ETSI GS QKD 008 V1.1.1 (2010-12)

Group Specification

Quantum Key Distribution (QKD); QKD Module Security Specification

Disclaimer

This document has been produced and approved by the Quantum Key Distribution (QKD) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG. It does not necessarily represent the views of the entire ETSI membership

Reference
DGS/QKD-0008

Keywords
analysis, protocols, Quantum Key Distribution,
security, system

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group on Quantum Key Distribution systems (QKD - ISG).

Introduction

The present document specifies the security requirements for QKD modules utilized within security systems to protect sensitive information in telecommunication systems. The present document has been developed by the ETSI Quantum Key Distribution Industry Specification Group (QKD-ISG) composed of both operators and vendors. The working group has identified requirements for QKD modules to provide data security.

Following the methodology used in conventional cryptographic security modules and systems, eleven security aspects have been identified, and the present document will establish the minimum requirements that QKD modules will fulfil to be in accordance with the present document. Because of the particular requirements and final quality that the Quantum Key Distribution systems have, the present document has not considered the possibility of having different security levels included in the present document, and it does not consider different degrees of data sensitivity nor different application environments.

In the present document, software requirements are given great prominence because software controls all the actual measurement and communications systems and software security appears as the cornerstone of the general system security. In the present document, requirements that protect the QKD modules against non-invasive attacks are also provided.

While the security requirements specified in the present document are intended to maintain the security provided by a QKD module, conformance to them is necessary but not sufficient to ensure that a particular module is secure. The operator of a QKD module is responsible for ensuring that the security provided by the module is sufficient and acceptable to the owner of the information that is being protected, and that any residual risk is acknowledged and accepted. Similarly, the use of a validated QKD module in a computer or telecommunications system is not sufficient to ensure the security of the overall system.

The importance of security awareness and of making information security a management priority should be communicated to all users, managers and system administrators. Since information security requirements vary for different applications and scenarios, organizations should identify their information resources and determine the sensitivity to and the potential impact of losses.

Controls should be based on the potential risks and should be selected from available controls, including administrative policies and procedures, physical and environmental controls, information and data controls, software development and acquisition controls, as well as backup and contingency planning.

1 Scope

The present document aims to establish the necessary requirements for a QKD module to have a high probability of detecting and responding precisely and timely to attempts of direct physical access, and use or modification of modules inside. The principal objective is to detect any possible penetration with high probability, and resulting in the immediate zeroization of all Critical Security Parameters in plain text.

This objective requires mechanisms to provide a complete envelope of protection around the QKD module, and sensors and circuits to detect and respond in time to all unauthorized attempts of physical access. This can be obtained using strong enough enclosures and redundant tamper detection and response circuitry that zeroizes all plaintext Critical Security Parameters. Enclosure's integrity can be controlled using tamper-evident coatings or seals, and pick-resistant locks on all removable covers or doors of the module.

Strong enclosures must be opaque to all visual and non-visual radiation examination and the tamper detection and zeroization circuitry is protected against disablement. When zeroization is required, Public and Critical Security Parameters are zeroized.

Access and module operation must require identity-based authentication mechanisms that enhance a role-based organization. This authentication must require at least two-factor authentication for operator authentication (secret password, physical key or token, biometric.). The proper operation requires the operator's identity authentication and to verify that he is authorized to assume a specific role and perform a corresponding set of services.

Entry or output of Critical Security Parameters must be done using ports that are physically separated from other ports, or trough interfaces that are logically separated using a trusted-channel from any other interfaces.

All QKD secure modules must be protected against environmental conditions or fluctuations outside of the module's normal operating ranges, because such deviations can be seen as an attack, or they can increase the module failure probability and that can compromise the module security and its operation. The environmental magnitudes to control must be darkness (when required), temperature, voltage, pressure, humidity, atmosphere chemical composition, mechanical vibrations and the presence of nuclear and any other ionizing radiation. Because QKD modules include optical and electro-optical subsystems, it is necessary to control any environmental variable that could affect specifically to that components and the way that they perform, no matter if it is temporally or permanently.

A QKD module is required to either include special environmental protection features designed to detect fluctuations and zeroize Critical Security Parameters, or to undergo rigorous Environmental Failure Testing to provide a reasonable assurance that the module will not be affected by fluctuations outside of the normal operating range in a manner that can compromise its security.

In particular, all QKD modules require the protection of Critical Security Parameters against Timing Analysis attacks, Simple Power Analysis, Differential Power Analysis attacks, Electromagnetic Emanation Attacks and any attack performed through the optical channels.

QKD modules must use strong cryptographic protection to detect and prevent the disclosure and modification of Public Security Parameters as well as Critical Security Parameters when the module is inactive.

To be sure that every time the module is operating in a safe mode, the module must have a clear indication that the module is operating in an Approved Mode.

Because software has the final control in any QKD module, this component must provide robust and tested solutions for the encryption and authentication of all the Critical Security Parameters, all the Sensitive Security Parameters in the system and also to provide secure integrity tests for the software code when the module is not in use.

QKD Module software components can be executed on a general purpose computing system if the operating system provides the auditing of all operator accesses to the audit data, to all requests to use authentication data management mechanisms, all use of security-relevant Crypto Officer Functions, and to all requests to access authentication data associated with the QKD module. In particular, the operating system running the general purpose computing system has to:

- prevent operators in the user role from modifying software, system Sensitive Security Parameters (SSPs), and audit data stored in the operational environment of the module;
- communicate all SSPs, authentication data, control inputs, and status outputs via a trusted channel; and
- audit the operation of any trusted channel.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

The following referenced documents are necessary for the application of the present document.

Not applicable.

2.2 Informative references

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

Not applicable.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

approved data authentication technique: approved method that may include the use of a digital signature, message authentication code or keyed hash

EXAMPLE: RSA, ECDSA and hMAC

approved mode of operation: mode of the QKD module that employs only Approved or Allowed security functions

NOTE: Not to be confused with a specific mode of an Approved security function.

EXAMPLE: AES in CBC mode.

approved security function: security functions are cryptographic algorithms that can be tested, cryptographic key management techniques or authentication techniques

NOTE: In QKD systems the full protocol, the optical or quantum communication and the algorithm derived with the Information Theoretical analysis, should be perfectly described as an Approved Security Functions.

bypass capability: ability of a service to partially or wholly circumvent encryption, cryptographic authentication or any other security function

NOTE: If, as a result of one or more service invocations, the module can output a particular data or status item in encrypted or cryptographically authenticated form, but instead (as a result of module configuration or operator intervention) outputs the item in a non-protected form, then a bypass capability exists.

compromise: unauthorized disclosure, modification, substitution, or use of sensitive data or an unauthorized breach of physical security

conditional test: test performed by a QKD module when the conditions specified for the test occur

confidentiality: property that sensitive information is not made available or disclosed to unauthorized individuals, entities, or processes

Configuration Management System (CMS): management of security features and assurances through control of changes made to hardware, software and documentation of a QKD module

control information: information that is entered into a QKD module for the purposes of directing the operation of the module

Critical Security Parameter (CSP): security-related information (e.g. secret and private cryptographic keys, optical hardware configuration and authentication data such as passwords and PINs) whose disclosure or modification can compromise the security of a QKD module

cryptographic officer: operator or process (subject), acting on behalf of the operator, performing cryptographic initialization or management functions

cryptographic algorithm: well-defined computational procedure that takes variable inputs, which may include cryptographic keys, and produces an output

cryptographic boundary: explicitly defined continuous perimeter that establishes the physical bounds of a QKD module and contains all the hardware and software components of a QKD module

cryptographic hash function: computationally efficient function that maps binary strings of arbitrary length to binary strings of fixed length, such that it is computationally infeasible to invert it, or to find two distinct values that hash into a common value

cryptographic key (**key**)**:** parameter used in conjunction with a cryptographic algorithm that determines such operations as:

- the transformation of plaintext data into ciphertext data;
- the transformation of ciphertext data into plaintext data;
- a digital signature computed from data;
- the verification of a digital signature computed from data;
- an authentication code computed from data; or
- an exchange agreement of a shared secret.

cryptographic key component (**key component**)**:** parameter used in conjunction with other key components in an Approved Security Function to form a plaintext cryptographic key or perform a cryptographic function

cryptographic module (**module**)**:** set of hardware and/or software that implements Approved Security Functions (including cryptographic algorithms and key generation) and is contained within the cryptographic boundary

cryptographic module Security Policy: description of how the specific module meets the security requirements of the standard, including the rules derived from the requirements of the present document and additional rules imposed by the vendor

cryptographically protected Critical Security Parameter (CSP): Critical Security Parameter (CSP) that is cryptographically protected against unauthorized disclosure, modification and substitution, and for which the protection mechanism's strength rationale relies only on Approved Security Functions

cryptographically protected Public Security Parameter (PSP): Public Security Parameter (PSP) that is cryptographically protected against unauthorized modification and substitution and for which the protection mechanism's strength rationale relies only on Approved Security Functions

cryptographically protected Sensitive Security Parameter (SSP): either a cryptographically protected Critical Security Parameter (CSP) or a cryptographically protected Public Security Parameter (PSP)

data path: physical or logical route over which data passes (a physical data path may be shared by multiple logical data paths)

Differential Power Analysis (**DPA**)**:** analysis of the variations of the electrical power consumption of a QKD module, using advanced statistical methods and/or other techniques, for the purpose of extracting information correlated to cryptographic keys used in a cryptographic algorithm or to any sensitive physical and logical internal state of the QKD module

digital signature: result of a cryptographic transformation of data which, when properly implemented, provides the services of:

- origin authentication;
- data integrity; and
- signer non-repudiation.

ElectroMagnetic Emanations (**EMEs**)**:** intelligence-bearing signal, which, if intercepted and analyzed, potentially discloses the information that is transmitted, received, handled, or otherwise processed by any information-processing equipment

electronic key entry: entry of cryptographic keys into a QKD module using electronic methods such as a smart card or a key-loading device

NOTE: The operator entering the key may have no knowledge of the value of the key being entered.

electronic key transport: transport of cryptographic keys, usually in encrypted form, using electronic means such as a computer network

EXAMPLE: Key transport/agreement protocols.

ElectroStatic Discharge (**ESD**)**:** sudden and momentary electric current that flows when an excess of electric charge, stored on an electrically insulated object, finds a path to an object at a different electrical potential (such as ground)

encrypted key: cryptographic key that has been encrypted using an approved security function with a key encrypting key

entity: person, a group, a device, or a process

entropy: uncertainty of a random variable

Environmental Failure Protection (EFP): use of features to protect against a compromise of the security of a QKD module due to environmental conditions or fluctuations outside of the module's normal operating range

Environmental Failure Testing (EFT): use of specific test methods to provide reasonable assurance that the security of a QKD module will not be compromised by environmental conditions or fluctuations outside of the module's normal operating range

Error Detection Code (EDC): code computed from data and comprised of redundant bits of information designed to detect, but not correct, unintentional changes in the data

Finite State Model (**FSM**)**:** mathematical model of a sequential machine that is comprised of a finite set of input events, a finite set of output events, a finite set of states, a function that maps states and input to output, a function that maps states and inputs to states (a state transition function), and a specification that describes the initial state

hard/hardness: relative resistance of a metal or other material to denting, scratching, bending or penetration; physically toughened; rugged, and durable

hardware: physical equipment within the QKD boundary used to process programs and data (includes non-reprogrammable software)

hardware module: module composed primarily of hardware, which may also contain some software

hash value: output of a cryptographic hash function

hybrid module: module whose cryptographic functionality is primarily contained in software, which also includes some special purpose hardware within the cryptographic boundary of the module

Initialization Vector: vector used in defining the starting point of a cryptographic process within a cryptographic algorithm

input data: information that is entered into a QKD module for the purposes of transformation or computation using an Approved security function

integrity: property that sensitive data has not been modified or deleted in an unauthorized manner without detection

interface: logical entry or exit point of a QKD module that provides access to the module for logical information flows representing physical signals

key agreement: key establishment procedure (either manual or electronic) where the resultant key is a function of information securely contributed by two or more participants, so that no party can predetermine the value of the key independently of the other party's contribution

key encrypting key: cryptographic key that is used for the encryption or decryption of other keys

key establishment: process by which cryptographic keys are securely established among QKD modules using key transport and/or key agreement procedures

key loader: self-contained unit that is capable of storing at least one plaintext or encrypted cryptographic key or key component that can be transferred, upon request, into a QKD module

key management: activities involving the handling of cryptographic keys and other related security parameters (e.g. Initialization Vectors (IVs) and passwords) during the entire life cycle of the keys, including their generation, storage, establishment, entry and output, and zeroization

key transport: secure transport of cryptographic keys (Critical Security Parameters) from one QKD entity to another entity

logical protection: protection against unauthorized access (including unauthorized use, modification, substitution, and, in the case of Critical Security Parameters, disclosure) by means of the Module Software Interface under operating system control

NOTE: Logical protection of software Sensitive Security Parameters does not protect against physical tampering.

manual key (**Sensitive Security Parameter**) **entry:** entry of cryptographic keys into a QKD module, using devices such as a keyboard

Message Authentication Code (MAC): cryptographic checksum on data that uses a symmetric key to detect both accidental and intentional modifications of data

EXAMPLE: A Hash Based Message Authentication Code.

microcode: elementary processor instructions that correspond to an executable program instruction

min-entropy: worst-case (that is, greatest lower bound) measure of uncertainty for a random variable

modifiable operational environment: operational environment that is designed to contain some non-validated software

