Joint Interpretation Library

CEM Refinements for POI Evaluation

Version 1.0 (for trial use)
27th May 2011

Acknowledgments:

The organisations listed below and organised within the Joint Interpretation Working Group (JIWG) provide JIWG Supporting documents in order to assist the consistent application of the criteria and methods between SOG-IS Evaluation and Certification Schemes.

France:	Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI)
Germany:	Bundesamt für Sicherheit in der Informationstechnik (BSI)
Italy:	Organismo di Certificazione della Sicurezza Informatica (OCSI)
Netherlands:	Netherlands National Communications Security Agency (NLNCSA)
Spain:	Centro Criptológico Nacional (CCN)
United Kingdom:	Communications-Electronics Security Group (CESG)

They acknowledge the contribution of the work on the development of this version of this JIWG Supporting document done by evaluation labs, several POI terminal vendors, and other companies organised within the

- Groupe d’Evaluation de la Sécurité des Terminaux électroniques (GESTe)[1] and the
- JIL Terminal Evaluation Methodology Subgroup (JTEMS).

[1] "GESTe (Groupe d'Evaluation de la Sécurité des Terminaux Electroniques) is a French ANR (Agence Nationale pour la Recherche) project started in March 2009 and ending in September 2011, involving the following partners; Ingenico, CEA LETI, EMSE, GIE- Cartes-Bancaires, Maxim, Thales-CESTI, Trusted Labs and University of Caen"

Table of Contents

This page is intentionally left blank

1 Introduction to the CEM refinements

1 This document provides refinements for the current version of the Common Criteria Methodology [CEM] for the evaluation of POIs in relation to the assurance components required by the Common Criteria POI Protection Profile [CC POI PP]. The acronym POI stands for Point of Interaction. It designates the Target of Evaluation (TOE) of the [CC POI PP]. The certification bodies, evaluation facilities and payment schemes of SEPA (Single Euro Payment Area) created [CC POI PP] with the objective to meet CC requirements and payment scheme needs.

2 According to [CC POI PP] the set of [CC] assurance components of EAL POI has to be used for the evaluation of a POI. This document refines or interprets these assurance components for each work unit of the [CEM] in order to meet payment schemes needs for POI security evaluation.

3 Guidance for the application of the assurance components is given in form of refinements or interpretation of work units of the [CEM].

4 The guidance refers to requirements of [CAS]. If not stated otherwise the refinement is valid for the complete TOE, i.e. for all the TSF parts that compose it.

5 In the following the notion “security relevant components of the TOE” is used. Security relevant components of the TOE are all components implementing the TOE security functionality (TSF).

6 There are two different sorts of requirements in this guidance.

7 In [CC POI PP] section 8.2 Security Assurance Requirements, the assurance class ADV components required by the EAL POI are marked as “STANDARD” whereas components like ALC_DVS.2 are marked as “REFINED”. These terms refer to [CAS] requirements inclusion in EAL POI. In a “STANDARD” assurance component, no [CAS] requirement has been added. In a “REFINED” component, [CAS] requirements are included and must be mandatorily checked by the evaluator.

8 This guidance comprises interpretations (also called explicative refinements) for the ADV class and its “STANDARD” components. They bring explanations and examples dedicated to POI evaluations and aim at a better understanding of the CEM by the developers, e.g. subsystems, TSFIs and SFRs are described in [CC POI PP] terms. They must be understood as pure examples and are in no way mandatory. The developper is free to write the ADV documentation without using them. Mandatory requirements on the POI under evaluation are in [CC POI PP] alone.

9 The mandatory [CAS] refinements or the explicative refinements are all marked with **bold letters**.

2 Scope

10 This document is a guidance for the application of the assurance components defined in EAL POI (cf. [CC POI PP]) during POI pilot evaluations. The experience acquired during the pilot phase shall become an input for the update of this document

11 The EAL POI comprises the following assurance components (c.f. section 8.2 Security Assurance Requirements in [CC POI PP]):

- Development: ADV_ARC.1, ADV_FSP.2, ADV_TDS.1
- Guidance: AGD_OPE.1, AGD_PRE.1
- Life cycle: ALC_CMC.2, ALC_CMS.2, ALC_DEL.1, ALC_DVS.2
- Vulnerability : AVA_POI.1, AVA_POI.2, AVA_POI.3, AVA_POI.4
- Test: ATE_COV.1, ATE_FUN.1, ATE_IND.2

12 The current version of this document does not provide any guidance on the ATE class. The ATE components text can be found in the original [CEM], without additional refinements.

13 CAS security requirements, which include PCI security requirements as well as security requirements on payment transaction data and external communication, have been translated in Common Criteria requirements, functional and assurance to be used in [CC POI PP]. Still, this guidance does not depend on PCI or CAS requirements. The references made to PCI or CAS requirements in this guide comes from the necessary compliance to [CC POI PP],

3 Class ADV: Development

3.1 Security Architecture (ADV_ARC)

3.1.1 Evaluation of sub-activity (ADV_ARC.1)

3.1.1.1 Objectives

14 The objective of this sub-activity is to determine whether the TSF is structured such that it cannot be tampered with or bypassed, and whether TSFs that provide security domains isolate those domains from each other.

3.1.1.2 Input

15 The evaluation evidence for this sub-activity is:

a) the ST;

b) the functional specification;

c) the TOE design;

d) the security architecture description;

e) the implementation representation (if available);

f) the operational user guidance;

3.1.1.3 Action ADV_ARC.1.1E

ADV_ARC.1.1C **The security architecture description shall be at a level of detail commensurate with the description of the SFR-enforcing abstractions described in the TOE design document.**

ADV_ARC.1-1 The evaluator ***shall examine*** the security architecture description to determine that the information provided in the evidence is presented at a level of detail commensurate with the descriptions of the SFR-enforcing abstractions contained in the functional specification and TOE design document.

16 With respect to the functional specification, the evaluator should ensure that the self-protection functionality described cover those effects that are evident at the TSFI. Such a description might include protection placed upon the executable images of the TSF, and protection placed on objects (e.g., files used by the TSF). The evaluator ensures that the functionality that might be invoked through the TSFI is described.

17 The evaluator ensures the security architecture description contains information on how any subsystems that contribute to TSF domain separation work.

18 This work unit fails if the security architecture description mentions any module, subsystem, or interface that is not described in the functional specification or TOE design document.

ADV_ARC.1.2C ***The security architecture description shall describe the security domains maintained by the TSF consistently with the SFRs.***

ADV_ARC.1-2 The evaluator ***shall examine*** the security architecture description to determine that it describes the security domains maintained by the TSF.

19 Security domains refer to environments supplied by the TSF for use by potentially-harmful entities; for example, a typical secure operating system supplies a set of resources (address space, per-process environment variables) for use by processes with limited access rights and security properties. The evaluator determines that the developer's description of the security domains takes into account all of the SFRs claimed by the TOE.

20 **If the POI_DATA group is included in the set of evaluated SFR, the security architecture description shall describe the security domains that result from the application separation principle (requirement CAS G2). It shall describe how isolation of payment application data is achieved, how the correct execution of the payment application is enforced as well as the management of Cardholder communication interface during payment application execution and how interference from other applications is avoided.**

21 **Except for PED ONLY configuration, which does not comprise POI_DATA group, application isolation principle shall be ensured. Especially for payment applications, the following requirements must be met, in conformance with the SFRs, FDP_ACC.1/POI_DATA Subset access control, FDP_ACF.1/POI_DATA Security attribute based access control, FDP_RIP.1/POI_DATA Subset residual information protection in [CC POI PP]:**

a) **The security of payment application in the TOE must not be impacted by any other application. Payment application isolation shall be ensured: no other application shall have unauthorized access to application data (Payment Transaction Data, TOE Management Data, TOE secret keys) (CAS G2.1).**

b) **The security of payment application in the TOE must not be impacted by any other application. Payment application isolation shall be ensured: it shall not be possible for another application to interfere with the execution of the payment application, by ac-**

cessing internal data (such as state machine or internal variables) (CAS G2.2).

c) **Payment application isolation shall be ensured: it shall not be possible for another application to deceive the Cardholder during execution of the payment application, by accessing Cardholder communication interface (e.g. display, beeper, printer) used by the payment application (CAS G2.3).**

~~For some TOEs such domains do not exist because all of the interactions available to users are severely constrained by the TSF. A packet-filter firewall is an example of such a TOE. Users on the LAN or WAN do not interact with the TOE, so there need be no security domains; there are only data structures maintained by the TSF to keep the users' packets separated. The evaluator ensures that any claim that there are no domains is supported by the evidence and that no such domains are, in fact, available~~.

ADV_ARC.1.3C ***The security architecture description shall describe how the TSF initialisation process is secure.***

ADV_ARC.1-3 The evaluator ***shall examine*** the security architecture description to determine that the initialisation process preserves security.

