.

2.4 Lightweight Crypto Algorithms

Accepted lightweight algorithms (on limited systems with constrained resources, where the use of conventional cryptographic standards is not applicable):

- Block ciphers (ISO/IEC 29192-2)
  - . PRESENT, 80-bit or 128-bit key lengths.

[3] Curve448 is accepted for ADVANCED even though it operates at 224-bit security level because of its increased performance, resistance to a wide-range of side channel attacks and ease of implementation.

  - CLEFIA, 128-bit, 192-bit or 256-bit key lengths.

- Stream ciphers (ISO/IEC 29192-3)
  - Enocoro, 80-bit or 128-bit key lengths.
  - Trivium, 80-bit key length.
- Asymmetric mechanisms (ISO/IEC 29192-4)
  - Unilateral authentication mechanisms.
  - ALIKE key exchange.
  - Identity-based signature.
- Hash functions (ISO/IEC 29192-5)
  - PHOTON, 80-bit, 128-bit, 160-bit, 224-bit or 256-bit output sizes.
  - SPONGENT, 88-bit, 128-bit, 160-bit, 224-bit or 256-bit output sizes.
  - Lesamnta-LW, 256-bit output size.
- Message Authentication Code (MAC) as in ISO/IEC 29192-6
  - Tsudik's keymode, hash-based.
  - Chaskey12, 128-bit key length.

3 Cryptographic Schemes

3.1 Block Cipher Modes of Operation

Accepted block cipher modes of operation:

- Counter Mode (CTR) as in NIST SP800-38A.
- Cipher Block Chaining (CBC) as in NIST SP800-38A.
- XEX Tweakable Block Cipher with Ciphertext Stealing (XTS-AES) as in NIST SP800-38E.
- Output Feedback (OFB) as in NIST SP 800-38A.
- Cipher Feedback (CFB) as in NIST SP 800-38A.

Common notes:

- Cipher Block Chaining (CBC) is only accepted for MODERATE.

3.2 Message Authentication Codes (MAC)

Accepted message authentication codes:

- Cipher-based MAC (CMAC) as in NIST SP 800-38B
  - The scheme should be used for at most 2^{48} messages.
  - Used only in applications where no party learns all-0 string enciphering in the block cipher underlying the MAC.
  - A tag length of at least 96 bit with protected authentication keys.
- Hash-based MAC (HMAC) as in FIPS PUB 198-1
  - A tag length of at least 96 bit with protected authentication keys.
  - To be used with SHA-2 and SHA-3 (as mentioned in Sub-Section 2.3).

3.3 Authenticated Encryption with Associated Data (AEAD)

Accepted authenticated encryption with associated data:

- Galois Counter Mode (GCM) as in NIST SP 800-38D
  - A nonce generation length of at least 96 bits.
  - IV to be unique within the key change period.
  - The use of short authentication tags is not accepted.
- Counter with CBC-MAC (CCM) as in NIST SP 800-38C.

3.4 Key Wrap Functions

Accepted key wrap functions:

- Key Wrap (KW) as in NIST SP 800-38 F.
- Key Wrap with Padding (KWP) as in NIST SP 800-38 F.

3.5 Key Derivation Functions (KDF)

Accepted key derivation functions:

- RFC 5869 (HKDF)[4].
- IKE-v2-KDF[4].
- TLS-v1.2-KDF[4].
- X9.63-KDF[4].

KDFs must comply with one of the following:

- NIST-800-56 A/B KDF (Single Step)[4].
- NIST-800-56 C KDF (Extract-then-expand)[4].
- NIST-800-108[4].

3.6 Key Agreement and Key Transport

3.6.1 Symmetric Schemes

Accepted key agreement schemes:

- Symmetric key agreement schemes must rely solely on shared long-term secrets.

Accepted key transport schemes:

- All symmetric encryption schemes and primitives can be used as per Section 2 and Section 3.
- Combination of an encryption scheme with a MAC in encrypt-then-MAC mode (refer to Section 2 and Section 3).