Module Software Interface (**MSI**)**:** set of commands used to request the services of the module, including parameters that enter or leave the module's cryptographic boundary as part of the requested service

multiple-chip embedded module: physical embodiment in which two or more integrated circuit chips are interconnected and are embedded within an enclosure or a product that may not be physically protected

EXAMPLE: Adapters and expansion boards.

multiple-chip standalone module: physical embodiment in which two or more integrated circuit chips are interconnected and the entire enclosure is physically protected

EXAMPLE: Encrypting routers or secure radios.

non-invasive attack: attack that can be performed on a QKD module without direct physical contact with the module

non-modifiable operational environment: operational environment that is designed to contain only validated software

opaque: (i.e. to light within a given wavelength range, to ionizing radiation within a given energy range, etc.) impenetrable by the specified radiation neither transparent nor translucent

operational environment: set of all software and hardware required for the module to operate securely

operator: individual accessing a QKD module or a process operating on behalf of the individual, regardless of the assumed role

output data: information that is produced from a QKD module

passivation: process in the construction of semiconductor devices in which junctions, surfaces of components and integrated circuits are afforded a means of protection against the modification of circuit behaviour

password: string of characters (letters, numbers, and other symbols) used to authenticate an identity or to verify access authorization

Personal Identification Number (PIN): numeric code, used to authenticate an identity

physical protection: safeguarding of a QKD module, cryptographic keys, or Critical Security Parameters using physical means

plaintext key: unencrypted cryptographic key

port: physical entry or exit point of a QKD module that provides access to the module for physical signals represented by logical information flows (physically separated ports do not share the same physical pin or wire)

pre-operational test: test performed by a QKD module between the time a QKD module is powered on and the time that the QKD module uses a function or provides a service using the function being tested

private key: cryptographic key, used with a public key cryptographic algorithm, that is uniquely associated with an entity and is not made public

production grade: industry standard manufacturing

public key: cryptographic key used with a public key cryptographic algorithm that is uniquely associated with an entity and that may be made public

NOTE: Public keys are not considered Critical Security Parameters.

public key certificate: set of data that contains a unique identifier associated with an entity, contains the public key associated with the identifier, and is digitally signed by a trusted party, thereby binding the public key to the identifier

public key (**asymmetric**) c**ryptographic algorithm:** cryptographic algorithm that uses two related keys, a public key and a private key

NOTE: The two keys have the property that deriving the private key from the public key is computationally infeasible.

Public Security Parameter (PSP): security-related public information whose modification can compromise the security of a QKD module

QKD module: set of hardware and software components that implements cryptographic functions and quantum optical processes, including cryptographic algorithms and protocols and key generation, and is contained within a defined cryptographic boundary

radiation hardening: improving the ability of a device or piece of equipment to withstand nuclear or other radiation; applies mainly to dielectric and semiconductor materials

Random Bit Generator (**RBG**)**:** device or algorithm that outputs a sequence of bits that appears to be statistically independent and unbiased

removable cover: part of a QKD module's enclosure that permits physical access to the contents of the module

secret (**symmetric**) **key:** cryptographic key, used with a symmetric secret key cryptographic algorithm that is uniquely associated with one or more entities and should not be made public

security policy: See cryptographic module security policy.

security strength: number associated with the amount of work (that is, the number of operations) that is required to break a cryptographic algorithm or module

NOTE: The average amount of work needed is $2^{\text{sec strength} - 1}$.

seed key: secret value used to initialize a cryptographic function or operation

sensitive data: data that, in user's view, requires protection

Sensitive Security Parameters (SSPs): Critical Security Parameters and Public Security Parameters

service input: all data or control information utilized by the cryptographic module that initiates or obtains specific operations or functions

service output: all data and status information that results from operations or functions initiated or obtained by service input

service: any externally invoked operation and/or function that can be performed by a QKD module

Simple Power Analysis (**SPA**)**:** direct (primarily visual) analysis of patterns of instruction execution (or execution of individual instructions), obtained through monitoring the variations in electrical power consumption of a QKD module, for the purpose of revealing the features and implementations of cryptographic and non-cryptographic algorithms and subsequently the values of cryptographic keys

software: programs within the cryptographic boundary, usually stored on erasable media that can be dynamically written and modified or reprogrammed

EXAMPLE: Ferro-electric and magneto resistive RAM, EEPROM, Flash Memory, magnetic disk.

software module: module that is composed solely of software

split knowledge: process by which a cryptographic key is split into multiple key components, individually providing no knowledge of the original key, which can be subsequently input into, or output from, a QKD module by separate entities and combined to recreate the original cryptographic key

status information: information that is output from a QKD module for the purposes of indicating certain operational characteristics or states of the module

strong: not easily defeated; having strength or power greater than average or expected; able to withstand attack; solidly built

system software: special software within the cryptographic boundary (e.g. operating system, compilers or utility programs) designed for a specific computer system or family of computer systems to facilitate the operation and maintenance of the computer system, associated programs, and data

tamper detection: automatic determination by a QKD module that an attempt has been made to compromise the physical security of the module

tamper evidence: external indication that an attempt has been made to compromise the physical security of a QKD module

NOTE: The evidence of the tamper attempt should be observable by an operator subsequent to the attempt.

tamper response: automatic action taken by a QKD module when a tamper attempt has been detected

Timing Analysis (TA): attack on a QKD module that is based on an analysis of time periods between the time a command is issued and the time the result is obtained

trusted channel: mechanism through which a QKD module provides a trusted, safe and discrete communication pathway for Sensitive Security Parameters and other critical information, between the QKD module and the module's intended communications endpoint

NOTE: A trusted channel exhibits a verification component that the operator or module may use to confirm that the trusted channel exists. A trusted channel protects against eavesdropping, as well as physical or logical tampering by unwanted operators/entities, processes or other devices, both within the module and along the module's communication link with the intended endpoint (e.g. the trusted channel will not allow man-in-the-middle or replay types of attacks). A trusted channel may be realized in one or more of the following ways:

- A communication pathway between the QKD module and endpoint that is entirely local, directly attached to the QKD module and has no intervening systems.
- A mechanism that cryptographically protects Sensitive Security Parameters during entry and output and does not allow misuse of any transitory Sensitive Security Parameters.

two-factor authentication: type of authentication that requires two independent methods to establish identity and authorization to perform services

NOTE: The three most recognized factors are:

- "something you are" (e.g. biometrics);
- "something you know" (e.g. password);
- "something you have" (e.g. smart card).

user: individual or process (subject) acting on behalf of the individual that accesses a QKD module in order to obtain cryptographic services

validated: validated by the validation authority

validation authority: entity that will validate the testing results for conformance to the present document

zeroization: method of erasing electronically stored data to prevent the recovery of the data

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

CMS	Configuration Management System
CSP	Critical Security Parameter
DPA	Differential Power Analysis
DSA	Digital Signature Algorithm
ECDSA	Elliptic Curve Digital Signature Algorithm
EDC	Error Detection Code
EFP	Environmental Failure Protection
EFT	Environmental Failure Testing
EME	ElectroMagnetic Emanation
ESD	Electrostatic Discharge
FIPS	Federal Information Processing Standard
FSM	Finite State Model
HDL	Hardware Description Language

HMAC	Hash-Based Message Authentication Code
IV	Initial Value
IV	Initialization Vector
KAT	Known Answer Test
LDPC	Low-Density Parity-Check
MAC	Message Authentication Code
MRI	Magnetic Resonance Imaging
MSI	Module Software Interface
NIST	National Institute of Standards and Technology
OS	Operative System
PIN	Personal Identification Number
PSP	Public Security Parameter
RBG	Random Bit Generator
SPA	Simple Power Analysis
SSP	Sensitive Security Parameter
TA	Timing Analysis
URL	Uniform Resource Locator
VHDL	VHSIC Hardware Description Language
VHSIC	Very-High-Speed Integrated Circuits

4 Functional security objectives

The security requirements specified in the present document relate to the secure design and implementation of a QKD module. The requirements are derived from the following high-level functional security objectives:

- To employ a correct QKD protocol and a correct implementation of it, that allow the user to be sure that only two communicating entities generate and share the same secret random binary sequence, and no other related information is available to others; this is to say that is the user the one who chooses the security level whatever its definition as long as the user agrees with it.
- To employ and correctly implement the security functions, cryptographic and quantum communication related, required for the protection of sensitive information included or generated inside the QKD module.
- To protect a QKD module from unauthorized operation or use.
- To prevent the unauthorized disclosure of the contents of the QKD module.
- To prevent the unauthorized and undetected modification of the QKD module and cryptographic algorithms, including the unauthorized modification, substitution, insertion, and deletion of Sensitive Security Parameters (SSPs).
- To provide true indications of the operational state of the QKD module.
- To ensure that the QKD module always performs properly when operating in an Approved Mode of Operation of the module.
- To detect errors in the operation of the QKD module and to prevent the compromise or the modification of sensitive data and SSPs resulting from these errors.
- To ensure the proper design, distribution and implementation of the QKD module.

4.1 Security requirements

This clause specifies the security requirements that shall be satisfied by QKD modules conforming to the present document. The full QKD protocols, in particular their quantum optical part and algorithmic, are not directly referenced in the present document. The security of any possible alternative QKD protocol must be probed and tested before it is included in the Approved Security Function suite, and their particular parameters and uses will be particularized in the Approved Mode of Operation document. The present document specifies the requirements and measures that will probe that required conditions are always fulfilled or then they do not and the system has to be zeroized, stopped and put into an error state.

The security requirements in the present document cover areas related to the design, implementation and operation of a QKD module. These areas include QKD module specification; module ports and interfaces; roles, services, and authentication; software security; operational environment; physical security; security against non-invasive attacks; sensitive security parameter management; self-tests; and life-cycle assurance.

A QKD module shall be tested against the requirements of each area addressed in this clause. The QKD module shall be independently validated in each area. The overall validation will indicate the minimum of the independent validations received in the areas.

All documentation, including copies of the user and installation manuals, shall be provided to the testing laboratory by the vendor. Many of the security requirements of the present document include specific documentation requirements that are summarized in annexes A and C.

4.2 QKD module specification

A QKD module shall be a set of hardware and software components that implements cryptographic functions and quantum optical processes, including cryptographic algorithms and protocols and key generation, and is contained within a defined cryptographic boundary.

In an Approved Mode of Operation, a QKD module shall implement at least one Approved or Allowed Security Function (see QKD Module specification requirements listed in annex A).

Allowed Security Functions used in an Approved Mode of Operation shall meet all of the applicable requirements specified in annex B. The operator shall be able to determine when an approved mode of operation is selected. All approved modes of operation shall be specified in the module Security Policy (see annex B.)

The hardware and software of a QKD module can be excluded from the requirements of the present document if the vendor can demonstrate that the excluded hardware and software does not affect the security of the module.

The QKD module Security Policy shall specify when a QKD module is performing in an approved mode of operation. In addition, a QKD module shall indicate when an approved mode of operation is selected.

4.2.1 Types of QKD modules

A QKD module shall be defined as a *Hybrid module*; that is, a module whose functionality is primarily contained in software, which also includes and uses some special purpose optical and cryptographic hardware within the cryptographic boundary of the module.

4.2.2 Cryptographic boundary

A cryptographic boundary shall consist of an explicitly defined perimeter that establishes the physical boundary of a QKD module. The requirements of the present document shall apply to all components within this boundary, including all hardware and software. The cryptographic boundary shall include the processor(s), optics and other hardware components that provide for the operational environment of the module.

4.2.3 Multiple approved modes of operations

A QKD module may be designed to support multiple approved modes of operation. For a QKD module to implement more than one approved mode of operation, the following shall apply:

- The overall security of the module shall not be changed when configured for different approved modes of operation.
- The Security Policy shall describe each approved mode of operation implemented in the QKD module and how each mode is configured.
- Upon re-configuration from one approved mode of operation to another, the QKD module shall perform the pre-operational self-tests (clause 4.10.1).
- Pre-operational self-tests shall be performed for all approved and allowed security functions used in the selected approved mode of operation.

- If re-configuration from one approved mode of operation to another alters the physical security of the module without changing the overall security of the cryptographic module, then the cryptographic module shall perform a zeroization of all CSPs within the module.

4.2.4 Degraded functionality

A QKD module may be designed to support degraded functionality within an approved mode of operation. Example of it would be to switch from one encryption algorithm to use another if the initial one may fail the self-test. Other case could be that quantum communication through a particular band does not reach the minimum quality level, and it is switched to use a different one. Any change in the approved mode of operation degrades its functionality.

For a QKD module to implement a degraded functionality in an approved mode of operation, the following shall apply:

- Degraded operation shall be entered only upon the failure of pre-operational self-tests.
- When the QKD module operates with degraded functionality, each operational security function shall pass all applicable self-tests.
- Non-operational security functions shall be isolated from the remaining security functions of the QKD module.
- The module shall remain in the degraded mode until failed test(s) have all been passed.

4.2.5 Security strength of the module

The security strength of the QKD module shall be specified. The security strength of the QKD module shall be no larger than the minimum security strength of the Approved and Allowed Security Functions and SSPs in the Approved Mode of Operation.

4.3 QKD module physical ports and logical interfaces

QKD module is defined above as contained within a crypto boundary which is defined as continuous physical perimeter containing all hardware and software. A QKD module shall restrict all information flow and physical access points to physical ports and logical interfaces that define all entry and exit points to and from its cryptographic boundary. The QKD module interfaces shall be logically distinct from each other although they may share one physical port (e.g. input data may enter and output data may exit via the same port) or may be distributed over one or more physical ports (e.g. input data may enter via both a serial and a parallel port).

