CEN/TC 224

Date: 2016-11-29 (v0.15)

Proposed draft for Evaluation of prEN 419 221-5

CEN/TC 224

Secretariat: AFNOR

Protection profiles for TSP Cryptographic modules - Part 5

Cryptographic Module for Trust Services

ICS:

Descriptors:

Contents

List of Tables

List of Figures

Foreword

This document (prEN 419221-5:2015) has been prepared by Technical Committee CEN/TC 224 “Personal identification, electronic signature and cards and their related systems and operations”, the secretariat of which is held by AFNOR.

Revision History

PRE-RELEASE HISTORY FOR EDITORIAL TRACKING ONLY, REPLACE FOR FINAL PP

v0.01	2013-10-04	Initial draft of introduction
v0.02	2014-03-28	Initial draft of Security Problem Description
v0.03	2014-05-16	Updated following comments at April 2014 meeting of CEN TC 224 WG17, and adding initial draft of Security Objectives
v0.04	2014-05-29	Updated following editorial review.
v0.05	2014-08-29	Updated to add draft SFRs and rationales
v0.06	2014-11-24	Updated following discussion and review
v0.07	2015-02-24	Updated following discussion and review
v0.08	2015-03-31	Updated following discussion and review – changes include: removed auditor role and associated actions (because these are to be handled by the higher-level system using the HSM); noted that an ST may identify additional channels to applications with only integrity and authentication properties for FTP_TRP.1 but must provide at least one channel with confidentiality; removed Master Keys and made only restore of the TOE subject to dual-control; removed the ability to import/export Assigned keys
v0.09	2015-05-13	Minor editorial updates
v0.10	2015-05-26	Updated following discussion and review
v0.11	2015-07-02	Updated following discussion and review
v0.12a	2015-11-27	Updated following discussion and review
v0.13a	2016-04-04	Changes to address ITSEF observation report and comments at March 2016 meeting of CEN TC 224 WG17
v0.14	2016-05-04	Changes to align FIA_UAU.6/KeyAuth with latest definition of Signature Activation Data and Signature Activation Protocol
v0.15	2016-11-29	Updated in response to Certification Body comments and further comments discussed in June 2016 and September 2016 meetings of CEN TC 224 WG17

Scope

This part of EN 419 221 specifies a Protection Profile for cryptographic modules which is intended to be suitable for use by trust service providers supporting electronic signature and electronic sealing operations, certificate issuance and revocation, time stamp operations, and authentication services, as identified by the (EU) No 910/2014 regulation of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (eIDAS) in [Regulation]. The Protection Profile also includes optional support for protected backup of keys.

Correspondence and comments on this document should be referred to:

CONTACT ADDRESS

CEN/ISSS Secretariat
Avenue Marnix 17,
1000 Brussels, Belgium

Tel +32 2 550 0813
Fax +32 2 550 0966

Email isss@cenorm.be

Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 15408-1:2009 Information technology – Security techniques – Evaluation criteria for IT security – Part 1: Introduction and general model.

ISO/IEC 15408-2:2008 Information technology – Security techniques – Evaluation criteria for IT security – Part 2: Security functional components.

ISO/IEC 15408-3:2008 Information technology – Security techniques – Evaluation criteria for IT security – Part 3: Security assurance components.

NOTE The following are equivalent to the aforementioned ISO/IEC 15408 standards:

Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; Version 3.1, Revision 3. CCMB-2009-07-001, July 2009.

Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; Version 3.1, Revision 3. CCMB-2009-07-002, July 2009.

Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Components; Version 3.1, Revision 3. CCMB-2009-07-003, July 2009.

ISO/IEC 19790:2012 Information technology – Security techniques – Security requirements for cryptographic modules

Conventions and Terminology

Conventions

The document follows the rules and conventions laid out in Common Criteria part 1 [CC1], Annex B "Specification of Protection Profiles".

Terminology

For the purposes of this document, the acronyms, terms and definitions given in EN 419221-1 apply.

Common Criteria terms and definitions are given in [CC1].

Additional terms defined for the purposes of this document are listed below.

Assigned Key

A key (usually a secret key) with the 'Assigned Flag' attribute set to 'assigned', meaning that:

- the 'Re-authorisation conditions' and 'Key Usage' attributes (see sections 1.3.1.2 and 6.3.6) cannot be changed
- the Authorisation Data attribute can only be changed by presentation of the current Authorisation Data – it cannot be changed or reset by an Administrator
- the key cannot be imported or exported.

These properties of an Assigned Key support the sole control of a key that is required for secret keys used to create digital signatures.

Authorisation Data

Data, including data particular to the user, which is used to control access to (and thus use of) a key.

Data particular to the user may include data derived from a secret known only by the user, data derived from a device held by the user and/or data derived from biometric features of the user. Other parts of the authorisation data may include data held within the cryptographic module, data held by administrator(s) or data provided by the application.

An illustration of authorisation data in support of signature activation for server signing, as specified in [CEN TS 419 241], is illustrated in the following figure:

Electronic Seal

Data in electronic form which is attached to or logically associated with other data in electronic form to ensure the latter's origin and integrity.

Electronic Timestamp

Data in electronic form which binds other data in electronic form to a particular time establishing evidence that the latter data existed at that time.

Secret Key

Either a secret key used in symmetric cryptographic functions, or a private key used in asymmetric cryptographic functions.

Trust Service

Electronic service which enhances trust and confidence in electronic transactions

NOTE: Such trust services are typically but not necessarily using cryptographic techniques or involving confidential material.

Document Structure

Section 1 provides the introductory material for the Protection Profile.

Section 2 provides the conformance claim

Section 3 provides the Security Problem Definition. It presents the Assets, Threats, Organisational Security Policies and Assumptions related to the TOE.

Section 4 defines the security objectives for both the TOE and the TOE environment.

Section 5 presents the extended components that will be used in this PP.

Section 6 contains the functional requirements and assurance requirements derived from the Common Criteria (CC), Part 2 [CC2] and Part 3 [CC3] that must be satisfied by the TOE.

Section 7 provides rationales to demonstrate that:

- Security Objectives satisfy the policies and threats
- SFR match the security Objectives
- SFR dependencies are satisfied
- The SARs are appropriate.

A reference section is provided to identify background material.

An acronym list is provided in Appendix A to define frequently used acronyms.

A Mapping to the EU ‘Requirements For Qualified Electronic Signature Creation Devices’ is provided in Appendix B.

1 Introduction

This section provides document management and overview information that is required to carry out protection profile registration. Section 1.1 “PP Reference” gives labelling and descriptive information necessary for registering the Protection Profile (PP). Section 1.2 “Protection Profile Overview” summarises the PP in narrative form. Section 1.3 "TOE Overview" summarises the TOE in a narrative form. As such, these sections give an overview to the potential user to decide whether the PP is of interest.

1.1 Protection Profile Reference

Title	Common Criteria Protection Profile – Cryptographic Module for Trust Service Providers
CC revision	v3.1 release 4
PP version	0.15
Authors	WG17
Publication Date	xx / yy / 2016 ***[**TBD]***
Keywords	cryptographic module
Registration	xxx/yyy ***[**TBD]***

1.2 Protection Profile Overview

This Protection Profile (PP) defines the security requirements for cryptographic modules used by trust service providers supporting electronic signing and sealing operations and authentication services. It includes optional support for protected backup of keys.

The protection profile is aimed at supporting trust services providers as identified by the proposed regulation of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (eIDAS) in [Regulation].

The Cryptographic Module, which is the Target of Evaluation (TOE), generates and/or protects secret keys and other sensitive data, and allows controlled use of these data for one or more cryptographic services in support of TSP trust services.

This PP is Common Criteria Part 2 extended and Common Criteria Part 3 conformant. The assurance level for this PP is EAL4, augmented with AVA_VAN.5 (Advanced methodical vulnerability analysis)

1.2.1 EU Qualified Electronic Signature / Seal Creation Device

Cryptographic Modules certified to this PP are intended to meet the security assurance requirements of Qualified Electronic Signature, and Electronic Seal, Creation Devices for use by trust service providers as specified in Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market [Regulation], although its use is not necessarily limited to such services. For further information see Appendix B.

This Protection Profile is established by CEN for use by trust services including qualified trust services as identified in [Regulation].

1.3 TOE Overview

1.3.1 TOE type

The TOE is a cryptographic module suitable for use by trust service providers supporting electronic signature and electronic sealing operations, certificate issuance and revocation, time stamp operations, and authentication services (including support of authentication of client applications or authorised users of secret keys, and support of authentication for electronic identification), as identified by the (EU) No 910/2014 regulation of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (eIDAS) in [Regulation]. The TOE may also support protected backup of keys.

The TOE provides cryptographic functions that support trust services but is not, in general, aware of the context in which a cryptographic function is used. Any such context is therefore the responsibility of client applications used by the trust service provider, and these client applications need to use the cryptographic functions in an appropriate way. In general this will be achieved by suitable configuration of the TOE and its stored data (for example: to ensure that secret keys intended for electronic signature creation are only available for use by the signatory to whom they are linked, the client application must follow an appropriate process to generate the key pair, to maintain sole control of the secret key by the intended signatory, and to ensure that the key can only be used for signing). As well as providing cryptographic functions, the TOE manages and protects the cryptographic keys used by these functions[1].

The TOE is therefore a set of configured software and hardware. Due to the generic TOE definition in this PP, the particular hardware/software/firmware required by the TOE is not defined by this PP. A generic TOE architecture is shown in Figure 1.

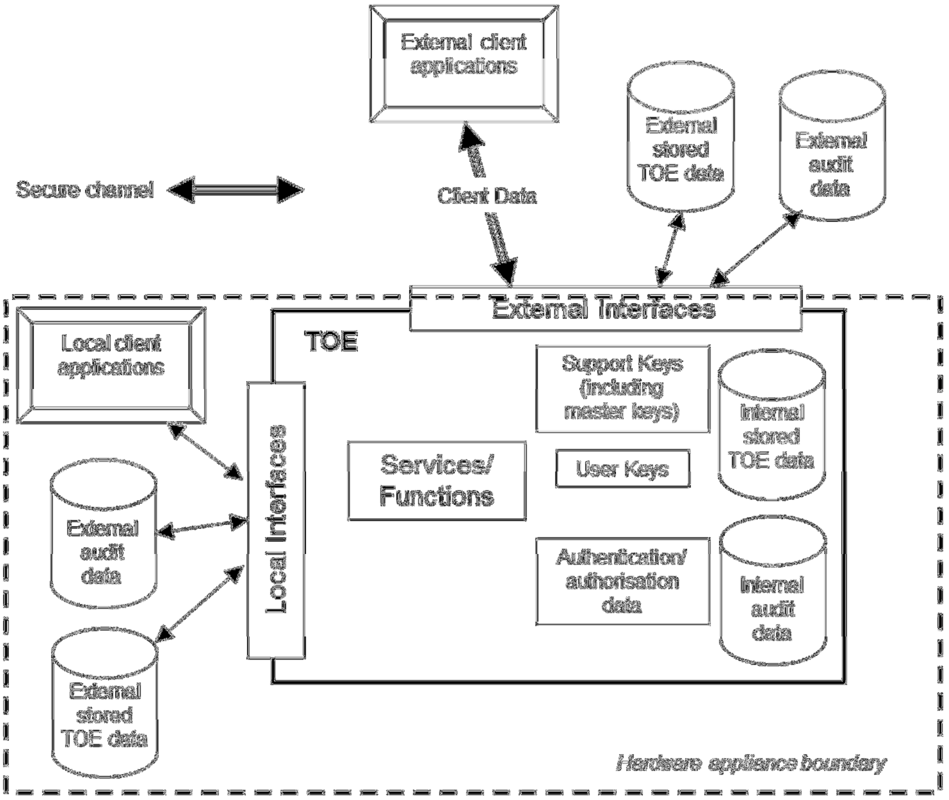


Figure 1: Generic TOE Architecture

The hardware appliance boundary in Figure 1 represents the enclosure of the computing appliance which hosts the TOE. This can be a server, a PC or equivalent.

Local client applications reside in the same hardware appliance as the TOE, e.g. in the case of the TOE being a PCIe card inside a server, local client applications are the applications running within the same server boundary and using the TOE's services through the PCIe bus. Another example of local client application is an embedded application running inside the physical boundary of the TOE.

[1] As described in footnote 6, this Protection Profile includes a refinement to ADV_ARC.1 to consider support keys used in the implementation of the TOE and its protection measures.

External client applications communicate remotely with the TOE through a network connection.

In all cases, the Client Application is outside the scope of the TOE.

A specific TOE will not necessarily include all of the elements shown in Figure 1. A TOE that comprises a PCIe card located in a server may have only local interfaces, e.g. for local client applications and storage of audit and TOE data within the server hardware boundary (which in this case is the hardware appliance boundary in Figure 1), but a dedicated cryptographic module might not include any such local storage and may use only external interfaces. The Security Target for each specific TOE is required to make clear what resources and channels are provided by that TOE.

The TOE is intended to support the provision of cryptographic functions for use by trust service providers.

The TOE implements separate authentication or authorisation[2] of the following distinct types of entity:

- administrators of the TOE
- application users of TOE cryptographic functions (local or external client applications, authenticated by their use of secure channels)
- users of secret keys (which in at least some cases need to have their use limited to a certain natural person or legal person[3]).

Acceptable authentication mechanisms include but are not limited to:

- Shared secret (e.g. password or key)
- Authentication based on asymmetric cryptography
- Physical tokens
- Biometrics
- One time password.

More specific requirements on authentication may be applicable in the case of a TOE performing remote signing, as noted in section 1.3.2.2, but these requirements are based on conformance with further Protection Profiles or other system security requirements directed specifically at remote signing.

If the TOE supports external client applications, then they are required to use a channel that provides authentication of its end-points and protection of confidentiality and integrity of data sent on the channel[4]. Where local client applications are connected to the TOE by a channel such as a PCIe bus within the same hardware appliance protected by measures in the physical environment, then the secure environment may be considered sufficient to provide the authentication, confidentiality and integrity protection needed for communication between the TOE and local applications. Secure channels may also exist between external and local client applications, but these are not within the scope of this Protection Profile.

Authorisation as a user of a secret key is always separately required before a key can be used in a cryptographic function (or exported), regardless of any other authorisation that may have been established for administrators or client applications. This requirement reflects the distinct activities that are being authorised in each case. Authorisation to act as an administrator is an authorisation to carry out management activities on the TOE, but not to *use* keys (in fact the requirement to be able to support sole control of a signature key means that in such cases an administrator must not have access to use keys or to be able to access their values, unless the administrator happens also to demonstrate authorisation as the owner of that key). Client applications are authorised to connect to the TOE in order to be able to invoke cryptographic functions, but the ownership of keys used in such

[2] In this document ‘authentication’ implies that the user is specifically identified, whereas ‘authorisation’ implies that the authority of the user to use the key is established but the identity of the individual may not be known (e.g. where a single key is available to a number of individuals using a shared passphrase). As noted elsewhere, it is the responsibility of client applications to ensure that they use the correct mechanism for the context of the relevant keys and cryptographic functions.

[3] More details of these requirements and the definitions of natural and legal persons can be found in [Regulation].

[4] A TOE may provide some additional channels that provide only authentication and integrity protection, but it must provide at least one channel that is also capable of protecting confidentiality.

functions must be separately controlled and checked, since the keys will in general be controlled by a variety of individual users with interests that are distinct from the client application itself (for example the client application may supply a signature service to a number of different users).

The requirement for authorisation at the level of individual keys also means that a cryptographic function will only be carried out by the TOE if authorisation is obtained for use with a key that can be used with that cryptographic function. Thus, a request by a client to use a specific cryptographic function may fail if the attributes of the key supplied do not allow its use for that operation. The authorisation data supplied in order to use a key will vary according to the environment in which the TOE is used, and the services which it supplies. The authorisation data could, for example, combine multiple factors or could itself be encrypted (requiring decryption by the TOE before use). The authorisation data may reach the TOE in a variety of ways, including transmission over a secure channel or direct entry at an input device connected to the TOE.

1.3.1.1 Cryptographic Functions

The TOE provides one or more of the following cryptographic functions:

- Digital signature generation and verification
- Message digest generation
- Message authentication code generation and verification
- Encryption and decryption (symmetric and asymmetric)
- Key generation
- Key agreement and distribution
- Key derivation
- Generation of shared secret values
- Cryptographic support for one time password and other non-PKI based authentication mechanisms
- Random number generation.

These functions may also be used to support TSP system functions to create electronic seals and electronic timestamps. From the perspective of this Protection Profile, specific cryptographic purposes such as electronic signatures and electronic seals are not distinguished: they both consist of a series of cryptographic functions (such as creating message digests, or encrypting data) using specific keys[5]. A Security Target that conforms to this Protection Profile will identify the precise cryptographic operations (including details of algorithms, key lengths and modes, as appropriate) provided by a specific TOE to carry out these purposes.

1.3.1.2 Key Management

The TOE supports the secure management of cryptographic keys[6] necessary for its implemented cryptographic functions, including:

- Key establishment (including key generation)
- Protection of keys held within the TOE and held externally (for use by the TOE);
- Control of access and use of keys by the cryptographic functions within the TOE
- Deletion of keys within the TOE.

The TOE supports at least one of the following techniques for establishing keys[7]:

[5] Some cryptographic operations, such as creating message digests, do not require keys.

[6] This Protection Profile distinguishes support keys from user keys, as described in the refinement to ADV_ARC.1 in section 6.4.1. Requirements are placed by the Protection Profile on user keys, and support keys are considered an aspect of the TOE implementation that is therefore required to support the requirements for user keys, but where different structures and mechanisms (including aspects such as critical attributes) may be used.

1. Generation of cryptographic keys using a random number generator and implementing the key generation algorithms depending on the intended use of the keys
2. Import of cryptographic keys in encrypted form or cryptographic key components using split-knowledge procedures
3. Key agreement protocols establishing common secrets with external entities
4. Derivation of keys from shared knowledge.

Secret keys are associated with attributes that determine their use, such that the correct association between the key and its attributes must be protected against unauthorised modification. The specific key attributes maintained by a particular TOE are required to be specified in its Security Target. In generic terms these attributes include[8]:

- The identifier of the key (this enables it to be linked by an application to a particular owner)
- The type of the key (e.g. whether the key is a secret key of a symmetric cryptographic algorithm or the secret (commonly called private) key of an asymmetric cryptographic algorithm)
- Authorisation data that enables access to the key (required only for secret keys)
- Re-authorisation conditions such as determining a time period or number of uses of a key that are enabled by a single presentation of the correct authorisation data for the key, after which the authorisation will have to be re-presented in order to authorise any further uses of the key (re-authorisation conditions are required only for secret keys, and may not be the same for all types of secret keys: the details of the re-authorisation conditions for a specific TOE are described by completing the selections and assignments in FIA_UAU.6/KeyAuth in section 6.3.2)
- Key usage constraints that determine which cryptographic functions that can use the key (e.g. encryption or signature)
- Whether the key is allowed to be exported
- Whether the key is an Assigned Key (see further discussion of assigned keys in the definition of FMT_MSA.1/AKeys in section 6.3.6)
- Integrity protection data that protects the integrity of the key value, the values of the key attributes, and the binding of the key to its attributes.