22 The information provided in the security architecture description relating to TSF initialisation is directed at the TOE components that are involved in bringing the TSF into an initial secure state (i.e. when all parts of the TSF are operational) when power-on or a reset is applied. This discussion in the security architecture description should list the system initialisation components and the processing that occurs in transitioning from the “down” state to the initial secure state.

23 It is often the case that the components that perform this initialisation function are not accessible after the secure state is achieved; if this is the case then the architectural design identifies the components and explains how they are not reachable by untrusted entities after the TSF has been established. In this respect, the property that needs to be preserved is that these components either 1) cannot be accessed by untrusted entities after the secure state is achieved, or 2) if they provide interfaces to untrusted entities, these TSFI cannot be used to tamper with the TSF.

24 The TOE components related to TSF initialisation, then, are treated themselves as part of the TSF, and analysed from that perspective. It should be noted that even though these are treated as part of the TSF, it is likely that a justification (as allowed by TSF internals (ADV_INT)) can be made that they do not have to meet the internal structuring requirements of ADV_INT.

ADV_ARC.1.4C ***The security architecture description shall demonstrate that the TSF protects itself from tampering.***

ADV_ARC.1-4 The evaluator ***shall examine*** the security architecture description to determine that it contains information sufficient to support a determination that the TSF is able to protect itself from tampering by untrusted active entities.

25 **In particular, the security architecture description shall demonstrate that:**

a) **PCI A2: If the PED or ICC reader permits access to internal areas (e.g. for service or maintenance), then it is not possible using this access area to insert a pin disclosing bug. Immediate access to sensitive data such as PIN or cryptographic data is either prevented by the design of the internal areas (e.g., by enclosing components with sensitive data into tamper resistant/responsive enclosures), or it has a mechanism so that access to internal areas causes the immediate erasure of sensitive data.**

b) **PCI A4: Sensitive functions or information are only used in the protected area(s) of the PED.**

c) **PCI A10: The design of the PED or ICC reader is such that it is not practical to construct a duplicate PED or ICC reader from commercially available components. For example, the casing used to house the device's electronic components is not commonly available.**

d) **PCI D1: It is not feasible to penetrate the IC Card Reader to make any additions, substitutions, or modifications to either the IC Card Reader's hardware or software, in order to determine or modify any sensitive data.**

e) **PCI D2.1: The slot of the ICC reader into which the IC card is inserted does not have sufficient space to hold a PIN-disclosing “bug” when a card is inserted, nor can it feasibly be enlarged to provide space for a PIN-disclosing “bug.” It is not possible for both an IC card and any other foreign object to reside within the card insertion slot. The slot of the ICC reader into which the IC card is inserted does not have sufficient space to hold a PIN-disclosing “bug” when a card is inserted, nor can it feasibly be enlarged to provide space for a PIN-disclosing “bug.” It is not possible for both an IC card and any other foreign object to reside within the card insertion slot.**

f) **PCI D2.2: The opening for the insertion of the IC card is in full view of the Cardholder during card insertion so that any untoward obstructions or suspicious objects at the opening are detectable.**

g) **PCI D3: The ICC reader is constructed so that wires running out of the slot of the IC Card Reader to a recorder or a transmitter (an external bug) can be observed by the Cardholder.**

26 ”Self-protection” refers to the ability of the TSF to protect itself from manipulation from external entities that may result in changes to the TSF. For TOEs that have dependencies on other IT entities, it is often the case that the TOE uses services supplied by the other IT entities in order to perform its functions. In such cases, the TSF alone does not protect itself because it depends on the other IT entities to provide some of the protection. For the purposes of the security architecture description, the notion of *self-protection* applies only to the services provided by the TSF through its TSFI **(the IC Card Reader and the PIN keypad are examples of POI-TSFI, see examples of self-protection below),** and not to services provided by underlying IT entities that it uses.

27 Self-protection is typically achieved by a variety of means, ranging from physical and logical restrictions on access to the TOE; to hardware-based means (e.g. **design or mechanism preventing access to internal areas of IC Card Reader in FPT_PHP.3/ICCardReader: PCI A2**); to software-based means (e.g. **authentication of each user before allowing access to sensitive services FIA_UAU.2/PIN_ENTRY**). The evaluator determines that all such mechanisms are described.

28 The evaluator determines that the design description covers how user input is handled by the TSF in such a way that the TSF does not subject itself to being corrupted by that user input. For example, the TSF might implement the notion of privilege and protect itself by using privileged-mode routines to handle user data. The TSF might make use of processor-based separation mechanisms such as privilege levels or rings. The TSF might implement software protection constructs or coding conventions that contribute to implementing separation of software domains, perhaps by delineating user address space from system address space. And the TSF might have reliance its environment to provide some support to the protection of the TSF.

29 All of the mechanisms contributing to the domain separation functions are described. The evaluator should use knowledge gained from other evidence (functional specification, TOE design, TSF internals description, other parts of the security architecture description, or implementation representation, as included in the assurance package for the TOE) in determining if any functionality contributing to self-protection was described that is not present in the security architecture description.

30 Accuracy of the description of the self-protection mechanisms is the property that the description faithfully describes what is implemented. The evaluator

should use other evidence (functional specification, TOE design, TSF Internals documentation, other parts of the security architectural description, as included in the ST for the TOE) in determining whether there are discrepancies in any descriptions of the self-protection mechanisms.. If an evaluator cannot understand how a certain self-protection mechanism works or could work in the system architecture, it may be the case that the description is not accurate.

ADV_ARC.1.5C ***The security architecture description shall demonstrate that the TSF prevents bypass of the SFR-enforcing functionality.***

ADV_ARC.1-5 The evaluator ***shall examine*** the security architecture description to determine that it presents an analysis that adequately describes how the SFR-enforcing mechanisms cannot be bypassed.

31 **In particular, the security architecture description shall demonstrate that:**

a) **PCI A1.2: Failure of a single security mechanism does not compromise PED security. Protection against a threat is based on a combination of at least two independent security mechanisms.**

b) **PCI A8.1: All prompts for non-PIN data entry are under the control of the cryptographic unit of the PED. If the prompts are stored inside the cryptographic unit, they cannot feasibly be altered without causing the erasure of the unit's cryptographic keys. If the prompts are stored outside the cryptographic unit, cryptographic mechanisms must exist to ensure the authenticity and the proper use of the prompts and that modification of the prompts or improper use of the prompts are prevented.**

32 Non-bypassability is a property that the security functionality of the TSF (as specified by the SFRs) is always invoked. For example, **for some TSF parts, TSF self test is asked with a given frequency (FPT_TST.1 instances). There must be no interface available to cancel the self test functionality or to specify another frequency than the one given at initialisation.** If access control to files is specified as a capability of the TSF via an SFR, there must be no interfaces through which files can be accessed without invoking the TSF's access control mechanism (such as an interface through which a raw disk access takes place).

33 Describing how the TSF mechanisms cannot be bypassed generally requires a systematic argument based on the TSF and the TSFIs. The description of how the TSF works (contained in the design decomposition evidence, such as the functional specification, TOE design documentation) - along with the information in the TSS - provides the background necessary for the evaluator to understand what resources are being protected and what security functions are being provided. The functional specification provides descriptions of the TSFIs through which the resources/functions are accessed.

34 The evaluator assesses the description provided (and other information provided by the developer, such as the functional specification) to ensure that no available interface can be used to bypass the TSF. This means that every available interface must be either unrelated to the SFRs that are claimed in the ST (and does not interact with anything that is used to satisfy SFRs) or else uses the security functionality that is described in other development evidence in the manner described. For example, a game would likely be unrelated to the SFRs, so there must be an explanation of how it cannot affect security. **Another example is the entering of prompts, in an integrated architecture comprising only one keypad. Prompts will be checked by the PED keypad security module, whether the data entered is the PIN or applicative information. Prompt control is detailed in this respect in the SFRs FDP_ACC.1/PEDPromptControl and FDP_ACF.1/PEDPromptControl so that prompts cannot be misused.** ~~Access to user data, however, is likely to be related to access control SFRs, so t~~The explanation would describe how the security functionality works when invoked through **the PED keypad**. Such a description is needed for every available interface.

35 An example of a description follows. **The TSF provide authentication means to users through secure channels (in order to update POI configuration or download a new application for instance). The TSFI here would be the secure channel authentication step. The evaluator should be able to determine from the vendor-provided description that this TSFI invokes the same protection mechanism whether, for instance, the authentication step is accessed via an USB device or whether the authentication step is accessed via Wi-Fi if the POI is attended remotely.** ~~Suppose the TSF provides file protection. Further suppose that although the "traditional" system call TSFIs for open, read, and write invoke the file protection mechanism described in the TOE design, there exists a TSFI that allows access to a batch job facility (creating batch jobs, deleting jobs, modifying unprocessed jobs). The evaluator should be able to determine from the vendor-provided description that this TSFI invokes the same protection mechanisms as do the "traditional" interfaces. This could be done, for example, by referencing the appropriate sections of the TOE design that discuss *how* the batch job facility TSFI achieves its security objectives.~~

36 Using this same example, suppose there is a TSFI whose sole purpose is to display the time of day. The evaluator should determine that the description adequately argues that this TSFI is not capable of manipulating any protected resources and should not invoke any security functionality.