A QKD module may utilize multiple independent communication channels. The data output, for all communications channels, shall be disabled while performing key generation, manual key entry, self-tests, software loading and zeroization.

LOGICAL INTERFACES

A QKD module shall have the following four logical interfaces ("input" and "output" are indicated from the perspective of the module):

- **Data output interface:** All output data (except status data output via the status output interface) from a QKD module (including plaintext, ciphertext, SSPs, and control information for another module) shall exit via the "data output" interface. For a given communication channel, all data output via the "data output" interface shall be prohibited when an error state exists and prior to successfully passing the pre-operational Software Integrity Test (clause 4.10.1).
- **Data input interface:** All input data (except control data entered via the control input interface) processed by a QKD module (including plaintext, ciphertext, SSPs, and status information from another module) shall enter via the "data input" interface.
- **Control input interface:** All input commands, signals, and control data (including function calls and manual controls such as switches, buttons, and keyboards) used to control the operation of a QKD module shall enter via the "control input" interface.

- **Status output interface:** All output signals, indicators, and status data (including return codes and physical indicators such as Light Emitting Diodes and displays) used to indicate the status of a QKD module shall exit via the "status output" interface. Status output may be either implicit or explicit.

The QKD module shall clearly distinguish between data and control information for input, and data and status information for output.

All electrical power externally provided to a QKD module (including power from an external power source or batteries) shall enter via a Power Port. A power port is not required when all power is provided or maintained within the cryptographic boundary of the QKD module (e.g. by an internal battery).

During manual SSP entry, the entered values may be temporarily displayed, in a controlled way, to allow visual verification to improve accuracy.

To prevent the inadvertent output of sensitive information, two independent internal actions shall be required to output CSPs. These two independent internal actions shall be dedicated to mediating the output of the CSPs.

The QKD module shall utilize a separate, dedicated physical port for the input or output of Critical Security Parameters, or a Trusted Channel shall be utilized to protect the Critical Security Parameters entering and leaving the QKD module. If a Trusted Channel is used, the documentation shall specify the security strength of the Trusted Channel.

4.4 Roles, authentication, and services

A QKD module shall support authorized roles for operators and corresponding services within each role.

4.4.1 Roles

A QKD module shall support a Cryptographic Officer Role. The Cryptographic Officer Role shall be assumed to perform all initialization or management functions and general security services (e.g. module initialization, management of cryptographic keys, Critical Security Parameters, and audit functions).

A QKD module may support a User Role. If the QKD module supports a *User Role*, then the *User Role* shall be assumed to perform general security services, including cryptographic operations and other Approved security functions.

A QKD module may support other roles in addition to the roles specified above.

Multiple roles may be assumed by a single operator. If a QKD module supports concurrent operators, then the module shall internally maintain the separation of the roles assumed by each operator and the corresponding services.

Authorized roles are applicable to all callable services utilizing Approved security functions or where the security of the module is affected. An operator is not required to assume an authorized role to perform services where Critical Security Parameters are not used, modified, disclosed, or substituted and PSPs are not used, modified or substituted (e.g. *show status* or other services that do not affect the security of the module).

Documentation shall specify all authorized roles supported by the QKD module.

4.4.2 Operator authentication

Authentication mechanisms may be required within a QKD module to authenticate an operator accessing the module and to verify that the operator is authorized to assume the requested role and perform services within that role. A QKD module shall support the following mechanisms to control access to the module:

- **Role-Based Authentication:** If role-based authentication mechanisms are supported by a QKD module, the module shall require that one or more roles either be implicitly or explicitly selected by the operator and shall authenticate the assumption of the selected role (or set of roles).

- **Identity-Based Authentication:** If identity-based authentication mechanisms are supported by a QKD module, the module shall require that the operator be individually and uniquely identified, shall require that one or more roles either be implicitly or explicitly selected by the operator, and shall authenticate the identity of the operator and the authorization of the operator to assume the selected role or set of roles. The authentication of the identity of the operator, selection of roles, and the authorization of the assumption of the selected roles may be combined. If a QKD module permits an operator to change roles, then the module shall verify the authorization of the identified operator to assume any role that was not previously authorized.

A QKD module may permit an authenticated operator to perform all of the services allowed within an authorized role, or may require separate authentication for each service or for different sets of services. When a QKD module is powered off and subsequently powered on, the results of previous authentications shall not be retained and the module shall require the operator to be re-authenticated.

Various types of authentication data may be required by a QKD module to implement the supported authentication mechanisms, including (but not limited to) the knowledge or possession of a password, PIN, cryptographic key, or equivalent; possession of a physical key, token, or equivalent; or verification of personal characteristics (e.g. biometrics). Authentication data within a QKD module shall be protected against unauthorized disclosure, modification, and substitution.

If a QKD module does not contain the authentication data required to authenticate the operator for the first time the module is accessed, then other authorized methods (e.g. procedural controls or use of factory-set or default authentication data) shall be used to control access to the module and initialize the authentication mechanisms. If default authentication data is used to control access to the module, then default authentication data shall be replaced upon first-time authentication. This default authentication data does not need to meet the zeroization requirements (see clause 4.9.6).

The authentication mechanism may be a group of mechanisms of different authentication properties that jointly meet the strength of authentication requirements of this clause. If the QKD module uses cryptographic functions to authenticate the operator, then those cryptographic functions shall be Approved or Allowed Cryptographic Functions. The combined strength of the authentication mechanism shall conform to the following specifications:

- For each attempt to use the authentication mechanism, the probability shall be equal to or less than one in 2^{27} (134 217 728) that a single attempt will succeed or a false acceptance will occur (e.g. guessing a password, false acceptance error rate of a biometric device, or some combination of authentication methods).

- For multiple attempts to use the authentication mechanism during a one-minute period, the probability shall be equal to or less than one in 2^{32} (4 294 967 296) that a single attempt will succeed or a false acceptance will occur. Time between consecutive attempts will be no less than 2 seconds.

- Authentication strength requirements shall be met by the module's implementation and shall not rely on documented procedural controls or security rules (e.g. password size restrictions).

- If passwords are utilized as an authentication mechanism, then restrictions shall be enforced by the module on password selection to prevent the use of weak passwords that are more susceptible to attacks (e.g. dictionary attacks).

- Feedback of authentication data to an operator shall be obscured during authentication (e.g. no visible display of characters when entering a password). Non-significant characters may be displayed in place of the actual authentication data; even the number of characters may remain obscured.

- Feedback provided to an operator during an attempted authentication shall not weaken the strength of the authentication mechanism beyond the required authentication strength.

If the module employs default authentication data to control access to the module for first-time authentication, then the default authentication data shall be unique per module unit delivered.

A QKD module shall employ *identity-based* authentication mechanisms to control access to the module and, in addition, it shall also meet the following requirement: The QKD module shall enforce two-factor identity-based authentication.

4.4.3 Services

A QKD module shall provide the following services to operators:

- **Show Status:** Output the current status of the QKD module. This may include the output of status indicators in response to a service request.
- **Show the Module's Version Number:** Output the name and the version number of the QKD module.
- **Perform Self-Tests:** Initiate and run pre-operational self-tests as specified in clause 4.10.
- **Perform Approved Security Function:** Perform at least one Approved or Allowed security function used in an Approved mode of operation, as specified in clause 4.1.
- **Zeroize:** Perform zeroization as specified in clause 4.9.6.
- **Bypass Capability:** It is the ability of a service to partially or wholly circumvent a cryptographic function. If the module can output a particular data or status item in a cryptographically protected form, but instead (as a result of module configuration or operator intervention) can also output the item in a non-protected form, then a bypass capability shall be defined.

  If a QKD module implements a bypass capability, then:

  - The operator shall assume an authorized role before configuring the bypass capability.
  - Two independent internal actions shall be required to deactivate the mechanisms that are designed to prevent the inadvertent bypass of security functions due to a single error. The two independent internal actions shall alter software and/or hardware behaviour that is dedicated to mediate the bypass.
  - The module shall show its status to indicate whether:
    - the module is providing services *without* the use of cryptographic functions (the bypass capability *is* activated); or
    - the module is providing services *with* the use of a cryptographic function (the bypass capability *is not* activated).
- **External Software Loading:** If a QKD module has the capability of loading software from an external source then:

  - The logic performing the external software loading shall be logically disconnected from all data output.
  - The QKD module shall not execute the loaded code until after the *Software Load Test* specified in clause 4.10.2 has successfully verified the validity of the externally loaded code.
  - The QKD module shall not execute any loaded Approved security functions until the Cryptographic Algorithm self-tests specified in clause 4.10.1 have been successfully executed.
  - The module shall support an Approved authentication technique to verify the validity of software that may be loaded. Defining a limited or non-modifiable operational environment by means of procedurally-enforced security rules prohibiting the use of the external software loading capability shall not be permitted.

A QKD module may provide other services, both Approved and non-Approved, in addition to the services specified above. Specific services may be provided in more than one role (e.g. key entry services may be provided in the User role and the Crypto-Officer role).

4.5 Software security

The requirements of this clause apply to QKD modules because they containing software. The following requirements shall apply to software contained within a QKD module:

- All cryptographic code within the module shall be in executable form.
- A cryptographic mechanism using an Approved integrity technique (e.g. an approved message authentication code or a digital signature algorithm) that uses a cryptographic key shall be applied to all software within the QKD module. The key may reside within the module.
- The input and output of the module shall be directed through a defined Module Software Interface (MSI).
- The MSI shall not permit the operator of the service to read the software.
- The MSI shall not permit the operator to modify module software without invoking the Software Load Test as specified in clause 4.10.2.
- Any modifications to module software other than a complete reload shall pass the Software Load Test as specified in clause 4.10.2.
- If a specific format for externally provided data is expected, then the module shall verify the format.
- The Approved integrity technique used in the Software Integrity Test shall consist of the generation of a digital signature using an Approved digital signature algorithm. The entity requesting validation shall generate the private key used to sign the code and the public key used to verify the code. The private signing key shall not reside within the module. The public verification key may reside with the module code (clarify, alternatives, etc.)
- An MSI command (i.e. callable service) permitting a cryptographic officer to initiate the Software Integrity Test without instituting a power-down of the module shall be incorporated. The MSI command shall return an indication as to whether the Software Integrity Test was successful and a newly computed hash value.

NOTE: Initially, the hash value on the module software may be transmitted to the cryptographic officer independently of the module. The cryptographic officer may manually compare the newly computed hash value to the one provided by the module vendor. If the hash values do not match or the digital signature does not validate, the cryptographic officer should assume that the module software is not valid.

- The hash value of the module's software shall be zeroized from the module upon completion of the MSI command which initiates the Software Integrity Test.
- The module shall have the capability to decrypt portions of the software that is encrypted when the module is first loaded. All CSPs as well as the Software Integrity Test software (including the public verification key and digital signature) shall be encrypted by the vendor using a symmetric key. The symmetric key, or key components, shall initially be generated by the vendor (clause 4.9.2) and transported to the module site (clauses 4.9.3 and 4.9.4). The symmetric key shall not be retained within the module when the module is transported to the customer. When the software is loaded into the module, the Cryptographic Officer(s) shall enter the symmetric key or key components (clause 4.9.4) to decrypt the encrypted portions. The Software Integrity Test, including the symmetric key (as data), shall then be performed as part of the pre-operational tests.
- Before the module subsequently transitions to the pre-operational state, the Cryptographic Officer(s) may supply a new symmetric key, or key components (otherwise the current symmetric key shall be used). The CSPs, and Software Integrity Test software (including the public verification key and digital signature) shall be encrypted and all plaintext copies of these values within the module shall be automatically zeroized.
- A new key pair used by the Software Integrity Test, and a new symmetric encryption key shall be initially generated (clause 4.9.2) for each instance of the QKD module.
- The mode of encryption used to protect CSPs and the Software Integrity Test software (including the public verification key and digital signature) shall be Approved encryption with an authentication mode.
- In addition to all Critical Security Parameters and the Software Integrity Test software (including the public verification key and digital signature), the symmetric encryption shall be applied to all PSPs.

4.6 Operational environment

The requirements of this clause apply only to modules containing software that run in a modifiable operational environment.

The operational environment of a QKD module is the set of all software and hardware required for the module to operate securely. For example, the operational environment of a software module includes the module itself, the processor on which the software is executed, and the operating system that controls the execution of the software. An operational environment can be non-modifiable or modifiable.

A non-modifiable operational environment is designed to contain only validated software. This environment may be software operating in a non-programmable computer (e.g. a non-programmable PC card or non-programmable smartcard), or software whose update is controlled using Approved data authentication processes (i.e. through the Software Load Test specified in clause 4.10.2). If the operation environment is non-modifiable, then the operational environment components that enforce the non-modifiability shall be bound to the software module.

A modifiable operational environment is designed to allow loading of non-validated software. This environment may include general purpose operating system capabilities (e.g. use of a computer OS or configurable smart card OS). Operating systems are considered to be modifiable operational environments if software can be modified by the operator and/or the operator can load and execute software (e.g. a word processor) that was not included as part of the validation module.

Some examples of non-modifiable and modifiable operational environments are provided in table 1.