3.6.2 Asymmetric Schemes: Key Exchange Algorithms

Accepted key exchange algorithms:

- Diffie-Hellman (DH) as in RFC 3526
  - At least 3072 bits for key length for MODERATE.
  - Not accepted for ADVANCED.
- Elliptic Curve Diffie-Hellman (ECDH) as in NIST SP 800-56A
  - 256-bit to 384-bit key for MODERATE.
  - 512-bit key or using Curve448 for ADVANCED.
  - With forward secrecy and authenticated key establishment implementation.
- RSA Key Establishment (NIST 800-56B)
  - At least 3072 bits for key length for MODERATE.
  - Not accepted for ADVANCED.

[4] European Commission, "eCrypt Algorithms, Key Size and Protocols Report," in *eCrypt Algorithms*, Key Size and Protocols Report, 2018.

3.7 Hybrid Encryption Schemes

Accepted hybrid encryption schemes:

- Elliptic Curve Integrated Encryption Scheme (ECIES)
  - For MODERATE and ADVANCED.
- Discrete Logarithm Integrated Encryption Scheme (DLIES)
  - For MODERATE.
  - Not accepted for ADVANCED.
- RSA with Optimal Asymmetric Encryption Padding (RSA-OAEP) as in PKCS #1 v2.1. RSA
  - For MODERATE.
  - Not accepted for ADVANCED.

3.8 Public Key Signatures

Accepted public key signatures:

- Digital Signature Algorithm (DSA) as in FIPS PUB 186-4
  - At least 3072 bits for modulus key for MODERATE.
  - Not accepted for ADVANCED.

- Elliptical Curve Digital Signature Algorithm (ECDSA) as in FIPS PUB 186-4
  - 256-bit to 384-bit key for MODERATE.
  - 512-bit key or using Curve448 for ADVANCED.
  - Accepted message authentication algorithms will be dependent on system and use case.

- RSA signatures (RSA-PSS and RSA Digital Signature Scheme) as in PKCS, RSA Cryptographic Standard v2.2
  - RSA-PSS and RSA Digital Signature Scheme (RSA-DS2) are accepted[5], [6].
  - Modulus lengths of 3072 bits for MODERATE.
  - Not accepted for ADVANCED.

- Merkle signatures
  - Accepted hash functions described in Section 2 must be used.
  - The required pseudo-random function family must be constructed with the HMAC construction based on the hash function used.

[5] PKCS, "RSA Cryptographic Standard. Version 2.2," 2012.

[6] ISO, "ISO/IEC 9796-2-2010. Information technology - Security techniques - Digital Signature Schemes. Part 2: Integer Factorization based mechanisms.," 2010.

4 Commonly Used Cryptographic Protocols

In this section, a list of prevalent cryptography protocols is presented along with the requirements for using them. Any protocols that are not listed here must follow the accepted primitives and schemes in Section 2 and Section 3. In addition, future versions of protocols listed below will be accepted if they follow the accepted primitives and schemes in Section 2 and Section 3.

4.1 IP Security (IPsec)

Accepted algorithms for IPsec:

For authentication, Authentication Header (AH) and Encapsulating Security Payload (ESP) must be used with one of the following MAC:

- HMAC-SHA2-384, HMAC-SHA3-256, HMAC-SHA3-384 for MODERATE.
- HMAC-SHA3-512 for ADVANCED.

For confidentiality, ESP must be used with one of the above MAC algorithms and one of the following encryption algorithms[7] :

- AES-CTR
- CAMELLIA-CTR

As an alternative to use the above authentication and confidentiality processes, an authenticated encryption can be used from the following modes[7]:

- AES-CCM_* (with * as size in bytes of Integrity Check Value (ICV))
  - . 12 or 16 bytes is accepted.
- CAMELLIA-CCM_* (with * as size in bytes of ICV)
  - . 12 or 16 bytes is accepted.
- AES-GCM_* (with * as size in bytes of ICV)
  - . 12 or 16 bytes is accepted.