37 Another example of bypass is when the TSF is supposed to maintain confidentiality of a cryptographic key (one is allowed to use it for cryptographic operations, but is not allowed to read/write it). If an attacker has direct physical access to the device, he might be able to examine side-channels such as the power usage of the device, the exact timing of the

device, or even any electromagnetic emanations of the device and, from this, infer the key.

38 If such side-channels may be present, the demonstration should address the mechanisms that prevent these side-channels from occurring, such as random internal clocks, dual-line technology etc. Verification of these mechanisms would be verified by a combination of purely design-based arguments and testing.

39 The evaluator should also ensure that the description is comprehensive, in that each interface is analysed with respect to the entire set of claimed SFRs. This may require the evaluator to examine supporting information (functional specification, TOE design, other parts of the security architectural description, operational user guidance, and perhaps even the implementation representation, as provided for the TOE) to determine that the description has correctly capture all aspects of an interface. The evaluator should consider what SFRs each TSFI might affect (from the description of the TSFI and its implementation in the supporting documentation), and then examine the description to determine whether it covers those aspects.

40 **With respect to the [CC POI PP], an example for a generic list of TSF parts with their corresponding groups of TSFIs is provided below:**

a) MSR TSFIs: MSR

b) PED Middle TSF TSFIs: PED prompt control (e.g. the PED keypad), the IC Card Reader, the Display

c) Middle TSF TSFIs: secure channel for payment application updates and management

d) Core TSF TSFIs: PED keypad restricted to PIN entry and secure channel for PIN transfer and secure channel for PIN related data (except the keys) download and update

e) Core TSF keys TSFIs: secure channel to update and initialize the PIN keys

This list is not exhaustive, it represents a minimal set of TSF parts and groups of TSFIs usable in a POI PP conformant product.

3.2 Functional specification (ADV_FSP)

3.2.1 Evaluation of sub-activity (ADV_FSP.2)

3.2.1.1 Objectives

41 The objective of this sub-activity is to determine whether the developer has provided a description of the TSFIs in terms of their purpose, method of use, and parameters. In addition, the SFR-enforcing actions, results and error messages of each TSFI that is SFR-enforcing are also described.

3.2.1.2 Input

42 The evaluation evidence for this sub-activity that is required by the work-units is:

a) the ST;

b) the functional specification;

c) the TOE design.

43 The evaluation evidence for this sub-activity that is used if included in the ST for the TOE is:

a) the security architecture description;

b) the operational user guidance;

3.2.1.3 Action ADV_FSP.2.1E

ADV_FSP.2.1C ***The functional specification shall completely represent the TSF.***

ADV_FSP.2-1 The evaluator shall examine the functional specification to determine that the TSF is fully represented.

44 The identification of the TSFI is a necessary prerequisite to all other activities in this sub-activity. The TSF must be identified (done as part of the TOE design (ADV_TDS) work units) in order to identify the TSFI. This activity can be done at a high level to ensure that no large groups of interfaces have been missed **(interfaces with ICC, Cardholder Verification devices, magstripe reader, terminal management system, acquirer system and other local devices if any**) ~~network protocols, hardware interfaces, configuration files)~~, or at a low level as the evaluation of the functional specification proceeds.

45 In making an assessment for this work unit, the evaluator determines that all portions of the TSF are addressed in terms of the interfaces listed in the functional specification. All portions of the TSF should have a corresponding interface description, or if there are no corresponding interfaces for a portion of the TSF, the evaluator determines that that is acceptable.

ADV_FSP.2.2C ***The functional specification shall describe the purpose and method of use for all TSFI.***

ADV_FSP.2-2 The evaluator ***shall examine*** the functional specification to determine that it states the purpose of each TSFI.

46 The purpose of a TSFI is a general statement summarising the functionality provided by the interface. It is not intended to be a complete statement of the actions and results related to the interface, but rather a statement to help the reader understand in general what the interface is intended to be used for. The evaluator should not only determine that the purpose exists, but also that it accurately reflects the TSFI by taking into account other information about the interface, such as the description of actions and error messages.

47 For instance, if a login and password are asked while using the secure channel for updating data or keys, the form expected for the login and password may be specified: first letter of name plus full first name for login and at least one number and one special character for a password of 8 characters for the password; along with the number of tries, an error message ("your login or password is incorrect, please try again, you have performed one of the three permitted tries"), etc…

ADV_FSP.2-3 The evaluator ***shall examine*** the functional specification to determine that the method of use for each TSFI is given.

48 The method of use for a TSFI summarises how the interface is manipulated in order to invoke the actions and obtain the results associated with the TSFI. The evaluator should be able to determine, from reading this material in the functional specification, how to use each interface. This does not necessarily mean that there needs to be a separate method of use for each TSFI, as it may be possible to describe in general how kernel calls are invoked, for instance, and then identify each interface using that general style. Different types of interfaces will require different method of use specifications. APIs, network protocol interfaces, system configuration parameters, and hardware bus interfaces all have very different methods of use, and this should be taken into account by the developer when developing the functional specification, as well as by the evaluator evaluating the functional specification.

49 For administrative interfaces whose functionality is documented as being inaccessible to untrusted users, the evaluator ensures that the method of making the functions inaccessible is described in the functional specification. It

should be noted that this inaccessibility needs to be tested by the developer in their test suite.

50 The evaluator should not only determine that the set of method of use descriptions exist, but also that they accurately cover each TSFI.

51 Below are listed examples of sets of TSFI corresponding to each of the three POI configurations described in the [CC POI PP] (table 1):

a) **PED ONLY TSFI: MSR, PED Keypad, IC Card Reader, Display, a secure channel (for PIN transfer, update of PIN keys, download of PIN related data)**

b) **POI-COMPREHENSIVE TSFI: MSR, PED Keypad, IC Card Reader, Display, a secure channel (for PIN transfer, update of PIN keys, download of PIN related data) and the Middle TSF TSFI, i.e. a secure channel for download and update of payment application**

c) **POI-OPTION TSFI (e.g. POI-COMPREHENSIVE TSFI minus MSR): PED Keypad, IC Card Reader, Display, a secure channel (for PIN transfer, update of PIN keys, download of PIN related data) and Middle TSF TSFI, i.e. a secure channel for download and update of payment application**

52 Note that two types of secure channels are listed among these examples of TSFI: a "PIN" dedicated secure channel and a "payment application" secure channel.

ADV_FSP.2.3C ***The functional specification shall identify and describe all parameters associated with each TSFI.***

ADV_FSP.2-4 The evaluator ***shall examine*** the presentation of the TSFI to determine that it completely identifies all parameters associated with every TSFI.

53 The evaluator examines the functional specification to ensure that all of the parameters are described for each TSFI. Parameters are explicit inputs or outputs to an interface that control the behaviour of that interface. For examples, parameters are the arguments supplied to an API; the various fields in packet for a given network protocol; the individual key values in the Windows Registry; the signals across a set of pins on a chip; **the digits numbers to provide upon PIN entry;** etc.

54 In order to determine that all of the parameters are present in the TSFI, the evaluator should examine the rest of the interface description (actions, error messages, etc.) to determine if the effects of the parameter are accounted for in the description. The evaluator should also check other evidence provided

for the evaluation (e.g., TOE design, security architecture description, operational user guidance, implementation representation) to see if behaviour or additional parameters are described there but not in the functional specification.

ADV_FSP.2-5 The evaluator ***shall examine*** the presentation of the TSFI to determine that it completely and accurately describes all parameters associated with every TSFI.

55 Once all of the parameters have been identified, the evaluator needs to ensure that they are accurately described, and that the description of the parameters is complete. A parameter description tells what the parameter is in some meaningful way. For instance, the interface foo(i) could be described as having “parameter i which is an integer"; this is not an acceptable parameter description. A description such as “parameter i is an integer that indicates the number of users currently logged in to the system” is much more acceptable.

56 In order to determine that the description of the parameters is complete, the evaluator should examine the rest of the interface description (purpose, method of use, actions, error messages, etc.) to determine if the descriptions of the parameter(s) are accounted for in the description. The evaluator should also check other evidence provided (e.g., TOE design, architectural design, operational user guidance, implementation representation) to see if behaviour or additional parameters are described there but not in the functional specification.

ADV_FSP.2.4C ***For each SFR-enforcing TSFI, the functional specification shall describe the SFR-enforcing actions associated with the TSFI.***

ADV_FSP.2-6 The evaluator ***shall examine*** the presentation of the TSFI to determine that it completely and accurately describes the SFR-enforcing actions associated with the SFR-enforcing TSFIs.

57 If an action available through an interface can be traced to one of the SFRs levied on the TSF, then that interface is SFR-enforcing. Such policies are not limited to the access control policies, but also refer to any functionality specified by one of the SFRs contained in the ST. Note that it is possible that an interface may have various actions and results, some of which may be SFR-enforcing and some of which may not.