Table 1

Configuration	Operational Environment
A QKD module that does not permit the loading of software and does not permit operators to modify the configuration of the operating system or QKD module.	Non-modifiable
A QKD module that allows the loading of additional software that is authenticated and meets all applicable requirements of the present document.	Non-modifiable
Software on a computer that does not isolate input data.	Modifiable
Software on a processor that allows the input of non-validated executable code.	Modifiable
Software on a computer whose operating system is reconfigurable by the operator allowing the removal of the security protections.	Modifiable

The goal of the requirements in clause 4.5 is to logically protect the QKD module running in a modifiable operational environment from unauthorized access (execute, modify, or read) by untrusted processes. Clause 4.5 does not address physical protection to the module.

Documentation shall specify the operational environment for a QKD module, including, if applicable, the operations system employed by the module.

4.6.1 Operating system requirements for modifiable operational environments

The following requirements shall apply to operating systems restricted to a single operator session at any given time:

- All MSI commands in a session shall be run on behalf of a single operator.
- All CSPs shall be zeroized before each operator's session is terminated and a new operator's session is begun.
- Processes that are spawned by the QKD module shall be owned by the module and shall not be owned by external processes/operators.

If the operating system allows multiple concurrent operators, then the following requirements apply:

- All cryptographic software, SSPs, and control and status information shall be under the control of an operating system that implements discretionary access controls that protect against unauthorized execution, modification, and reading.

- To protect plaintext data, cryptographic software, SSPs, and authentication data, the access control mechanisms of the operating system shall be configured to:
  - Enforce the set of roles that can execute stored cryptographic software.
  - Enforce the set of roles that can modify (i.e. write, replace, and delete) the following QKD module software stored within the cryptographic boundary: cryptographic programs, cryptographic data (e.g. cryptographic audit data), SSPs, and plaintext data.
  - Enforce the set of roles that can read the following cryptographic software stored within the cryptographic boundary: cryptographic data (e.g. cryptographic audit data), CSPs, and plaintext data.
  - Enforce the set of roles that can enter SSPs.
- The following specifications shall be consistent with the roles and services as defined in the Security Policy.
  - The operating system shall prevent all operators and executing processes from modifying executing cryptographic processes (i.e. loaded and executing cryptographic program images). In this case, executing processes refer to all non-operating system processes (i.e. operator-initiated), cryptographic or not.
  - The operating system shall prevent operators from gaining either read or write access to SSPs of other operators.
  - The operating system shall prevent operators and external executing processes from reading cryptographic software stored within the cryptographic boundary.
  - The configuration of the operating system to meet the above requirements shall be specified in a Crypto Officer guideline. The Crypto Officer guideline shall state that the operating system must be configured as specified, before the module contents can be considered as protected.
- The operating system shall provide an audit mechanism to record modifications, accesses, deletions, and additions of cryptographic data and SSPs. If audit information is stored outside of the module, then the module shall use Approved cryptographic mechanisms to protect the information when external to the module from unauthorized disclosure and modification.
- The following events shall be recorded by the audit mechanism:
  - attempts to provide invalid input for Cryptographic Officer functions; and
  - addition or deletion of an operator to and from a cryptographic Officer role.
- The audit mechanism shall be capable of auditing the following events:
  - all operator read or write accesses to audit data stored in the audit trail;
  - requests to use authentication data management mechanisms;
  - the use of a security-relevant crypto officer function;
  - requests to access authentication data associated with the cryptographic module;
  - the use of an authentication mechanism (e.g. login) associated with the cryptographic module; and
  - explicit requests to assume a crypto officer role.
- The module Security Policy shall specify whether identification and authentication of module operators is performed by operating system code or vendor supplied code. In either case, the identification and authentication mechanism shall meet the requirements of clause 4.4.2.
- The operating system shall be configured to prevent operators in the user role (if supported) from modifying cryptographic module software, system SSPs, and audit data stored within the operational environment of the module.
- A Trusted Channel shall be implemented between the authenticated operators and the QKD module.

- All SSPs, authentication data, control inputs, and status outputs shall be communicated via a Trusted Channel. Communications via this Trusted Channel shall be activated exclusively by an operator or the QKD module. The Trusted Channel shall provide source authentication and shall prevent unauthorized modification, substitution, disclosure, and playback of sensitive security parameters.
- In addition to the previous audit requirements, the following events shall be recorded by the audit mechanism:
  - Attempts to use the trusted channel function.
  - Identification of the initiator and target of a trusted channel.
- The audit mechanism shall be permanently configured so that the following events are always audited:
  - All operators read or write accesses to audit data stored in the audit trail.
  - Requests to use authentication data management mechanisms.
  - The use of a security-relevant Cryptographic Officer functions.
  - Requests to access authentication data associated with the QKD module.

4.7 Physical security

A QKD module shall employ physical security mechanisms in order to restrict unauthorized physical access to the contents of the module and to deter unauthorized use or modification of the module when installed. All hardware, software, and SSPs within the cryptographic boundary shall be protected.

Depending on the physical security mechanisms of a QKD module, unauthorized attempts at physical access, use, or modification will have a high probability of being detected:

- subsequent to an attempt by leaving visible signs (i.e. tamper evidence); and/or
- during an access attempt so that appropriate immediate actions shall be taken by the QKD module to protect SSPs (i.e. tamper response).

Table 2: Physical security requirements

General Requirements	Multiple-Chip Standalone QKD Modules
Production-grade components.	Production-grade enclosure.
Tamper response and zeroization circuitry.	Opaque enclosure with uniquely numbered tamper-evident seals.
Vents protected from probing.	Pick-resistant locks for doors or removable covers.
EFP for temperature and voltage.	Hard opaque potting material encapsulation of multiple chip circuitry embodiments.
Opaque to non-visual radiation examination (e.g. x-rays, MRI, etc.).	Strong opaque enclosure of multiple chip circuitry embodiments, with removal/penetration attempts causing serious damage.
ESD and Radiation Fault-Induction.	Tamper detection envelope.
	Tamper response and zeroization capability. Tamper detection response circuitry mitigation.

In general, physical protection requires providing tamper-evident mechanisms and the inability to gather information about the internal operations of the critical areas of the module (opaqueness). Also it requires the use of strong enclosures with tamper detection and response mechanisms for removable covers and doors and resistance to probing via ventilation openings. The use of strong enclosures with tamper detection and response mechanisms for the entire enclosure in required, as well as either environmental failure protection from non-visual radiation examination, protection from electro-static discharge and radiation fault induced attacks, as well as protection of the tamper detection response circuitry from disablement.

Security requirements are specified for a maintenance access interface when a QKD module is designed to permit physical access (e.g. by the module vendor or other authorized individuals).

Tamper detection and tamper response are not substitutes for tamper evidence.

4.7.1 General physical security requirements

The following requirements shall apply to all physical embodiments:

- Documentation shall specify the physical embodiment and the security level for which the physical security mechanisms of a QKD module are implemented.
- Whenever zeroization is performed for physical security purposes, the zeroization shall occur in a sufficiently small time period so as to prevent the recovery of the sensitive data between the time of detection and the actual zeroization.
- If a module includes a maintenance role that requires physical access to the contents of the module or if the module is designed to permit physical access (e.g. by the module vendor or other authorized individual), then:
  - A maintenance access interface shall be defined.
  - The maintenance access interface shall include all physical access paths to the contents of the QKD module, including any removable covers or doors.
  - Any removable covers or doors included within the maintenance access interface shall be safeguarded using the appropriate physical security mechanisms.
  - All CSPs and PSPs shall be zeroized when the maintenance access interface is accessed.
- The QKD module shall consist of production-grade components that shall include standard passivation techniques (e.g. a conformal coating or a sealing coat applied over the module's circuitry to protect against environmental or other physical damage).
- When performing physical maintenance, all CSPs contained in the cryptographic module shall be zeroized. Zeroization shall either be performed procedurally by the operator or automatically by the QKD module.
- The cryptographic module shall provide evidence of tampering (e.g. on the cover, enclosure, or seal) when physical access to the module is attempted.
- The tamper-evident material, coating or tamper-evident enclosure shall be opaque within the visible spectrum (i.e. light of wavelength range of 400 nm to 750 nm) to prevent the gathering of information about the internal operations of the critical areas of the module.
- If the cryptographic module contains ventilation holes or slits, then the holes or slits shall be constructed in a manner to prevent the gathering of information by direct visual observation using artificial light sources of the module's internal construction or components.
- If the QKD module contains ventilation holes or slits, then the holes or slits shall be constructed in manner to prevent the gathering of information of the module's internal construction or components by direct visual observation using artificial light sources in the visual spectrum, then the module shall contain tamper response and zeroization circuitry. The tamper response and zeroization circuitry shall immediately zeroize all CSPs when a door is opened, a cover is removed, or when the maintenance access interface is accessed. The tamper response and zeroization circuitry shall remain operational when CSPs are contained within the cryptographic module.
- If the cryptographic module contains ventilation holes or slits, then the holes or slits shall be constructed in a manner that prevents undetected physical probing inside the enclosure (e.g. require at least one 90 degree bend or obstruction with a substantial blocking material).
- The QKD module shall be protected either by a hard opaque removal-resistant coating or by a tamper detection envelope with tamper response and zeroization capability.
- The module shall either include EFP (Environmental Failure Protection) features or undergo EFT (Environmental Failure Testing).
- The QKD module shall include EFP features for both temperature and voltage.

- The QKD module shall be opaque to non-visual radiation examination (e.g. x-rays, MRI, THz, thermal imaging, etc.).
- The QKD module shall include fault-tolerant features to provide protection from electrostatic discharge and electromagnetic radiation induced faults.

4.7.2 Multiple-chip embedded QKD modules

In addition to the general security requirements specified in clause 4.7.1, the following requirements are specific to multiple-chip embedded QKD modules.

If the QKD module is contained within an enclosure or within an enclosure that has a door or a removable cover, then a production-grade enclosure or enclosure with a door or a removable cover shall be used.

The module shall satisfy one of the following requirements:

- the module's components shall be covered with a tamper-evident coating or potting material (e.g. etch-resistant coating or bleeding paint) to deter direct observation or manipulation of module components and to provide evidence of attempts to tamper with or remove module components; or
- the module's components shall be contained in a tamper-evident enclosure to deter direct observation or manipulation of module components and to provide evidence of attempts to tamper it or remove module components; or
- the module shall be entirely contained within a metal, hard plastic or equivalent production-grade material enclosure that may include doors or removable covers.

If the enclosure includes any doors or removable covers, then the doors or covers shall be locked with pick-resistant mechanical locks employing physical or logical keys or shall be protected with uniquely numbered tamper-evident seals (e.g. uniquely numbered evidence tape or uniquely numbered holographic seals).

The module shall be contained within a strong enclosure such that attempts at removal or penetration of the enclosure will have a high probability of causing serious damage to the module (i.e. the module will not function).

The QKD module components shall be covered by potting material or contained within an enclosure encapsulated by a tamper detection envelope (e.g. a flexible Mylar printed circuit with a serpentine geometric pattern of conductors or a wire-wound package or a non-flexible, brittle circuit or a strong enclosure) that shall detect tampering by means such as cutting, drilling, milling, grinding, or dissolving of the potting material or enclosure to an extent sufficient for accessing or modifying the internal components and the SSPs of the module.

The QKD module shall contain tamper response and zeroization circuitry that shall continuously monitor the tamper detection envelope and, upon the detection of tampering, shall immediately zeroizes all CSPs. The tamper response and zeroization circuitry shall remain operational when CSPs are contained within the cryptographic module.

CSPs shall be protected from disclosure if the tamper detection response circuitry or components are disabled.

Possible attacks against the QKD module include but are not limited to the catastrophic and sudden disabling of the tamper detection response circuitry or components. If the disabling method renders the response circuitry disabled such that CSPs are no longer protected from disclosure, this requirement is not met. If the disabling method renders the response circuitry disabled and either concurrently zeroizes the CSPs and PSPs or renders the CSPs and PSPs destroyed then this requirement is met.

4.7.3 Multiple-chip standalone QKD modules

In addition to the general security requirements specified in clause 4.7.1, the following requirements are specific to multiple-chip standalone QKD modules.

- The QKD module shall be entirely contained within a metal, hard plastic or equivalent production-grade material enclosure that may include doors or removable covers.

- If the enclosure of the QKD module includes any doors or removable covers, then the doors or covers shall be locked with pick-resistant mechanical locks employing physical or logical keys or shall be protected with uniquely numbered tamper-evident seals (e.g. uniquely numbered evidence tape or uniquely numbered holographic seals).

- The module shall be contained within a strong enclosure such that attempts at removal or penetration of the enclosure will have a high probability of causing serious damage to the module (i.e. the module will not function).

- The potting material or enclosure of the QKD module shall be encapsulated within a tamper detection envelope that uses tamper detection mechanisms such as cover switches (e.g. micro-switches, magnetic Hall effect switches, permanent magnetic actuators, etc.), motion detectors (e.g. ultrasonic, infrared, or microwave), or other tamper detection mechanisms as described above for multiple-chip embedded QKD modules. The tamper detection mechanisms shall detect tampering by means such as cutting, drilling, milling, grinding, or dissolving of the potting material or enclosure, to an extent sufficient for accessing the contents of the module.

- The QKD module shall contain tamper response and zeroization circuitry that shall continuously monitor the tamper detection envelope and, upon the detection of tampering, shall immediately zeroize CSPs. The tamper response and zeroization circuitry shall remain operational when CSPs are contained within the cryptographic module.

- The QKD module tamper detection response circuitry or components shall be protected from disablement, or CSPs shall be protected from disclosure if the tamper detection response circuitry or components are disabled.