4.2 Transport Layer Security (TLS)

Accepted TLS versions:

- It is acceptable to use TLS version 1.2 cipher suites with the strongest cryptographic primitives and schemes to ensure compatibility with Section 2 and 3 in addition to implementing a configuration that does not allow downgrading[8].

[7] European Union Agency for Network and Information Security "Study on cryptographic protocols," 2014.

[8] E. Ronen, "The 9 Lives of Bleichenbacher's CAT. New Cache Attacks on TLS Implementations.," 2018.

- While the transition to the new TLS 1.3 standard will take some additional time, it is recommended for future applications.

Accepted cipher suites for TLS 1.2:

- TLS_[*1]_[*2]_WITH_AES_128_CCM, TLS_[*1]_[*2]_WITH_AES_128_CCM_8, TLS_[*3]_[*4]_WITH_Camellia_256_GCM_SHA-384, TLS_[*3]_[*4]_WITH_AES_256_GCM_SHA-384 for MODERATE.
- TLS_[*3]_[*4]_WITH_AES_256_CCM, TLS_[*3]_[*4]_WITH_AES_256_CCM_8 for ADVANCED.

With the following:

[*1] as the underlying key agreement scheme: ECDH, ECDHE, DH or DHE

[*2] as the underlying signature scheme: EC_DSA, RSA or DSS

[*3] as the underlying key agreement scheme: ECDH or ECDHE

[*4] as the underlying signature scheme: EC_DSA

Accepted cipher suites for TLS 1.3:

- TLS_AES_256_GCM_SHA384 for both MODERATE and ADVANCED.

4.3 DNSSEC (Domain Name System Security)

Accepted algorithms and schemes for zone data signing:

- ECDSA_P384_SHA384 for both MODERATE and ADVANCED[9, 10] .

Accepted Message authentication algorithms:

- HMAC-SHA384 for MODERATE.
- HMAC-SHA512 for ADVANCED[9].

4.4 Secure Shell (SSH)

Accepted SSH versions: SSH-2.

Accepted SSH encryption and MAC algorithms for SSH-2:

- AEAD_AES_128_GCM for MODERATE.
- AEAD_AES_256_GCM for ADVANCED.

4.5 Bluetooth

Accepted Bluetooth versions: 4.1 or higher with the following requirements (NIST SP 800-121r2):

- Use Security Mode 4, Level 4 (Authenticated link key using secure connection).
- Use encryption algorithm AES-CCM.
- Use secure connection feature with P-256 elliptic curve for link key generation.
- Use encryption Mode 3 (all traffic is encrypted).
- For low energy feature of Bluetooth, use version 4.2 or higher with Low Energy Security Mode 1 Level 4.

[9] As SHA3-512 is not implemented in this protocol, this is a special case for ADVANCED. Also HMAC-SHA-512 is not vulnerable to Length Extension Attacks.

[10] As ECC with key length of 512 bits is not implemented in this protocol, this is a special case for ADVANCED.

Common notes:

- Organizations must use the strongest Bluetooth security mode that is available for their devices.

4.6 Universal Mobile Telecommunications System (UMTS) / Long Term Evolution (LTE) / 5G

Accepted algorithms:

- For UMTS, 128-UEA1 and 128-UIA1 must be used.
- For LTE, 128-EEA2 and 128-EIA2 must be used.
- For 5G, 128-NEA2 and 128-NIA2 must be used.
- EIA0 and NIA0 can be used in the exceptional case of unauthenticated emergency calls in limited service mode.
- Only the cryptographic primitives and schemes listed in sections 2 and 3 must be used. However, KASUMI and both ECIES profiles are accepted, including the elliptic curves used in the profile, which are accepted as exceptions, solely for use under 3GPP systems.

4.7 WPA (Wi-Fi Protected Access)

Accepted versions:

- WPA3-Enterprise

Common notes:

- WPA3-Enterprise is the only accepted version and must be applied once it is available.

4.8 Kerberos Protocol

Accepted Kerberos encryption types:

- CAMELLIA128-CTS-CMAC, AES256-CTS-HMAC-SHA384 for MODERATE.
- CAMELLIA256-CTS-CMAC for ADVANCED.