58 The developer is not required to “label” interfaces as SFR-enforcing, and likewise is not required to identify actions available through an interface as SFR-enforcing. It is the evaluator's responsibility to examine the evidence provided by the developer and determine that the required information is present. In the case where the developer has identified the SFR-enforcing TSFI and SFR-enforcing actions available through those TSFI, the evaluator must judge completeness and accuracy based on other information supplied for the evaluation (e.g., TOE design, security architecture description, operational

user guidance), and on the other information presented for the interfaces (parameters and parameter descriptions, error messages, etc.).

59 In this case (where the developer has provided only the SFR-enforcing information for SFR-enforcing TSFI) the evaluator also ensures that no interfaces have been mis-categorised. This is done by examining other information supplied for the evaluation (e.g., TOE design, security architecture description, operational user guidance), and the other information presented for the interfaces (parameters and parameter descriptions, for example) not labelled as SFR-enforcing.

60 In the case where the developer has provided the same level of information on all interfaces, the evaluator performs the same type of analysis mentioned in the previous paragraphs. The evaluator should determine which interfaces are SFR-enforcing and which are not, and subsequently ensure that the SFR-enforcing aspects of the SFR-enforcing actions are appropriately described.

61 The SFR-enforcing actions are those that are visible at any external interface and that provide for the enforcement of the SFRs being claimed. For example, if audit requirements are included in the ST, then audit-related actions would be SFR-enforcing and therefore must be described, even if the result of that action is generally not visible through the invoked interface (as is often the case with audit, where a user action at one interface would produce an audit record visible at another interface).

62 The level of description that is required is that sufficient for the reader to understand what role the TSFI actions play with respect to the SFR. The evaluator should keep in mind that the description should be detailed enough to support the generation (and assessment) of test cases against that interface. If the description is unclear or lacking detail such that meaningful testing cannot be conducted against the TSFI, it is likely that the description is inadequate.

ADV_FSP.2.5C ***For each SFR-enforcing TSFI, the functional specification shall describe direct error messages resulting from processing associated with the SFR-enforcing actions.***

ADV_FSP.2-7 The evaluator ***shall examine*** the presentation of the TSFI to determine that it completely and accurately describes error messages that may result from SFR-enforcing actions associated with each SFR-enforcing TSFI.

63 This work unit should be performed in conjunction with, or after, work unit ADV_FSP.2-6 in order to ensure the set of SFR-enforcing TSFI and SFR-enforcing actions is correctly identified. The developer may provide more information than is required (for example, all error messages associated with each interface), in which the case the evaluator should restrict their assess-

ment of completeness and accuracy to only those that they determine to be associated with SFR-enforcing actions of SFR-enforcing TSFI.

64 Errors can take many forms, depending on the interface being described. For an API, the interface itself may return an error code, set a global error condition, or set a certain parameter with an error code. For a configuration file, an incorrectly configured parameter may cause an error message to be written to a log file. For a hardware PCI card, an error condition may raise a signal on the bus, or trigger an exception condition to the CPU.

65 Errors (and the associated error messages) come about through the invocation of an interface. The processing that occurs in response to the interface invocation may encounter error conditions, which trigger (through an implementation-specific mechanism) an error message to be generated. In some instances this may be a return value from the interface itself; in other instances a global value may be set and checked after the invocation of an interface. It is likely that a TOE will have a number of low-level error messages that may result from fundamental resource conditions, such as “disk full” or “resource locked”. While these error messages may map to a large number of TSFI, they could be used to detect instances where detail from an interface description has been omitted. For instance, a TSFI that produces a “disk full” message, but has no obvious description of why that TSFI should cause an access to the disk in its description of actions, might cause the evaluator to examine other evidence (Security Architecture (ADV_ARC), TOE design (ADV_TDS)) related that TSFI to determine if the description is accurate.

66 In order to determine that the description of the error messages of a TSFI is accurate and complete, the evaluator measures the interface description against the other evidence provided for the evaluation (e.g., TOE design, security architecture description, operational user guidance), as well as other evidence available for that TSFI (parameters, analysis from work unit ADV_FSP.2-6).

ADV_FSP.2.6C ***The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.***

ADV_FSP.2-8 The evaluator ***shall check*** that the tracing links the SFRs to the corresponding TSFIs.

67 The tracing is provided by the developer to serve as a guide to which SFRs are related to which TSFIs. This tracing can be as simple as a table; it is used as input to the evaluator for use in the following work units, in which the evaluator verifies its completeness and accuracy.

3.2.1.4 Action ADV_FSP.2.2E

ADV_FSP.2-9 The evaluator ***shall examine*** the functional specification to determine that it is a complete instantiation of the SFRs.

68 To ensure that all SFRs are covered by the functional specification, as well as the test coverage analysis, the evaluator may build upon the developer's tracing (see ADV_FSP.2-8 a map between the TOE security functional requirements and the TSFI. Note that this map may have to be at a level of detail below the component or even element level of the requirements, because of operations (assignments, refinements, selections) performed on the functional requirement by the ST author.

69 For example, the FDP_ACC.1 component contains an element with assignments. If the ST contained, for instance, ten rules in the FDP_ACC.1 assignment, and these ten rules were covered by three different TSFI, it would be inadequate for the evaluator to map FDP_ACC.1 to TSFI A, B, and C and claim they had completed the work unit. Instead, the evaluator would map FDP_ACC.1 (rule 1) to TSFI A; FDP_ACC.1 (rule 2) to TSFI B; etc. It might also be the case that the interface is a wrapper interface (e.g., IOCTL), in which case the mapping would need to be specific to certain set of parameters for a given interface.

70 The evaluator must recognise that for requirements that have little or no manifestation at the TSF boundary (e.g., FDP_RIP) it is not expected that they completely map those requirements to the TSFI. The analysis for those requirements will be performed in the analysis for the TOE design (ADV_TDS) when included in the ST. It is also important to note that since the parameters, actions, and error messages associated with TSFIs must be fully specified, the evaluator should be able to determine if all aspects of an SFR appear to be implemented at the interface level.

ADV_FSP.2-10 The evaluator ***shall examine*** the functional specification to determine that it is an accurate instantiation of the SFRs.

71 For each functional requirement in the ST that results in effects visible at the TSF boundary, the information in the associated TSFI for that requirement specifies the required functionality described by the requirement. For example, if the ST contains **more than one instantiation of the** requirement **for a specific inter-TSF trusted channel (FTP_ITC.1), that state several communication initiators (FTP_ITC.1.2) for several types of secure channels and these secure channels or their authorized initiators are not specifically addressed in the functional specification, then the functional specification is not accurate with respect to the requirements.** ~~for access control lists, and the only TSFI that map to that requirement specify functionality for Unix-style protection bits, then the functional specification is not accurate with respect to the requirements.~~

72 The evaluator must recognise that for requirements that have little or no manifestation at the TSF boundary (e.g., FDP_RIP) it is not expected that the evaluator completely map those requirements to the TSFI. The analysis for those requirements will be performed in the analysis for the TOE design (ADV_TDS) when included in the ST.

3.3 TOE design (ADV_TDS)

3.3.1 Evaluation of sub-activity (ADV_TDS.1)

3.3.1.1 Input

73 The evaluation evidence for this sub-activity is:

a) the ST;

b) the functional specification;

c) security architecture description;

d) the TOE design.

3.3.1.2 Action ADV_TDS.1.1E

ADV_TDS.1.1C ***The design shall describe the structure of the TOE in terms of subsystems.***

ADV_TDS.1-1 The evaluator ***shall examine*** the TOE design to determine that the structure of the entire TOE is described in terms of subsystems.

74 The evaluator ensures that all of the subsystems of the TOE are identified. This description of the TOE will be used as input to work unit ADV_TDS.1-2, where the parts of the TOE that make up the TSF are identified. That is, this requirement is on the entire TOE rather than on only the TSF.

75 The TOE (and TSF) may be described in multiple layers of abstraction (i.e. subsystems and modules) Depending upon the complexity of the TOE, its design may be described in terms of subsystems and modules, as described in CC Part 3 Annex A.4, ADV_TDS: Subsystems and Modules. At this level of assurance, the decomposition only need be at the “subsystem” level.

76 In performing this activity, the evaluator examines other evidence presented for the TOE (e.g., ST, operator user guidance) to determine that the description of the TOE in such evidence is consistent with the description contained in the TOE design.

ADV_TDS.1.2C ***The design shall identify all subsystems of the TSF.***

ADV_TDS.1-2 The evaluator ***shall examine*** the TOE design to determine that all subsystems of the TSF are identified.

77 In work unit ADV_TDS.1-1 all of the subsystems of the TOE were identified, and a determination made that the non-TSF subsystems were correctly characterised. Building on that work, the subsystems that were not characterised as non-TSF subsystems should be precisely identified. The evaluator determines that, of the hardware and software installed and configured according to the Preparative procedures (AGD_PRE) guidance, each subsystem has been accounted for as either one that is part of the TSF, or one that is not.