Possible attacks against the QKD module include but are not limited to the catastrophic and sudden disabling of the tamper detection response circuitry or components. If the disabling method renders the response circuitry disabled such that CSPs are no longer protected from disclosure, this requirement is not met. If the disabling method renders the response circuitry disabled and either concurrently zeroizes the CSPs or renders the CSPs destroyed this requirement is met.

4.7.4 Environmental failure protection/testing

The electronic devices, circuitry and hardware in general, are designed to operate within a particular range of environmental conditions. Deliberate or accidental excursions outside the specified normal operating ranges of voltage and temperature can cause erratic operation or failure of the electronic devices or circuitry that can compromise the security of the QKD module. Reasonable assurance that the security of a QKD module cannot be compromised by extreme environmental conditions can be provided by having the module employ EFP features or undergo EFT.

A QKD module shall employ EFP features for both temperature and voltage.

4.7.4.1 Environmental failure protection features

EFP features shall protect a QKD module against unusual environmental conditions or fluctuations (accidental or induced) outside of the module's normal operating range that can compromise the security of the module.

The QKD module shall monitor and correctly respond to fluctuations in the operating *temperature* and *voltage* outside of the specified normal operating ranges.

The EFP features shall involve electronic circuitry or devices that continuously measure the operating temperature and voltage of a QKD module. If the temperature or voltage falls outside of the cryptographic module's normal operating range, the protection circuitry shall either:

- shut down the module to prevent further operation; or

- immediately zeroize all CSPs and PSPs.

Documentation shall specify the normal operating ranges of a QKD module and the EFP features employed by the module.

4.7.4.2 Environmental failure testing procedures

Environmental Failure Testing (EFT) procedures shall involve a combination of analysis, simulation, and testing of a QKD module to provide reasonable assurance that environmental conditions or fluctuations (accidental or induced) outside the module's normal operating ranges for temperature and voltage will not compromise the security of the module.

EFT shall demonstrate that, if the operating conditions falls outside the normal operating range of the QKD module resulting in a failure, at no time shall the security of the cryptographic module be compromised. All QKD module operation under environmental conditions or fluctuations outside of the module's normal operating ranges must be seen as an attack and they increase the module failure probability. Both cases can compromise the module security and its operation. The environmental magnitudes to control must be darkness (when required), temperature, voltage, pressure, humidity, atmospheric chemical composition, mechanical vibrations and the presence of nuclear and any other ionizing radiation. Because QKD modules include optical and electro-optical subsystems, it is necessary to control any environmental variable that could affect specifically to that components and the way that they perform, no matter if it is temporally or permanently.

The temperature tested shall be gradually decreasing from a temperature within the normal operating temperature range to a lower temperature that either (1) shuts down the module to prevent further operation or (2) immediately zeroizes all CSPs and PSPs; and shall be gradually increasing from a temperature within the normal operating temperature range to a higher temperature that either (1) shuts down the module to prevent further operation or (2) immediately zeroize all CSPs and PSPs.

The temperature range tested shall be from -100 °C to +200 °C; however, the test shall be interrupted as soon as either:

1) the module is shutdown to prevent further operation;

2) all CSPs and PSPs are immediately zeroized; or

3) the module enters a failure mode.

The voltage range tested shall be gradually decreasing from a voltage within the normal operating voltage range to a lower voltage that either (1) shuts down the module to prevent further operation or (2) immediately zeroizes all CSPs (also, PSPs if Security Level 5); and shall be gradually increasing from a voltage within the normal operating voltage range to a higher voltage that either (1) shuts down the module to prevent further operation or (2) immediately zeroizes all CSPs (also, PSPs if Security Level 5), including reversing the polarity of the voltages.

Documentation shall specify the normal environmental operating ranges of the QKD module in all aspects and the environmental failure tests performed. For the QKD systems, this include to control and describe environmental and operational magnitudes as pressure, humidity, atmospheric chemical composition, mechanical vibrations and the presence of nuclear and any other ionizing radiation, etc.

4.8 Physical Security - Non-Invasive Attacks

Attacks on the operations of the module that are physical (not logical) in nature and do not require physical contact or direct observation of the module are specified in this section. These attacks include Simple Power Analysis, Differential Power Analysis, Electromagnetic Emanation and Timing Analysis. Other non-invasive attacks may exist but defence against them is currently considered optional:

- The QKD module shall protect the module's CSPs against TA (time analysis) attacks. Documentation shall specify the mitigation techniques against applicable TA attacks.

- The QKD module shall protect the CSPs against SPA attacks. Documentation shall specify the mitigation techniques against applicable SPA attacks.

- The cryptographic module shall protect the module's CSPs against DPA attacks. Documentation shall specify the mitigation techniques against applicable DPA attacks.

- The QKD module shall protect the module's CSPs against EME. Documentation shall specify the mitigation techniques against applicable EME attacks.

4.9 Sensitive Security Parameter (SSP) management

Sensitive Security Parameters (SSPs) consist of Critical Security Parameters and Public Security Parameters.

The security requirements for SSP management encompass the entire lifecycle of SSPs employed by the module. SSP management includes random bit generators, SSP generation, SSP establishment, SSP entry/output, SSP storage, and SSP zeroization. A module may contain one or more embedded modules each performing SSP management functions.

Encrypted CSPs refer to CSPs that are encrypted using an approved security function. CSPs encrypted using a non-Approved security function are considered plaintext within the scope of the present document.

CSPs shall be protected within the module from unauthorized disclosure, modification, and substitution.

PSPs shall be protected within the module against unauthorized modification and substitution.

Keys used only to test the cryptographic algorithms as specified in clause 4.10.2 are PSPs.

For a software module, the Software Integrity Test key is a CSP. For a hardware module that contains software components, the Software Integrity Test key is a PSP. For the hybrid module, the key used for the Software Integrity Test is a CSP. If another key is used to test the integrity of software in the hardware portion of the hybrid module, this key is not a SSP.

Documentation shall specify all SSPs employed by a module.

4.9.1 Random bit generators

A QKD module must contain RBGs, a chain of RBGs, and/or one or more RBG entropy sources. The QKD module may be considered as a single RBG or an RBG entropy source for other systems.

Documentation shall list each RBG and RBG entropy source contained in the module. All RBGs used in an Approved mode shall be Approved and listed in annex A.

If a module contains an RBG or an RBG entropy source in an Approved mode then:

- RBG entropy sources shall be subject to the RBG Entropy Source Test as specified in clause 4.10.2.
- Deterministic components of an RBG shall be subject to the Cryptographic Algorithm Test in clause 4.10.1.
- Data output from the RBG shall pass the Continuous RBG Test as specified in clause 4.10.2.

If entropy is provided from outside of the module then the claimed minimum entropy value shall be provided to the module. The module shall verify that the claimed minimum entropy provided by the RBG entropy source is sufficient to support the intended security strength of RBG that uses the entropy.

If random values are required in an Initial Value (IV), used by an Approved security function(s), then an approved RBG shall be used to generate this Initial Value.

4.9.2 SSP Generation

A module may generate SSPs internally or they may be loaded from an external source. Documentation shall specify each SSP generation method employed by a module. Documentation shall specify each SSP generation method that makes use of an RBG.

Any SSPs (other than seeds and seed keys) generated in the approved mode of the module using an RBG shall be generated using an approved RBG meeting the requirements specified in clause 4.10.1.

When using an approved RBG to generate SSPs, the security strength of the RBG shall be sufficient to support the security strength of the cryptographic security function that makes use of the SSPs.

SSPs (other than seed keys) generated by the module for use by an Approved security function shall be generated using an Approved SSP generation method.

4.9.3 SSP Establishment

QKD module documentation shall specify all SSP establishment methods employed by a module. SSP establishment may be performed by electronic SSP establishment methods (i.e. using SSP transport or SSP agreement schemes). All electronic SSP establishment methods employed in an Approved mode of operation shall be Approved or Allowed for use in an Approved mode.

If an SSP establishment method in an Approved mode requires random values as an input, an Approved RBG shall be used to provide these values.

If an SSP transport method is used by a module, the SSPs transported in the process shall meet the requirements of clause 4.9.4.

4.9.4 SSP Entry and Output

SSPs may be entered into or output from a QKD module. If SSPs are entered into or output from a module, the entry or output of SSPs is performed using manual (e.g. entered via a keyboard or output via a visual display) or electronic (e.g. via a smart card/tokens, PC card, other electronic key loading device, or the module operating system) methods or some combination thereof. Documentation shall specify the SSP entry and output methods employed by a module.

A module shall associate an SSP entered into or output from the module with the correct entity (i.e. person, group, role, or process) to which the SSP is assigned.

All encrypted SSPs, entered into or output from a module and used in an Approved mode of operation, shall be encrypted using an Approved security function.

During manual SSP entry, the entered values may be temporarily displayed to allow visual verification and to improve accuracy. If encrypted CSPs are manually entered into the module, then the plaintext values of the CSPs shall not be displayed. Manually entered (plaintext or encrypted) cryptographic keys (including seed keys) shall be verified during entry into a module for accuracy using the Manual Key Entry Test specified in clause 4.10.2.

For software modules, CSPs may be entered into or output from the module in either encrypted or plaintext form under control of the module operating system provided that the CSPs are maintained within the operational environment. PSPs may be entered into or output from a module in plaintext form.

Electronically transported CSPs shall enter into and output from a module in encrypted form and their integrity shall be protected (e.g. by an Approved security function or an Approved or Allowed key establishment method). Electronically transported PSPs shall enter into and output from the module with their integrity protected by either an Approved digital signature algorithm or an Approved MAC or an Approved key transport method.

Non-electronically transported PSPs may be entered into or output from a module in plaintext form and need not be cryptographically authenticated regardless of whether they are entered manually or electronically.

Non-electronically transported CSPs shall be entered into or output from a module either (1) in encrypted form or (2) using split knowledge procedures (i.e. as two or more plaintext components.)

If split knowledge procedures are used:

- The module shall separately authenticate the operator entering or outputting each component as a separate identity.
- The module shall verify that no two operators entering or outputting key components have the same identities.
- In order to prevent misuse of any SSP, a QKD module shall utilize a Trusted Channel for the input or output of all SSPs, whether or not cryptographically protected. If a Trusted Channel is established and maintained using the cryptographic algorithms, the algorithms shall by Approved and meet or exceed the documented security strength of the module.
- At least two components shall be required to reconstruct the original CSP.

- Documentation shall demonstrate that if knowledge of *n* components is required to reconstruct the original CSP, then knowledge of any *n-1* components provides no information about the original CSP other than the length.
- Documentation shall specify the split knowledge procedures employed by a module.

4.9.5 SSP Storage

SSPs stored within a module may be stored either in plaintext form or encrypted form. A module shall associate every SSP stored within the module with the correct entity (e.g. operator, role, or process) to which the SSP is assigned. An SSP may also be stored within an embedded cryptographic module that meets or exceeds the requirements of the standard relative to the larger module.

Documentation shall specify:

- The SSPs stored in the module.
- How CSPs are protected from unauthorized access when stored in the module.
- How PSPs are protected from unauthorized modification and when stored within the module.
- How the module associates a PSP stored in the module with the entity (operator, role, or process) to which the parameter is assigned.

Plaintext CSPs shall not be accessible to unauthorized operators from outside the module. PSPs shall not be modifiable by unauthorized operators from outside the module.

4.9.6 SSP Zeroization

A QKD module shall provide methods to actively zeroize all CSPs (including temporarily stored values) within the module. Once a CSP is zeroized, the CSP shall not be retrievable from the module.

Zeroization of PSPs, encrypted CSPs, or CSPs otherwise physically or logically protected within an additional embedded validated module (meeting the requirements of the present document) are required.

Keys used only to perform pre-operational self-tests shall be considered as PSPs. Hash values of passwords that, if known, would be subject to an off-line exhaustion attack shall be considered as CSPs. RBG state information shall be considered a CSP.

Documentation shall specify the CSP zeroization method(s) employed by a module and the rationale as to why the method(s) prevent the retrieval and reuse the zeroized CSPs.

Temporary CSPs (e.g. ephemeral keys) shall be zeroized as soon as they are no longer in use.

The zeroization of CSPs may be performed procedurally, and independent of the module's control. For example, the operator executes the destruction of the module (e.g. reformatting of a hard drive, the atmospheric destruction of a module during re-entry).

The QKD module shall control the zeroization of the CSPs.

The following security requirements shall be met:

- A module shall provide methods to zeroize all PSPs (including temporarily stored values) within the module.
- Documentation shall specify the PSP zeroization methods employed by a module and the rationale as to why the methods prevent the retrieval and reuse of the zeroized data.
- Temporary PSPs shall be zeroized when they are no longer needed.

4.10 Self-Tests

A QKD module shall perform pre-operational self-tests, conditional self-tests and, if applicable, critical functions tests to ensure that the module is functioning properly. The pre-operational self-tests must be performed and passed successfully prior to the module providing any services. Conditional self-tests shall be performed when an applicable security function is invoked (i.e. security functions for which self-tests are required). A QKD module may perform other tests in addition to the tests specified in the present document.

If a QKD module fails a self-test, the module shall enter an error state and shall output an error indicator via the status output interface. The QKD module shall not perform any cryptographic operations or output data via the data output interface while in an error state. The QKD module shall not utilize any functionality that relies upon a function or algorithm that failed a self-test until the relevant self-test has been repeated and successfully passed.

Documentation shall specify:

- The self-tests performed by a QKD module.
- The error states that a QKD module can enter when a self-test fail.
- The conditions and actions necessary to exit the error states and resume normal operation of a QKD module (e.g. this may include maintenance of the module, re-powering the module, automatic module recovery, or returning the module to the vendor for servicing).

