Swedish Certification Body for IT Security

188 Scheme Crypto Policy

Issue: 14.0, 2025-mar-25

Authorisation: Mats Engquist, Quality Manager , CSEC

Template: CSEC_mall_doc_ssl.dot, 7.0

Table of Contents

1 Preface

This document is one of the governing documents for the Swedish Certification Body for IT Security (CSEC).

In this document, "the Scheme" refers to any or all of the Certification Schemes under which CSEC performs certifications and issues certificates.

The Scheme has been established by the Swedish Certification Body for IT Security (CSEC) to evaluate and certify the trustworthiness of security features in IT products and the suitability of protection profiles (PP) to define implementation-independent sets of IT security requirements.

The objectives of the Scheme are to ensure that all evaluations are performed to high and consistent standards and are seen to contribute significantly to confidence in the security of those products and protection profiles; to improve the availability of evaluated IT products and protection profiles; and to continuously improve the efficiency and cost-effectiveness of the evaluation and certification process for IT products and protection profiles.

This document is part of a series of documents that provide a description of aspects of the Scheme and procedures applied under it. This document is of value to all participants under the Scheme, i.e., to anyone concerned with the development, procurement, or accreditation of IT systems for which security is a consideration, as well as those already involved in the Scheme, i.e., Scheme employees, evaluators, current customers, contractors, and security consultants.

The Scheme documents and further information can be obtained from the Swedish Certification Body for IT Security here:

Swedish Certification Body for IT Security
FMV / CSEC
Postal address: SE-115 88 Stockholm, Sweden
Visiting address: Banérgatan 62

Telephone: +46-8-782 4000 E-mail: csec@fmv.se
Web: www.csec.se

1.1 Purpose

This document provides instructions for evaluations of targets of evaluation (TOE) with cryptographic functionality, including a list of cryptographic algorithms that may be subject to Common Criteria (CC) evaluation, instructions how to define the target of evaluation boundaries, and rules for specification of security functional requirements (SFR) in a protection profile (PP) or security target (ST).

General information about the Scheme is published in External publications EP-001 *Certification and Evaluation - Overview*, and EP-301 *Certification and Evaluation – EUCC – Overview*.

1.2 Terminology

Abbreviations commonly used by CSEC are described in EP-001 *Certification and Evaluation - Overview.*

The following terms are used to specify requirements:

SHALL	Within normative text, “SHALL” indicates “requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.” (ISO/IEC).
SHOULD	Within normative text, “SHOULD” indicates “that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required.” (ISO/IEC) The CC interprets 'not necessarily required' to mean that the choice of another possibility requires a justification of why the preferred option was not chosen.
MAY	Within normative text, “MAY” indicates “a course of action permissible within the limits of the document.” (ISO/IEC).
CAN	Within normative text, “CAN” indicates “statements of possibility and capability, whether material, physical or causal.” (ISO/IEC).

2 Scheme Crypto Policy

This Scheme Publication deals with some issues related to the use of cryptography in CC evaluations.

- Cryptographic requirements when claiming a cPP or a NIAP PP
- Using FIPS CAVP testing in CC evaluations
- Third party cryptographic functionality in the operational environment
- Selection of cryptographic algorithms etc. in an ST
- How to specify cryptographic functionality in an ST
- Specifying many cryptographic mechanisms in “one” SFR

2.1 Collaborative PPs and NIAP PPs

In collaborative PPs (cPPs), and NIAP PPs, crypto modules often are part of the TOE. In such cases, the ST writer may not place the crypto module in the operative environment.

In general, when such a PP, or its companion supporting document, has specific claims that conflicts with CSECs general requirements, the PP and its supporting document has precedence.

However, please note that:

- To achieve SOGIS recognition, all requirements of the CEM must also be met.
- In the ST, all operations on the SFRs must be completed. Even under exact conformance, an ST may not contain statements like “keylengths 2048 bit and greater”. The ST writer has to select a specific subset of the mechanisms offered in the PP. Each claimed keylenght must be specified in an SFR in the ST. Open requirements are only allowed in PPs.

2.2 FIPS CAVP Testing

The FIPS CAVP testing aims to verify that a cryptographic implementation/module complies with certain cryptographic standards. FIPS CAVP testing corresponds to the testing of internal interfaces between sub-systems, and between modules, required by CEM at EAL3 and higher (i.e. when there is an ATE_DPT requirement in the claimed assurance package).

Note that the cryptographic behaviour of the TOE itself must always be tested (i.e. testing corresponding to the ATE_COV requirement).

FIPS CAVP testing that is part of the developer testing, may be performed by a third party on the developer’s behalf.

At higher EALs, the evaluators are expected to repeat or otherwise verify a greater amount of the developer’s tests, some of which may be FIPS CAVP tests. Also in cPP/NIAP PP evaluations, FIPS CAVP testing may occur as part of the evaluator testing.

Note that evaluator tests always have to be performed under full control of the evaluators. The tested version of the cryptographic implementation/module has to be identical to the version used in the TOE.

2.3 Third Party Cryptography

Some products use cryptographic mechanisms, implemented in a third party component, to protect assets. Considering such third party components to be part of the TOE in a CC evaluation is not always possible because:

- at EAL4, and higher, source code for the entire TOE shall be made available to the evaluators. For third party components, this may not be available to the TOE developer, and therefore not to the evaluator.
- at EAL3, it is expected that the TOE developer keeps all source code under CM control, which requires that the source code is available to the TOE developer. The third party source code is not necessarily available to the TOE developer.
- when an ALC_FLR requirement is present in an ST, the developer is expected to have procedures to ensure that known vulnerabilities are mitigated or eliminated. Removing vulnerabilities in third party components depend on the procedures of the third party and removing them cannot be guaranteed by the TOE developer.