78 The description of the TSF subsystems shall be described in sufficient details such that they can be linked to one of the following TSF parts: CoreTSF, CoreTSFKeys, PED MiddleTSF, Middle TSF, MSR (as an option).

79 Additionaly, the description may include external IT entities that the POI interacts with such as:

80 Application/Acquirer System: Entity operated by the Application Provider resp. Acquirer or the Acquirer Processor with whom the POI exchanges transaction data.

81 Terminal Management System: Entity used to administrate (installation, maintenance) a set of POIs. It is used by the Terminal Administrator.

82 Local Devices: Any device that is not a peripheral device and that either inputs or outputs payment transaction data. Examples of Local Devices are the Electronic Cash Register (ECR), a Vending Machine Controller or a Pump Controller for Petrol Outdoor Configurations. The connections to these external devices may be implemented by various means such as private or public network, etc.

ADV_TDS.1.3C ***The design shall describe the behaviour of each SFR-supporting or SFR-non-interfering TSF subsystem in sufficient detail to determine that it is not SFR-enforcing.***

ADV_TDS.1-3 The evaluator ***shall examine*** the TOE design to determine that each SFR-supporting or SFR-non-interfering subsystem of the TSF is described such that the evaluator can determine that the subsystem is SFR-supporting or SFR-non-interfering.

83 SFR-supporting and SFR-non-interfering subsystems do not need to be described in detail as to how they function in the system. However, the evaluator makes a determination, based on the evidence provided by the developer, that the subsystems that do not have high-level descriptions are SFR-supporting or SFR-non-interfering. Note that if the developer provides a uniform level of detailed documentation then this work unit will be largely satisfied, since the point of categorising the subsystems is to allow the developer

to provide less information for SFR-supporting and SFR-non-interfering subsystems than for SFR-enforcing subsystems.

84 An SFR-supporting subsystem is one that is depended on by an SFR-enforcing subsystem in order to implement an SFR, but does not play as direct a role as an SFR-enforcing subsystem. An SFR-non-interfering subsystem is one that is not depended upon, in either a supporting or enforcing role, to implement an SFR.

85 **In a POI-type TOE, it means that the descriptions must detail the TSF parts which are the POI SFR-enforcing abstractions. The subsystems may correspond to these entities (Core TSF, Middle TSF, PED Middle TSF, etc…). They also may be smaller parts of the TSF abstractions. The Core TSF for instance may easily be separated in smaller entities. The definition of subsystems is up to the developer and as long as the rationale provided is consistent, subsystems can be smaller subsets of TSF parts (the keypad may be a subsystem for instance). The subsystems may vary depending on the POI configuration and the security features they embed (see table 1 in [CC POI PP]). Each Security Target based on [CC POI PP] shall detail precisely the security features corresponding to the POI configuration applying to the ST TOE.**

ADV_TDS.1.4C ***The design shall summarise the SFR-enforcing behaviour of the SFR-enforcing subsystems.***

ADV_TDS.1-4 The evaluator ***shall examine*** the TOE design to determine that it provides a complete, accurate, and high-level description of the SFR-enforcing behaviour of the SFR-enforcing subsystems.

86 The developer may designate subsystems as SFR-enforcing, SFR-supporting, and SFR non-interfering, but these “tags” are used only to describe the amount and type of information the developer must provide, and can be used to limit the amount of information the developer has to develop if their engineering process does not produce the documentation required. Whether the subsystems have been categorised by the developer or not, it is the evaluator's responsibility to determine that the subsystems have the appropriate information for their role (SFR-enforcing, etc.) in the TOE, and to obtain the appropriate information from the developer should the developer fail to provide the required information for a particular subsystem.

87 SFR-enforcing behaviour refers to how a subsystem provides the functionality that implements an SFR. A high-level description need not refer to specific data structures (although it may), but instead talks about more general data flow, message flow, and control relationships within a subsystem. The goal of these descriptions is to give the evaluator enough information to understand how the SFR-enforcing behaviour is achieved. Note that the evaluator should find unacceptable asserts of SFR-enforcement in the TOE design

documentation for this work unit. It should be noted that it is the evaluator's determination with respect to what “high-level” means for a particular TOE, and the evaluator obtains enough information from the developer to make a sound verdict for this work unit.

88 **Here follow several examples relevant to this evaluation task:**

a) **An example is when the PED embeds a functionality that causes the PED to become immediately inoperable (PCI A1.1 in FPT_PHP.3) in erasing any secret which may be stored in the PED (PIN, secret cryptographic keys, administration passwords, etc.). The PED, if it permits access to internal areas (e.g., for service or maintenance) prevents access to internal data such as PIN or cryptographic data by the design of these areas or by a mechanism which causes immediate erasure of sensitive data.**

b) **The subsystem which provides the TRNG (True Random Generator) shall be identified. The description shall include details on the information data processing that causes the subsystem to generate a TRNG which should demonstrate that the TRNG generates numbers sufficiently impredictable (PCI B9 in FCS_RND.1).**

c) The subsystem which provides PIN encipherment shall be identified, demonstrating that it enforces the FCS_COP.1 SFR, stating precisely which cryptographic algorithm is used for the PIN cipherment/decipherment, with its corresponding padding and/or parameters and the conformance to the standard ISO 9564 clearly established.

Note that the subsystem for the TRNG and the one for the PIN cipherment can be the same unique subsystem or that they can be two separated subsystems, depending on the decomposition chosen by the developer.

d) **Describing in which manner the Middle TSF is SFR-enforcing, through the secure channel and the update of POI software or applicative software. A description of how the authentication is performed at each end of the secure channel and how data is exchanged, stored and used by the Middle TSF.**

89 To determine completeness and accuracy, the evaluator examines other information available (e.g., functional specification, security architecture description, implementation representation). Descriptions of functionality in these documents should be consistent with what is provided for evidence for this work unit

ADV_TDS.1.5C ***The design shall provide a description of the interactions among SFR-enforcing subsystems of the TSF, and between the SFR-enforcing subsystems of the TSF and other subsystems of the TSF.***

ADV_TDS.1-5 The evaluator ***shall examine*** the TOE design to determine that interactions between the subsystems of the TSF are described.

90 The goal of describing the interactions between the SFR-enforcing subsystems and other subsystems is to help provide the reader a better understanding of how the TSF performs it functions. These interactions do not need to be characterised at the implementation level (e.g., parameters passed from one routine in a subsystem to a routine in a different subsystem; global variables; hardware signals (e.g., interrupts) from a hardware subsystem to an interrupt-handling subsystem), but the data elements identified for a particular subsystem that are going to be used by another subsystem need to be covered in this discussion. Any control relationships between subsystems (e.g., a subsystem responsible for configuring a rule base for a firewall system and the subsystem that actually implements these rules) should also be described.

91 The evaluators need to use their own judgement in assessing the completeness of the description. If the reason for an interaction is unclear, or if there are SFR-related interactions (discovered, for instance, in examining the descriptions of subsystem behaviour) that do not appear to be described, the evaluator ensures that this information is provided by the developer. However, if the evaluator can determine that interactions among a particular set of subsystems, while incompletely described by the developer, will not aid in understanding the overall functionality nor security functionality provided by the TSF, then the evaluator may choose to consider the description sufficient, and not pursue completeness for its own sake.

ADV_TDS.1.6C ***The mapping shall demonstrate that all TSFIs trace to the behaviour described in the TOE design that they invoke.***

ADV_TDS.1-6 The evaluator ***shall examine*** the TOE design to determine that it contains a complete and accurate mapping from the TSFI described in the functional specification to the subsystems of the TSF described in the TOE design.

92 The subsystems described in the TOE design provide a description of how the TSF works at a detailed level for SFR-enforcing portions of the TSF, and at a higher level for other portions of the TSF. The TSFI provide a description of how the implementation is exercised. The evidence from the developer identifies the subsystem that is initially involved when an operation is requested at the TSFI, and identify the various subsystems that are primarily responsible for implementing the functionality. Note that a complete “call tree” for each TSFI is not required for this work unit.

93 The evaluator assesses the completeness of the mapping by ensuring that all of the TSFI map to at least one subsystem. The verification of accuracy is more complex.

94 The first aspect of accuracy is that each TSFI is mapped to a subsystem at the TSF boundary. This determination can be made by reviewing the subsystem description and interactions, and from this information determining its place in the architecture. The next aspect of accuracy is that the mapping makes sense. For instance, mapping a TSFI dealing with access control to a subsystem that checks passwords is not accurate. The evaluator should again use judgement in making this determination. The goal is that this information aids the evaluator in understanding the system and implementation of the SFRs, and ways in which entities at the TSF boundary can interact with the TSF. The bulk of the assessment of whether the SFRs are described accurately by the subsystems is performed in other work units.

11.8.1.3 Action ADV_TDS.1.2E

ADV_TDS.1-7 The evaluator ***shall examine*** the TOE security functional requirements and the TOE design, to determine that all ST security functional requirements are covered by the TOE design.