Authorisation to change the attributes of a key is, in general, distinct from authorisation to use the key for cryptographic functions. For example, a signature key may need to require that some or all of its attributes cannot be changed after initial definition (e.g. because such changes might enable subjects beyond the signatory alone to access the key, or might allow the permitted use of the key to be changed) – this is supported by the definition of an ‘Assigned Key’ which cannot be imported or exported, for which the re-authorisation conditions and key usage cannot be changed, and for which the authorisation data can only be changed on successful validation of the current authorisation data.

Keys may leave the TOE in one of three possible situations:

- External storage of keys

  The TOE may allow external storage of keys for later use by the TOE (or another instance of the TOE within the same authorised security infrastructure operated by a TSP). This reflects the fact that when dealing with large numbers of keys then a cryptographic module may not have sufficient internal storage to hold them all internally. Keys stored in this way correspond to ‘external stored TOE data’ in Figure 1, and the form in which the key is stored must be sufficient to protect the confidentiality (for at least secret keys) and integrity of the key and the binding of the key to its attributes (in particular the requirements of the SFRs FDP_IFF.1/KeyBasics, FDP_ACF.1/KeyUsage and FDP_SDI.1 in section 6.3 apply to keys even when they are externally stored). The type and format of this storage for normal operational purposes may be used as part of a secure backup as described in section 1.3.1.4.

[7] SFRs are defined in section 6 only for random number generation and import; however a Security Target may add SFRs for additional techniques supported by the TOE.

[8] In particular these attributes must be sufficient to allow a secret key to be identified as one that is used to produce qualified electronic signatures and qualified electronic seals that meet the requirements of [Regulation], as interpreted for a specific TOE according to the definition in item 1 of the refinement to AGD_OPE.1 in section 6.4.1.

- Export of keys

  The TOE may allow export of keys for use by authorised client applications, provided that they are not Assigned Keys, that other key attributes do not prohibit export, and that the correct authorisation data for the key has been supplied. Although the TOE checks key attributes to determine whether to allow export, the appropriate values to use for the key attributes will depend on the application context in which the key is used, and the security measures (technical, physical and procedural) that apply to that context. Keys that are exported are not included in the 'external stored TOE data' in Figure 1.

  Keys might be imported or exported as part of providing general cryptographic functions (e.g. in support of client applications that use the TOE to support their own authentication mechanisms), but the TOE also allows individual secret keys to be identified as non-exportable. Assigned keys cannot be imported or exported, and represent a more strongly controlled type of key that is intended to be used only within the TOE for operations such as electronic signature or electronic seal generation.

- Backup

  The TOE may provide facilities for secure backup and restore of the TSF state, as described in section 1.3.1.4.

A distinction is drawn between export of keys (as a means of storing for future use by the TOE, or for passing to client applications) and creation of backups: the TOE may use separate mechanisms for these operations.

The TOE supports at least one of the acceptable authentication mechanisms in section 1.3.1 above to be used as a basis for authorisation to access and use secret keys.

1.3.1.3 Cryptographic Algorithms

Only algorithms and algorithm parameters (e.g. key length) approved for the identified purpose shall be used by the TOE to carry out cryptographic operations for trust services. The Security Target author should therefore consult the notified body or the relevant national certification body for the admissible algorithms, cryptographic key sizes and other parameters for algorithms, and standards for trust services.

An exemplary list of algorithms and parameters can be found in [TS 119 312] or [SOG-IS-Crypto].

1.3.1.4 Backup

The TOE may support backup and restoration of the TSF state necessary to re-establish an operational state after failure. This is not a mandatory capability of the TOE, but if a backup mechanism is provided then it must preserve the security requirements on keys. Backups may include their own copies of keys, or may make use of a copy of the externally stored form of the keys (i.e. 'external stored TOE data' in Figure 1). The TOE will protect the confidentiality of the backup data and detect loss of the integrity of the backup data (including the attributes of the keys). It is assumed that the availability is supported by the IT environment outside the scope of the TOE. The TOE shall also ensure that any backup data supports the necessary controls over access to secret keys (including *use* of the key) as required for the intended use (i.e. the application context) of the key[9].

Because the TOE is intended to operate as part of a TSP system that operates under strong environmental and procedural controls, there is a requirement that, if the TOE provides a backup capability, then the corresponding 'restore' operation can only be carried out under at least dual person control, i.e. the restore must be approved by at least two separate administrators.

1.3.1.5 Audit

The cryptographic module is assumed to be part of a larger system that manages audit data for the system as a whole (integrating audit records from a number of individual components). The TOE therefore logs audit records for its own actions, and it is assumed that these are collected, maintained

[9] The TOE may provide a single protection method for backups, provided that this is consistent with the other requirements on protection of keys, or may provide different methods according to differences in the controls needed for keys.

and reviewed in the larger system. Hence there is no separate auditor role within the cryptographic module TOE, but the role of System Auditor is assumed to exist in the larger system. An example of the audit process would be for the cryptographic module TOE to export audit data to a separate audit server that is monitored and controlled by the System Auditor.

1.3.2 Usage and major security features of the TOE

In most cases the TOE will be a separate component with its own hardware and software, communicating via a well-defined physical and logical interface with the client application in the TSP system. Examples of physical interfaces that may be used to connect the TOE to client applications include the PCI bus, the SCSI bus, USB or Ethernet. Several instances of a TOE may be combined in a single domain under a common infrastructure, but the nature of this combination and common infrastructure is beyond the scope of this Protection Profile.

The threat environment the TOE is designed for is one of high threat of network compromise, and low threat of physical compromise (for example, a Certification Authority facility with a high degree of physical protection, but an operational requirement to be connected to an untrusted network such as the internet).

The environment is assumed to prevent prolonged unauthorised physical access to the TOE (including theft). The TOE provides physical protection mechanisms to deter undetected compromise of its security functions by low attack potential individuals that do have physical access to the TOE (for example disgruntled employees with legitimate access to the TOE).

The TOE is responsible for protecting the keys against logical attacks that would result in disclosure, compromise and unauthorised modification, and for ensuring that the TOE services are only used in an authorised way.

Client applications request cryptographic functions from the TOE, typically using a key managed by the TOE[10], once the appropriate authorisation has been provided.

Two distinct use cases for the deployment of a cryptographic module conforming to this Protection Profile are described below. These are not necessarily the only use cases for which a cryptographic module certified against the PP will be suitable, but these are the ones that have been considered in developing this PP.

1.3.2.1 Use Case 1: Local signing

This use case is aimed at trust service providers applying its own electronic signatures or seals. Examples include TSPs issuing certificates and time-stamps, as well as TSPs supporting application services such as e-Invoicing and registered e-mail where the TSP applies its own seal / signature.

The TOE performs local cryptographic operations, and associated key management, which can be used by a client application to create qualified electronic signatures and qualified electronic seals for a natural or legal person representing a TSP. The same TSP is responsible for the security of the environment in which the TOE is used and managed (including the client application, which is outside the TOE). The signing / sealing request is passed from a signature / seal creation client application under control of the TSP and executing on an appliance in the same local operational environment as the TOE (i.e. all communications involved in creating, receiving and executing the signing / sealing request take place within the network environment controlled by the TSP, and do not involve uncontrolled networks). Apart from its support keys (e.g. to protect local secure channels to the signature creation application), the TOE generates, stores and uses only keys that belong to and represent the TSP (e.g. for signing other keys). In this use case the TOE by itself is intended to be used as a qualified electronic signature creation, or seal, device compliant to Annex II of Regulation EU 910/2014 [Regulation]. See Appendix B for further details.

1.3.2.2 Use Case 2: Support for Remote Server Signing

This use case is aimed at TSPs supporting requirements for remote signing, or sealing, as specified in Regulation 910/2014. In this case the TOE on its own is not intended to meet the requirements for

[10] All cryptographic operations in the scope of this Protection Profile are carried out using keys managed by the TOE, and therefore any use of other keys is outside the scope of the Protection Profile.

QSCDs in the context of remote signing set out in Annex II of (EU) No 910/2014. It is expected that the TOE would be used in conjunction with the protection profile to be defined in EN 419 241-2, and any other related protection profiles, to meet the requirements for Sole Control Assurance Level 2 as defined in EN 419 241-1. These security requirements may govern aspects such as the definition of specific user identification and authentication methods (e.g. multi-factor authentication) used within the signing system and may affect the type and form of the authorisation data that is passed to the cryptographic module in order to authorise use of a key.

The TOE performs local cryptographic operations, and associated key management, which can be used by an application using server signing, as defined in EN 419 241-1, to create qualified electronic signatures and qualified electronic seals on behalf of a legal or natural person which is distinct from and remote from the TSP which manages the TOE. The TOE generates, stores and uses signing / sealing keys in a way that maintains the remote control of an identified signatory or seal creator who operates through the use of a client application. The TOE deals with ensuring the security of keys and their use for signature or seal creation. Non-cryptographic functionality concerned with assuring sole control of these keys, for example authentication, is provided by other assured functionality outside the scope of the TOE.

1.3.3 Available non-TOE hardware/software/firmware

The TOE is a Cryptographic Module comprising its own hardware and software, though it may be supported by additional non-TOE hardware (e.g. a surrounding hardware appliance, physical authentication factors) and non-TOE software (e.g. utilities, management software or interface libraries).

2 Conformance Claim

2.1 CC Conformance Claim

This protection profile is conformant to Common Criteria version 3.1 revision 4.

More precisely, this protection profile is:

- CC Part 1 [CC1],
- CC Part 2 extended [CC2],
- CC Part 3 conformant [CC3].

The assurance requirement of this Protection Profile is **EAL4 augmented**.

Augmentation results from the selection of:

- AVA_VAN.5 Advanced methodical vulnerability analysis

2.2 PP Claim

This PP does not claim conformance to any another Protection Profile.

2.3 Conformance Rationale

Since this Protection Profile is not claiming conformance to any other protection profile, no rationale is necessary here.

2.4 Conformance Statement

This Protection Profile requires strict conformance of any Security Target or Protection Profile that claims conformance to this Protection Profile.

3 Security Problem Definition

3.1 Assets

The assets that need to be protected by the TOE are identified below.

R.SecretKey: secret keys used in symmetric cryptographic functions and private keys used in asymmetric cryptographic functions, managed and used by the TOE in support of the cryptographic services that it offers. This includes user keys, owned and used by specific users, and support keys used in the implementation and operation of the TOE. The asset also includes copies of such keys made for external storage and/or backup purposes. The confidentiality and integrity of these keys must be protected.

R.PubKey: public keys managed and used by the TOE in support of the cryptographic services that it offers (including user keys and support keys). This asset includes copies of keys made for external storage and/or backup purposes. The integrity of these keys must be protected.

R.ClientData: data supplied by a client for use in a cryptographic function. Depending on the context, this data may require confidentiality and/or integrity protection.

R.RAD: reference data held by the TOE that is used to authenticate an administrator (hence to control access to privileged administrator functions such as TOE backup, export of audit data) or to authorise a user for access to secret and private keys (R.SecretKey). This asset includes copies of authentication/authorisation data made for external storage and/or backup purposes. The integrity of the RAD must be protected; its confidentiality must also be protected unless the authentication method used means that the RAD is public data (such as a public key).

3.2 Subjects

The types of subjects identified in this PP are:

S.Application: a client application, or process acting on behalf of a client application and that communicates with the TOE over a local or external interface. Client applications will in some situations be acting directly on behalf of end users (see S.User).

S.User: an end user of the TOE who can be associated with secret keys and authentication/authorisation data held by the TOE. An end user communicates with the TOE by using a client application (S.Application).

S.Admin: an administrator of the TOE. Administrators are responsible for performing the TOE initialisation, TOE configuration and other TOE administrative functions.

Each type of subject may include many individual members, for example a single TOE will generally have many users who are all included as members of the type S.User.

3.3 Threats

The following threats are defined for the TOE. The attacker (i.e. the ‘threat agent’) described in each of the threats is a subject who is not authorised for the relevant action, but who may present themselves as either a completely unknown user, or as one of the subjects in section 3.2 (but in this case the attacker will not have access to the authentication or authorisation data for the subject).

T.KeyDisclose **Unauthorised disclosure of secret/private key**

An attacker obtains unauthorised access to the plaintext form of a secret key (R.SecretKey), enabling either direct reading of the key or other copying into a form that can be used by the attacker as though the key were their own. This access may be gained during generation, storage, import/export, use of the key, or backup if supported by the TOE.

T.KeyDerive **Derivation of secret/private key**

An attacker derives a secret key (R.SecretKey) from publicly known data, such as the corresponding public key or results of cryptographic functions using the key or any other data that is generally available outside the TOE.

T.KeyMod **Unauthorised modification of a key**

An attacker makes an unauthorised modification to a secret or public key (R.SecretKey or R.PubKey) while it is stored in, or under the control of, the TOE, including export and backups if supported. This includes replacement of a key as well as making changes to the value of a key, or changing its attributes such as required authorisation, usage constraints or identifier (changing the identifier to the identifier used for another key would allow unauthorised substitution of the original key with a key known to the attacker). The threat therefore includes the case where an attacker is able to break the binding between a key and its critical attributes[11].

T.KeyMisuse **Misuse of a key**

An attacker uses the TOE to make unauthorised use of a secret key (R.SecretKey) that is managed by the TOE (including the unauthorised use of a secret key for a cryptographic function that is not permitted for that key[12]), without necessarily obtaining access to the value of the key.

T.KeyOveruse **Overuse of a key**

An attacker uses a key (R.SecretKey) that has been authorised for a specific use (e.g. to make a single signature) in other cryptographic functions that have not been authorised.

T.DataDisclose **Disclosure of sensitive client application data**

An attacker gains access to data that requires protection of confidentiality (R.ClientData, and possibly R.RAD) supplied by a client application during transmission to or from the TOE or during transmission between physically separate parts of the TOE.

T.DataMod **Unauthorised modification of client application data**

An attacker modifies data (R.ClientData such as DTBS/R, authentication/authorisation data, or a public key (R.PubKey)) supplied by a client application during transmission to the TOE or during transmission between physically separate parts of the TOE, so that the result returned by the TOE (such as a signature or public key certificate) does not match the data intended by the originator of the request.

T.Malfunction **Malfunction of TOE hardware or software**

The TOE may develop a fault that causes some other security property to be weakened or to fail. This may affect any of the assets and could result in any of the other threats being realised. Particular causes of faults to be considered are:

- Environmental conditions (including temperature and power)
- Failures of critical TOE hardware components (including the RNG)
- Corruption of TOE software.

[11] See OT.KeyIntegrity in section 4.1 for further discussion of critical attributes of a key.

[12] This therefore means that the threat includes unauthorised use of a cryptographic function that makes use of a key.

3.4 Organisational Security Policies

P.Algorithms Use of approved cryptographic algorithms

The TOE offers key generation functions and other cryptographic functions provided for users that are endorsed by recognised authorities as appropriate for use by TSPs.

Application Note 1

The relevant authorities and endorsements are determined by the context of the client applications that use the TOE. For digital signatures within the European Union this is as indicated in [Regulation] and an exemplary list of algorithms and parameters is given in [TS 119 312] or [SOG-IS-Crypto] (see also section 1.3.1.3).

P.KeyControl Support for control of keys

The life cycle of the TOE and any secret keys that it manages (where such keys are associated with specific entities, such as the signature creation data associated with a signatory or the seal creation data associated with a seal creator[13]), shall be implemented in such a way that the secret keys can be reliably protected by the legitimate owner against use by others, and in such a way that the use of the secret keys by the TOE can be confined to a set of authorised cryptographic functions.

Application Note 2

This policy is intended to ensure that the TOE can be used for qualified electronic seals and qualified electronic signatures as in [Regulation], but recognises that not all keys are used for such purposes. Therefore, although the TOE must be able to support the necessary strong controls over keys in order to create such seals and signatures, not all keys need the same level and type of control.

P.RNG Random Number Generation

The TOE is required to generate random numbers that meet a specified quality metric, for use by client applications. These random numbers shall be suitable for use as keys, authentication/authorisation data, or seed data for another random number generator that is used for these purposes.

P.Audit Audit trail generation

The TOE is required to generate an audit trail of security-relevant events, recording the event details and the subject associated with the event.

Application Note 3

The cryptographic module TOE is assumed to be part of a larger system that manages audit data. The TOE therefore logs audit records, and it is assumed that these are collected, maintained and reviewed in the larger system. Hence there is no separate auditor role within the cryptographic module TOE, but the role of System Auditor is assumed to exist in the larger system – cf. A.AuditSupport in section 3.5.

3.5 Assumptions

A.ExternalData Protection of data outside TOE control

Where copies of data protected by the TOE are managed outside of the TOE, client applications and other entities must provide appropriate protection for that data to a level required by the application context and the risks in the deployment environment.

[13] A seal creator may be a *legal person* (see [Regulation]) rather than a *natural person*, and seal creation data may therefore be authorised for use by a number of natural persons, depending on the nature and requirements of the trust service provided.

In particular, any backups of the TOE and its data are maintained in a way that ensures appropriate controls over making backups, storing backup data, and using backup data to restore an operational TOE. The number of sets of backup data does not exceed the minimum needed to ensure continuity of the TSP service. The ability to restore a TOE to an operational state from backup data requires at least dual person control (i.e. the participation and approval of more than one authenticated administrator).

A.Env Protected operating environment

The TOE operates in a protected environment that limits physical access to the TOE to authorised Administrators. The TOE software and hardware environment (including client applications) is installed maintained by Administrators in a secure state that mitigates against the specific risks applicable to the deployment environment.

A.DataContext Appropriate use of TOE functions

Any client application using the cryptographic functions of the TOE will ensure that the correct data are supplied in a secure manner (including any relevant requirements for authenticity, integrity and confidentiality). For example, when creating a digital signature over a DTBS the client application will ensure that the correct (authentic, unmodified) DTBS/R is supplied to the TOE, and will correctly and securely manage the signature received from the TOE; and when certifying a public key the client application will ensure that necessary checks are made to prove possession of the corresponding private key. The client application may make use of appropriate secure channels provided by the TOE to support these security requirements. Where required by the risks in the operational environment a suitable entity (possibly the client application) performs a check of the signature returned from the TOE, to confirm that it relates to the correct DTBS.

Client applications are also responsible for any required logging of the uses made of the TOE services, such as signing (or sealing) events.

Similar requirements apply in local use cases where no client application need be involved, but in which the TOE and its user data (such as keys used for signatures) need to be configured in ways that will support the need for security requirements such as sole control of signing keys.