4.10.1 Pre-Operational Self-Tests

The pre-operational tests shall be performed by a QKD module between the time a QKD module is powered on, either from a power-off state or a quiescent state (e.g. low power, suspend or hibernate) and the time that the QKD module uses a function or provides a service using the function to be tested. Prior to using a security function, the pre-operational test(s) of that security function shall pass successfully. The pre-operational self-tests shall be initiated automatically and shall not require operator intervention. The vendor shall specify a critical time period that specifies the maximum operational time before pre-operational tests must be repeated. When a pre-operational test is completed, the results (i.e. indications of success or failure) may be output via the "status output" interface. If a module does not output an error status upon failure of a module self-test, the operator of the module shall be able to determine if the module has entered an error state through a procedure documented in the Security Policy.

A QKD module shall permit operators to initiate the pre-operational tests on demand for periodic testing of the module.

A QKD module shall repeat the pre-operational self-tests as documented. Documentation shall specify the time period and the policy regarding the interruption of the module's operations.

A QKD module shall perform the following pre-operational tests, as applicable: Software Integrity Test, Cryptographic Algorithm Test, and Pre-Operational Bypass Test:

- **Software Integrity Test:** A test using an approved data authentication technique shall be applied to all validated software within a QKD module when the module is powered up. This pre-operational self-test shall be successfully completed before the QKD module provides any services.

  The Software Integrity Test is not required for any software excluded from the security requirements of the present document or for any executable code stored in non-reconfigurable memory. If the integrity of the executable code cannot be verified, the Software Integrity Test shall fail.

  The Approved data authentication technique shall include the use of a digital signature.

- **Cryptographic Algorithm Test:** This test shall be conducted for all approved and allowed cryptographic algorithms (e.g. encryption, decryption, data authentication and random bit generation) of each cryptographic algorithm implemented by a QKD module via any of the following methods:

  - A Known-Answer Test involves operating the cryptographic algorithm on data for which the correct output is already known and comparing the calculated output with the previously generated output (the known answer). If the calculated output does not equal the known answer, the known-answer test shall fail. Cryptographic algorithms whose outputs do not vary for a given set of inputs (i.e. no random data is obtained and used during the execution of the algorithm) shall be tested using a Known Answer Test (KAT).

- Public key cryptographic algorithms whose outputs vary for a given set of inputs (e.g. the DSA or the ECDSA) shall be tested using a known-answer test if the random number responsible for the variability of the output can be fixed, or shall be tested using a Pair-Wise Consistency Test (see clause 4.10.2) with a fixed pair of public and private keys.

- If a QKD module includes two independent implementations of the same cryptographic algorithm, then the module shall:

  - continuously compare the outputs of the two implementations, and, if the outputs of the two implementations are not equal, the Cryptographic Algorithm Test shall fail; or

  - perform a KAT for each cryptographic algorithm and mode to be tested in accordance with the specified condition. A KAT is not required for the security function in the approved data authentication technique used by the Software Integrity Test.

- **Pre-Operational Bypass Test:** If a QKD module implements a bypass capability, then the module shall ensure the correct operation of the logic governing activation of the bypass capability by exercising that logic. This test shall be performed before the bypass capability is first exercised.

4.10.2 Conditional Self-Tests

Conditional tests shall be performed by a QKD module when the conditions specified for the following tests occur: Pair-Wise Consistency Test, Software Load Test, Manual Key Entry Test, Continuous RBG Test, RBG Entropy Source Test, and Conditional Bypass Test:

- **Pair-Wise Consistency Test** (for public and private keys)**:** If a QKD module generates public or private keys (e.g. as needed in firmware updating, QKD module management, etc.), then the following pair-wise consistency tests for every pair of generated public and private keys shall be performed:

  - If the keys are used to perform key transport, then the public key shall encrypt a plaintext value. The resulting ciphertext value shall be compared to the original plaintext value. If the two values are equal, then the test shall fail. If the two values differ, then the private key shall be used to decrypt the ciphertext and the resulting value shall be compared to the original plaintext value. If the two values are not equal, the test shall fail.

  - If the keys are used to perform the calculation and verification of digital signatures then the consistency of the keys shall be tested by the complete calculation and verification of a digital signature. If the digital signature cannot be verified, the test shall fail.

  - If the keys are used to perform key agreement, then the arithmetic validity of the keys shall be tested by verifying the correct mathematical relationship between the public key and private key values.

- **Software Load Test:** If software can be externally loaded into a QKD module, then the following Software Load Tests shall be performed:

  - An approved digital signature technique shall be applied to all validated software when externally loaded into a QKD module. The Software Load Test is not required for any software that is loaded onto and solely executed on hardware which has been excluded from the security requirements of the present document (see clause 4.1).

  - The applied approved data authentication technique shall be successfully verified or the Software Load Test shall fail.

  - Before the newly loaded software is operationally used, the requirements of clause 4.10.1 shall be satisfied.

- **Manual Key Entry Test:** If cryptographic keys or key components are manually entered into a QKD module or if error on the part of the human operator could result in the incorrect entry of the intended key, then the following manual key entry tests shall be performed:

  - The cryptographic key or key components shall have an error detection code (EDC) applied, or shall be entered using duplicate entries.

- If an EDC is used, the EDC shall be at least 32 bits in length.
- If the EDC cannot be verified, or the duplicate entries do not match, the test shall fail.

- **Continuous RBG Test:** If a QKD module employs an approved RBG or an RBG entropy source in an approved mode of operation, the module shall perform the following continuous random bit generator test on each RBG and RBG entropy source that tests for failure to a constant value:
  - If each call to a RBG produces blocks of *n* bits (where n > 63), the first *n*-bit block generated after power-up, initialization, or reset shall not be used, but shall be saved for comparison with the next *n*-bit block to be generated. Each subsequent generation of an *n*-bit block shall be compared with the previously generated block. The test shall fail if any two compared *n*-bit blocks are equal.
  - If each call to a RBG produces fewer than 64 bits, the first *n* bits generated after power-up, initialization, or reset (for some $n > 63$) shall not be used, but shall be saved for comparison with the next *n* generated bits. Each subsequent generation of *n* bits shall be compared with the previously generated *n* bits. The test fails if any two compared *n*-bit sequences are equal.

- **RBG Entropy Source Test:** If an RNG entropy source is contained within the operational environment, then the min-entropy assessment shall be performed on each output of the entropy source. This test shall fail if the assessed min-entropy is less than the min-entropy required by the Approved RBGs.

- **Conditional Bypass Test:** If a QKD module maintains internal information that governs the bypass capability, then the module shall verify the integrity of the governing information through an approved integrity technique immediately preceding modification of the governing information, and shall generate a new integrity value using the Approved integrity technique immediately following the modification.

  Documentation shall specify the mechanism or logic governing the bypass capability.

4.10.3 Critical Functions Tests

There may be other security functions critical to the secure operation of the QKD module that shall be tested either when the module is powered up or when certain conditions are met. Documentation shall specify all identified critical functions and testing methods.

4.11 Life-Cycle Assurance

Life-cycle assurance refers to the use of best practices by the vendor of a QKD module during the design, deployment, and operation of a QKD module, providing assurance that the module is properly developed, tested, configured, delivered, and installed, and that the proper operator guidance documentation is provided. Security requirements are specified for configuration management, design, finite state model, development, testing, delivery and operation, and guidance documentation.

4.11.1 Configuration Management

The configuration management specifies the security requirements for a configuration management system implemented by a QKD module vendor, providing assurance that the integrity of the QKD module is preserved by requiring discipline and control in the processes of refinement and modification of the QKD module and related documentation. A configuration management system is put in place to prevent accidental or unauthorized modifications to, and provide change traceability for, the QKD module and related documentation.

The following security requirement shall apply to QKD modules:

- A configuration management system shall be implemented for a QKD module and module components within the cryptographic boundary, and for associated module documentation.
- Each version of each configuration item (e.g. QKD module, module hardware parts, module software components, module HDL, user guidance, Security Policy, etc.) that comprises the module and associated documentation shall be assigned and labelled with a unique identification number.
- The configuration management system shall track and maintain the changes to the identification and version or revision of each configuration item throughout the life-cycle of the validated QKD module.

- Documentation shall specify and describe the configuration management system used for the QKD module.
- The configuration items shall be managed using an automated configuration management system.

4.11.2 Design

A design is an engineering solution that addresses the Functional Specification for a QKD module. The design is intended to provide assurance that the functional specification of a QKD module corresponds to the intended functionality described in the Security Policy.

QKD modules shall be designed to allow the testing of the implemented functionality to the present document, where possible without compromising the security of the module, so that all the services of the QKD module can be tested.

The following requirements shall apply to a QKD module:

- Documentation shall specify the correspondence between the design of the hardware and/or software of a QKD module, and the QKD module's Security Policy and Finite State Model.
- Documentation shall specify a Functional Specification that informally describes the QKD module, the functionality of the QKD module, the external physical ports and logical interfaces of the QKD module, and the purpose of the physical ports and logical interfaces.
- Documentation shall specify the detailed design that describes the internal functionality of the QKD module's major components, the internal component interfaces, the purpose of the component interfaces, and the internal information flow (within the cryptographic boundary as a whole and also within the major components).
- Documentation shall specify an informal proof (including the pre-conditions and the post-conditions) of the correspondence between the design of the QKD module and the functional specification.
- Documentation shall specify a formal model that describes the rules and characteristics of the QKD module Security Policy. The formal model shall be specified using a formal specification language that is a rigorous notation based on established mathematics, such as first order logic or set theory.
- Documentation shall specify a rationale that demonstrates the consistency and completeness of the formal model with respect to the QKD module Security Policy.
- Documentation shall specify an informal proof of the correspondence between the formal model and the functional specification.

4.11.3 Finite State Model

The operation of a QKD module shall be specified using a Finite State Model (or equivalent) represented by a state transition diagram and/or a state transition table and state descriptions. The FSM shall be sufficiently detailed to demonstrate that the QKD module complies with all of the requirements of the present document.

Documentation shall include the FSM (or equivalent) using a state transition diagram and/or state transition table and state descriptions that shall specify:

- The operational and error states of a QKD module.
- The corresponding transitions from one state to another.
- The input events, including data inputs and control inputs, which cause transitions from one state to another.
- The output events, including internal module conditions, data outputs, and status outputs, resulting from transitions from one state to another.

The Finite State Model of a QKD module shall include the following operational and error states:

- **Power on/off state:** a state in which the module is powered off or in standby mode, and in which primary, secondary, or backup power is applied to the module. This state may distinguish between power sources being applied to a QKD module.

- **General initialization state:** a state in which the QKD module is initializing non-cryptographic services.
- **Crypto-Officer state:** a state in which the Crypto-Officer services are performed (e.g. QKD initialization, secure administration, and key management).
- **CSP entry state:** a state for entering the CSPs into the QKD module.
- **User state:** (if a User role is implemented) a state in which authorized users obtain security services, perform cryptographic operations, or perform other Approved or non-Approved functions.
- **Approved state:** a state in which Approved security functions are performed.
- **Self-test state:** a state in which the QKD module is performing self-tests.
- **Error state:** a state when the QKD module has encountered an error condition (e.g. fails a self-test or attempt to encrypt without operational keys or CSPs). There may be one or more error conditions that result in a single module error state. Error states may include "hard" errors that indicate an equipment malfunction and that may require maintenance, service or repair of the QKD module, or recoverable "soft" errors that may require initialization or resetting of the module. Recovery from error states shall be possible, except for those caused by hard errors that require maintenance, service, or repair of the QKD module.

Each distinct QKD module service, security function use, error state, self test, or operator authentication shall be depicted as a separate state.

A QKD module may contain other states including, but not limited to, the following:

- **Bypass state:** a state in which a service, as a result of module configuration or operator intervention, causes the plaintext output of a particular data or status item that would normally be output in encrypted form.
- **Quiescent state:** a state in which the QKD module is dormant (e.g. low power, suspended or in hibernation.)

4.11.4 Development

A proper development process provides assurance that the implementation of a QKD module corresponds to the module functional specification and Security Policy, that the QKD module is maintainable, and that the validated QKD module is reproducible. This section specifies the security requirements for the representation of a QKD module's security functionality at various levels of abstraction from the functional specification to the implementation representation.

The following requirements shall apply to QKD modules:

- If a QKD module contains software, documentation shall specify the compilers, configuration settings, and methods to compile the source code into an executable form. The documentation shall also include the source code for the software, annotated with comments that depict the correspondence of the software to the design of the module.
- If a cryptographic module contains hardware, documentation shall specify the schematics and/or Hardware Description Language (HDL), as applicable. The HDL shall be annotated with comments that depict the correspondence of the hardware to the design of the module.
- All software within a QKD module shall be implemented using a high-level, non-proprietary language, except that the limited use of a low-level language (e.g. assembly language or microcode) is allowed if essential to the performance of the module or when a high-level language is not available.
- Custom integrated circuits within a QKD module shall be implemented using a high-level HDL (e.g. VHDL or Verilog).
- For each QKD module hardware and software component, the documentation shall be annotated with comments that specify:

    1) the pre-conditions required upon entry into the module component, function, or procedure in order to execute correctly; and

2) the post-conditions expected to be true when the execution of the module component, function, or procedure is complete.

The pre-conditions and post-conditions may be specified using any notation that is sufficiently detailed to completely and unambiguously explain the behaviour of the cryptographic module component, function, or procedure.