95 The evaluator may construct a map between the TOE security functional requirements and the TOE design. This map will likely be from a functional requirement to a set of subsystems. Note that this map may have to be at a level of detail below the component or even element level of the requirements, because of operations (assignments, refinements, selections) performed on the functional requirement by the ST author.

96 The set of subsystems are at least among the following ones: CoreTSF, CoreTSFKeys, PED MiddleTSF, Middle TSF, MSR (as an option). The mapping may so be applied from SFR groups and/or individual SFR. to the set of subsystems above.

97 Note that if the TOE decomposition in subsystems is more detailed than the one in TSF parts, it should still be consistent with table 13 (SFR packages in each PP configuration) in [CC POI PP].

98 For example, the FDP_ACC.1 Subset access control component contains an element with assignments. If the ST contained, for instance, ten rules in the FDP_ACC.1 Subset access control assignment, and these ten rules were implemented in specific places within fifteen modules, it would be inadequate for the evaluator to map FDP_ACC.1 Subset access control to one subsystem and claim the work unit had been completed. Instead, the evaluator would map FDP_ACC.1 Subset access control (rule 1) to subsystem A, behaviours x, y, and z; FDP_ACC.1 Subset access control (rule 2) to subsystem A, behaviours x, p, and q; etc.

ADV_TDS.1-8 The evaluator ***shall examine*** the TOE design to determine that it is an accurate instantiation of all security functional requirements.

99 The evaluator ensures that each security requirement listed in the TOE security functional requirements section of the ST has a corresponding design de-

scription in the TOE design that accurately details how the TSF meets that requirement. This requires that the evaluator identify a collection of subsystems that are responsible for implementing a given functional requirement, and then examine those subsystems to understand how the requirement is implemented. Finally, the evaluator would assess whether the requirement was accurately implemented.

100 As an example, if the ST requirements specified a role-based access control mechanism, the evaluator would first identify the subsystems that contribute to this mechanism's implementation. This could be done by in-depth knowledge or understanding of the TOE design or by work done in the previous work unit. Note that this trace is only to identify the subsystems, and is not the complete analysis.

101 The next step would be to understand what mechanism the subsystems implemented. For instance, if the design described an implementation of access control based on UNIX-style protection bits, the design would not be an accurate instantiation of those access control requirements present in the ST example used above. If the evaluator could not determine that the mechanism was accurately implemented because of a lack of detail, the evaluator would have to assess whether all of the SFR-enforcing subsystems have been identified, or if adequate detail had been provided for those subsystems.

4 Class AGD: Guidance documents

4.1 Operational user guidance (AGD_OPE)

4.1.1 Evaluation of sub-activity (AGD_OPE.1)

4.1.1.1 Objectives

102 The objectives of this sub-activity are to determine whether the user guidance describes for each user role the security functionality and interfaces provided by the TSF, provides instructions and guidelines for the secure use of the TOE, addresses secure procedures for all modes of operation, facilitates prevention and detection of insecure TOE states, or whether it is misleading or unreasonable.

4.1.1.2 Input

103 The evaluation evidence for this sub-activity is:

a) the ST;

b) the functional specification;

c) the TOE design, if applicable;

d) the user guidance;

4.1.1.3 Action AGD_OPE.1.1E

AGD_OPE.1.1C ***The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.***

AGD_OPE.1-1 The evaluator ***shall examine*** the operational user guidance to determine that it describes, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

104 The configuration of the TOE may allow different user roles to have dissimilar privileges in making use of the different functions of the TOE. This means that some users are authorised to perform certain functions, while other users may not be so authorised. These functions and privileges should be described, for each user role, by the user guidance.

105 The user guidance identifies, for each user role, the functions and privileges that must be controlled, the types of commands required for them, and the reasons for such commands. The user guidance should contain warnings regarding the use of these functions and privileges. Warnings should address expected effects, possible side effects, and possible interactions with other functions and privileges.

106 User roles defined in the [CC POI PP] are the following: Cardholder, Attendant, Merchant, Terminal Administrator, Acquirer System, Terminal Management System, IC Card, Magnetic Stripe Card, Local Device, Payment application. The repartition of user roles in function of the POI configuration is detailed in table 4 of [CC POI PP].

AGD_OPE.1.2C ***The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.***

AGD_OPE.1-2 The evaluator ***shall examine*** the operational user guidance to determine that it describes, for each user role, the secure use of the available interfaces provided by the TOE.

107 The user guidance should provide advice regarding effective use of the TSF (e.g. reviewing password composition practises, suggested frequency of user file backups, discussion on the effects of changing user access privileges).

AGD_OPE.1.3C ***The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.***

AGD_OPE.1-3 The evaluator ***shall examine*** the operational user guidance to determine that it describes, for each user role, the available security functionality and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

108 The user guidance should contain an overview of the security functionality that is visible at the user interfaces.

109 The user guidance should identify and describe the purpose, behaviour, and interrelationships of the security interfaces and functionality.

110 For each user-accessible interface, the user guidance should:

a) describe the method(s) by which the interface is invoked (e.g. command-line, programming-language system call, menu selection, command button);

b) describe the parameters to be set by the user, their particular purposes, valid and default values, and secure and insecure use settings of such parameters, both individually or in combination;

c) describe the immediate TSF response, message, or code returned.

111 The evaluator should consider the functional specification and the ST to determine that the TSF described in these documents is consistent to the operational user guidance. The evaluator has to ensure that the operational user guidance is complete to allow the secure use through the TSFI available to all types of human users. The evaluator may, as an aid, prepare an informal mapping between the guidance and these documents. Any omissions in this mapping may indicate incompleteness.

AGD_OPE.1.4C ***The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.***

AGD_OPE.1-4 The evaluator ***shall examine*** the operational user guidance to determine that it describes, for each user role, each type of security-relevant event relative to the user functions that need to be performed, including changing the security characteristics of entities under the control of the TSF and operation following failure or operational error.

112 All types of security-relevant events are detailed for each user role, such that each user knows what events may occur and what action (if any) he may have to take in order to maintain security. Security-relevant events that may occur during operation of the TOE (e.g. audit trail overflow, system crash, updates to user records, such as when a user account is removed when the user leaves the organisation) are adequately defined to allow user intervention to maintain secure operation.

AGD_OPE.1.5C ***The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.***

AGD_OPE.1-5 The evaluator ***shall examine*** the operational user guidance and other evaluation evidence to determine that the guidance identifies all possible modes of operation of the TOE (including, if applicable, operation following failure or operational error), their consequences and implications for maintaining secure operation.

113 The operational user guidance shall identify all possible modes of operation of the TOE, i.e. data on production and personalisation, physical and chronological whereabouts, repair and maintenance, removal from operation and loss or theft. It shall identify their consequences and implications for maintaining secure operation (CAS F5).

114 Moreover, the opening for the insertion of the IC Card is in full view of the Cardholder during card insertion so that any untoward obstructions or suspicious objects at the opening are detectable (PCI D2.2).

115 Other evaluation evidence, particularly the functional specification, provide an information source that the evaluator should use to determine that the guidance contains sufficient guidance information.

116 If test documentation is included in the assurance package, then the information provided in this evidence can also be used to determine that the guidance contains sufficient guidance documentation. The detail provided in the test steps can be used to confirm that the guidance provided is sufficient for the use and administration of the TOE.

117 The evaluator should focus on a single human visible TSFI at a time, comparing the guidance for securely using the TSFI with other evaluation evidence, to determine that the guidance related to the TSFI is sufficient for the secure usage (i.e. consistent with the SFRs) of that TSFI. The evaluator should also consider the relationships between interfaces, searching for potential conflicts.

AGD_OPE.1.6C ***The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.***

AGD_OPE.1-6 The evaluator ***shall examine*** the operational user guidance to determine that it describes, for each user role, the security measures to be followed in order to fulfil the security objectives for the operational environment as described in the ST.

118 The evaluator analyses the security objectives for the operational environment in the ST and determines that for each user role, the relevant security measures are described appropriately in the user guidance.

119 The security measures described in the user guidance should include all relevant external procedural, physical, personnel and connectivity measures.

120 Note that those measures relevant for secure installation of the TOE are examined in Preparative procedures (AGD_PRE).

AGD_OPE.1.7C ***The operational user guidance shall be clear and reasonable.***

AGD_OPE.1-7 The evaluator ***shall examine*** the operational user guidance to determine that it is clear.

121 The guidance is unclear if it can reasonably be misconstrued by an administrator or user, and used in a way detrimental to the TOE, or to the security provided by the TOE.

AGD_OPE.1-8 The evaluator ***shall examine*** the operational user guidance to determine that it is reasonable.

122 The guidance is unreasonable if it makes demands on the TOE's usage or operational environment that are inconsistent with the ST or unduly onerous to maintain security.

4.2 Preparative procedures (AGD_PRE)

4.2.1 Evaluation of sub-activity (AGD_PRE.1)

4.2.1.1 Objectives

123 The objective of this sub-activity is to determine whether the procedures and steps for the secure preparation of the TOE have been documented and result in a secure configuration.