Appropriate procedures are defined for the initial creation of data and continuing operation of the TOE according to the specific risks applicable to the deployment environment and the ways in which the TOE is used.

A.UAuth Authentication of application users

Any client application using the cryptographic services of the TOE will correctly and securely gather identification and authentication/authorisation data from its users and securely transfer it to the TOE (protecting the confidentiality of the authentication/authorisation data as required) when required to authorise the use of TOE assets and services.

A.AuditSupport Audit data review

The audit trail generated by the TOE will be collected, maintained and reviewed by a System Auditor according to a defined audit procedure for the TSP.

Application Note 4

As noted for P.Audit in section 3.4, the TOE is assumed to exist as part of a larger system and the System Auditor is a role within this larger system.

A.AppSupport Application security support

Procedures to ensure the ongoing security of client applications and their data will be defined and followed in the environment, and reflected in use of the appropriate TOE cryptographic functions and parameters, and appropriate management and administration actions on the TOE. This includes, for example, any relevant policies on algorithms, key generation methods, key lengths, key access, key import/export, key usage limitations, key activation, cryptoperiods and key renewal, and key/certificate revocation.

4 Security Objectives

This section identifies and defines the security objectives for the TOE and its operational environment.

Security objectives reflect the stated intent and counter the identified threats, as well as comply with the identified organisational security policies and assumptions.

4.1 Security Objectives for the TOE

The following security objectives describe security functions to be provided by the TOE.

OT.PlainKeyConf Protection of confidentiality of plaintext secret keys

The plaintext value of secret keys is not made available outside the TOE (except where the key has been exported securely in the manner of OT.ImportExport). This includes protection of the keys during generation, storage (including external storage), and use in cryptographic functions, and means that even authorised users of the keys and administrators of the TOE cannot directly access the plaintext value of a secret key.

OT.Algorithms Use of approved cryptographic algorithms

The TOE offers key generation functions and other cryptographic functions provided for users that are endorsed by recognised authorities as appropriate for use by TSPs. This ensures that the algorithms used do not enable publicly known data to be used to derive secret keys.

Application Note 5

See note under P.Algorithms (section 3.4) on relevant references for digital signatures within the European Union.

OT.KeyIntegrity Protection of integrity of keys

The value and critical attributes of keys (secret or public) have their integrity protected by the TOE against unauthorised modification (unauthorised modifications include making unauthorised copies of a key such that the attributes of the copy can be changed without the same authorisation as for the original key). Critical attributes in this context are defined to be those implementation-level attributes of a key that could be used by an attacker to cause the equivalent of a modification to the key value by other means (e.g. including changing the cryptographic functions for which a key can be used, the users with access to the key, or the identifier of the key). This objective includes protection of the keys during generation, storage (including external storage), and use.

OT.Auth Authorisation for use of TOE functions and data

The TOE carries out an authentication/authorisation check on all subjects before allowing them to use the TOE. The following types of entity are distinguished for the purposes of authorisation (i.e. each type has a distinct method of authorisation):

- administrators of the TOE
- users of TOE cryptographic functions (client applications using secure channels)
- users of secret keys.

In particular, the TOE always requires authorisation before using a secret key.

Application Note 6

Local client applications within a suitable security environment (such as client applications that are connected to the TOE by a channel such as a PCIe bus within the same hardware appliance) do not require authentication to communicate with the TOE, as noted in section 1.3.1. However, use of a secret key always requires prior authorisation.

OT.KeyUseConstraint Constraints on use of keys

Any key (secret or public) has an unambiguous definition of the purposes for which it can be used, in terms of the cryptographic functions or operations (e.g. encryption or signature) that it is permitted to be used for. The TOE rejects any attempt to use the key for a purpose that is not permitted. The TOE also has an unambiguous definition of the subjects that are permitted to access the key (and the purposes for which this access can be used) and allows this to be set to the granularity of an individual subject – these access constraints apply to *use* of the key even where the key value is not accessible. This objective means that the TOE also prevents unauthorised use of any cryptographic functions that use a key.

OT.KeyUseScope Defined scope for use of a key after authorisation

The TOE is required to define and apply clearly stated limits on when authorisation and re-authorisation are required in order for a secret key to be used[14]. For example the TOE may allow secret keys to be used for a specified time period or number of uses after initial authorisation, or for may allow the key to be used until authorisation is explicitly rescinded. As another example, the TOE may implement a policy that requires re-authorisation before every use of a secret key.

Application Note 7

Such limits on the use of a key after initial authorisation are termed "re-authorisation conditions" in this PP. A wide range of policies and re-authorisation conditions are allowed, and different policies may be applied to different types of secret key, but the re-authorisation conditions for all types of secret key must be unambiguously defined in the Security Target. The decision to use supported re-authentication conditions is made on the basis of the application context. Making appropriate use of re-authorisation conditions supports client applications in meeting their requirements for OE.DataContext and OE.AppSupport.

OT.DataConf Protection of confidentiality of sensitive client application data

The TOE provides secure channels to client applications that can be used to protect the confidentiality of sensitive data (such as authentication/authorisation data) during transmission between the client application and the TOE, or during transmission between separate parts of the TOE where that transmission passes through an insecure environment.

Application Note 8

Protection of secret keys (as a specific type of sensitive data) is also subject to additional protection specified in other TOE objectives. Any requirements for secure storage and control of access to other types of client application data within the TOE rely on the client application using appropriate interfaces and cryptographic functions to protect it, as required by OE.DataContext and OE.AppSupport. For example, if a client application uses the TOE to perform cryptographic functions on data that represent a passphrase value and the passphrase value is to be stored on the TOE, then the client application would need to use an appropriate encryption function before storing the data on the TOE.

OT.DataMod Protection of integrity of client application data

The TOE provides secure channels to client applications that can be used to protect the integrity of sensitive data (such as data to be signed, authentication/authorisation data or public key certificates) during transmission between the client application and the TOE.

[14] Any attempt to use the key in cryptographic functions that are not permitted for that key is addressed by OT.KeyUseConstraint.

Application Note 9

Any requirements for integrity protection of client application data within the TOE rely on the client application using appropriate interfaces and cryptographic functions to protect it, as required by OE.DataContext and OE.AppSupport.

OT.ImportExport Secure import and export of keys

The TOE allows import and export of secret keys only by using a secure method that protects the confidentiality and integrity of the data during transmission – in particular, secret keys must be exported only in encrypted form (it is not sufficient to rely on properties of a secure channel to provide the protection: the key itself must be encrypted). The TOE also allows individual secret keys under its control to be identified as non-exportable, in which case any attempt to export them will be rejected automatically. Public keys may be imported and exported in a manner that protects the integrity of the data during transmission.

Assigned keys cannot be imported or exported.

OT.Backup Secure backup of user data

Any method provided by the TOE for backing up user data, including secret keys, preserves the security of the data and is controlled by authorised Administrators. The secure backup process preserves the confidentiality and integrity of the data during creation, transmission, storage and restoration of the backup data. Backups also preserve the integrity of the attributes of keys.

OT.RNG Random number quality

Random numbers generated and provided to client applications for use as keys, authentication/authorisation data, or seed data for another random number generator that is used for these purposes shall meet a defined quality metric in order to ensure that random numbers are not predictable and have sufficient entropy.

OT.TamperDetect Tamper Detection

The TOE shall provide features to protect its security functions against tampering. In particular the TOE shall make any physical manipulation within the scope of the intended environment (adhering to OE.Env) detectable for the administrators of the TOE.

OT.FailureDetect Detection of TOE hardware or software failures

The TOE detects faults that would cause some other security property to be weakened or to fail, including:

- Environmental conditions outside normal operating range (including temperature and power)
- Failures of critical TOE hardware components (including the RNG)
- Corruption of TOE software.

On detection of a fault, the TOE takes action to maintain its security and the security of the data that it contains and controls.

OT.Audit Generation of audit trail

The TOE creates audit records for security-relevant events, recording the event details and the subject associated with the event. The TOE ensures that the audit records are protected against accidental or malicious deletion or modification of records by providing tamper protection (either prevention or detection) for the audit log.

4.2 Security Objectives for the Operational Environment

The following security objectives relate to the TOE environment. This includes client applications as well as the procedure for the secure operation of the TOE.

OE.ExternalData Protection of data outside TOE control

Where copies of data protected by the TOE are managed outside of the TOE, client applications and other entities shall provide appropriate protection for that data to a level required by the application context and the risks in the deployment environment. This includes protection of data that is exported from, or imported to, the TOE (such as audit data and encrypted keys).

In particular, any backups of the TOE and its data shall be maintained in a way that ensures appropriate controls over making backups, storing backup data, and using backup data to restore an operational TOE. The number of sets of backup data shall not exceed the minimum needed to ensure continuity of the TSP service. The ability to restore a TOE to an operational state from backup data shall require at least dual person control (i.e. the participation and approval of more than one authenticated administrator).

OE.Env Protected operating environment

The TOE shall operate in a protected environment that limits physical access to the TOE to authorised Administrators. The TOE software and hardware environment (including client applications) shall be installed and maintained by Administrators in a secure state that mitigates against the specific risks applicable to the deployment environment, including (where applicable):

- Protection against loss or theft of the TOE or any of its externally stored assets
- Inspections to deter and detect tampering (including attempts to access side-channels, or to access connections between physically separate parts of the TOE, or parts of the hardware appliance)
- Protection against the possibility of attacks based on emanations from the TOE (e.g. electromagnetic emanations) according to risks assessed for the operating environment
- Protection against unauthorised software and configuration changes on the TOE and the hardware appliance
- Protection to an equivalent level of all instances of the TOE holding the same assets (e.g. where a key is present as a backup in more than one instance of the TOE).

OE.DataContext Appropriate use of TOE functions

Any client application using the cryptographic functions of the TOE shall ensure that the correct data are supplied in a secure manner (including any relevant requirements for authenticity, integrity and confidentiality). For example, when creating a digital signature over a DTBS the client application shall ensure that the correct (authentic, unmodified) DTBS/R is supplied to the TOE, and shall correctly and securely manage the signature received from the TOE; and when certifying a public key the client application shall ensure that necessary checks are made to prove possession of the corresponding private key. The client application may make use of appropriate secure channels provided by the TOE to support these security requirements. Where required by the risks in the operational environment a suitable entity (possibly the client application) shall perform a check of the signature returned from the TOE, to confirm that it relates to the correct DTBS.

Client applications shall be responsible for any required logging of the uses made of the TOE services, such as signing (or sealing) events.

Similar requirements shall apply in local use cases where no client application need be involved, but in which the TOE and its user data (such as keys used for signatures) need to be configured in ways that will support the need for security requirements such as sole control of signing keys.

Appropriate procedures shall be defined for the initial creation of data and continuing operation of the TOE according to the specific risks applicable to the deployment environment and the ways in which the TOE is used.

OE.Uauth Authentication of application users

Any client application using the cryptographic services of the TOE shall correctly and securely gather identification and authentication/authorisation data from its users and securely transfer it to the TOE (protecting the confidentiality of the authentication/authorisation data as required) when required to authorise the use of TOE assets and services.

OE.AuditSupport Audit data review

The audit trail generated by the TOE will be collected, maintained and reviewed by a System Auditor according to a defined audit procedure for the TSP.

Application Note 10

As noted for P.Audit in section 3.4, the TOE is assumed to exist as part of a larger system and the System Auditor is a role within this larger system.

OE.AppSupport Application security support

Procedures to ensure the ongoing security of client applications and their data shall be defined and followed in the environment, and reflected in use of the appropriate TOE cryptographic functions and parameters, and appropriate management and administration actions on the TOE. This includes, for example, any relevant policies on algorithms, key generation methods, key lengths, key access, key import/export, key usage limitations, key activation, cryptoperiods and key renewal, and key/certificate revocation.

5 Extended Components Definitions

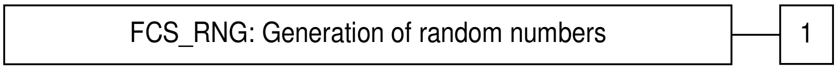
5.1 Generation of random numbers (FCS_RNG)

This family describes the functional requirements for random number generation used for cryptographic purposes.

Family behaviour

This family defines quality requirements for the generation of random numbers which are intended to be use for cryptographic purposes.

Component levelling:



Management: FCS_RNG.1

There are no management activities foreseen.

Audit: FCS_RNG.1

There are no actions defined to be auditable.

FCS_RNG.1 *Generation of random numbers*

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1 The TSF shall provide a [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*] random number generator that implements: [assignment: *list of security capabilities*].

FCS_RNG.1.2 The TSF shall provide [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*] that meet [assignment: *a defined quality metric*].

Application Note 11

A physical random number generator (RNG) produces the random number by a noise source based on physical random processes. A non-physical true RNG uses a noise source based on non-physical random processes like human interaction (key strokes, mouse movement). A deterministic RNG uses an random seed to produce a pseudorandom output. A hybrid RNG combines the principles of physical and deterministic RNGs where a hybrid physical RNG produces at least the amount of entropy the RNG output may contain and the internal state of a hybrid deterministic RNG output contains fresh entropy but less than the output of RNG may contain.

5.2 Basic TSF Self Testing (FPT_TST_EXT.1)

The extended component defined here is a simplified version of FPT_TST.1 in [CC2]

Family behaviour

Components in this family address the requirements for self-testing the TSF for selected correct operation.

Component levelling:

Management: FPT_TST_EXT.1

There are no management activities foreseen.

Audit: FPT_TST_EXT.1

The following actions should be auditable if FAU_GEN Security audit data generation is included in the PP/ST:

- Indication that TSF self test was completed.

FPT_TST_EXT.1 *Basic TSF Self Testing*

Hierarchical to:	No other components.
Dependencies:	No dependencies.

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [selection: *during initial start-up (on power on), periodically during normal operation, at the request of the authorised user, at the conditions [assignment: conditions under which self-tests should occur]*] to demonstrate the correct operation of the TSF: [assignment: *list of self-tests run by the TSF*].

6 Security Requirements

This chapter gives the security functional requirements (SFR) and the security assurance requirements (SAR) for the TOE and the environment.

Security functional requirements components given in section 6.3 "*TOE security functional requirements*" are drawn from Common Criteria part 2 [CC2]. Some security functional requirements represent extensions to [CC2], with a reasoning given in section 6.5. Operations for assignment, selection and refinement have been made. Operations not performed in this PP are identified in order to enable instantiation of the PP to a Security Target (ST).

The TOE security assurance requirements statements given in section 6.4 "*TOE Security Assurance Requirement*" are drawn from the security assurance components from Common Criteria part 3 [CC3].

6.1 Typographical Conventions

The following conventions are used in the definitions of the SFRs:

- Refinements are denoted in one of two ways, depending on whether they add detail to an SFR ('explanatory refinements') or update the text of an SFR element ('element refinements'). Explanatory refinements follow the SFR that they update and are marked by the word "Refinement" in bold followed by text describing the refinement. Element refinements are indicated by bold text within an SFR element, with the original text indicated in a footnote.

- Selections and assignments made in this PP are italicised, and the original text is indicated in a footnote. Selections and assignments that are left to be filled in by the Security Target author appear in square brackets with an indication that a selection or assignment is to be made, [selection:] or [assignment:], and the description of selection options or assignment description are *italicized*.

6.2 SFR Architecture

6.2.1 SFR Relationships

Figure 2 and Figure 3 give a graphical presentation of the connections between the Security Functional Requirements (SFRs) from section 6.3 below and the underlying functional areas and operations that the TOE provides. The diagrams provide a context for SFRs that relates to their use in the TOE, whereas section 6.3 defines the SFRs grouped by the abstract class and family groupings in [CC2].

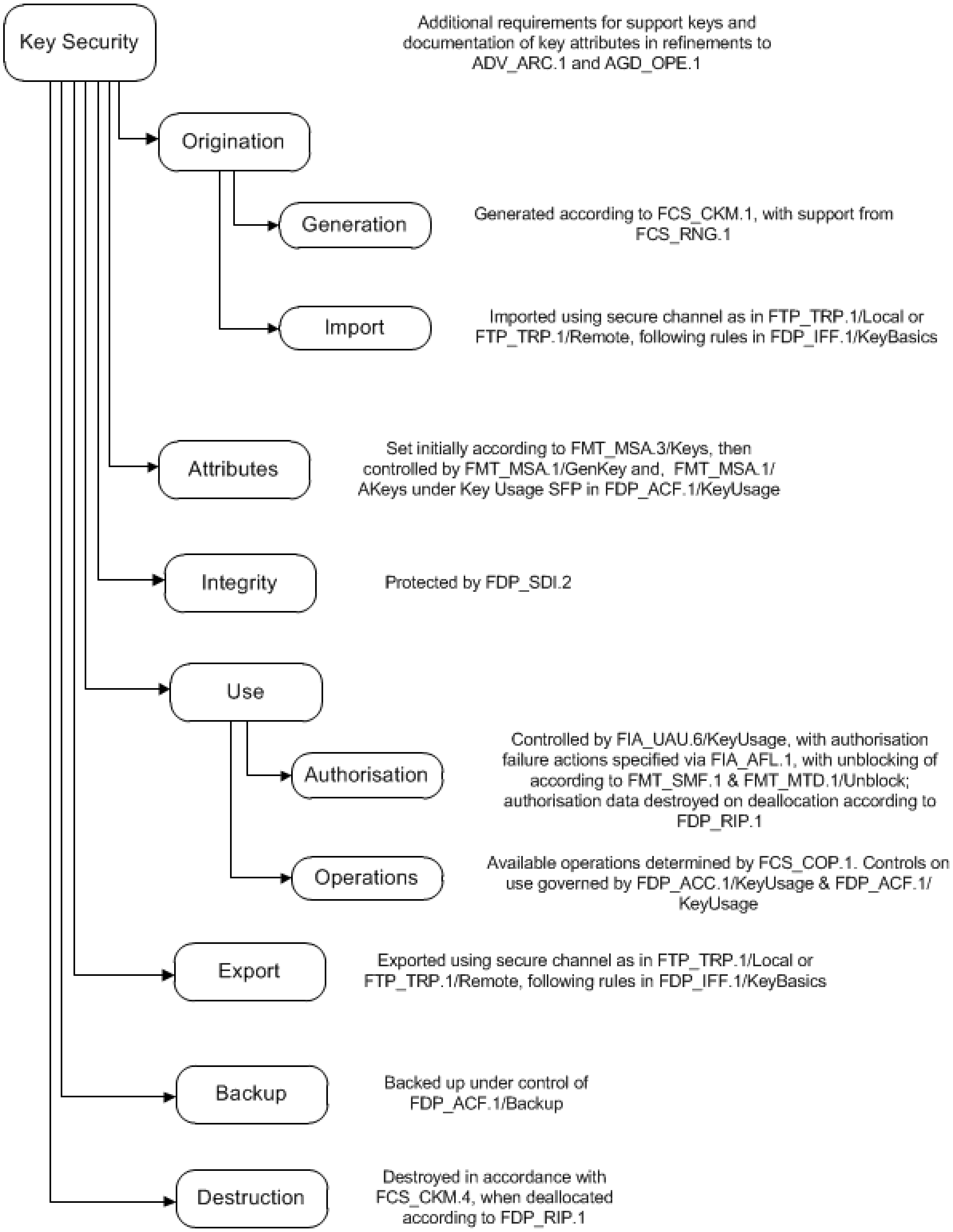

Key Security
Additional requirements for support keys and documentation of key attributes in refinements to ADV_ARC.1 and AGD_OPE.1
Origination
Generation
Generated according to FCS_CKM.1, with support from FCS_RNG.1
Import
Imported using secure channel as in FTP_TRP.1/Local or FTP_TRP.1/Remote, following rules in FDP_IFF.1/KeyBasics
Attributes
Set initially according to FMT_MSA.3/Keys, then controlled by FMT_MSA.1/GenKey and, FMT_MSA.1/ AKeys under Key Usage SFP in FDP_ACF.1/KeyUsage
Integrity
Protected by FDP_SDI.2
Use
Authorisation
Controlled by FIA_UAU.6/KeyUsage, with authorisation failure actions specified via FIA_AFL.1, with unblocking of according to FMT_SMF.1 & FMT_MTD.1/Unblock; authorisation data destroyed on deallocation according to FDP_RIP.1
Operations
Available operations determined by FCS_COP.1. Controls on use governed by FDP_ACC.1/KeyUsage & FDP_ACF.1/ KeyUsage
Export
Exported using secure channel as in FTP_TRP.1/Local or FTP_TRP.1/Remote, following rules in FDP_IFF.1/KeyBasics
Backup
Backed up under control of FDP_ACF.1/Backup
Destruction
Destroyed in accordance with FCS_CKM.4, when deallocated according to FDP_RIP.1