5 Public Key Infrastructure (PKI)

5.1 Algorithms for Certificates

Accepted algorithms for Root CA certificates:

- RSA
  - At least 4096 bits for key length.
- Elliptic Curve Cryptography (ECC)
  - NIST P-384, NIST P-521, Curve448, BrainpoolP384r1 or BrainpoolP512r1.

Accepted algorithms for Intermediate and End User certificates:

- RSA
  - At least 3072 bits for key length.
- ECC
  - NIST P-256, NIST P-384, NIST P-521, Curve25519, Curve448, BrainpoolP256r1, BrainpoolP384r1 or BrainpoolP512r1.

Common notes:

- Certificates, Certificate Revocation Lists (CRLs) and Authority Revocation Lists (ARLs) must comply with X.509 PKI Certificate (RFC 5280).
- SHA-2 or SHA-3 families of hash algorithms must be used as mentioned in Section 2.
- Match the strength of the asymmetric key algorithms with the strength of the hash algorithm.

5.2 Validity of the Certificates

Accepted Root CA certificate validity life span:

- At most 20 years[11].

Accepted Intermediate CA, Subordinate CA and Issuing CA certificate validity life span:

- At most 10 years[11].

Accepted end user certificate validity life span:

- At most 5 years for MODERATE[11].
- At most 3 years for ADVANCED[11].

[11] NIST, "X.509 Certificate Policy for the U.S. Federal PKI Common Policy Framework, Federal Public Key Infrastructure Policy Authority," *NIST*, 2015

6 Key Lifecycle Management (KLM)

6.1 Key Protection and Lifetime

Accepted key protection lifetime:

- Using Hardware cryptographic modules:
  - Private keys should not be valid for more than 5 years (this does not limit CA certificates' lifetime) for MODERATE[12].
  - Private keys should not be valid for more than 3 years (this does not limit CA certificates' lifetime) for ADVANCED[12].
- Using Software cryptographic modules:
  - Private keys must not be valid longer than 2 years for MODERATE[12].
  - Not accepted for ADVANCED.

6.2 KLM Processes

Table 1 lists the required KLM processes from generation till destruction.

Table 1 KLM Processes Requirements

KLM processes	Requirements
Key Generation	• Keys (secret and private) must not be vulnerable to prediction or bias. • Weak keys must not be used. • Private and public keys require prime number generation with extra mathematical properties.
Key Registration/ Certification	• Keys must be associated with their owner (user) with a certificate. • Root certificates must be distributed to relying parties and signatories. • A trusted CA must be used.
Key Distribution and Installation	• Keys must be distributed to their users securely and must be under the users control. • Keys must be transported securely by protecting their confidentiality and authenticity. • All copies of keys must be securely installed and stored. • Public keys must be transported and authenticity (but not privacy) must be protected by using certificates. • Private keys must be protected and authorized by the owner/ third party or CA.

[12] E. Barker, "Recommendation for Key Management: Part 1 - General," NIST, NIST Special Publication 800-57 Part 1 Revision 5, May 2020. https://doi.org/10.6028/NIST.SP.800-57pt1r5.

Key Use	• Keys must be protected against unauthorized use during their lifetimes. • Keys must be protected against misuse from the owners, with key storage on secure hardware and/or secure software (authorization checks).
Key Storage	• Organizations must require secure backups of keys (for internal or law enforcement use) when encryption is supported. • Keys used for non-repudiation must be under the sole control of the user.
Key Revocation/ Validation	• In distributed systems, special measures such as updated versions of the Certificate Revocation List (CRL) and the Online Certificate Status Protocol (OCSP) must be implemented to avoid relying on keys that have expired. • Key validation must be done by checking the CRL or OCSP servers.
Key Archive	• The archival process must be secured, and confidentiality must be ensured to preserve the secrecy of information encrypted with archived keys. • Expired keys must be archived to keep old data accessible when encryption is supported. • Archival systems must follow the retention periods as per relevant regulations.
Key Destruction	• When key lifetime expires and there is no need for it to be archived or stored, it must be removed from hardware via a secure deletion process. • Media storage systems storing keys must be sanitized before disposal using a process compliant with NIST SP 800-88r1 or NSA/CSS Storage Device Sanitization Manual.
Key Accounting	• There must be accounting for all asymmetric keys. • Use of asymmetric keys must be monitored. • There must be accounting for key use.