4.2.1.2 Input

124 The evaluation evidence for this sub-activity is:

a) the ST;

b) the TOE including its preparative procedures;

c) the description of developer's delivery procedures, if applicable;

4.2.1.3 Application notes

125 The preparative procedures refer to all acceptance and installation procedures, that are necessary to progress the TOE to the secure configuration as described in the ST.

126 **Developing and manufacturing of the TOE are part of the developer phase. During the developer phase, at least the initial cryptographic keys are loaded. If required other cryptographic keys may be loaded into the POI during user phase. Additionally, cryptographic keys can also be loaded during the user phase. The Security Target author shall define precisely where the developer phase ends and where the user phase begins in relation to cryptographic key loading.**

127 **Thus, the scope of this sub-activity may vary depending on the technical choices made by the developer and specified by the Security Target.**

4.2.1.4 Action AGD_PRE.1.1E

AGD_PRE.1.1C ***The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.***

AGD_PRE.1-1 The evaluator ***shall examine*** the provided acceptance procedures to determine that they describe the steps necessary for secure acceptance of the TOE in accordance with the developer's delivery procedures.

128 If it is not anticipated by the developer's delivery procedures that acceptance procedures will or can be applied, this work unit is not applicable, and is therefore considered to be satisfied.

129 The acceptance procedures should include as a minimum, that the user has to check that all parts of the TOE as indicated in the ST have been delivered in the correct version.

130 The acceptance procedures should reflect the steps the user has to perform in order to accept the delivered TOE that are implied by the developer's delivery procedures.

131 The acceptance procedures should provide detailed information about the following, if applicable:

a) making sure that the delivered TOE is the complete evaluated instance;

b) detecting modification/masquerading of the delivered TOE.

AGD_PRE.1.2C ***The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.***

AGD_PRE.1-2 The evaluator ***shall examine*** the provided installation procedures to determine that they describe the steps necessary for secure installation of the TOE and the secure preparation of the operational environment in accordance with the security objectives in the ST.

132 If it is not anticipated that installation procedures will or can be applied (e.g. because the TOE may already be delivered in an operational state), this work unit is not applicable, and is therefore considered to be satisfied.

133 The installation procedures should provide detailed information about the following, if applicable:

a) minimum system requirements for secure installation.

b) requirements for the operational environment in accordance with the security objectives provided by the ST;

c) the steps the user has to perform in order to get to an operational TOE being commensurate with its evaluated configuration. Such a description shall include - for each step - a clear scheme for the decision on the next step depended on success, failure or problems at the current step;

d) changing the installation specific security characteristics of entities under the control of the TSF (for example parameters, settings, passwords);

e) handling exceptions and problems.

5 Class ALC: Life-cycle support

5.1 CM capabilities (ALC_CMC)

5.1.1 Evaluation of sub-activity (ALC_CMC.2)

5.1.1.1 Objectives

944 The objectives of this sub-activifty are to determine whether the developer uses a CM system that uniquely identifies all configuration items.

5.1.1.2 Input

945 The evaluation evidence for this sub-activity is:

a) the ST;

b) the TOE suitable for testing;

c) the configuration management documentation.

5.1.1.3 Application notes

946 This component contains an implicit evaluator action to determine that the CM system is being used. As the requirements here are limited to identification of the TOE and provision of a configuration list, this action is already covered by, and limited to, the existing work units. At Evaluation of sub-activity (ALC_CMC.3) the requirements are expanded beyond these two items, and more explicit evidence of operation is required.

5.1.1.4 Action ALC_CMC.2.1E

ALC_CMC.2.1C ***The TOE shall be labelled with its unique reference.***

ALC_CMC.2-1 The evaluator ***shall check*** that the TOE provided for evaluation is labelled with its reference and that **each TOE security related component shall have a unique visible identifier affixed to it (CAS F4). The unique identifier applies to tamper-resistant boundaries, e.g. the PED and IC Card Reader. Their respective identifiers must be visible without opening.**

947 The evaluator should ensure that the TOE contains the unique reference which is stated in the ST. This could be achieved through labelled packaging or media, or by a label displayed by the operational TOE. This is to ensure that it would be possible for consumers to identify the TOE (e.g. at the point of purchase or use).

948 The TOE may provide a method by which it can be easily identified. For example, a software TOE may display its name and version number during the start up routine, or in response to a command line entry. A hardware or firmware TOE may be identified by a part number physically stamped on the TOE.

949 Alternatively, the unique reference provided for the TOE may be the combination of the unique reference of each component from which the TOE is comprised (e.g. in the case of a composed TOE and certified as being free from hidden and unauthorized).

ALC_CMC.2-2 The evaluator ***shall check*** that the TOE references used are consistent.

950 If the TOE is labelled more than once then the labels have to be consistent. For example, it should be possible to relate any labelled guidance documentation supplied as part of the TOE to the evaluated operational TOE. This ensures that consumers can be confident that they have purchased the evaluated version of the TOE, that they have installed this version, and that they have the correct version of the guidance to operate the TOE in accordance with its ST.

951 The evaluator also verifies that the TOE reference is consistent with the ST.

952 If this work unit is applied to a composed TOE, the following will apply. The composed IT TOE will not be labelled with its unique (composite) reference, but only the individual components will be labelled with their appropriate TOE reference. It would require further development for the IT TOE to be labelled, i.e. during start-up and/or operation, with the composite reference. If the composed TOE is delivered as the constituent component TOEs, then the TOE items delivered will not contain the composite reference. However, the composed TOE ST will include the unique reference for the composed TOE and will identify the components comprising the composed TOE through which the consumers will be able to determine whether they have the appropriate items.

953 Typically, in an architecture where the keyboard and the IC Card Reader are separate entities, each of these components must be labelled with their appropriate TOE component reference, as stated in the ST on the one hand. On the other hand, the keyboard reference must be consistent with the IC Card Reader reference, with a consistency properly defined in user and/or administrator guidance.

ALC_CMC.2.2C ***The CM documentation shall describe the method used to uniquely identify the configuration items.***

ALC_CMC.2-3 The evaluator ***shall examine*** the method of identifying configuration items to determine that it describes how configuration items are uniquely identified.

954 Procedures should describe how the status of each configuration item can be tracked throughout the life-cycle of the TOE. The procedures may be detailed in the CM plan or throughout the CM documentation. The information included should describe:

a) the method how each configuration item is uniquely identified, such that it is possible to track versions of the same configuration item;

b) the method how configuration items are assigned unique identifiers and how they are entered into the CM system;

c) the method to be used to identify superseded versions of a configuration item.

ALC_CMC.2.3C ***The CM system shall uniquely identify all configuration items.***

ALC_CMC.2-4 The evaluator ***shall examine*** the configuration items to determine that they are identified in a way that is consistent with the CM documentation.

955 Assurance that the CM system uniquely identifies all configuration items is gained by examining the identifiers for the configuration items. For both configuration items that comprise the TOE, and drafts of configuration items that are submitted by the developer as evaluation evidence, the evaluator confirms that each configuration item possesses a unique identifier in a manner consistent with the unique identification method that is described in the CM documentation.

956 CM scope (ALC_CMS)

5.1.2 Evaluation of sub-activity (ALC_CMS.2)

5.1.2.1 Objectives

957 The objective of this sub-activity is to determine whether the configuration list includes the TOE, the parts that comprise the TOE, and the evaluation evidence. These configuration items are controlled in accordance with CM capabilities (ALC_CMC).

5.1.2.2 Input

958 The evaluation evidence for this sub-activity is:

a) the ST;

b) the configuration list.

5.1.2.3 Action ALC_CMS.2.1E

ALC_CMS.2.1C ***The configuration list shall include the following: the TOE itself; the evaluation evidence required by the SARs; and the parts that comprise the TOE.***

ALC_CMS.2-1 The evaluator ***shall check*** that the configuration item list includes the set of items required by the CC.

959 The list includes at least the following:

a) the TOE itself;

b) the parts that comprise the TOE;

c) the evaluation evidence required by the SARs.

ALC_CMS.2.2C ***The configuration list shall uniquely identify the configuration items.***

ALC_CMS.2-2 The evaluator ***shall examine*** the configuration list to determine that it uniquely identifies each configuration item.

960 The configuration list contains sufficient information to uniquely identify which version of each item has been used (typically a version number). Use of this list will enable the evaluator to check that the correct configuration items, and the correct version of each item, have been used during the evaluation.

961 The configuration list shall show that PED software implementation and any changes thereafter, have been inspected and reviewed using a

documented and auditable process and certified[2] as being free from hidden and unauthorized or undocumented functions (PCI B3).

ALC_CMS.2.3C ***For each TSF relevant configuration item, the configuration list shall indicate the developer of the item.***

ALC_CMS.2-3 The evaluator ***shall check*** that the configuration list indicates the developer of each TSF relevant configuration item.

962 If only one developer is involved in the development of the TOE, this work unit is not applicable, and is therefore considered to be satisfied.