Figure 2: Architecture of Key Protection SFRs

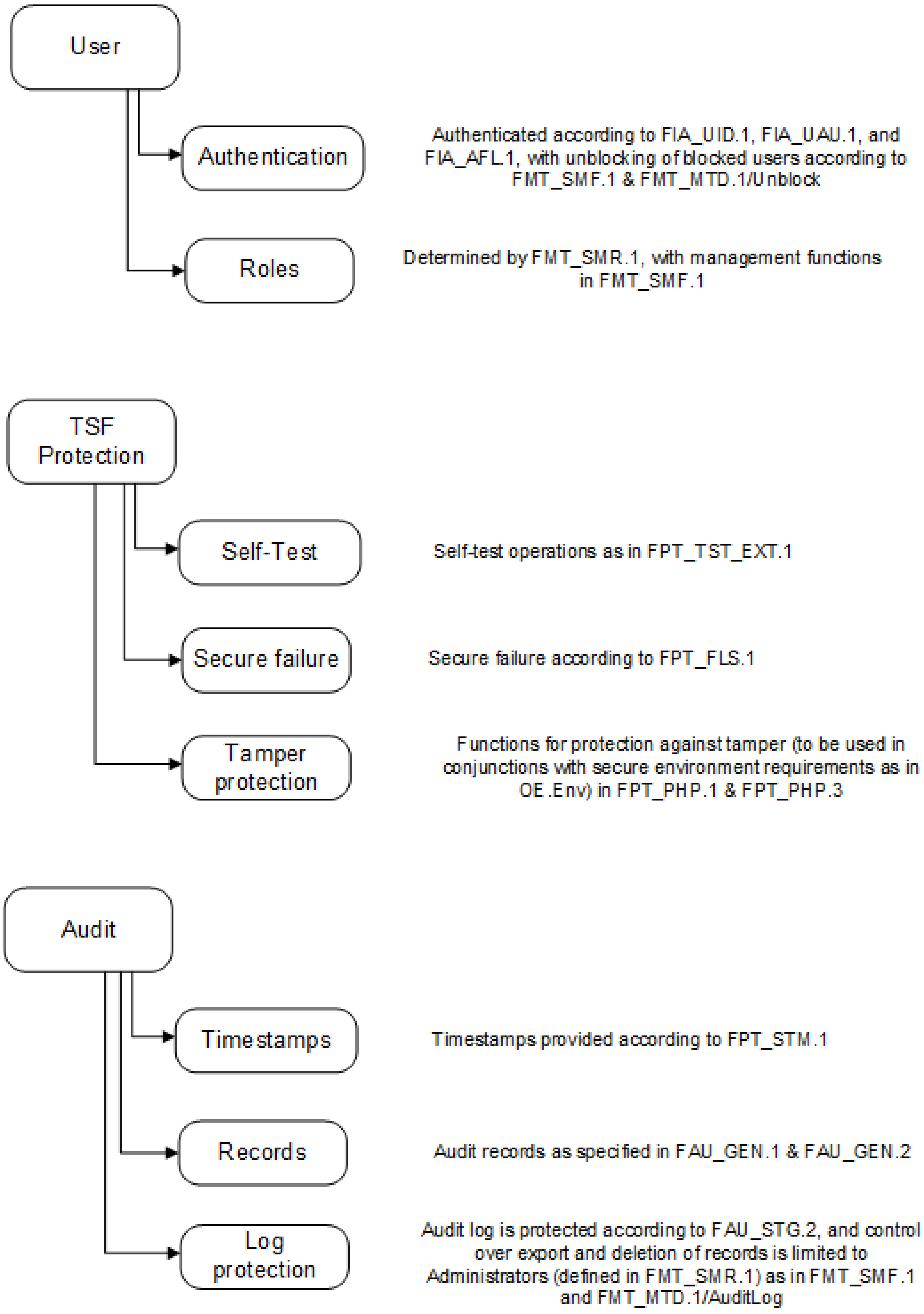

User
Authentication
Authenticated according to FIA_UID.1, FIA_UAU.1, and FIA_AFL.1, with unblocking of blocked users according to FMT_SMF.1 & FMT_MTD.1/Unblock
Roles
Determined by FMT_SMR.1, with management functions in FMT_SMF.1
TSF Protection
Self-Test
Self-test operations as in FPT_TST_EXT.1
Secure failure
Secure failure according to FPT_FLS.1
Tamper protection
Functions for protection against tamper (to be used in conjunctions with secure environment requirements as in OE.Env) in FPT_PHP.1 & FPT_PHP.3
Audit
Timestamps
Timestamps provided according to FPT_STM.1
Records
Audit records as specified in FAU_GEN.1 & FAU_GEN.2
Log protection
Audit log is protected according to FAU_STG.2, and control over export and deletion of records is limited to Administrators (defined in FMT_SMR.1) as in FMT_SMF.1 and FMT_MTD.1/AuditLog

Figure 3: Architecture of User, TSF Protection & Audit SFRs

6.2.2 SFRs and the Key Lifecycle

The generic lifecycle for a key is illustrated in Figure 4. This shows the methods by which a key may arrive in the TOE (import, generation or restore from backup), resulting in binding of a set of attributes to the key and storage of the key, and finally the ways in which a stored key may then be processed (export, use in a cryptographic function, backup, or destruction). The SFRs related to each of these aspects are then described below Figure 4.

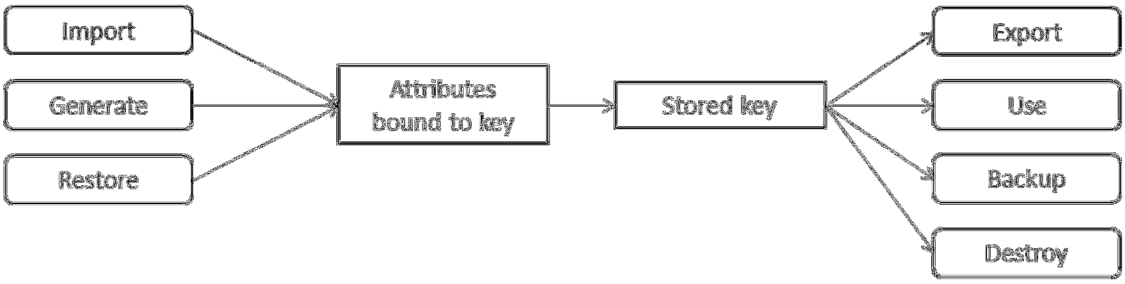


Figure 4: Generic Key Lifecycle and Related SFRs

Import:

- FDP_IFF.1/KeyBasics requires a secure channel (FTP_TRP.1) and import in encrypted form or by using at least two components
- FAU_GEN.1 requires audit of import

Generate:

- FCS_CKM.1 requires approved algorithms
- FCS_RNG.1 defines requirements on random number generation
- FMT_MSA.3/Keys defines requirements on key attribute initialisation
- FAU_GEN.1 requires audit of generation (and of failure of RNG)

Restore:

- FDP_ACF.1/Backup requires only an Administrator can restore from a backup, all backups must preserve confidentiality and integrity of keys (as appropriate to key type) and their attributes, and any restore must be under dual person control
- FAU_GEN.1 requires auditing of a restore (or of any integrity failure during a restore attempt)

Attributes bound to key:

- FMT_MSA.3/Keys defines requirements on key attribute initialisation
- FDP_ACF.1/KeyUsage, FMT_MSA.1/GenKeys and FMT_MSA.1/AKeys define requirements on key attribute modification
- FAU_GEN.1 requires audit of changes to key attributes

Stored key:

- FDP_IFF.1/KeyBasics requires no plaintext access
- FDP_SDI.2 requires protection of the integrity of keys and their attributes
- FAU_GEN.1 requires audit of integrity errors detected

Export:

- FDP_IFF.1/KeyBasics requires a secure channel (FTP_TRP.1), authorisation before export, no export of Assigned Keys, export controlled by the export flag attribute, and export in encrypted form
- FAU_GEN.1 requires audit of export

Use:

- FIA_AFL.1 requires blocking of access to a key on reaching an authorisation failure threshold (FDP_IFF.1/KeyBasics and FMT_MTD.1/Unblock define requirements on unblocking)
- FDP_ACF.1/KeyUsage requires authorisation before use of a key and that the key can only be used as identified in its Key Usage attribute
- FIA_UAU.6/KeyAuth requires authorisation before initial use of a key and describes any additional requirements for re-authorisation conditions such as expiry of a time period or number of uses of a key (or when the authorisation period has been explicitly ended)
- FDP_RIP.1 requires protection of authorisation data on deallocation
- FDP_IFF.1/KeyBasics requires no access to intermediate values in any operation using a secret key

- FCS_COP.1 requires the use of approved algorithms
- FAU_GEN.1 requires audit of authorisation failure (and blocking or unblocking)

Backup:

- FDP_ACF.1/Backup requires only Administrator can make a backup; all backups must preserve confidentiality and integrity of keys (as appropriate to key type) and their attributes
- FAU_GEN.1 requires auditing of a backup

Destroy:

- FDP_RIP.1 requires key to be protected on deallocation
- FCS_CKM.4 requires key zeroisation on deallocation
- FAU_GEN.1 requires audit of key destruction

6.3 Security Functional Requirements

The individual security functional requirements are specified in the sections below.

6.3.1 Cryptographic Support (FCS)

FCS_CKM.1 *Cryptographic key generation*

Hierarchical to: No other components.

Dependencies: [FCS_CKM.2 Cryptographic key distribution or FCS_COP.1 Cryptographic operation]

FCS_CKM.4 Cryptographic key destruction

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *[*assignment: *cryptographic key generation algorithm]* and specified cryptographic key sizes [assignment*: cryptographic key sizes*] that meet the following: [assignment*: list of standards*].

Application Note 12

The Security Target must include all key generation operations that are intended to support TSP operations using one or more iterations of FCS_CKM.1.

The relevant authorities and endorsements for completion of the SFRs are determined by the context of the client applications that use the TOE. For digital signatures within the European Union this is as indicated in [Regulation] and an exemplary list of algorithms and parameters is given in [TS 119 312] or [SOG-IS-Crypto] (see also section 1.3.1.3).

Note that key generation needs to be linked to the setting of security attributes of a key (including the link to a subject who owns the key, via the setting of authorisation data) as in FMT_MSA.1/GenKeys and FMT_MSA.1/AKeys,

FCS_CKM.4 *Cryptographic key destruction*

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *zeroisation*[15] that meets the following: [assignment*: list of standards*].

Application Note 13

The Security Target must specify the method(s) of secure destruction of all secret keys and all support keys[16]*, and must ensure that all are covered by a secure destruction method. If necessary then more than one iteration of FCS_CKM.4 may be included to describe different standards for secure deletion. The 'list of standards' in the final assignment may be met in the Security Target by simply providing a description of the action taken to zeroise the keys rather than referencing an external standard.*

FCS_COP.1 *Cryptographic operation*

Hierarchical to: No other components.

Dependencies: [FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]

FCS_CKM.4 Cryptographic key destruction

FCS_COP.1.1 The TSF shall perform [assignment: *list of cryptographic operations*] in accordance with a specified cryptographic algorithm [assignment*: cryptographic algorithm*] and cryptographic key sizes [assignment*: cryptographic key sizes*] that meet the following: [assignment: *list of standards*].

Application Note 14

The Security Target must include all cryptographic functions that are intended to support TSP operations using one or more iterations of FCS_COP.1. This includes cryptographic operations for digital signatures and seals, implementing trusted paths (FTP_TRP.1) and secure channels (FTP_TRP.1), key encryption (e.g. FDP_IFF.1/KeyBasics), and any backups (FDP_ACF.1/Backup) that the TOE creates. If the TOE supports software or firmware updates then the iterations must include the cryptographic operations used to support the validation of digital signatures on the updates as described in the refinement to ADV_ARC.1 in section 6.4.1.

The relevant authorities and endorsements for completion of each of these iterations are determined by the context of the client applications that use the TOE. For digital signatures and seals within the European Union this is as indicated in [Regulation] and an exemplary list of algorithms and parameters is given in [TS 119 312] or [SOG-IS-Crypto] (see also section 1.3.1.3).

FCS_RNG.1 *Generation of random numbers*

Hierarchical to: No other components.

Dependencies: No dependencies.

FCS_RNG.1.1 The TSF shall provide a [selection: *physical, non-physical true, deterministic, hybrid physical, hybrid deterministic*] random number generator that implements: [assignment: *list of security capabilities*].

FCS_RNG.1.2 The TSF shall provide [selection: *bits, octets of bits, numbers [assignment: format of the numbers]*] that meet [assignment: *a defined quality metric*].

[15] [assignment*: cryptographic key destruction method*]

[16] See the description of 'support keys' in the refinement of ADV_ARC.1 in section 6.4.

Application Note 15

For more information on the selections and assignments see the SFR definition in section 5.1.

The Security Target describes the uses made of the RNG and its relationship to other SFRs such as FCS_CKM.1, and to any random number generation function/service made available to users or clients applications.

6.3.2 Identification and authentication (FIA)

FIA_UID.1 *Timing of identification*

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UID.1.1 The TSF shall allow

(1) *Self test according to FPT_TST_EXT.1*

(2) [assignment: *list of additional TSF-mediated actions*][17]

on behalf of the user to be performed before the user is identified.

FIA_UID.1.2 The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.

Application Note 16

The 'list of additional *TSF-mediated actions' may be left empty (equivalent to an assignment of 'None') if applicable.*

FIA_UAU.1 *Timing of authentication*

Hierarchical to: No other components.

Dependencies: FIA_UID.1 Timing of identification.

FIA_UAU.1.1 The TSF shall allow

(1) Self-test according to FPT_TST_EXT.1,

(2) *Identification of the user by means of TSF required by FIA_UID.1*

(3) [assignment: *list of additional TSF-mediated actions*][18]

on behalf of the user to be performed before the user is authenticated.

FIA_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Application Note 17

The Security Target must separately identify any different types of identification and authentication, e.g. for Administrators, local users, application users, using separate iterations of the FIA_UID.1 and FIA_UAU.1 SFRs where the methods differ. The Security Target must also separately identify the difference between authentication of users and authorisation for use of keys as required for FIA_UAU.6/KeyAuth. Separate iterations of FIA SFRs may be necessary to capture these separate cases.

[17] [assignment: *list of TSF-mediated actions*]

[18] [assignment: *list of TSF-mediated actions*]

The ‘list of additional TSF-mediated actions’ in FIA_UAU.1.1 may be left empty (equivalent to an assignment of ‘None’) if applicable.

FIA_AFL.1 *Authentication failure handling*

Hierarchical to: No other components.

Dependencies: FIA_UAU.1 Timing of authentication

FIA_AFL.1.1 The TSF shall detect when [selection: *[assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]*] unsuccessful authentication **or authorisation** attempts occur related to *consecutive failed authentication or authorisation attempts*[19].

FIA_AFL.1.2 When the defined number of unsuccessful authentication **or authorisation** attempts has been [selection: *met, surpassed*], the TSF shall *block access to [assignment: description of the relevant functionality] until [selection: unblocked by [assignment: identification of the authorised subject or role], a time period [assignment: time period] has elapsed]*[20].

Application Note 18

The Security Target must separately identify the different types of authentication or authorisation to which failure responses apply, and this should include all of the different types of authentication identified for FIA_UAU.1 and failed authorisation attempts related to attempts to use keys as in FIA_UAU.6/KeyAuth. Where different authentication/authorisation failure responses apply then the SFR should be iterated.

The unblocking of functionality blocked as described in each iteration of FIA_AFL.1.2 must be described in a corresponding iteration of FMT_MTD.1 (cf. section 6.3.6).

FIA_UAU.6/KeyAuth *Re-authenticating*

Hierarchical to: No other components.

Dependencies: No dependencies.

FIA_UAU.6.1/KeyAuth The TSF shall **authorise and re-authorise**[21] the user **for access to a secret key** under the conditions

(1) Authorisation in order to be granted initial access to the key; and

(2) [selection:

- *Re-authorisation of [assignment: identification of secret keys that are subject to re-authorisation conditions below] under the following conditions: [selection:*
  - *after expiry of the time period (as specified in the secret key’s attributes) for which the secret key was last authorised;*
  - *after the number of uses of the secret key (as specified in the secret key’s attributes) for which the secret key was last authorised has already been made;*

[19] [assignment: *list of authentication events*]

[20] [assignment: *list of actions*]

[21] re-authenticate

– *after explicit rescinding of previous authorisation for access to the secret key];*

- *[assignment: list of other conditions under which authorisation and re-authorisation for access to secret keys is required];*

- *Authorisation on every subsequent access to the key]*[22].

Application Note 19

Note that any use of a key requires an initial authorisation by presentation of the correct authorisation data. Subsequent uses may require re-authorisation on every use (in this case 'Authorisation on every subsequent access to the key' is selected in FIA_UAU.6.1/KeyAuth (2)), or else the TOE may allow some uses of the key without further authorisation until one of the specified re-authorisation conditions occurs.