7 Appendices

7.1 Pseudo Random Number Generation (PRNG)

Pseudo Random Number Generation is required in several aspects of cryptography, including the creation of asymmetric key-pairs and symmetric keys[13].

Software library functions, such as `random()` in the programming languages, must not be used, as such functions tend to be based on weak Linear Congruential Generators (LCG). Dedicated cryptographic PRNG implementations are required for cryptographic applications. The PRNGs should be tested with test packages, such as the randomness tests from NIST [14] and the Dieharder test package[15]. International standards such as ISO 28640:2010 "Random variate generation methods" and NIST SP 800-90A "Recommendation for Random Number Generation Using Deterministic Random Bit Generators" provide some recommendations and requirements for the PRNG.

7.2 Post-Quantum Cryptography

Many commonly used cryptosystems are expected to be broken by quantum capabilities once quantum computers exist with enough qubits. That will have a major impact on popular public-key cryptographic systems, including RSA, DSA and ECDSA[16].

Even though quantum computers with enough qubits are not yet known to be available or used commercially, the risk they impose on the confidentiality of encrypted data is significant. Such a risk is acknowledged and it makes it very important to consider Post-Quantum Cryptography.

However, no international standards for Post-Quantum Cryptography are available yet, and it is expected that they will be issued by international standardization bodies over the next three years. Post-Quantum Cryptography will be considered in the upcoming versions of the NCS.

[13] European Commission, “eCrypt Algorithms, Key Size and Protocols Report,” in *eCrypt Algorithms*, Key Size and Protocols Report, 2018.

[14] NIST, “Special Publication 800-22. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications,” *NIST*, 2010.

[15] R. Brown, “Robert G. Brown’s General Tools Page,” 2019. [Online]. Available: http://webhome.phy.duke.edu/~rgb/General/dieharder.php.

[16] NIST, “NISTIR 8105. Report on Post-Quantum Cryptography,” 2016.

7.3 Side-Channel Attacks

Side-channel attacks on a cryptographic system make use of the results of the system's physical measurements, such as energy consumption, electromagnetic emanation and time consumption, in order to gain insight into sensitive data[17]. Attacks can be carried out by remote and passive adversaries which are difficult to be detected and may lead to a significant and unnoticed leakage of data.

In order to prevent these attacks, the first step to minimize information leakage is to ensure that for the side channel signals, the signal-to-noise ratio is as low as possible. Further, it must also be ensured that the information leakages from side-channels are not useful for the adversaries[18]. For example, eliminate any correlation between the binary representation of secret key and side channel signals, i.e. use dummy operations to mask any potential correlation.

In order to minimize the risk of side-channel attacks, it is important to satisfy the following requirements[19]:

- Cryptographic operations must be performed in certified hardware components, e.g. smart cards to protect the secret and/or private keys.
- Analyse thoroughly the effects of such side channels on certified hardware components in a specialized laboratory during the development process.
- Protect ciphertexts using MAC and verify ciphertexts before performing any other cryptographic operations.

Side-channel attacks cover a wide range of threat scenarios. Therefore, an extensive review of relevant threat scenarios should be carried out for the proper secure implementation.

[17] P. C. Kocher, J. Jaffe and B. Jun, "Differential power analysis," *in Differential power analysis*, 2011.

[18] A. Vega, P. Bose and A. Buyuktosunoglu, "Rugged Embedded Systems: Computing in Harsh Environments", *Morgan Kaufmann*, 2017.

[19] BSI, "Cryptographic Mechanisms: Recommendations and Key Lengths", BSI - Technical Guideline, 2020.