Implementation of software within a QKD module using the recommended development practices listed in annex C will facilitate the analysis of the software for conformance to the requirements in the present document and will reduce the chance of design errors.

4.11.5 Vendor Testing

This clause specifies the security requirements for vendor testing of the QKD module, including testing the security functionality implemented in the QKD module, providing assurance that the module behaves in accordance with the module Security Policy and functional specifications.

The QKD module documentation shall specify the functional testing performed on the QKD module. Functional testing refers to the testing of the QKD module functionality as defined by the Functional Specification required by clause 4.10.2.

The QKD module documentation shall specify the procedures for and the results of low-level testing performed on the QKD module. Low-level testing refers to the testing of the individual components or group of components of the QKD module and their physical ports and logical interfaces as defined by the documentation required by clause 4.10.2.

4.11.6 Delivery and Operation

This clause specifies the security requirements for the secure delivery, installation, and startup of a QKD module, providing assurance that the module is securely delivered to authorized operators, and is installed and initialized in a correct and secure manner.

The QKD module documentation shall specify the procedures for secure installation, initialization, and startup of the QKD module.

The QKD module documentation shall specify the procedures required for maintaining security while distributing and delivering versions of a QKD module to authorized operators. The procedures shall specify how to detect tamper during the delivery of the module to the authorized operators.

The QKD module procedures shall require the authorized operator to authenticate to the module using authentication data provided by the vendor.

4.11.7 Guidance Documents

The requirements in this section are intended to ensure that all entities using the QKD module have adequate guidance and procedures to administer and use the module in a secure manner. Guidance documentation consists of administrator and non-administrator guidance.

Administrator guidance is written material that is used by the Crypto Officer and/or other administrative roles for the correct configuration, maintenance, and administration of the QKD module. The administrator guidance contains information and procedures for administering the QKD module in a secure manner.

Administrator guidance shall specify:

- The administrative functions, security events, security parameters (and parameter values, as appropriate), physical ports, and logical interfaces of the QKD module available to the Crypto-Officer and/or other administrative roles.
- Procedures required keeping independent operator authentication mechanisms functionally independent.
- Procedures on how to administer the QKD module in a secure manner.
- Assumptions regarding User behaviour that are relevant to the secure operation of the cryptographic module.

Non-administrator guidance is written material that is used by the User and/or other non-administrative roles for operating the QKD module in a secure manner. The non-administrator guidance describes the security functions of the module and contains information and procedures for the secure use of the QKD module, including instructions, guidelines, and warnings.

Non-administrator guidance (if the User role is implemented) shall specify:

- The Approved and non-Approved security functions, physical ports, and logical interfaces available to the users of a QKD module.
- All User responsibilities necessary for the secure operation of a QKD module.

4.12 Mitigation of Other Attacks

Susceptibility of a QKD module to attacks not defined elsewhere in the present document depends on the module type, implementation, and implementation environment. Such attacks may be of particular concern for QKD modules implemented in hostile environments (e.g. where the attackers may be the authorized operators of the module). These attacks generally rely on the analysis of information obtained from sources that are physically external to the module. In all cases, the attacks attempt to determine some knowledge about the CSPs within the QKD module.

If a QKD module is designed to mitigate one or more specific attack(s), then the module's Security Policy or other supporting documents shall enumerate the attack(s) the module is designed to mitigate. The existence and proper functioning of the security mechanisms used to mitigate the attack(s) will be validated when requirements and associated tests are developed.

In addition, the following requirement shall apply to QKD modules:

- If the mitigation of other attacks is claimed, documentation shall specify the methods used to mitigate the attacks and the methods to test the effectiveness of mitigation techniques.

Annex A (normative): Summary of Documentation Requirements

The following check list summarizes the documentation requirements of the present document. All documentation shall be provided to the testing facility by the vendor of a QKD module.

QKD MODULE SPECIFICATION

- Specification of the hardware and software configuration items of a QKD module, specification of the cryptographic boundary surrounding these items, and description of the physical configuration of the module.
- Specification of any hardware or software configuration items of a QKD module that are excluded from the security requirements of the present document and an explanation of the rationale for the exclusion.
- Specification of the physical ports and logical interfaces of a QKD module.
- Specification of the manual or logical controls of a QKD module, physical or logical status indicators, and applicable physical, logical, and electrical characteristics.
- List of all security functions, both Approved and non-Approved, that are employed by a QKD module and specification of all modes of operation, both Approved and non-Approved.
- Block diagram depicting all of the major hardware components of a QKD module and component interconnections, including any microprocessors, input/output buffers, plaintext/ciphertext buffers, control buffers, key storage, working memory, and program memory.
- Specification of the design of the hardware and software of a QKD module.
- Specification of all security-related information, including secret and private cryptographic keys (both plaintext and encrypted), authentication data (e.g. passwords, PINs), other CSPs, and other protected information (e.g. audited events, audit data) whose disclosure or modification can compromise the security of the QKD module.
- Specification of a QKD module Security Policy including the rules derived from the requirements of the present document and the rules derived from any additional requirements imposed by the vendor.

QKD MODULE PHYSICAL PORTS AND LOGICAL INTERFACES

- Specification of the physical ports and logical interfaces of a QKD module and all defined input and output data paths.

ROLES, AUTHENTICATION, AND SERVICES

- Specification of all authorized roles supported by a QKD module.
- Specification of the services, operations, or functions provided by a QKD module, both Approved and non-Approved. For each service, specification of the service input, corresponding service output, and the authorized role(s) in which the service can be performed.
- Specification of any services provided by a QKD module for which the operator is not required to assume an authorized role, and how these services do not modify, disclose, or substitute cryptographic keys and other CSPs, or otherwise affect the security of the module.
- Specification of the authentication mechanisms supported by a QKD module, the types of authentication data required to implement supported authentication mechanisms, the authorized methods used to control access to the module for the first time and initialize the authentication mechanism, and the strength of the authentication mechanisms supported by the module, including the rationale supporting the use of multiple authentication mechanisms.

SOFTWARE SECURITY

- Documentation shall specify which approved software integrity techniques are used.
- Documentation shall specify the MSI commands employed by the module.

OPERATIONAL ENVIRONMENT

- Specification of the operational environment for the QKD module.

PHYSICAL SECURITY

- Specification of the physical embodiment and security level for which the physical security mechanisms of a QKD module are implemented. Specification of the physical security mechanisms that are employed by a module.
- If a QKD module includes a maintenance role that requires physical access to the contents of the module, or if the module is designed to permit physical access, specification of the maintenance access interface and how plaintext secret and private keys and other CSPs are to be zeroized when the maintenance access interface is accessed.
- Specification of the normal operating ranges of a QKD module. Specification of the environmental failure protection features employed by a QKD module or specification of the environmental failure tests performed.

PHYSICAL SECURITY - NON-INVASIVE ATTACKS

- Specification of the mitigation techniques against applicable Timing Analysis attacks.
- Specification of the mitigation techniques against applicable SPA attacks.
- Specification of the mitigation techniques against applicable DPA attacks.
- Specification of the mitigation techniques against applicable EME attacks.

SENSITIVE SECURITY PARAMETER MANAGEMENT

- Specification of all cryptographic keys, cryptographic key components, and other CSPs employed by a QKD module.
- Specification of each RBG (Approved RBGs and non-Approved RBG entropy sources) employed by a QKD module.
- Specification of each of the key generation methods (Approved and non-Approved) employed by a cryptographic module.
- Specification of the key establishment methods employed by a QKD module.
- Specification of the key entry and output methods employed by a QKD module.
- If split knowledge procedures are used, proof that if knowledge of n key components is required to reconstruct the original key, then knowledge that any n-1 key components provides no information about the original key other than the key's length.
- Specification of the SSP storage methods employed by a QKD module.
- Specification of the SSP zeroization methods employed by a QKD module.

SELF-TESTS

- Specification of self-tests performed by a QKD module, including pre-operational, conditional, and critical functions tests.
- Specification of the error states that a QKD module can enter when a self-test fails, and the conditions and actions necessary to exit the error states and resume normal operation of a module.
- Specification of all security functions critical to the secure operation of a QKD module and identification of the applicable pre-operational, conditional, and critical functions tests performed by the module.
- If a QKD module implements a bypass capability, specification of the mechanism or logic governing the switching procedure.

LIFE-CYCLE ASSURANCE

- Specification of procedures for secure installation, generation, and start-up of a QKD module.
- Specification of the procedures for maintaining security while distributing and delivering versions of a QKD module to authorized operators.
- Specification of the correspondence between the design of the hardware and software of a QKD module and the QKD module Security Policy (i.e. the rules of operation).
- If a QKD module contains software, specification of the source code for the software, annotated with comments that clearly depict the correspondence of the software to the design of the module.
- If a cryptographic module contains hardware, specification of the schematics and/or the HDL listings for the hardware.
- Functional specification that informally describes a QKD module, the external ports and interfaces of the module, and the purpose of the interfaces.
- Specification of a formal model that describes the rules and characteristics of the QKD module Security Policy, using a formal specification language that is a rigorous notation based on established mathematics, such as first order logic or set theory.
- Specification of a rationale that demonstrates the consistency and completeness of the formal model with respect to the QKD module Security Policy.
- Specification of an informal proof of the correspondence between the formal model and the functional specification.
- For each hardware and software component, source code annotation with comments that specify:

  1) the pre-conditions required upon entry into the module component, function or procedure in order to execute correctly; and

  2) the post-conditions expected to be true when the execution of the module component, function, or procedure is complete.

- Specification of an informal proof of the correspondence between the design of the QKD module (as reflected by the pre-condition and post-condition annotations) and the functional specification.
- For Cryptographic Officer guidance, specification of:

  - the administrative functions, security events, security parameters (and parameter values, as appropriate), physical ports, and logical interfaces of the QKD module available to the crypto officer;
  - procedures on how to administer the QKD module in a secure manner; and
  - assumptions regarding user behaviour that is relevant to the secure operation of the QKD module.

- For User guidance, specification of:
  - the Approved security functions, physical ports, and logical interfaces available to the users of the QKD module; and
  - all user responsibilities necessary for the secure operation of the module.

MITIGATION OF OTHER ATTACKS

- If a QKD module is designed to mitigate one or more specific attacks, specification in the module's Security Policy of the security mechanisms employed by the cryptographic module to mitigate the attack(s).

SECURITY POLICY

See annex B.

Annex B (normative): QKD Module Security Policy

A QKD module Security Policy shall be included in the documentation provided by the vendor. The following clauses outline the required contents of the security policy.

B.1 Definition of QKD Module Security Policy

A QKD module security policy shall consist of:

- A specification of the security rules, under which a QKD module shall operate, including the security rules derived from the requirements of the standard and the additional security rules imposed by the vendor.

The specification shall be sufficiently detailed to answer the following questions:

- What access does operator *X*, performing service *Y* while in role *Z*, have to security-relevant data item *W* for every role, service, and security-relevant data item contained in the QKD module?
- What physical security mechanisms are implemented to protect a QKD module and what actions are required to ensure that the physical security of a module is maintained?
- What security mechanisms are implemented in a QKD module to mitigate against attacks for which testable requirements are not defined in the standard?

B.2 Purpose of QKD Module Security Policy

There are two major reasons for developing and following a precise QKD module security policy:

- To provide a specification of the cryptographic and general security that will allow individuals and organizations to determine whether a QKD module, as implemented, satisfies a stated security policy.
- To describe to individuals and organizations the capabilities, protection, and access rights provided by the QKD module, thereby allowing an assessment of whether the module will adequately serve the individual or organizational security requirements.

B.3 Specification of a Cryptographic Module Security Policy

A QKD module security policy shall be expressed in terms of roles, services, and cryptographic keys and CSPs. At a minimum, the following shall be specified:

- an identification and authentication (I&A) policy;
- an access control policy;
- a physical security policy; and
- a security policy for mitigation of other attacks.

B.3.1 Identification and Authentication Policy

The QKD module security policy shall specify an Identification and Authentication Policy, including:

- all roles (e.g. user, crypto officer, and maintenance) and associated type of authentication (e.g. identity-based, role-based, or none); and
- the authentication data required of each role or operator (e.g. password or biometric data) and the corresponding strength of the authentication mechanism.

B.3.2 Access Control Policy

The QKD module security policy shall specify an Access Control Policy. The specification shall be of sufficient detail to identify the cryptographic keys and CSPs that the operator has access to while performing a service, and the type(s) of access the operator has to the parameters.

The security policy shall specify:

- all roles supported by a cryptographic module;
- all services provided by a cryptographic module;
- all cryptographic keys and CSPs employed by the cryptographic module, including:
  - secret, private, and public cryptographic keys (both plaintext and encrypted);
- authentication data such as passwords or PINs; and
- other security-relevant information (e.g. audited events and audit data);
- for each role, the services an operator is authorized to perform within that role; and
- for each service within each role, the type(s) of access to the cryptographic keys and CSPs.

B.3.3 Physical Security Policy

The QKD module security policy shall specify a Physical Security Policy, including:

- the physical security mechanisms that are implemented in a QKD module (e.g. tamper-evident seals, locks, tamper response and zeroization switches, and alarms); and
- the actions required by the operator(s) to ensure that physical security is maintained (e.g. periodic inspection of tamper-evident seals or testing of tamper response and zeroization switches).

B.3.4 Mitigation of Other Attacks Policy

The QKD module security policy shall specify a Security Policy for Mitigation of other Attacks, including the security mechanisms implemented to mitigate the attacks.