The TOE may also allow different re-authorisation conditions for different types of secret key. The types of secret keys may be identified (in the first assignment in (2)) as individual keys, or in terms of a generic definition (e.g. 'all non-Assigned keys'). Where different re-authorisation conditions apply to different types of key then the second assignment in (2) may be used to specify the other types of key and the conditions that apply to them in a similar manner.

The explicit rescinding of an authorisation period in (2) ensures that client applications or users can decide to revoke a previous authorisation in (2) that may still be in force. If the TOE intends to allow unlimited uses of a secret key after initial authorisation, until authorisation is rescinded by a client application or user, then the selection 'after explicit rescinding of previous authorisation for access to the secret key' is chosen in the Security Target without any accompanying selections for time periods or number of uses. The Security Target describes the method or methods used for such rescinding (such as particular API commands).

It is the responsibility of the client application to make appropriate use of any re-authentication conditions according to the application context (cf. OE.DataContext and OE.AppSupport).

Each 'use' of a key is expected to relate to one cryptographic function carried out with the key. If there are circumstances where a different interpretation may be placed on the 'use' of a key then this must be identified and explained in the Security Target and the Operational Guidance. The intention here is to make clear any situations that are relevant to a key owner who can be held responsible for use of the key (such as any case where a single authorisation for use of a key could allow the creation of more than one signature using the authorised key). Note that in order to make qualified electronic signatures under [Regulation] then the user/application must be able to precisely control the signatures that can be made under each authorisation.

Actions taken by the TOE in the case of successive authorisation failures must be specified using an iteration of FIA_AFL.1.

[22] [assignment: *list of conditions under which re-authentication is required*]

6.3.3 User data protection (FDP)

FDP_IFC.1/KeyBasics *Subset information flow control*

Hierarchical to: No other components.

Dependencies: FDP_IFF.1 Simple security attributes

FDP_IFC.1.1/KeyBasics The TSF shall enforce the *Key Basics SFP*[23] on

(1) subjects: all

(2) information: keys

(3) operations: all[24].

FDP_IFF.1/KeyBasics *Simple security attributes*

Hierarchical to: No other components.

Dependencies: FDP_IFC.1 Subset information flow control
FMT_MSA.3 Static attribute initialisation

FDP_IFF.1.1/KeyBasics The TSF shall enforce the *Key Basics SFP*[25] based on the following types of subject and information security attributes:

(1) whether a key is a secret or a public key

(2) whether a secret key is an Assigned Key

(3) whether channels selected to export keys are secure

(4) the value of the Export Flag of a key[26].

FDP_IFF.1.2/KeyBasics The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold:

(1) Export of secret keys shall only be allowed provided that the secret key is not an Assigned Key, that the secret key is encrypted, and that a secure channel (providing authentication and integrity protection) is used for the export

(2) Public keys shall always be exported with integrity protection of their key value and attributes

(3) Keys shall only be imported over a secure channel (providing authentication and integrity protection)

(4) A secret key can only be imported if it is a non-Assigned key

(5) Secret keys shall only be imported in encrypted form or using split-knowledge procedures requiring at least two key components to reconstruct the key, with key components

[23] [assignment: *information flow control SFP*]

[24] [assignment: *list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*]

[25] [assignment: *information flow control SFP*]

[26] [assignment: *list of subjects and information controlled under the indicated SFP, and for each, the security attributes*]

supplied by at least two separately authenticated users

(6) *Unblocking access to a key shall not allow any subject other than those authorised to access the key at the time when it was blocked*[27].

Application Note 20

A secure channel for export of keys in FDP_IFF.1.2/KeyBasics (1) or for import of keys in FDP_IFF.1.2/KeyBasics (3) is one that meets the requirements of FTP_TRP.1/Local or FTP_TRP.1/External.

The encrypted form required for keys imported or exported over a secure channel requires encryption of the key itself, in addition to any encryption provided by the secure channel.

Unblocking a key as in FDP_IFF.1.2/KeyBasics (6) is intended only to restore the ability of subjects to authorise for access to a key by presenting the correct authorisation data. As noted for FMT_MTD.1/Unblock, the subject who unblocks the key must not be able also to use the key as a result of the unblocking (unless of course they are able to supply the correct authorisation data). This is a part of ensuring that sole control of secret keys can be achieved.

FDP_IFF.1.3/KeyBasics The TSF shall enforce the **following additional information flow control rules:** *none*[28].

FDP_IFF.1.4/KeyBasics The TSF shall explicitly authorise an information flow based on the following rules: *none*[29].

FDP_IFF.1.5/KeyBasics The TSF shall explicitly deny an information flow based on the following rules:

(1) *No subject shall be allowed to access the plaintext value of any secret key directly.*

(2) *No subject shall be allowed to export a secret key in plaintext.*

(3) *No subject shall be allowed to export an Assigned Key.*

(4) *No subject shall be allowed to export a secret key without submitting the correct authorisation data for the key*

(5) *No subject shall be allowed to access intermediate values in any operation that uses a secret key*

(6) *A key with an Export Flag value marking it as non-exportable shall not be exported*[30]

Application Note 21

The requirements of FDP_IFF.1/KeyBasics apply regardless of how the key is stored by the TOE, including when the key is externally stored (cf. section 1.3.1.2).

Direct access to a key value in FDP_IFF.1.5/KeyBasics (1) is access that makes the value available for reading or modification – this includes operations that would subsequently allow reading or modification of the key (e.g. making a copy of the key with different attributes, or with a different object

[27] [assignment: *for each operation, the security attribute-based relationship that must hold between subject and information security attributes*]

[28] [assignment: *additional information flow control SFP rules*]

[29] [assignment: *rules, based on security attributes, that explicitly authorise information flows*]

[30] [assignment: *rules, based on security attributes, that explicitly deny information flows*]

type that would then allow direct read access). Note that this PP assumes that key values are never modified after they have been generated.

Export of a key as in FDP_IFF.1.5/KeyBasics (1), (2), (4) and (6) is not the same as backup (governed by FDP_ACF.1/Backup) or external storage of keys under continuing TOE control (governed by other parts of the Key Basics SFP in FDP_IFF.1/KeyBasics, and the Key Usage SFP in FDP_ACF.1/KeyUsage). Thus an Export Flag of 'non-exportable' does not prevent backup or external storage of the keys under continuing TOE control.

The Security Target and/or Operational Guidance shall specify how any attributes not supplied with an imported key are set when the key is imported (or alternatively how such keys are rejected). Similarly the Security Target and/or Operational Guidance shall describe how the key's attributes are represented when exported, so that their meaning can be understood by the receiver.

If the TOE does not provide facilities to import or export keys then the relevant part of the SFR is trivially satisfied, and this should be stated in the Security Target.

FDP_ACC.1/KeyUsage *Subset access control*

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

FDP_ACC.1.1/KeyUsage The TSF shall enforce the *Key Usage SFP*[31] on

(1) *subjects: all*

(2) *objects: keys*

(3) *operations: all*[32].

FDP_ACF.1/KeyUsage *Security attribute based access control*

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/KeyUsage The TSF shall enforce the *Key Usage SFP*[33] to objects based on the following:

(1) *whether the subject is currently authorised to use the secret key*

(2) *whether the subject is currently authorised to change the attributes of the secret key*

(3) *the cryptographic function that is attempting to use the secret key*[34].

[31] [assignment: *access control SFP*]

[32] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

[33] [assignment: *access control SFP*]

[34] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

Application Note 22

Whether a subject is currently authorised for access to a secret key is determined by whether the subject has submitted the correct authorisation data for the key, and whether this authorisation is yet subject to one or more of the re-authorisation conditions in FIA_UAU.6/KeyAuth.

Whether a subject is currently authorised to change the attributes of a secret key is determined by the iterations of FMT_MSA.1 in section 6.3.6.

FDP_ACF.1.2/KeyUsage The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

(1) *Attributes of a key shall only be changed by an authorised subject, and only as permitted in the Key Attributes Modification Table*

(2) *Only subjects with current authorisation for a specific secret key shall be allowed to carry out operations using the plaintext value of that key*

(3) *Only cryptographic functions permitted by the secret key's Key Usage attribute shall be carried out using the secret key*[35].

Application Note 23

FDP_ACF.1.2/KeyUsage (1) refers to controls over changing attributes that are specified in more detail in the iterations of FMT_MSA.1.

FDP_ACF.1.2/KeyUsage (2) requires that a key can only be used when the relevant subject has been authorised either by presenting the correct authorisation data for the key as part of the request for the operation or else the authorisation has previously been presented by the subject and the current use of the key does not yet require re-authorisation according to FIA_UAU.6/KeyAuth (meaning that the current usage is therefore within the usage constraints for time and number of uses since the last authorisation of use of the key). The reference to use of the plaintext value of the key does not imply that a subject has access to that value, only that it can be used to carry out operations within the TOE – reference to operations of this sort are thus distinguished from operations that may use an encrypted form of a secret key (e.g. for external storage of keys) and that are not necessarily restricted in this way.

FDP_ACF.1.3/KeyUsage The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *none*[36].

FDP_ACF.1.4/KeyUsage The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *none*[37]

Application Note 24

The requirements of FDP_ACF.1/KeyUsage apply regardless of how the key is stored by the TOE, including when the key is externally stored (cf. section 1.3.1.2).

FDP_ACC.1/Backup *Subset access control*

Hierarchical to: No other components.

Dependencies: FDP_ACF.1 Security attribute based access control

[35] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

[36] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]

[37] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

FDP_ACC.1.1/Backup The TSF shall enforce the *Backup SFP*[38] on

(1) subjects: all

(2) objects: keys

(3) operations: backup, restore[39].

FDP_ACF.1/Backup *Security attribute based access control*

Hierarchical to: No other components.

Dependencies: FDP_ACC.1 Subset access control
FMT_MSA.3 Static attribute initialisation

FDP_ACF.1.1/Backup The TSF shall enforce the *Backup SFP*[40] to objects based on the following:

(1) whether the subject is an administrator[41].

FDP_ACF.1.2/Backup The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed:

(1) Only authorised administrators shall be able to perform any backup operation provided by the TSF to create backups of the TSF state or to restore the TSF state from a backup

(2) Any restore of the TSF shall only be possible under at least dual person control, with each person being an administrator

(3) Any backup and restore shall preserve the confidentiality and integrity of the secret keys, and the integrity of public keys

(4) Any backup and restore operations shall preserve the integrity of the key attributes, and the binding of each set of attributes to its key [42].

Application Note 25

Preserving the binding of a set of attributes to its key (in FDP_ACF.1.2/Backup (4)) means that it is not possible for the attributes to be changed during a backup operation, or by modification of the backup data while it is away from the TSF.

Backups may contain keys whose export flag attribute marks them as 'non-exportable'.

The ST author specifies the cryptographic operations used to protect confidentiality and integrity of any supported backups using one or more iterations of FCS_COP.1.

FDP_ACF.1.3/Backup The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *none*[43].

[38] [assignment: *access control SFP*]

[39] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

[40] [assignment: *access control SFP*]

[41] [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

[42] [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]

FDP_ACF.1.4/Backup The TSF shall explicitly deny access of subjects to objects based on the following additional rules: *none*[44]

Application Note 26

If the TOE does not provide backup and restore operations then the Security Target shall include FDP_ACC.1/Backup and FDP_ACF.1/Backup but shall state in an Application Note for each of these SFRs that the relevant security requirements are trivially met because no backup facility is provided.

FDP_SDI.2	*Stored data integrity monitoring and action*

Hierarchical to: FDP_SDI.1 Stored data integrity monitoring.

Dependencies: No dependencies.

FDP_SDI.2.1 The TSF shall monitor user data stored in containers controlled by the TSF for *integrity errors*[45] on all **keys (including security attributes)**[46], based on the following attributes: *integrity protection data*[47].

FDP_SDI.2.2 Upon detection of a data integrity error, the TSF shall

(1) prohibit the use of the altered data

(2) *notify the error to the user*[48].

Application Note 27

No specific requirement is placed here on the nature of the integrity protection data, but the Security Target shall describe this protection measure, and shall identify the iteration of FCS_COP.1 that covers any cryptographic algorithm used.

This SFR may also be used in the implementation of the mechanism for protection against modification access to the value of a secret key in FDP_IFF.1.5/KeyBasics, and in the requirement for export of public keys with integrity protection in FDP_IFF.1.2/KeyBasics.

The integrity protection data in FDP_SDI.2.1 is included in the list of attributes identified in FMT_MSA.1/GenKeys and FMT_MSA.1/AKeys, and protects the value of the key and of its other security attributes, including when the key is externally stored by the TOE (cf. section 1.3.1.2).

FDP_RIP.1	*Subset residual information protection*

Hierarchical to: No other components.

Dependencies: No dependencies.

FDP_RIP.1.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the *deallocation of the resource from*[49] the following objects:

- *authorisation data*

[43] [assignment: *rules, based on security attributes, that explicitly authorise access of subjects to objects*]

[44] [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*]

[45] [assignment: *integrity errors*]

[46] objects

[47] [assignment: *user data attributes*]

[48] [assignment: *action to be taken*]

[49] [selection: *allocation of the resource to, deallocation of the resource from*]

- *secret keys*[50].

Application Note 28

Authorisation data is not to be stored persistently in the TOE; the refinements to ADV_ARC.1 in section 6.4.1 require the approach to minimising the time that this data is held before deallocation according to FDP_RIP.1.

6.3.4 Trusted path/channels (FTP)

FTP_TRP.1/Local *Trusted Path*

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_TRP.1.1/Local The TSF shall provide a communication path between itself and *local*[51] **client applications**[52] that is logically distinct from other communication paths and provides assured **authentication**[53] of its end points and protection of the communicated data from *modification and disclosure*[54].

FTP_TRP.1.2/Local The TSF shall permit [selection: *the TSF, local client applications*][55] to initiate communication via the trusted path.

FTP_TRP.1.3/Local The TSF shall require the use of the trusted path for *[assignment: services for which trusted path is required]*[56].

Application Note 29

FTP_TRP.1/Local must be completed in a Security Target to identify the local client applications and to reflect the way that the TOE communicates with them, and to justify the security of this communication path. Where the TOE and local client applications are located within the physical boundary of the same hardware appliance (e.g. local applications running on a server and communicating with a PCI card on the server's internal PCI bus) then the trusted path may be mapped in the Security Target to the physical configuration, and no additional authentication or cryptographic protection are required (because of the physical security assumed in the appliance environment).

If the TOE does not provide an interface for local client applications, then this SFR is not applicable and is trivially satisfied. This should be stated in the Security Target.

The TOE may provide other additional channels that provide only authentication and integrity protection (not confidentiality), in which case other iterations of FTP_TRP.1 may be added in the ST, allowing the selection of only modification protection in FTP_TRP.1.1 for these additional iterations.

The Security Target shall identify in an application note the iterations of FCS_COP.1 that provide any cryptographic functions that contribute to the implementation of the trusted path, and the SFRs that provide the authentication of the end points.

[50] [assignment: *list of objects*]

[51] [selection: *remote, local*]

[52] users

[53] identification

[54] [selection: *modification, disclosure, [assignment: other types of integrity or confidentiality violation]*]

[55] [selection: *the TSF, local users, remote users*]

[56] [selection: *initial user authentication, [assignment: other services for which trusted path is required]*]

FTP_TRP.1/External *Trusted Path*

Hierarchical to: No other components.

Dependencies: No dependencies.

FTP_TRP.1.1/External The TSF shall provide a communication path between itself and *remote*[57] **external client applications**[58] that is logically distinct from other communication paths and provides assured **authentication**[59] of its end points and protection of the communicated data from *modification and disclosure*[60].

FTP_TRP.1.2/External The TSF shall permit [selection: *the TSF, remote* ***external client applications***][61] to initiate communication via the trusted path.

FTP_TRP.1.3/External The TSF shall require the use of the trusted path for *[assignment: services for which trusted path is required]*[62].

Application Note 30

FTP_TRP.1/External must be completed in a Security Target to identify the external client applications and to reflect the way that the TOE communicates with them, and to justify the security of this communication path. The word "remote" in FTP_TRP.1.1/External and FTP_TRP.1.2/External refers to client applications that are described as "external" in the rest of this PP.

If the TOE does not provide an interface for external client applications, then this SFR is not applicable and is trivially satisfied. This should be stated in the Security Target.

The TOE may provide other additional channels that provide only authentication and integrity protection (not confidentiality), in which case other iterations of FTP_TRP.1 may be added in the ST, allowing the selection of only modification protection in FTP_TRP.1.1 for these additional iterations.

The Security Target shall identify in an application note the iterations of FCS_COP.1 that provide any cryptographic functions that contribute to the implementation of the trusted path, and the SFRs that provide the authentication of the end points.

6.3.5 Protection of the TSF (FPT)

FPT_STM.1 *Reliable time stamps*

Hierarchical to: No other components.

Dependencies: No dependencies

FPT_STM.1.1 The TSF shall be able to provide reliable time stamps.

Application Note 31

The TOE must provide timestamps suitable for supporting the time in an audit record for FAU_GEN.1. If the TOE provides additional timestamping services for client applications, or other record of the time of an operation for client applications, then these should be covered in one or more separate iterations of the SFR, with an Application Note added to define any specific requirement for reliability of the time information for that service.

[57] [selection: *remote, local*]

[58] users

[59] identification

[60] [selection: *modification, disclosure, [assignment: other types of integrity or confidentiality violation]*]

[61] [selection: *the TSF, local users, remote users*]

[62] [selection: *initial user authentication, [assignment: other services for which trusted path is required]*]

FPT_TST_EXT.1	*Basic TSF Self Testing*

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests *during initial start-up (or power-on) and [selection: periodically during normal operation, at the request of the authorised user, at the conditions [assignment: conditions under which self-tests should occur]]*[63] to demonstrate the correct operation of the TSF:

- *At initial start-up (or power-on):*
  - *Software/firmware integrity test*
  - *Cryptographic algorithm tests*
  - *Random number generator tests*
- *[assignment: list of additional self-tests run by the TSF]*[64].

Application Note 32

Completion of the selection in FPT_TST_EXT.1.1 may be by 'None' (in which case the 'and' preceding the selection should be deleted and no selection text included). Completion of the list of additional tests in the final assignment may include tests performed at initial start-up (or power-on) and/or tests run under the conditions specified in the earlier selection and assignment. The term 'start-up (or power-on) means that the tests should be executed at least any time that the TOE is powered-on.

The tests of the cryptographic functions shall include all cryptographic functions covered by FCS_COP.1. The Operational Guidance shall include a description of the errors that may arise from self-test and the actions that should be taken in response to each.

FPT_PHP.1	*Passive detection of physical attack*

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_PHP.1.1 The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.

FPT_PHP.1.2 The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.