7.4 Definitions

Table 2- Terms and Definitions[20]

Terms	Definitions
Asymmetric Algorithm	A cryptographic algorithm that uses one cryptographic key for encryption, and another key for decryption. The two keys are called private and public keys.
Authentication	Verifying the identity of a user, process or device, often as a prerequisite to allowing access to resources in a system.
Authenticity	The property of being genuine and being able to be verified and trusted.
Block Cipher Algorithm	A symmetric key cipher method that segments data into groups or blocks, then encrypts each one separately.
Certificate	A set of data that uniquely identifies an entity's public key and other information that is digitally signed by a Certification Authority (i.e., a trusted party), thereby binding the public key to the owner.
Certificate Revocation List (CRL)	A list of revoked certificates issued by a Certification Authority.
Certification Authority (CA)	A trusted entity that is responsible for issuing and revoking public key certificates.
Collision	Two or more distinct inputs produce the same output.
Confidentiality	A property of preventing information from being available or disclosed to unauthorized individuals, entities or processes.
Cryptographic Primitive	A low-level cryptographic algorithm used as a basic building block for higher-level cryptographic algorithms.
Cryptography	Principles, means and methods of applying data transformation algorithms for security purposes including integrity, confidentiality, authentication, authenticity and non-repudiation.
Cybersecurity	Cybersecurity is the protection of networks, IT systems, operational technologies systems and their components of hardware and software, their services and the data they contain, from any penetration, disruption, unauthorized modification, access, use or exploitation. The concept of cybersecurity also includes information security and digital security, etc [21].
Elliptic Curve Cryptography (ECC)	Public-key cryptographic methods that use operations in an elliptic curve group.

[20] NIST Glossary, unless otherwise stated for a specific term.

[21] Official NCA Charter, approved by Royal Order number 6801, dated October 31, 2017.

Terms	Definitions
Encryption	The process of transforming plaintext into ciphertext using a cryptographic algorithm and key.
Hash Function	A function that maps an input bit string of arbitrary length to a fixed-length output bit string. This output is often irreversible and serves as a condensed representation of the input.
Hash-based MAC (HMAC)	A message authentication code that uses an approved keyed-hash function.
Hybrid encryption	An application of cryptography that combines two or more encryption algorithms, particularly a combination of symmetric and asymmetric encryption.[22]
Initialization Vector	A known public vector used as an input to initialize an encryption algorithm to increase security and support synchronization.
Integrity	A property whereby data has not been altered in an unauthorized manner since it was created, transmitted or stored.
Integrity Check Value	Checksum capable of detecting modification of an information system.
Kerberos Protocols	An authentication system developed to enable two parties to exchange private information across a public network.
Key Agreement	A key-establishment procedure where resultant keying material is a function of information contributed by two or more participants, so that no party can predetermine the value of the keying material independently of any other party's contribution.
Key Archive	A function in the lifecycle of keying material; a repository for the long-term storage of keying material.
Key Derivation Functions	The process by which one or more keys are derived from either a pre-shared key or a shared secret and other information.
Key Destruction	Removing all traces of keying material so that it cannot be recovered by either physical or electronic means.
Key Distribution	See Key Transport.
Key Exchange	The process of exchanging public keys to establish secure communications.