B.4 Security Policy Check List Tables

The following check list tables may be used as guides to ensure the security policy is complete and contains the appropriate details.

Table B.1: Sample Roles and Required Identification and Authentication

Role	Type of Authentication	Authentication Data
Security officer	Biometric and token authentication.	Allowed PKI certificate and fingerprint template.

Table B.2: Sample Strengths of Authentication Mechanisms

Authentication Mechanism	Strength of Mechanism
RSA digital signature.	1 528 bits minimum key length.

Table B.3: Sample Services Authorized for Roles

Role	Authorized Services
Operator.	Power up/down the QKD module.
Configuration Officer.	Firmware update.

Table B.4: Sample Access Rights within Services

Service	Cryptographic Keys and CSPs	Type(s) of Access
QKD serial number ID.	None.	Read.
QKD pairing (Alice-Bob).	Biometric and token authentication.	Read, Write and Execute.

Table B.5: Sample Inspection/Testing of Physical Security Mechanisms

Physical Security Mechanisms	Recommended Frequency of Inspection/Test	Inspection/Test Guidance Details
Electronic Radiation Isolation Test.	Programmed once per month minimum plus randomly.	Check compliance with approved TEMPEST standard.
Power detector in the quantum channel.	Continuously during operation	Confirm the detector threshold fires when out of operational limits of the QKD module and detection circuit integrity check.

Table B.6: Sample Mitigation of Other Attacks

Other Attacks	Mitigation Mechanism	Specific Limitations
Optical flooding of the quantum channel	Shut down the quantum channel.	System unavailable during attack.

Annex C (informative): Recommended Software Development Practices

This annex is provided for informational purposes only and does not contain security requirements applicable to QKD modules within the scope of the present document.

Life-cycle software engineering recommendations (dealing with the specification, construction, verification, testing, maintenance, and documentation of software) should be followed. Software engineering practices may include documented unit testing, code reviews, explicit high-level and low-level design documents, explicit requirements and functional specifications, structure charts and data flow diagrams, function-point analysis, defect and resolution tracking, configuration management, and a documented software development process.

For all software development, both large and small, the following programming techniques are consistent with current practices and should be used to facilitate analysis of software components of a cryptographic module and to reduce chances of programming errors.

MODULAR DESIGN

- A modular design is recommended, especially for moderate to large-scale software development efforts. Each software module should have well-defined and readily understood logical interfaces.
- Software components should be constructed using the principles of data abstraction. If available, an object-oriented, high-level language that supports the construction of abstract data types should be used.
- The software should be hierarchically structured as a series of layers.

SOFTWARE MODULE/PROCEDURE INTERFACES

- Entries to a software module or procedure should be through external calls on explicitly defined interfaces.
- Each procedure should have only one entry point and at most two exit points, one for normal exits and one for error exits.
- Data should be communicated between software modules and between procedures through the use of argument lists and/or explicit return values. Global variables should not be used among procedures except where necessary for the implementation of abstract data types. Input values should be checked for range errors using assertion statements (if provided by the programming language in use).

INTERNAL CONSTRUCTION

- Each procedure should perform only a single, well-defined function.
- Control flow within a single thread of execution should be defined using only sequencing, structured programming constructs for conditionals (e.g. if-then-else or case), and structured constructs for loops (e.g. *while-do* or *repeat-until*).
- If concurrent execution is employed (e.g. via multiple threads, tasks, or processes), the software components should enforce limits on the maximum allowable degree of concurrency and should use structured synchronization constructs to control access to shared data.
- Equivalence of variables should not be used to permit multiple memory usage for conflicting purposes.
- Robust command parsing and range checking mechanisms should be implemented to guard against malformed requests, out-of-range parameters, and I/O buffer overflows.

IN-LINE DOCUMENTATION

- Each software module, procedure, and major programming construct should be documented specifying the functions performed along with a (formal or informal) specification of pre-conditions and post-conditions.
- Each loop should be preceded by a convincing argument (as a comment) that termination is guaranteed.
- Variable names should be used in only one context within the same procedure.
- Each variable should have an associated comment identifying the purpose of the variable and noting the range of allowable values, including if the range is unrestricted.
- If concurrency is employed, the documentation should specify how limits are enforced on the maximum allowable degree of concurrency and how accesses to shared data are synchronized in order to avoid (possibly undetected) run-time errors.

ASSEMBLY LANGUAGE

The following additional programming practices should be used when the implementation is in assembly language.

- All code should be position independent except where appropriate security concerns, efficiency, or hardware constraints require position dependency.
- All register references should use symbolic register names.
- Self-modifying code should not be used.
- All procedures should be responsible for saving and restoring the contents of any register that is used within the procedure.
- Control transfer instructions should not use numeric literals.
- Each unit of code should contain comments describing register use in the unit.

Annex D (informative): Approved Security Function Example: BB84

This annex contains an example of a possible Approved QKD Security Function. Since BB84 is the most widely known QKD protocol, it has been selected to exemplify this function. Sample requirements are given in order to illustrate some of the main parameters to control in QKD Security Functions, by no means they are intended to be complete.

Function General Description

- A general BB84 is described in detail in annex F [BB84]. The principals are Alice and Bob, the two partners linked by a quantum and classical channel.
- In what follows we assume a BB84 protocol executed with single photons and phase coding as qubit carriers.
- Alice device includes a single photon source, a phase modulator and the part of the interferometer corresponding to Alice, including fibre beam splitters and any other required components. It also includes a clock, a Random Bit Generator and associated electronics to drive the optoelectronics subsystems, computational and classical communications subsystems. The whole of the device is included inside a tamper resistant enclosure that defines the cryptographic boundary.
- Bob device includes two single photon detectors, a phase modulator, the corresponding part of the interferometer, electronics to drive the optoelectronics subsystems, computational and classical communications subsystems as in Alice. Again, a tamper resistant enclosure is required to define the cryptographic boundary.
- Alice and Bob have access to a classical communication channel and to an optical fibre used to implement the quantum channel. Both channels are not required to be of exclusive use.
- Awakening from the power-off state, all QKD modules have to pass successfully all the start-up test, the realization of that entire set of tests must not radiate, trough any possible channel, any information outside the cryptographic boundary.
- BB84 protocol has to major phases: Optical sessions, and Computational Key generation. QKD Optical Sessions is the QKD protocol part where both cooperating agents, Alice and Bob, send a prefixed number of qubits and measure to receive those qubits, respectively.

BB84 Approved Operation Mode

- The first step for both partners is the "System Startup":
  - Successful completion of power on self-tests to confirm that all the internal components are present and operating within the approved conditions (e.g. all inside allowed tolerance limits).
  - The system have to operate long enough to insure that no correlation between the initial state and the operational state survives (e.g. no less than 15 minutes, and no less than 10^{12} clock cycles).
- The second step starts with an Optical Session. In BB84 protocol consecutive optical sessions shall be repeated until a predefined size of the Bob storage is reached. The size will be dictated by the user and the security parameters that regulate the quality of the outcome (e.g. to limit finite size effects in the final key):
  - Both partners: Local clock synchronization. (e.g. frequency drift: less than 10^{-8} parts per clock cycle. Jitter: 10^{-2} % of a detector gate length):
    - Alice: As emitter, Alice must perform random and programmed single photon emitter test on its single photon source (e.g. average power test, and single photon pulse assurance test.
  - Both partners: Random and programmed tests of the internal interferometer, including the phase modulator, are required.
  - Bob part: Periodical test is required to insure that indistinguishability among the two single photon detectors from outside the cryptographic boundary is achieved.

- Bob part: Approved functioning requires that all the time, bob sensors will operate in single photon mode. Because of that, tests must be performed, random and periodically, to insure detectors working in single photon regime.
- Both partners: Bob and Alice start their respective BB84 main loops in synchrony using the previous clock synchronization step. Alice and Bob loops are repeated while the clock drift allows and a predefined number of iterations are not reached.
- Alice: Start main BB84 loop (note: tests shall interrupt the loop whenever required. A resume procedure must be specified to resume after test completion. The timing of the tests might be transmitted once the optical session has been completed using the public channel in order to increase its efficiency avoiding intentional service degradation attacks):
  - Pulse laser: emit single photon.
  - Use RBG: 1 bit sets Base value for coding.
  - Use RBG: 1 bit sets qubit Value.
  - Use Phase Modulator to set: Base, Value.
  - Indexed Storage: Base, Value.
  - Repeat.
- Bob: Start main BB84 loop (note: tests shall interrupt the loop whenever required. A resume procedure must be specified to resume after test completion. The timing of the tests might be transmitted once the optical session has been completed using the public channel in order to increase its efficiency avoiding intentional service degradation attacks).:
  - While (Ready State of Photon Detector Modules for duty cycle and):
    - Use RBG: 1 bit sets Base value.
    - Use Phase Modulator to set Base.
    - Use single Photon Detector Modules: Open Detection Gate. Get Value.
    - Indexed Storage: Base, Value:
      - Else: Procedure to reach Photon Detector Module Ready state (e.g.: Wait detector dead time).

- Third And Final Steps: Information reconciliation and Final key distillation.
  - After an agreed number of optical sessions that guarantee a minimum predefined length, blocks of bits specifying Index, Base and Value are located in Bob's and Alice's storage pool.
  - The Value field in blocks of a minimum predefined length in Bob and Alice Indexed Storage undergoes a information reconciliation procedure that uses the index and Base values through and approved procedure (e.g. base publication using the classical, integrity preserving channel followed by a CASCADE or LDPC based error correction).
  - Privacy amplification: An approved privacy amplification procedure might be used on the results of the reconciled blocks to increase the quality of the final key (e.g. hashing). This step might be integral to the information reconciliation procedure.
  - Final Step: Indexed Storage of final key blocks in Alice and Bob. Key blocks have an index that univocally address them such that the same indexes in Alice and Bob refers to identical key blocks.

Annex E (informative): Applicable Internet Uniform Resource Locators

- Communications Security Establishment (CSE): http://www.cse-cst.gc.ca.
- Cryptographic Module Validation Program (CMVP): http://www.nist.gov/cmvp.
- NIST Information Technology Laboratory (NIST ITL): http://www.nist.gov/itl.
- NIST Security Publications including FIPS and Special Publications: http://csrc.nist.gov/publications.
- National Technical Information Service (NTIS): http://www.ntis.gov.
- National Voluntary Laboratory Accreditation Program (NVLAP): http://ts.nist.gov/nvlap.
- National Information Assurance Partnership (NIAP): http://niap.nist.gov/.
- Validated Protection Profiles: http://niap.nist.gov/cc-scheme/PPRegistry.html.
- National Institute of Standards and Technology, *FIPS 140-2 Annex A: Approved Security Functions*, available at URL: http://www.nist.gov/cmvp.
- National Institute of Standards and Technology, *FIPS 140-2 Annex B: Approved Protection Profiles*, available at URL: http://www.nist.gov/cmvp.
- National Institute of Standards and Technology, *FIPS 140-2 Annex C: Approved Random Number Generators*, available at URL: http://www.nist.gov/cmvp.
- National Institute of Standards and Technology, *FIPS 140-2 Annex D: Approved Key Establishment Techniques*, available at URL: http://www.nist.gov/cmvp.

Annex F (informative): Bibliography

- American Bankers Association, Digital Signatures using Reversible Public Key Cryptography for the Financial Services Industry (rDSA), ANSI X9.31-1998, Washington, D.C., 1998.
- American Bankers Association, Triple Data Encryption Algorithm Modes of Operation, ANSI X9.52-1998, Washington, D.C., 1998.
- American Bankers Association, Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm, American National Standard X9.62-1998, Washington, D.C., 1998.
- Common Criteria Implementation Board (CCIB), International Standard (IS) 15408, Common Criteria for Information Technology Security Evaluation, Version 2, May 1998, ISO/IEC JTC 1 and Common Criteria Implementation Board.
- Keller, Sharon and Smid, Miles, Modes of Operation Validation System (MOVS): Requirements and Procedures, Special Publication 800-17, Gaithersburg, MD, National Institute of Standards and Technology, February 1998.
- Keller, Sharon, Modes of Operation Validation System for the Triple Data Encryption Algorithm (TMOVS): Requirements and Procedures, Special Publication 800-20, Gaithersburg, MD, National Institute of Standards and Technology, October 1999.
- Lee, Annabelle, Guideline for Implementing Cryptography in the Federal Government, Special Publication 800-21, Gaithersburg, MD, National Institute of Standards and Technology, November, 1999.
- [BB84] C. H. Bennett and G. Brassard, "Quantum Cryptography: Public key distribution and coin tossing", in Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing, Bangalore, p. 175 (1984)

Deterministic Random Number Generators

1) National Institute of Standards and Technology, NIST-Recommended Random Number Generator Based on ANSI X9.31 Appendix A.2.4 Using the 3-Key Triple DES and AES Algorithms, January 31, 2005.
2) National Institute of Standards and Technology, Recommendation for Random Number Generation Using Deterministic Random Bit Generators (Revised), Special Publication 800-90, March 2007.

Nondeterministic Random Number Generators

1) (none)

Annex G (informative): Authors and contributors

The following people have contributed to the present document:

Rapporteur:

Thomas Laenger
Austrian Institute of Technology
Vienna (Austria)

STF Expert:

Prof. Jorge Dávila Muro
Facultad de Informática
Universidad Politécnica de Madrid
Madrid (SPAIN)

Other contributors:

Prof. Vicente Martín
Facultad de Informática
Universidad Politécnica de Madrid
Madrid (SPAIN)

History

Document history		
V1.1.1	December 2010	Publication