Application Note 33

Passive detection of a physical attack is typically achieved by using physical seals and an appropriate physical design of the TOE that allows the TOE administrator to verify the physical integrity of the TOE as part of a routine inspection procedure.

Because of the requirement for a physically secure environment with regular inspections (cf. OE.Env), the level of protection (and hence resistance to attack potential) that is required by the implementation of FPT_PHP.1 for this TOE is equivalent to the physical security mechanisms for tamper detection and response required by section 7.7.2 Physical security general requirements and section 7.7.3 Physical security requirements for each physical security embodiment in ISO/IEC 19790:2012 for Security Level 3. (Cf. refinement of AVA_VAN.5 in section 6.4.1.)

[63] [selection: *during initial start-up (on power on), periodically during normal operation, at the request of the authorised user, at the conditions [assignment: conditions under which self-tests should occur]*]

[64] [assignment: *list of self-tests run by the TSF*]

FPT_PHP.3	*Resistance to physical attack*

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_PHP.3.1 The TSF shall resist [assignment: *physical tampering scenarios*] to the [assignment: *list of TSF devices/elements*] by responding automatically such that the SFRs are always enforced.

Application Note 34

This SFR is linked to the requirements for passive detection of physical attacks in FPT_PHP.1, and should identify the relevant responses of the TOE involved in meeting the key zeroisation requirements of ISO/IEC 19790:2012 Security Level 3. As in the case of FPT_PHP.1, because of the requirement for a physically secure environment with regular inspections (cf. OE.Env), the level of protection (and hence resistance to attack potential) that is required by the implementation of FPT_PHP.3 for this TOE is equivalent to the level of assessment for this aspect of tamper detection and response required for section 7.7.2 Physical security general requirements and section 7.7.3 Physical security requirements by each physical security embodiment in ISO/IEC 19790:2012 for Security Level 3. (Cf. refinement of AVA_VAN.5 in section 6.4.1.)

FPT_FLS.1	*Failure with preservation of secure state*

Hierarchical to: No other components.

Dependencies: No dependencies.

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur:

(1) *Self-test according to FPT_TST_EXT.1 fails*

(2) Environmental conditions are outside normal operating range (including temperature and power)

(3) Failures of critical TOE hardware components (including the RNG) occur

(4) Corruption of TOE software occurs

(5) [assignment: *list of other types of failures in the TSF*][65].

Application Note 35

The Operational Guidance shall include a description of the specific failures that are detected (e.g. the thresholds for environmental conditions, and the nature of the monitoring of specific critical TOE hardware components), how these failures are notified, and the actions that should be taken in response to each.

6.3.6 Security management (FMT)

For the purposes of specifying a minimum set security attributes of keys, and the constraints on initialisation and modification of these attributes in FMT_MSA.1 and FMT_MSA.3, two separate types of keys are defined: Assigned Keys (defined and recognised by having their ‘Assigned Flag’ attribute set to ‘assigned’), and general keys (keys that have their ‘Assigned Flag’ attribute set to ‘non-assigned’).

[65] [assignment: *list of types of failures in the TSF*]

Assigned Keys represent a type of key that can be more easily mapped to requirements for sole control because changes to some of their attributes are more tightly controlled (see FMT_MSA.1/AKeys, and the description of attributes below) and, since they are intended for use within the TOE, because they cannot be imported or exported[66]. In particular, an Administrator cannot avoid the need to provide the current authorisation data in order to use such a key, nor can an Administrator change the authorisation data (which would then allow use of the key by the Administrator). This enables a key to be generated and then to be made an Assigned Key at the point where it is assigned to an individual signatory or, in the case of a key used for the creation of electronic seals, to a group of key users[67].

In the FMT_MSA SFRs specified for keys below, the permitted values of assignments have been restricted to identify a minimum set of attributes that must be mapped to their implementation in a TOE, and to specify a minimum set of constraints on their initialisation and subsequent modification. Additional notes regarding these attributes are as follows:

- key identifier: this must be sufficient to uniquely identify the key within the system of which the TOE is a part
- key type: this identifies at a minimum whether the key is a secret key of a symmetric cryptographic algorithm or the secret (commonly called private) key of an asymmetric cryptographic algorithm
- authorisation data: value of data that allows the key to be used for cryptographic operations according to the rules in other SFRs such as FDP_IFF.1/KeyBasics, FDP_ACF.1/KeyUsage, and FDP_ACF.1/Backup. Authorisation data is required only for secret keys
- re-authorisation conditions: the constraints on uses of the key that can be made before re-authorisation is required according to FIA_UAU.6/KeyAuth, and which determines whether a subject is currently authorised to use a key as in FDP_ACF.1/KeyUsage. The types of secret key to which re-authorisation conditions apply, and the details of the re-authorisation conditions for a specific TOE are described in FIA_UAU.6/KeyAuth in section 6.3.2
- key usage: the cryptographic functions that are allowed to use the key as in FDP_ACF.1/KeyUsage
- export flag: indicates whether the key is allowed to be exported (cf. FDP_IFF.1/KeyBasics); allowed values are referred to in this PP as ‘true’ (meaning export is allowed) and ‘false’ (meaning export is not allowed) but may be mapped to other suitable binary values in TOE implementations
- assigned flag: indicates whether the key has currently been assigned. Once a key has been assigned by an Administrator then its authorisation data can only be changed on successful validation of the current authorisation data – it cannot be changed or reset by an Administrator – and the re-authorisation conditions and key usage attributes cannot be changed; allowed values are referred to in this PP as ‘assigned’ and ‘non-assigned’ but may be mapped to other suitable binary values in TOE implementations.

FMT_SMR.1 *Security roles*

Hierarchical to:	No other components.
Dependencies:	FIA_UID.1 Timing of identification.

[66] Assigned Keys may be stored externally in a form that protects the confidentiality and integrity of the key and the binding of the key to its attributes (in particular the requirements of the SFRs FDP_IFF.1/KeyBasics and FDP_SDI.1 apply to externally stored keys), as discussed in section 1.3.1.

[67] Secure operating procedures will be needed in order to ensure that the process from generation to assignment is suitable for maintaining any requirements for non-repudiation that may apply to the application context for use of the key (cf. OE.DataContext and the refinement to AGD_OPE.1 in section 6.4.1).

FMT_SMR.1.1 The TSF shall maintain the roles *Administrator, [selection: Local Client Application, External Client Application], Key User,* [*assignment*: *list of additional authorised identified roles*][68].

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Application Note 36

The Local Client Application role represents an identifiable subject that communicates locally with the TOE, i.e. within the same hardware appliance. The External Client Application role represents an identifiable subject that communicates remotely with the TOE over a secure channel. A TOE can support one or both types of Client Applications.

The Key User role represents a normal, unprivileged subject who can invoke operations on a key according to the other authorisation requirements for the key – this role may sometimes act through a client application.

FMT_SMF.1 *Security management functions*

Hierarchical to: No other components.

Dependencies: No dependencies.

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

(1) Unblock of access due to authentication or authorisation failures

(2) Modifying attributes of keys

(3) Export and deletion of the audit data, which can take place only under the control of the Administrator role

(4) [selection: backup and restore functions, no backup and restore functions]

(5) [selection: key import function, no key import function]

(6) [selection: key export function, no key export function][69].

Application Note 37

The unblocking of authentication or authorisation failures in FMT_SMF.1.1 (1) is related to the authentication failures described in FIA_AFL.1. The attributes of keys in FMT_SMF.1.1 (2) correspond to the attributes in FMT_MSA.1/GenKeys and FMT_MSA.1/AKeys. Export of audit data in FMT_SMF.1.1 (3) relates to the ability to export audit data from the TOE for preservation and storage elsewhere. The selections in FMT_SMF.1.1 (4), (5) and (6) identify whether or not the TOE provides the relevant functions (and must therefore correspond to the relevant statements in the ST for FDP_IFF.1.2/KeyBasics, FDP_ACC.1/Backup and FDP_ACF.1/Backup.

FMT_MTD.1/Unblock *Management of TSF data*

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MTD.1.1/Unblock The TSF shall restrict the ability to *unblock*[70] the [assignment: *list of TSF data*] to [assignment: *the authorised identified administrative roles*].

[68] [assignment: *the authorised identified roles*]

[69] [assignment: *list of management functions to be provided by the TSF*]

Application Note 38

The list of TSF data assigned must correspond to the relevant data blocked by authentication or authorisation failures according to the associated iteration(s) of FIA_AFL.1. For the purposes of unblocking, the TSF data in the assignment includes any key that can be affected by blocking due to failure of authorisation (as in FIA_UAU.6), as well as user accounts (as in FIA_UAU.1) blocked by authentication/authorisation failures.

There is a distinction between administrators authorised to unblock a key and users authorised to use the key. When unblocking a secret key, the unblocking process must not allow a subject to use the key other than a subject who is authorised by presentation of the current authorisation data. For example, an administrator who is able to unblock the key cannot then ***use*** *the key as a result of the unblocking (so the unblocking process does not itself allow the key to be used, nor does it enable the authorisation data to be changed without proving knowledge of the previous authorisation data). This is a part of ensuring that sole control of secret keys can be achieved.*

FMT_MTD.1/AuditLog *Management of TSF data*

Hierarchical to: No other components.

Dependencies: FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MTD.1.1/AuditLog The TSF shall restrict the ability to *control export and deletion of*[71] the *audit log records*[72] to *the Administrator role*[73].

Application Note 39

The control of export and deletion of the audit log records helps to ensure their protection against accidental or malicious deletion (deletion should normally occur only after the records have been exported and preserved outside the TOE). Note that this does not require the Administrator to carry out these export or delete operations manually as long as the actions are controlled *by the Administrator.*

FMT_MSA.1/GenKeys *Management of security attributes*

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1/GenKeys The TSF shall enforce the *Key Usage SFP*[74] to restrict the ability to *modify*[75] the security attributes [assignment: *list of security attributes, to include attributes as specified in the Key Attributes Modification Table*] [76] to [assignment: *list of subjects, objects, and operations among subjects and General Keys, to include at least the constraints specified in the Key Attributes Modification Table*][77].

[70] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]

[71] [selection: *change_default, query, modify, delete, clear, [assignment: other operations]*]

[72] [assignment: *list of TSF data*]

[73] [assignment: *the authorised identified roles*]

[74] [assignment: *access control SFP(s), information flow control SFP(s)*]

[75] [selection: *change_default, query, modify, delete, [assignment: other operations]*]

[76] [assignment: *list of security attributes*]

FMT_MSA.1/AKeys	*Management of security attributes*

Hierarchical to: No other components.

Dependencies: [FDP_ACC.1 Subset access control, or FDP_IFC.1 Subset information flow control]
FMT_SMR.1 Security roles
FMT_SMF.1 Specification of Management Functions

FMT_MSA.1.1/AKeys The TSF shall enforce the *Key Usage SFP*[78] to restrict the ability to *modify*[79] the security attributes [assignment: *list of security attributes, to include attributes as specified in the Key Attributes Modification Table*] [80] to [assignment: *list of subjects, objects, and operations among subjects and Assigned Keys to include at least the constraints specified in the Key Attributes Modification Table*][81].

Application Note 40

The Key Attributes Modification Table is referenced from FMT_MSA.1/GenKeys, and FMT_MSA.1/AKeys. The required constraints on security attribute modification specified in this PP are shown in Table 1; the Security Target completes the other parts not specified here (along with any other information for other security attributes relevant to a particular TOE). The specific attributes used by a particular TOE may vary, but the Security Target must make clear how control is achieved over the ability to modify attributes of keys in terms of the specific attributes and controls imposed by the TOE. Where applicable to the operational environment for a particular TOE, these controls should be described with reference to the ways that they are used to provide qualified electronic signatures and qualified electronic seals that meet the requirements of [Regulation] (cf. the refinement to AGD_OPE.1 in section 6.4.1).

Where a TOE does not support one of the individual types of key then the Security Target states this, and the requirements for that type of key are considered to be trivially satisfied.

Authorisation Data and Re-authorisation conditions are required for secret keys only. Re-authorisation conditions include the conditions specified for FIA_UAU.6.1/KeyAuth (matching the assignments and selections made for that SFR in the Security Target).

[77] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

[78] [assignment: *access control SFP(s), information flow control SFP(s)*]

[79] [selection: *change_default, query, modify, delete, [assignment: other operations]*]

[80] [assignment: *list of security attributes*]

[81] [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]

Key Attribute (MSA.1)	Assigned Key	General Key
Key ID	Cannot be modified	Cannot be modified
Key type	Cannot be modified	Cannot be modified
Authorisation Data	Modified only when modification operation includes successful validation of current (pre-modification) authorisation data	Modified only when modification operation includes successful validation of current (pre-modification) authorisation data, or by an Administrator
Re-authorisation conditions	Cannot be modified	---
Key Usage	Cannot be modified	---
Export Flag	Cannot be modified	---
Assigned Flag	Cannot be modified	Can be modified only by Administrator, and only to change from non-assigned to assigned
Integrity Protection Data	Cannot be modified by users (maintained automatically by TSF)	Cannot be modified by users (maintained automatically by TSF)

Table 1: Key Attributes Modification Table[82]

FMT_MSA.3/Keys *Static attribute initialisation*

Hierarchical to: No other components.

Dependencies: FMT_MSA.1 Management of security attributes
FMT_SMR.1 Security roles

FMT_MSA.3.1/Keys The TSF shall enforce the *Key Usage SFP*[83] to provide [selection, choose one of: *restrictive, permissive, [assignment: other property]*] default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2/Keys The TSF shall allow the [assignment: *the authorised identified roles, according to the constraints in the Key Attributes Initialisation Table*][84] to specify alternative initial values to override the default values when an object or information is created.

[82] It is acceptable for a Security Target to specify more restrictive modification conditions than listed in this table, but not to specify less restrictive modification conditions. Where no specific condition is specified (denoted by '---') then the Security Target is not constrained by this PP, but clearly the requirements of the system of which the cryptographic module is a part may have more detailed requirements for a specific deployment (i.e. operational environment).

[83] [assignment: *access control SFP, information flow control SFP*]

[84] [assignment: *the authorised identified roles*]

Key Attribute (MSA.1)	Assigned Key	Other Key
Key ID	Initialised by generation process	Initialised by generation process
Key type	Initialised by generation process	Initialised by generation process
Authorisation Data	Initialised by creator during generation	Initialised by creator during generation
Re-authorisation conditions	Initialised by Administrator during generation	---
Key Usage	Initialised by creator during generation	---
Export Flag	False (i.e. no export allowed)	---
Assigned Flag	Initialised by generation process	Non-assigned
Integrity Protection Data	Initialised automatically by TSF	Initialised automatically by TSF

Table 2: Key Attributes Initialisation Table[82]

Application Note 41

The Key Attributes Initialisation Table is referenced from FMT_MSA.3/Keys and matches the attributes covered by the separate iterations of FMT_MSA.1 above. The required constraints on security attribute initialisation specified in this PP are shown in Table 2; the Security Target completes the other parts not specified here (along with any other information for other security attributes relevant to a particular TOE). The specific attributes used by a particular TOE may vary, but the Security Target must make clear how control is achieved over the ability to modify attributes of keys in terms of the specific attributes and controls imposed by the TOE. Where applicable to the operational environment for a particular TOE, these controls should be described with reference to the ways that they are used to provide qualified electronic signatures and qualified electronic seals that meet the requirements of [Regulation] (cf. the refinement to AGD_OPE.1 in section 6.4.1).

Where a TOE does not support one of the individual types of key then the Security Target states this, and the requirements for that type of key are considered to be trivially satisfied.

Authorisation Data and Re-authorisation conditions are required for secret keys only, and only as described in the assignments and selections made in the Security Target for FIA_UAU.6/KeyAuth.

Attributes assigned by the TOE to any imported keys must be described in the Security Target and in operational user guidance (see the refinements to AGD_OPE.1 in section 6.4.1), noting that a secret key can only be imported if it is a non-Assigned key (cf. FDP_IFF.1/KeyBasics).

The Integrity Protection Data for a key is used to support FDP_SDI.2 and covers not only the key but also its other attributes.

6.3.7 Security audit data generation (FAU)

FAU_GEN.1 *Audit data generation*

Hierarchical to: No other components.

Dependencies: FPT_STM.1 Reliable time stamps

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events for the *not specified*[85] level of audit; and[86]

c) Startup of the TOE;

d) Shutdown of the TOE

e) Cryptographic key generation (FCS_CKM.1);

f) Cryptographic key destruction (FCS_CKM.4);

g) Failure of the random number generator (FCS_RND.1);

h) Authentication and authorisation failure handling (FIA_AFL.1): all unsuccessful authentication or authorisation attempts, the reaching of the threshold for the unsuccessful authentication or authorisation attempts and the blocking actions taken;

i) All attempts to import or export keys (FDP_IFF.1/KeyBasics);

j) All modifications to attributes of keys (FDP_ACF.1/KeyUsage, FMT_MSA.1/GenKeys and FMT_MSA.1/AKeys*);*

k) Backup and restore (FDP_ACF.1/Backup): use of any backup function, use of any restore function, unsuccessful restore because of detection of modification of the backup data;

l) Integrity errors detected for keys (FDP_SDI.2);

m) Failures to establish secure channels (FTP_TRP.1/Local, FTP_TRP.1/External);

n) Self-test completion (FPT_TST_EXT.1);

o) Failures detected by the TOE (FPT_FLS.1);

p) All administrative actions (FMT_SMF.1, FMT_MSA.1 (all iterations), FMT_MSA.3/Keys,);

q) Unblocking of access (FMT_MTD.1/Unblock);

r) Modifications to audit parameters (affecting the content of the audit log) (FAU_GEN.1)

s) [assignment: *other specifically defined auditable events*][87].

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the

[85] [selection, choose one of: minimum,basic, detailed, not specified]

[86] Levels of audit are not required to be defined in the Security Target.

[87] [assignment: *other specifically defined auditable events*]

functional components included in the PP/ST:

- [assignment: *other audit relevant information*].

Application Note 42

The Security Target is not required to identify separate levels of audit in FAU_GEN.1.1. However, the Operational Guidance is required to describe any configuration or other actions that apply to audit functions, and to make clear, in cases where logging of particular audit events is optional, how to ensure that any individual audit event is logged. Default logging actions of the TOE must also be described in Operational Guidance.

The Administrative Actions logged need not be limited to those related to FMT SFRs: other administrative actions affecting the operation of SFRs should also be included (and listed as part of the assignment in FAU_GEN.1.1).

FAU_GEN.2 *User identity association*

Hierarchical to: No other components.

Dependencies: FAU_GEN.1 Audit data generation
FIA_UID.1 Timing of identification

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

FAU_STG.2 *Guarantees of audit data availability*

Hierarchical to: FAU_STG.1 Protected audit trail storage

Dependencies: FAU_GEN.1 Audit data generation

FAU_STG.2.1 The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.