In a CC evaluation, a third party implementation of cryptographic mechanisms may be placed in the operational environment. If the TOE depends on these cryptographic mechanisms to protect assets, and if the TOE explicitly invokes the cryptographic mechanisms, an SFR must be present in the ST for each such invoked cryptographic mechanism used by the TOE. These SFRs represent the TOEs correct usage of the third party functionality, not the correctness of the third party implementation per se.

These third party cryptographic mechanisms may encompass trusted paths, trusted channels, signatures, data encryption, key management, Diffie-Hellman etc.

The third party implementation must be a distinct product with a unique identity (product name and version) specified in the ST. It is acceptable to specify several choices for the cryptographic third party components, but each of them have to be uniquely specified in the ST, and full testing has to be performed with each.

Dependencies from these SFRs (invocing cryptographic functionality implemented in the environment) must be satisfied by the TOE when these dependencies involves actions from the TOE. The dependency rationale in the ST has to consider all dependencies, satisfied by the TOE or not.

Each SFR for the TOE, implemented in the operational environment, should contain an “application note” stating that “This SFR corresponds to the correct invocation by the TOE, but not the implementation of cryptographic functionality”.

Note that even if the functionality is implemented in the operational environment, the dependencies (e.g. key generation, or key destruction) may be implemented in the TOE.

Testing of SFRs corresponding to cryptographic functionality in the environment should focus on whether the TOE invokes the intended functionality or not. If several choices are given for these cryptographic third party components, testing has to be performed for each choice. Testing is described in section 2.4.

When using a reference implementation to verify the cryptographic functionality, this reference implementation shall be independent from the implementation, or all relevant parts shall have been verified by a national certification/validation scheme of good standing.

2.4 Testing of Cryptographic Functionality

When testing 3rd party crypto implemented in the environment, the focus is on the TOE and aims to verify that the TOE properly implements the calls to the crypto implementation in the environment. Since the cryptographic implementation/module is in the environment, it will not have any internal module interfaces in the TOE.

The testing of cryptographic functionality, implemented in the TOE or in the environment, may be done using a reference implementation.

A reference implementation should have a code base that is separate from the cryptographic implementation used by the TOE. It is also acceptable to use a cryptographic implementation that has been validated/certified by a validation scheme appointed by any of the CCRA/SOGIS member nations. In this case the implementations do not need to be separate.

All claimed algorithms, key lengths, curves, modes, signature schemes, HMAC variants, cipher suites, etc. used by the TOE and claimed in the ST, are potentially subject for testing. When TOE is using several distinct crypto implementations, each of them should be tested fully. Please note that at lower EALs (EAL1 and EAL2) full coverage may not be required.

2.5 Selecting Cryptographic Algorithms for the ST

Cryptographic Algorithms, used in an ST should be well known, be in common use, and there should be proper reference implementations available for testing.

Algorithms with known weaknesses should be avoided. This also applies to cryptographic schemes and protocols.

The cryptographic algorithms, schemes and protocols claimed in the ST should be considered in the vulnerability analysis in AVA.

Guidance for choosing cryptographic mechanisms for an ST may be issued by national bodies such as “The Swedish Armed Forces’ National Communications Security Authority” (i.e. the Swedish NCSA).

Another recommended guide is “SOGIS Agreed Cryptographic Mechanisms” available from sogis.eu. The latest available version should be used.

2.6 How to Specify Cryptographic Functionality in an ST

The specification of cryptographic functionality can be very different in different STs, making the cryptographic specification in the ST harder to understand.

The preferred way is to specify one cryptographic operation at a time, including the algorithm used (including scheme or mode), all key lengths used, and standards that define how the algorithm works, and standards defining the scheme/mode. If there are no specific reasons not to, the latest standard versions should be referenced.

Key generation (FCS_CKM.1) is not always specified in detail, the standard specifying the algorithm may say “128 random bits”. In such cases, this statement in the algorithm standard may be sufficient. If there is an extended SFR for random bit generation, further information/references may be necessary.

FCS_COP.1 examples:

For an operation “data encryption in IPSec“, the algorithm and mode could be specified together as “AES with CBC”, the key length could be specified as “128 or 256 bit”, and the defining standards could be “FIPS 197, SP 800-38A”.

A second example:

Operation “Digital signature”, algorithm ”RSASSA PSS”, key length ”2048 bit”, and standard “PKCS#1 v2.2”.

FCS_CKM.1 example:

Key generation algorithm “key generation for AES”, cryptographic key sizes “128 and 256 bit”, standard “FIPS 197”

2.7 Specifying Many Cryptographic Mechanisms in “one” SFR

Sometimes cryptographic SFRs (such as FCS_COP.1 and FCS_CKM.1) are not specified one by one, but are written together as one SFR with a list or table specifying a large number of operations, algorithms, key lengths and standards.

Writing several cryptographic SFRs as one is acceptable as long as it is easy to see which operations, algorithms, key lengths and standards are intended, and how they relate to each other.

This does not mean that the expected test coverage is lower because all cryptography has been reduced to “one” Security Functional Requirement (SFR). Each separate mechanism, key length, curve etc. still counts as one requirement, though the requirements are presented in a more compact manner.