[22] SANS Glossary

Terms	Definitions
Key Generation	The process of generating keys for cryptography.
Key Lifecycle Management (KLM)	The activities involving the handling of cryptographic keys and other related security parameters (e.g., initialization vectors) during the entire lifecycle of the keys, including their generation, storage, establishment, entry and output, use and destruction.
Key Registration / Certification	A function in the lifecycle of keying material; the process of officially recording the keying material by a registration authority.
Key Revocation	A function in the lifecycle of keying material; a process whereby a notice is made available to the affected entities that the keying material should be removed from operational use prior to the end of the established cryptoperiod of that keying material.
Key Transport	A key-establishment procedure whereby one entity distributes the key to another entity.
Key Wrap	A method of encrypting keys (along with associated integrity information) that provides both confidentiality and integrity protection using a symmetric key algorithm.
Lightweight Crypto	A sub-category in the field of cryptography that intends to provide security solutions for resource-constrained devices.
Message Authentication Code (MAC)	A cryptographic checksum on data that uses a symmetric key to detect both accidental and intentional modifications of the data. MACs provide authenticity and integrity protection.
Non-repudiation	A service using a digital signature that is used to support a determination of whether a message was actually signed by a given entity.
Private Key	In an asymmetric algorithm, the private key is used for digital signing and decrypting data, and it must remain secret.
Public Key	In an asymmetric algorithm, the public key is used for verifying digital signature and encrypting data, and it is publicly known.
Public Key Infrastructure (PKI)	A framework that is established to issue, maintain and revoke public key certificates.
RSA	An asymmetric algorithm that is used for key establishment and digital signature generation and verification.

Terms	Definitions
Stream Cipher Algorithm	A symmetric key cipher method where each plaintext bit/word is encrypted one at a time with a corresponding pseudorandom stream (keystream) using a time variant internal state to produce ciphertext bit/word.
Strong Primes	In cryptography, strong primes are primes whose products are hard to factor. More specifically, a prime p is a strong prime if: (a) $p - 1$ has a large prime factor q, and (b) $q - 1$ has a large prime factor, and (c) $p + 1$ has a large prime factor.
Symmetric Algorithm	A cryptographic algorithm that uses a single secret key for both encryption and decryption.

7.5 Acronyms

Table 3- List of acronyms

Acronym	Full Term
AEAD	Authenticated Encryption with Associated Data
AES	Advanced Encryption Standards
AH	Authentication Header
ALIKE	Authenticated Lightweight Key Exchange
ARLs	Authority Revocation Lists
CA	Certificate Authority
CBC	Cipher Block Chaining
CCM	Counter with CBC-MAC
CFB	Cipher Feedback
CMAC	Cipher-based Message Authentication Code
CRLs	Certificate Revocation Lists
CTR	CounTeR
DH	Diffie-Hellman
DLIES	Discrete Logarithm Integrated Encryption Scheme
DNSSEC	Domain Name System Security
DSA	Digital Signature Algorithm
ECC	Elliptic Curve Cryptography
ECDLP	Elliptic Curve Discrete Logarithm Problem
ECDSA	Elliptical Curve Digital Signature Algorithm
ECIES	Elliptic Curve Integrated Encryption Scheme
EEA	EPS Encryption Algorithm
EIA	EPS Integrity Algorithm
EPS	Evolved Packet System
ESP	Encapsulating Security Payload
FIPS	Federal Information Processing Standards
GCM	Galois Counter Mode
HKDF	Hash-based Key Derivation Function
HMAC	Hash-based Message Authentication Code
ICV	Integrity Check Value
IKE-v2	Internet Key Exchange version 2
IPsec	Internet Protocol Security

Acronym	Full Term
ISO/IEC	International Organization for Standardization and the International Electrotechnical Commission
IV	Initialization Vector
KDF	Key Derivation Functions
KLM	Key Lifecycle Management
KW	Key Wrap
KWP	Key Wrap with Padding
LTE	Long-Term Evolution
MAC	Message Authentication Code
NEA	NR Encryption Algorithm
NIA	NR Integrity Algorithm
NIST	National Institution of Standard and Technology
NR	New Radio
OCSP	Online Certificate Status Protocol
OFB	Output Feedback
PKI	Public Key Infrastructure
RSA	Algorithm developed by Rivest, Shamir and Adelman
RSA-OAEP	RSA with Optimal Asymmetric Encryption Padding
SHA-2	Secure Hash Algorithm 2
SHA-3	Secure Hash Algorithm 3
SSH	Secure Shell
TLS	Transport Layer Security
UEA	UMTS Encryption Algorithm
UIA	UMTS Integrity Algorithm
UMTS	Universal Mobile Telecommunications System
WPA	Wi-Fi Protected Access

الهيئة الوطنية للأمن السيبراني
National Cybersecurity Authority