FAU_STG.2.2 The TSF shall be able to [selection, choose one of: *prevent, detect*] unauthorised modifications to the stored audit records in the audit trail.

FAU_STG.2.3 The TSF shall ensure that *all*[88] stored audit records will be maintained when the following conditions occur: *audit storage exhaustion*[89].

Application Note 43

The Operational Guidance is required to describe any use that the TOE makes of an external audit server, the situation regarding records held locally on the TOE and those held externally on an audit server (e.g. the TOE might accumulate records locally before transferring them to an external audit server), and the way in which audit records are maintained when local audit storage is exhausted (including description of the actions taken by the TOE when audit storage exhaustion is detected). The Operational Guidance shall describe the protection applicable to all records created by the TOE (in order to provide prevention or detection of unauthorised modifications as in FAU_STG.2.2), and shall identify any obligations for the environment in maintaining audit trail protection. The expectation is that this will comprise cryptographic methods of prevention or detection of unauthorised modification (including deletion) of audit records.

Control over export and deletion of the audit log records is limited to the Administrator role as specified in FMT_MTD.1/AuditLog.

[88] [assignment: *metric for saving audit records*]

[89] [selection: *audit storage exhaustion, failure, attack*]

6.4 Security Assurance Requirements

The security assurance requirement level is **EAL4** augmented with **AVA_VAN.5.** The assurance components are identified in the table below (with augmentations in bold). It is noted that due to the physically protected environment in which the TOE operates (as expressed in OE.Env), it is unlikely that physical attacks will be within the scope of an evaluation against this PP.

Assurance Class	**Assurance Components**
Security Target (ASE)	ST introduction (ASE_INT.1)
	Conformance claims (ASE_CCL.1)
	Security problem definition (ASE_SPD.1)
	Security objectives (ASE_OBJ.2)
	Extended components definition (ASE_ECD.1)
	Derived security requirements (ASE_REQ.2)
	TOE summary specification (ASE_TSS.1)
Development (ADV)	Security architecture description (ADV_ARC.1)
	Complete functional specification (ADV_FSP.4)
	Basic modular design (ADV_TDS.3)
	Implementation representation of the TSF (ADV_IMP.1)
Guidance documents (AGD)	Operational user guidance (AGD_OPE.1)
	Preparative procedures (AGD_PRE.1)
Life cycle support (ALC)	Production support, acceptance procedures and automation (ALC_CMC.4)
	Problem tracking CM coverage (ALC_CMS.4)
	Delivery procedures (ALC_DEL.1)
	Identification of security measures (ALC_DVS.1)
	Developer defined life-cycle model (ALC_LCD.1)
	Well-defined development tools (ALC_TAT.1)
Tests (ATE)	Functional testing (ATE_FUN.1)
	Analysis of coverage (ATE_COV.2)
	Testing: basic design (ATE_DPT.1)
	Independent testing – sample (ATE_IND.2)
Vulnerability assessment (AVA)	**Advanced methodical vulnerability analysis** (**AVA_VAN.5**)

Table 3: Security Assurance Requirements

6.4.1 Refinements of Security Assurance Requirements

The following refinements are made to selected assurance requirements in Table 3:

ADV_ARC.1 Security architecture description

Refinement:

The following specific topics must be addressed as part of ADV_ARC.1 for this Protection Profile. It is acceptable for references to deliverables supplied for other assurance families, such as ADV_FSP, to be used to meet these requirements, provided that the relationship of the relevant interface specifications to the concepts in the Protection Profile is clear. Note that in some cases, the requirement for description of these particular aspects under ADV_ARC is intended to make clear any differences between the full capabilities of the product and the scope of the Security Target.

1. In general cryptographic modules will make use of 'support keys' as part of their implementation of protection mechanisms, where these keys are generally not held on behalf of specific users[90] or client applications, but are used by the TOE to carry out its normal operations and as part of the implementation mechanism other SFRs and to protect the TSF itself. These support keys may be used for a variety of purposes (including aspects such as authentication, authorisation, secure channels, security of external storage, or internal data protection), For the purposes of this PP, support keys used by the TOE are treated as TSF data, and require a specific security rationale to be included as part of the ADV_ARC.1 deliverables. This rationale must include a description of the key architecture, identifying all support keys used by the TOE (at least in its evaluated configuration), their method of generation and storage, their purpose in TOE operation, and the ways in which they are protected so as to support the requirements of FDP_IFF.1/KeyBasics and FDP_ACF.1/KeyUsage (noting that the mechanisms used for support keys may differ from those used for user keys). Examples would be keys used for wrapping user keys in order to allow secure storage of the user keys, keys used to implement secure channels, and keys used to protect backups. The description must demonstrate that sufficient entropy has been used in the generation of each support key, and the source of that entropy. The rationale must demonstrate that these support keys cannot be exported/imported in a way that threatens the secure operation of the TOE. The evaluator shall include the description of the support keys in their analysis of the protection of user data (e.g. to confirm that it does not introduce vulnerabilities in the implementation of the SFRs).

2. If updates to the TOE software or firmware are supported then the ADV_ARC.1 deliverables must describe how the TOE is protected against unauthorised updates, by using digital signatures. This shall be confirmed by evaluator testing (if updates are supported) to confirm that updates with invalid signatures are rejected without being executed. The digital signature algorithms used to protect updates shall be included in the scope of FCS_COP.1 signature SFR(s).
3. The ADV_ARC.1 deliverables must in particular describe
    a. Any use that the TOE makes of an audit server
    b. The locations used for any externally stored keys and the structure and format of the externally stored keys including the cryptographic structures that protect the keys in their externally stored form, and that bind them to their attributes (support keys are separately addressed by the description required in item 1 above)
    c. All key import and/or export functions and the secure channels that they use
    d. The secure channels supported by the TOE and the authentication mechanisms that they use (cf. FTP_TRP.1/Local & FTP_TRP.1/External)
    e. All local and external interfaces used for communications with users, client applications, audit data, and stored TOE data (cf. Figure 1)
    f. The specific key attributes supported, their method of representation (e.g. the relevant data structures and permitted values) and the method by which they are bound to the corresponding key value (cf. FMT_MSA.1). This also includes identifying the types of keys (if any) that support re-authorisation conditions described in FIA_UAU.6/KeyAuth
    g. The user types and roles supported, the interfaces by which they interact with the TOE (e.g. a local administrator console or an externally available API), the authentication methods used (cf. FIA_UAU.1 and Application Note 17), and any privileges available to the user type/role
    h. All of the cryptographic functions provided (cf. section 1.3.1.1) and whether any non-endorsed cryptographic algorithms and/or cryptographic functions are available (cf. FCS_COP.1 and section 1.3.1.3)
    i. The authorisation methods used for keys (cf. FIA_UAU.6/KeyAuth & FDP_ACC.1/KeyUsage)

[90] Some support keys may be seen as being held on behalf of administrators, but the main intention of distinguishing support keys and user keys is for the ADV_ARC.1 deliverables to describe all the different types of key available, their properties, and their relationship to the SFRs in this Protection Profile.

j. Description of the way in which the TOE ensures that it only holds authorisation data for the minimum time possible before deallocating it according to FDP_RIP.1
k. If the TOE provides backup operations then the ADV_ARC deliverables shall describe the use of support keys by the backup and restore processes (cf. FDP_ACF.1/Backup), and in particular shall describe the ways in which confidentiality and integrity of the backup are provided, and the way in which the TOE rejects an attempt to carry out a restore process using backup data that has been modified
l. Any mechanisms that the TOE uses to support dual person control (cf. FDP_ACF.1/Backup).

AGD_OPE.1 Operational user guidance

Refinement:

The following specific topics must be addressed as part of the Operational Guidance for the TOE:

1. The specific ways in which the TOE needs to be configured and used in order to provide qualified electronic signatures and qualified electronic seals that meet the requirements of [Regulation]. This includes ways in which the TOE can ensure that the signatory can, with high level of confidence, have sole control over the use of the secret key that acts as his/her signature creation data. Thus, for example, it may be necessary for client applications to use TOE interfaces according to certain guidance in order to correctly implement the requirements on attributes of keys as described in this PP. It may be necessary for the TOE to define ways in which secret keys to be used for signing purposes can be created in a way that does not allow subsequent modification of some or all of their attributes, e.g. by an administrator, before they are assigned to the signatory (cf. FMT_MSA.1/AKeys). The intention of this aspect of the operational user guidance documentation is to identify the configuration and secure use required for a particular TOE, and how it is necessary to connect this with other aspects such as procedural controls and client applications in the operational environment.

   The evaluators shall test the identified ways of using the TOE for qualified electronic signatures and qualified electronic seals to demonstrate that the description in the Operational Guidance is suitably complete, and that the keys produced by following the Operational Guidance do indeed meet the requirements of requirements of [Regulation, Annex II & Annex III] for qualified electronic signatures and qualified electronic seals.

2. The use of trusted channels (cf. FTP_TRP.1/Local & FTP_TRP.1/External).

3. The available key attributes, their possible values, and the meaning of each of these values (cf. FMT_MSA.1/GenKeys and FMT_MSA.1/AKeys, including their use to constrain the period and number of uses that are enabled by authorisation of a key (cf. FIA_UAU.6/KeyAuth and Application Note 19).

4. Identification of any non-endorsed cryptographic algorithms and/or cryptographic functions that are available (cf. FCS_COP.1 and section 1.3.1.3).

5. Identification of any other cryptographic algorithms and operations that are not included in the scope of the Security Target.

6. Possible errors from the self-test process and the actions that should be taken in response to each (cf. FPT_TST_EXT.1 & Application Note 32).

7. Specific failures detected by the TOE (cf. FPT_FLS.1 & Application Note 35).

8. Audit functions and their configuration (including specification of the available audit records), along with any other actions that are associated with audit functions (e.g. archiving or viewing audit records, or use of an external audit server) (cf. FAU_GEN.1 & Application Note 42, FAU_STG.2 & Application Note 43, FMT_MTD.1/AuditLog & Application Note 39).

9. Any configuration and operation requirements for dual-control operations (cf. FDP_ACF.1/Backup).

10. If backup is provided by the TOE (cf. FDP_ACF.1/Backup), then the Operational Guidance shall describe the backup and restore functions, and the administrator roles that are required to carry them out.

11. If key import is provided by the TOE, then the Operational Guidance shall describe how attributes are defined for any imported keys (cf. FMT_MSA.3/Keys). The evaluators shall test the import process to demonstrate that the description in the Operational Guidance is suitably complete, and that the keys imported have attributes appropriately defined. Similarly if key export is provided by the TOE then the Operational Guidance shall describe whether attributes are exported with keys (and if so, then how the attributes are represented and associated with the exported key), and the evaluators shall test the export process to demonstrate that the description in the Operational Guidance is suitably complete, and that the handling of attributes is as described.

ATE_IND.2 Independent testing – sample

Refinement:

The following specific topics must be addressed as part of the independent testing of the TOE:

1. The evaluator shall execute the electronic signature and electronic seal operations provided by the TOE and shall confirm that the signatures and seals returned by the TOE correspond to the correct DTBS.
2. If software and/or firmware updates are supported by the TOE then the evaluator shall carry out tests to ensure that only updates with valid digital signatures can be installed on the TOE.

AVA_VAN.5 Advanced methodical vulnerability analysis

Refinement:

Regarding the protection of the TOE against physical attacks: because of the requirement for a physically secure environment with regular inspections (cf. OE.Env), the level of protection (and hence resistance to attack potential) that is required by the implementation of FPT_PHP.1 and FPT_PHP.3 for this TOE is equivalent to the level of assessment in section 7.7.2 Physical security general requirements and section 7.7.3 Physical security requirements for each physical security embodiment in ISO/IEC 19790:2012 for Security Level 3.

7 Rationales

7.1 Security Objectives Rationale

7.1.1 Security Objectives Coverage

The table below shows the mapping of Threats, Organisational Security Policies and Assumptions to Security Objectives for the TOE and for the TOE Environment.

	OT.PlainKeyConf	OT.Algorithms	OT.KeyIntegrity	OT.Auth	OT.KeyUseConstraint	OT.KeyUseScope	OT.DataConf	OT.DataMod	OT.ImportExport	OT.Backup	OT.RNG	OT.TamperDetect	OT.FailureDetect	OT.Audit	OE.ExternalData	OE.Env	OE.DataContext	OE.AppSupport	OE.Uauth	OE.AuditSupport
T.KeyDisclose	X		X				X		X	X		X			X	X				
T.KeyDerive		X									X									
T.KeyMod			X						X	X		X								
T.KeyMisuse				X	X															
T.KeyOveruse						X														
T.DataDisclose							X										X	X		
T.DataMod								X									X	X		
T.Malfunction													X							
P.Algorithms		X																		
P.KeyControl	X	X		X	X	X			X	X										
P.RNG											X									
P.Audit														X						
A.ExternalData															X					
A.Env																X				
A.DataContext																	X			
A.AppSupport																		X		
A.UAuth																			X	
A.AuditSupport																				X

Table 4: Security Problem Definition mapping to Security Objectives

7.1.2 Security Objectives Sufficiency

The following paragraphs describe the rationale for the sufficiency of the Security Objectives relative to the Threats, OSPs and Assumptions.

7.1.2.1 Threats

T.KeyDisclose is addressed by the requirement in OT.PlainKeyConf to keep plaintext secret keys unavailable, and this is supported in terms of controls over key attributes (which might threaten the confidentiality of the key if modified) in OT.KeyIntegrity. The confidentiality of secret keys that are exported is protected partly by the use of a secure channel as described in OT.DataConf and the requirements for import and export in OT.ImportExport (including the requirement to export secret keys only in encrypted form, or to be able to exclude the export of a key entirely). Physical tamper protection of the keys is provided by OT.TamperDetect (supported by an appropriate inspection procedure as required in OE.Env). Protection of secret key confidentiality during backup is ensured by OT.Backup. The environment also contributes to maintaining secret key confidentiality by protecting any versions of a secret key that may exist outside the TOE, as in OE.ExternalData, and by protecting the operation of the TOE itself by providing a secure environment, as in OE.Env.

T.KeyDerive is addressed by the choice of algorithms that have been endorsed for the appropriate purposes, and this is described in OT.Algorithms. Where keys are generated by the TOE then the use of a suitable random number generator is required by OT.RNG in order to mitigate the risk that an attacker can guess or deduce the key value.

T.KeyMod is addressed by requiring integrity protection of secret and public keys, and their critical attributes in OT.KeyIntegrity, and by requiring use of secure channels that protect integrity if a key is imported or exported (OT.ImportExport). Protection of key integrity during backup is ensured by OT.Backup. Physical tamper protection of the keys is provided by OT.TamperDetect (supported by an appropriate inspection procedure as required in OE.Env).

T.KeyMisuse raises the possibility of a secret key being used for an unintended and unauthorised purpose, and is addressed by the requirement in OT.Auth for the TOE to carry out an authorisation check before using a secret key. OT.KeyUseConstraint expands on this to set out requirements for the granularity of authorisation.

T.KeyOveruse is concerned with the possibility that more uses may be made of an authorised key than were intended, and this is addressed by the requirements of OT.KeyUseScope which requires controls to be specified and enforced for any re-authorisation conditions that the TOE allows a user to define.

T.DataDisclose is concerned with the transmission of data between client applications and the TOE, or between separate parts of the TOE where the transmission passes through an insecure environment. This is addressed by OT.DataConf, which requires the TOE to provide secure channels to protect such communications. The appropriate use of such channels is a requirement for the environment as expressed in OE.DataContext, as is the use of appropriate procedures in OE.AppSupport.

T.DataMod is concerned with the possibility of unauthorised modification of data transmitted between a client application and the TOE, and this is addressed by OT.DataMod which requires that the TOE provides secure channels that can be used to protect the integrity of data that they carry. As with T.DataDisclose, the appropriate use of such channels is a requirement for the environment as expressed in OE.DataContext, as is the use of appropriate procedures in OE.AppSupport.

T.Malfunction is addressed by the requirement in OT.FailureDetect for the TOE to detect certain types of fault.

7.1.2.2 Organisational Security Policies

P.Algorithms requires the use of key generation and other cryptographic functions that are endorsed by appropriate authorities, and this is addressed by OT.Algorithms.

P.KeyControl requires that the TOE can provide controls and support a key lifecycle to ensure that secret keys can be reliably protected against use by those other than the owner of the key, and that the keys can be confined to use for certain cryptographic functions. This is addressed by a combination of TOE objectives as follows:

- OT.PlainKeyConf protects the value of the secret key to prevent the possibility of it being used by unauthorised subjects
- OT.Algorithms ensures that endorsed algorithms that employ and support suitable properties and procedures are provided by the TOE
- OT.Auth, OT.KeyUseConstraint and OT.KeyUseScope ensure that the TOE can provide well-defined limits on the use of a key when it is authorised (as described above for T.KeyMisuse and T.KeyOveruse)

- OT.ImportExport and OT.Backup ensure protection of keys when they are transmitted outside the TOE to client applications or for backup purposes, including the prevention of export of Assigned Keys.

P.Audit requires the TOE to provide an audit trail and this is addressed directly by OT.Audit (which includes protection of the audit records).

7.1.2.3 Assumptions

Each of the Assumptions in section 3.5 is directly matched by a security objective for the operational environment in section 4.2. The wording of each objective for the operational environment includes the wording of each assumption, and no further rationale is therefore given here.

7.2 Security Requirements Rationale

7.2.1 Security Requirements Coverage

The table below summarises the mapping of Security Objectives for the TOE to SFRs.

	OT.PlainKeyConf	OT.Algorithms	OT.KeyIntegrity	OT.Auth	OT.KeyUseConstraint	OT.KeyUseScope	OT.DataConf	OT.DataMod	OT.ImportExport	OT.Backup	OT.RNG	OT.TamperDetect	OT.FailureDetect	OT.Audit
FCS_CKM.1		X												
FCS_CKM.4	X													
FCS_COP.1		X												
FCS_RNG.1											X			
FIA_UID.1				X										
FIA_UAU.1				X										
FIA_AFL.1				X										
FIA_UAU.6/KeyAuth				X		X								
FDP_IFC.1/KeyBasics	X				X				X					
FDP_IFF.1/KeyBasics	X		X		X				X					
FDP_ACC.1/KeyUsage					X	X								
FDP_ACF.1/KeyUsage					X	X								
FDP_ACC.1/Backup										X				
FDP_ACF.1/Backup										X				
FDP_SDI.2			X											
FDP_RIP.1	X				X									
FTP_TRP.1/Local			X	X			X	X	X					
FTP_TRP.1/External			X	X			X	X	X					
FPT_STM.1														X
FPT_TST_EXT.1													X	
FPT_PHP.1												X		
FPT_PHP.3												X		
FPT_FLS.1													X	
FMT_SMR.1				X										X
FMT_SMF.1				X										X
FMT_MTD.1/Unblock				X										
FMT_MTD.1/AuditLog														X
FMT_MSA.1/GenKeys					X									
FMT_MSA.1/AKeys					X									
FMT_MSA.3/Keys					X									

	OT.PlainKeyConf	OT.Algorithms	OT.KeyIntegrity	OT.Auth	OT.KeyUseConstraint	OT.KeyUseScope	OT.DataConf	OT.DataMod	OT.ImportExport	OT.Backup	OT.RNG	OT.TamperDetect	OT.FailureDetect	OT.Audit
FAU_GEN.1														X
FAU_GEN.2														X
FAU_STG.2														X

Table 5: TOE Security Objectives mapping to SFRs

OT.PlainKeyConf is addressed by the requirements in the Key Basics SFP defined in FDP_IFC.1/KeyBasics and FDP_IFF.1/KeyBasics (especially FDP_IFF.1.5/KeyBasics). Secure destruction of keys according to FCS_CKM.4 protects the key value at the end of its lifetime. FDP_RIP.1 protects secret keys from being accessed after they have been deallocated.

OT.Algorithms is addressed by the need to use endorsed standards for FCS_COP.1 (cf. Application Note 14) and the use of an appropriate random number generator in FCS_CKM.1. Note that the refinements to assurance components in section 6.4.1 also specify requirements that ensure clear documentation of endorsed and non-endorsed algorithms and functions provided by the TOE.

OT.KeyIntegrity is addressed primarily by FDP_SDI.2 which requires integrity protection of keys and their attributes by the TOE. FDP_IFF.1/KeyBasics requires that any importing or exporting of keys requires the use of secure channels and integrity protection (cf. the requirement for an integrity-protected channel as part of FTP_TRP.1/Local and FTP_TRP.1/External, which is linked to the Key Basics SFP by Application Note 20 under FDP_IFF.1/KeyBasics).

OT.Auth is addressed by FIA_UID.1, FIA_UAU.1 and FIA_AFL.1 for administrator authentication (with FMT_MTD.1/Unblock and its dependencies on FMT_SMR.1 and FMT_SMF.1 ensuring that appropriate roles and unblocking for authorisation and authentication failures are also provided). Authorisation for external client applications is provided by the requirements for authentication of endpoints in FTP_TRP.1/Local and FTP_TRP.1/External. Authorisation for the use of secret keys is addressed by FIA_UAU.6/KeyAuth.

OT.KeyUseConstraint is addressed by the requirements for well-defined (and securely initialised) key attributes in FMT_MSA.1/GenKeys, FMT_MSA.1/AKeys, and FMT_MSA.3/Keys, and the application of the attributes to operate constraints on the use of keys in FDP_IFC.1/KeyBasics, FDP_IFF.1/KeyBasics, FDP_ACC.1/KeyUsage and FDP_ACF.1/KeyUsage. FDP_RIP.1 protects authorisation data (which enables a key to be used) from being accessed after it has been deallocated.

OT.KeyUseScope is addressed by the Key Usage SFP in FDP_ACC.1/KeyUsage and FDP_ACF.1/KeyUsage and by the re-authorisation conditions for use of a secret key specified in FIA_UAU.6/KeyAuth.

OT.DataConf is addressed by the authentication and confidentiality requirements for secure channels in FTP_TRP.1/Local and FTP_TRP.1/External.

OT.DataMod is addressed by the authentication and integrity requirements for secure channels in FTP_TRP.1/Local and FTP_TRP.1/External.

OT.ImportExport is addressed by the requirements for the use of secure import/export through a secure channel and restrictions on how keys are imported and exported to protect confidentiality and integrity in the Key Basics SFP in FDP_IFC.1/KeyBasics and FDP_IFF.1/KeyBasics, and by the requirements on the secure channels themselves in FTP_TRP.1/Local and FTP_TRP.1/External.

OT.Backup separates out the requirements for any backup and restore properties that the TOE may provide and is addressed directly by the Backup SFP in FDP_ACC.1/Backup and FDP_ACF.1/Backup.

OT.RNG is addressed by the requirement in FCS_RNG.1 for a random number generator of an appropriate type, which meets appropriate randomness metrics.

OT.TamperDetect is addressed by the requirement for passive tamper detection in FPT_PHP.1 and the tamper response mechanisms in FPT_PHP.3.

OT.FailureDetect is addressed by the self-test requirements of FPT_TST_EXT.1 and secure failure requirements of FPT_FLS.1.

OT.Audit is addressed in terms of basic creation of audit records by the requirements for audit record generation in FAU_GEN.1 and FAU_GEN.2 and provision of timestamps for use in audit records in FPT_STM.1. Protection of the audit trail is ensured by FAU_STG.2, FMT_MTD.1/AuditLog and FMT_SMF.1. Support for the Administrator role that controls export and deletion of audit records from the TOE is required by FMT_SMR.1.

7.2.2 SFR Dependencies

The dependencies between SFRs are addressed as shown in Table 6. Where a dependency is not met in the manner defined in [CC2] then a rationale is provided for why the dependency is unnecessary or else met in some other way.

Requirement	**Dependencies**	**Fulfilled by**
FCS_CKM.1	[FCS_CKM.2 or FCS_COP.1] FCS_CKM.4	FCS_COP.1 FCS_CKM.4
FCS_CKM.4	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1]	FCS_CKM.1 See also note below on key attributes during import or export.
FCS_COP.1	[FDP_ITC.1 or FDP_ITC.2 or FCS_CKM.1] FCS_CKM.4	FCS_CKM.1 FCS_CKM.4 See also note below on key attributes during import or export.
FCS_RNG.1	No dependencies	
FIA_UID.1	No dependencies	
FIA_UAU.1	FIA_UID.1	FIA_UID.1
FIA_AFL.1	FIA_UAU.1	FIA_UAU.1
FIA_UAU.6/KeyAuth	No dependencies	
FDP_IFC.1/KeyBasics	FDP_IFF.1	FDP_IFF.1/KeyBasics
FDP_IFF.1/KeyBasics	FDP_IFC.1 FMT_MSA.3	FDP_IFC.1/KeyBasics FMT_MSA.3/Keys
FDP_ACC.1/KeyUsage	FDP_ACF.1	FDP_ACF.1/KeyUsage
FDP_ACF.1/KeyUsage	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/KeyUsage FMT_MSA.3/Keys
FDP_ACC.1/Backup	FDP_ACF.1	FDP_ACF.1/Backup

Requirement	Dependencies	Fulfilled by
FDP_ACF.1/Backup	FDP_ACC.1 FMT_MSA.3	FDP_ACC.1/Backup The dependency on FMT_MSA.3 is not relevant in this case since the attribute used in FDP_ACF.1/Backup is determined by the ability of the user to authenticate as an administrator according to FIA_UAU.1.
FDP_SDI.2	No dependencies	
FDP_RIP.1	No dependencies	
FTP_TRP.1/Local	No dependencies	
FTP_TRP.1/External	No dependencies	
FPT_STM.1	No dependencies	
FPT_TST_EXT.1	No dependencies	
FPT_FLS.1	No dependencies	
FMT_SMR.1	FIA_UID.1	FIA_UID.1
FMT_SMF.1	No dependencies	
FMT_MTD.1/Unblock	FMT_SMR.1 FMT_SMF.1	FMT_SMR.1 FMT_SMF.1
FMT_MTD.1/AuditLog	FMT_SMR.1 FMT_SMF.1	FMT_SMR.1 FMT_SMF.1
FMT_MSA.1/GenKeys	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1/KeyUsage FDP_IFC.1/KeyBasics FMT_SMR.1 FMT_SMF.1
FMT_MSA.1/AKeys	[FDP_ACC.1 or FDP_IFC.1] FMT_SMR.1 FMT_SMF.1	FDP_ACC.1/KeyUsage FDP_IFC.1/KeyBasics FMT_SMR.1 FMT_SMF.1
FMT_MSA.3/Keys	FMT_MSA.1 FMT_SMR.1	FMT_MSA.1/GenKeys, FMT_MSA.1/AKeys FMT_SMR.1
FAU_GEN.1	FPT_STM.1	FPT_STM.1
FAU_GEN.2	FAU_GEN.1 FIA_UID.1	FAU_GEN.1 FIA_UID.1
FAU_STG.2	FAU_GEN.1	FAU_GEN.1

Table 6: SFR Dependencies Rationale

Key attributes during import or export: the TOE may allow import or export of keys according to the rules in FDP_IFF.1/KeyBasics. For keys that may be imported or exported, the TOE does not place any specific requirements on whether attributes are imported and exported with keys. However, the refinement to AGD_OPE.1 in section 6.4.1 requires that the behaviour of the TOE in this situation is described in documentation, and that the evaluators confirm the behaviour that is documented.

Application Note 41 (for FMT_MSA.1) also requires that the initialisation of any attributes on import is described in the Security Target.

7.2.3 Rationale for SARs

The assurance level for this protection profile is **EAL4 augmented with AVA_VAN.5**.

EAL4 allows a developer to attain a reasonably high assurance level without the need for highly specialised processes and practices. It is considered to be the highest level that could be applied to an existing product line without undue expense and complexity. As such, EAL4 is appropriate for commercial products that can be applied to moderate to high security functions.

The TOE described in this protection profile is just such a product. Augmentation results from the selection of **AVA_VAN.5**. All the dependencies of AVA_VAN.5 are satisfied by other assurance components in the EAL4 assurance package.

7.2.4 AVA_VAN.5 Advanced methodical vulnerability analysis

The TOE generates uses and manages the highly sensitive data in the form of secret keys, at least some of which may be used as signature creation data. The protection of these keys and associated security of their attributes and use in cryptographic operations can only be ensured by the TOE itself. While the TOE environment is intended to protect against physical attacks, a high level of protection against logical attacks (especially those that might be carried out remotely) is also necessary, and is therefore addressed by augmenting vulnerability analysis to deal with High attack potential.

Bibliography

[CC1] Common Criteria for Information Technology Security Evaluation,
Part 1: Introduction and general model,
Version 3.1 Revision 4, September 2012,
CCMB-2012-09-001

[CC2] Common Criteria for Information Technology Security Evaluation,
Part 2: Security functional requirements,
Version 3.1 Revision 4, September 2012,
CCMB-2012-09-002

[CC3] Common Criteria for Information Technology Security Evaluation,
Part 3: Security assurance requirements,
Version 3.1 Revision 4, September 2012,
CCMB-2012-09-003

[CEM] Common Methodology for Information Technology Security Evaluation,
Evaluation Methodology,
Version 3.1 Revision 4, September 2012,
CCMB-2012-09-004

[CEN] CEN/ISSS WS/E-Sign; Area D1, CWA 14167-1: Security Requirements for Trustworthy Systems Managing Certificates for Electronic Signatures

[CEN TS 419 241] CEN TS 419 241
Requirements for Trustworthy Systems Supporting Server Signing

[CWA 14170] prEN 14170-1:2011
Protection profiles for signature creation and verification application
Part 1: Introduction to the European Norm

[Regulation] REGULATION (EU) No 910/2014 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC

[SOG-IS-Crypto] SOG-IS Crypto Evaluation Scheme Agreed Cryptographic Mechanisms, v1.0, May 2016

[TS 119 312] ETSI TS 119 312
Electronic Signatures and Infrastructures (ESI); Cryptographic Suites

Appendix A Acronyms

CC	Common Criteria
DTBS	Data To Be Signed
DTBS/R	Data to be signed or its unique representation
EAL	Evaluation Assurance Level
IT	Information Technology
PCIe	Peripheral Component Interconnect Express
PP	Protection Profile
RNG	Random Number Generator
SAR	Security assurance requirements
SFP	Security Function Policy
SFR	Security functional requirements
ST	Security Target
TOE	Target of Evaluation
TSF	TOE Security Functions
TSFI	TSF Interface
TSP	Trust Service Provider

Appendix B Mapping to [Regulation] (Informative)

This Appendix provides a mapping from the 'Requirements For Qualified Electronic Signature Creation Devices' in [Regulation, Annex II] to parts of this PP that address the requirements.

For remote signing, the TOE on its own is not intended to meet the requirements for QSCDs in the context of remote signing set out in Annex II of (EU) No 910/2014. It is expected that the TOE would be used in conjunction with the protection profile to be defined in EN 419 241-2, and any other related protection profiles, to meet the requirements for Sole Control Assurance Level 2 as defined in EN 419 241-1.

[Regulation, Annex II] requirement	PP coverage of requirement
1. Qualified electronic signature creation devices shall ensure, by appropriate technical and procedural means, that at least:	
(a) the confidentiality of the electronic signature creation data used for electronic signature creation is reasonably assured;	The PP addresses direct threats to the confidentiality of secret keys used as electronic signature creation data in the threats as follows: • T.KeyDisclose: adoption of this threat and the resulting security objectives and SFRs ensure that the risk of unauthorised access to a secret key is addressed. • T.KeyDerive: this threat and the resulting security objectives and SFRs ensure that the risk of confidentiality being breached by derivation of a secret key from generally available data is addressed. Potential disclosure of a secret key while held outside the TOE is addressed in: • A.ExternalData directly addresses the need to protect copies of sensitive data held outside the TOE • A.Env addresses the general need for security in the operational environment.
(b) the electronic signature creation data used for electronic signature creation can practically occur only once;	The risk of duplicating signature creation data maps to the risk of using a poor random number generator in the TOE. This is addressed by the following policy: • P.RNG: this requires a quality metric for random numbers to be specified and achieved by the TOE. In addition: the use of appropriate algorithms is addressed by: • P.Algorithms: this requires the use of algorithms endorsed by appropriate authorities, and therefore mitigates the risk of duplication related to use of inappropriate cryptographic algorithms or parameters.

[Regulation, Annex II] requirement	PP coverage of requirement
(c) the electronic signature creation data used for electronic signature creation cannot, with reasonable assurance, be derived and the electronic signature is reliably protected against forgery using currently available technology;	The PP addresses the potential for derivation of electronic signature creation data by: • T.KeyDerive: this states the threat of derivation of a secret key from generally available data. • P.Algorithms: this requires the use of algorithms endorsed by appropriate authorities, and therefore mitigates the risk of inappropriate cryptographic algorithms that might allow derivation of the secret key. Protection against forgery is achieved as a result of the protection of secret key confidentiality described in rows above, and protection against use of the secret key by users other than the legitimate signatory described in rows below, and by: • T.KeyMod: this ensures that an attacker cannot modify or substitute a known key that might enable them to produce a signature that appears to be associated with a different entity.
(d) the electronic signature creation data used for electronic signature creation can be reliably protected by the legitimate signatory against use by others.	The risk of unauthorised use of a signatory's secret key is addressed by: • T.KeyMisuse: this directly introduces the threat of unauthorised use of a secret key (or cryptographic function using that key) • T.KeyOveruse: this addresses the threat of a key authorised for a specific use then being used beyond that specific use (e.g. in order to produce more signatures than were authorised). • T.DataDisclose: this threat ensures that the PP covers potential exposure of sensitive data exchanged with applications, where this data might enable signatures to be produced by users other than the legitimate signatory. • T.DataMod: this threat ensures that the PP covers the risk of modification of sensitive data exchanged with applications, where this might enable modification or substitution with data known and/or controlled by an attacker and that might thereby enable signatures to be produced by users other than the legitimate signatory. • P.KeyControl: this requires the TOE to support lifecycles that enable protection of secret keys against use by others. • A.DataContext: addresses the need for client applications to use the TOE in a way appropriate to the security requirements of the client applications. • A.UAuth: addresses the need for client applications to identify and authenticate their users as part of ensuring that use of the TOE

[Regulation, Annex II] requirement	PP coverage of requirement
	data and functions are properly authorised.
2. Qualified electronic signature creation devices shall not alter the data to be signed or prevent such data from being presented to the signatory prior to signing.	The PP addresses the threat of alteration of the data to be signed by: • T.DataMod: this threat specifically identifies the possibility of modification of the DTBS/R and leads to TOE objectives that provide channels that would prevent such modification during transmission of DTBS/R between client applications and the TOE. • A.DataContext: this addresses the need for client applications to use TOE functions to provide appropriate security (e.g. by using secure channels to protect the DTBS/R). It also identifies that if justified by the risks in the operational environment then a suitable entity should check the signature returned from the TOE, to confirm that it relates to the correct DTBS. The PP also includes a refinement of ATE_IND.2 that requires the evaluators to perform sample electronic signatures and electronic seals, and to check their correspondence to the DTBS. (This establishes a baseline of confidence in the correctness of the signature process with regard to its preservation of the DTBS). The presentation of data to the signatory before signing is a responsibility of client applications rather than the TOE.
3. Generating or managing electronic signature creation data on behalf of the signatory may only be done by a qualified trust service provider.	Ensuring the use of the TOE by a qualified trust service provider is beyond the scope of the PP, and can only be achieved by examining the details of a service offering rather than testing and analysis of the cryptographic module product(s) used to provide the service. However, the PP supports use of the TOE by a qualified trust service provider by setting requirements for ways that the TOE can be used and the controls that it provides for the trust service provider and client applications to use. These are described in other rows of this table, but in particular those addressing 1(a) and 1(d) above. In addition the general requirements for audit contribute to the needs of a qualified trust service provider to demonstrate suitable operational practices: • P.Audit: requires the TOE to provide an audit trail of security-relevant events. • A.AuditSupport: leads to a requirement for the environment to adopt a procedure for reviewing the audit trail.

[Regulation, Annex II] requirement	PP coverage of requirement
4. Without prejudice to point (d) of point 1, qualified trust service providers managing electronic signature creation data on behalf of the signatory may duplicate the electronic signature creation data only for back-up purposes provided the following requirements are met:	
(a) the security of the duplicated datasets must be at the same level as for the original datasets;	The PP allows for an optional backup capability in a TOE: • T.KeyDisclose: the requirements of this threat are extended to any backup capability provided by the TOE (this is ensured by the definition of OT.Backup as one of the TOE security objectives required to mitigate T.KeyDisclose). • P.KeyControl: the requirements of this policy are also extended to any backup capability provided by the TOE (this is ensured by the definition of OT.Backup as one of the TOE security objectives required to meet P.KeyControl). In addition, the same requirements for security of the environment apply to backups in: • A.ExternalData • A.Env.
(b) the number of duplicated datasets shall not exceed the minimum needed to ensure continuity of the service.	This cannot be directly enforced by the TOE, but the PP identifies it as a requirement on the operational environment in A.ExternalData.

Table 7: Mapping between [Regulation, Annex II] and this PP

In addition to the specific mappings above, some parts of the PP address general requirements for the TOE that underlie many of the requirements in [Regulation, Annex II]:

- the threat T.Malfunction mitigates the risk of malfunction of the TOE leading to failure or weakening of other security properties
- the assumption A.AppSupport (and the corresponding objective on the environment OE.AppSupport) identify the need for the operational environment to use the TOE in a way that supports the security needs of the applications using the TOE services and data.