[2] Certified here means that the Firmware has been checked by the developer. Hence the Firmware that is part of the configuration items has been checked in integrity.

5.2 Delivery (ALC_DEL)

5.2.1 Evaluation of sub-activity (ALC_DEL.1)

5.2.1.1 Objectives

963 The objective of this sub-activity is to determine whether the delivery documentation describes all procedures used to maintain security of the TOE when distributing the TOE to the user.

5.2.1.2 Input

964 The evaluation evidence for this sub-activity is:

a) the ST;

b) the delivery documentation.

5.2.1.3 Action ALC_DEL.1.1E

ALC_DEL.1.1C ***The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to the consumer.***

ALC_DEL.1-1 The evaluator ***shall examine*** the delivery documentation to determine that it describes all procedures that are necessary to maintain security when distributing versions of the TOE or parts of it to the consumer.

965 The delivery documentation describes proper procedures to maintain security of the TOE during transfer of the TOE or its component parts and to determine the identification of the TOE. Typically, the loader and manufacturing procedures must be documented.

966 The delivery documentation should cover the entire TOE, but may contain different procedures for different parts of the TOE. The evaluation should consider the totality of procedures.

967 The delivery procedures should be applicable across all phases of delivery from the production environment to the installation environment (e.g. packaging, storage and distribution). Standard commercial practise for packaging and delivery may be acceptable. This includes shrink wrapped packaging, a security tape or a sealed envelope. For the distribution, physical (e.g. public mail or a private distribution service) or electronic (e.g. electronic mail or downloading off the Internet) procedures may be used.

968 Cryptographic checksums or a software signature may be used by the developer to ensure that tampering or masquerading can be detected. Tamper proof seals additionally indicate if the confidentiality has been broken. For software TOEs, confidentiality might be assured by using encryption. If availability is of concern, a secure transportation might be required.

969 Interpretation of the term “necessary to maintain security” will need to consider:

– The nature of the TOE (e.g. whether it is software or hardware).

– The overall security level stated for the TOE by the chosen level of the Vulnerability Assessment. If the TOE is required to be resistant against attackers of a certain potential in its intended environment, this should also apply to the delivery of the TOE. The evaluator should determine that a balanced approach has been taken, such that delivery does not present a weak point in an otherwise secure development process.

– The security objectives provided by the ST. The emphasis in the delivery documentation is likely to be on measures related to integrity, as integrity of the TOE is always important. However, confidentiality and availability of the delivery will be of concern in the delivery of some TOEs; procedures relating to these aspects of the secure delivery should also be discussed in the procedures.

5.2.1.4 Implied evaluator action

ALC_DEL.1.2D ***The developer shall use the delivery procedures.***

ALC_DEL.1-2 The evaluator ***shall examine*** aspects of the delivery process to determine that the delivery procedures are used.

970 The approach taken by the evaluator to check the application of delivery procedures will depend on the nature of the TOE, and the delivery process itself. In addition to examination of the procedures themselves, the evaluator seeks some assurance that they are applied in practise. Some possible approaches are:

c) a visit to the distribution site(s) where practical application of the procedures may be observed;

d) examination of the TOE at some stage during delivery, or after the user has received it (e.g. checking for tamper proof seals **or checking for the cryptographic protection applied to the keys loaded at the different key loading facilities**);

e) observing that the process is applied in practise when the evaluator obtains the TOE through regular channels;

f) questioning end users as to how the TOE was delivered.

971 For guidance on site visits see [CEM A.4, Site Visits]. **The evaluator shall confirm the use of delivery procedures by examination of the developer’s documentation and evidences. The delivery procedures involving the Initial Key Loading Facility shall be also checked during a site visit (cf ALC_DVS.2)**

972 It may be the case of a newly developed TOE that the delivery procedures have yet to be exercised. In these cases, the evaluator has to be satisfied that appropriate procedures and facilities are in place for future deliveries and that all personnel involved are aware of their responsibilities. The evaluator may request a “dry run” of a delivery if this is practical. If the developer has produced other similar products, then an examination of procedures in their use may be useful in providing assurance.

5.3 Development security (ALC_DVS)

5.3.1 Evaluation of sub-activity (ALC_DVS.2)

5.3.1.1 Objectives

973 The objective of this sub-activity is to determine whether the developer's security controls on the development environment are adequate to provide the confidentiality and integrity of the TOE design and implementation that is necessary to ensure that secure operation of the TOE is not compromised. Additionnally, sufficiency of the measures as applied is intended be justified.

5.3.1.2 Input

974 The evaluation evidence for this sub-activity is:

a) the ST;

b) the development security documentation.

975 In addition, the evaluator may need to examine other deliverables to determine that the security controls are well-defined and followed. Specifically, the evaluator may need to examine the developer's configuration management documentation (the input for the Evaluation of sub-activity (ALC_CMC.4) “Production support and acceptance procedures” and the Evaluation of sub-activity (ALC_CMS.4) “Problem tracking CM coverage”). Evidence that the procedures are being applied is also required.

5.3.1.3 Action ALC_DVS.2.1E

ALC_DVS.2.1C ***The development security documentation shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the confidentiality and integrity of the TOE design and implementation in its development environment.***

ALC_DVS.2-1 **The TOE development environment stands for the design, manufacturing, assembling and maintenance environments of the TOE components, including the final assembly and the initial key loading facilities.**

The evaluator ***shall examine*** the development security documentation to determine that it details all security measures used in the development environment that are necessary to protect the confidentiality and integrity of the TOE design and implementation.

976 The evaluator determines what is necessary by first referring to the ST for any information that may assist in the determination of necessary protection.

977 In particular when distributing security relevant components of the TOE before assembling, subsequent to production and prior to shipment and on the way to the Initial Key Loading facility (which can be at manufacturing, testing, pre-personalization phases).

978 **As long as the initialisation of the TOE is not completed, every place where key or software loading is performed must be accounted for in the delivery procedures, along with the description of the actors performing the loading (identification of the software loading agents for instance).**

979 **The development security documentation, shall describe all the physical, procedural, personnel, and other security measures that are necessary to protect the integrity of the design and implementation of the TOE security-related components in their development environment. The development security documentation shall provide evidence that these security measures are followed during the development and maintenance of the POI security-related components. The evidence shall justify that the security measures provide the necessary level of protection to maintain the integrity of the POI security-related components (CAS E8).**

980 **The development security documentation shall show that the security relevant components of the TOE are protected and stored in such a manner as to preclude unauthorized modification, e.g., using dual control or standardized cryptographic authentication procedures (PCI E2, CAS E2.a).**

981 **The development security documentation shall show that the TOE is assembled in a manner that the components used in the authenticating process are those components that were under configuration management (ALC_CMC) and in the configuration list (ALC_CMS), and that unauthorized substitutions have not been made. The vendor shall confirm this by giving an integration statement (PCI E3, CAS E3.a).**

982 **The development security documentation shall show that the production software that is loaded to TOE components at the time of manufacture is transported, stored, and used under the principle of dual control, preventing unauthorized modifications and/or substitutions (PCI E4, CAS E4.a). Subsequent to production but prior to shipment from the manufacturer's facility, the TOE and any of its components are stored in protected, access-controlled area or sealed within tamper-evident packaging to prevent undetected unauthorized access to the device or its components (PCI E5, CAS E5.a).**

983 **The development security documentation must provide means to the key loading facilities to assure the authenticity of the TOE security relevant components (CAS E7) e.g. if the manufacturer is in charge of initial-key-loading himself he must verify the authenticity of the TOE security enforcing components for himself (CAS E7.1), else if the manufacturer is not in charge of Initial Key Loading he must provide means to the initial-key-loading facility to assure the verification of the authenticity of the TOE security enforcing components (CAS E7.2).**

984 **If the TOE or security relevant components of the TOE will be authenticated at the key loading facility by means of secret information placed in the device during manufacturing, then the development security documentaton shall show that this secret information is unique to each TOE**

resp. security relevant components of the TOE, unknown and unpredictable to any person, and installed in the TOE resp. security relevant components under dual control to ensure that it is not disclosed during installation (PCI E6, CAS E6.a).

985 **The development security documentation shall show that while in transit from the manufacturer's facility to external facilities, security related TOE components are shipped and stored in tamper-evident packaging; and/or, shipped and stored containing a secret that is immediately and automatically erased if any physical or functional alteration to the device is attempted, that can be verified by the key loading facilities, but that cannot feasibly be determined by unauthorized personnel (PCI F3, CAS F3.a).**

986 **The development security documentation must show that the tools used to produce and manufacture the TOE software and hadware at the different sites of fabrication are managed by secured tools, for instance secured databases.**

987 **The development security documentation shall describe all the delivery procedures necessary to maintain the security of the TOE components before assembling, subsequent to production and prior to shipment and on the way to the Initial Key Loading Facility. The delivery procedures shall contribute enforcing the following requirements:**

a) PCI F1, CAS F1.a: The PED and PAL (POI Application Logic) security enforcing components are shipped from the manufacturer's facility to the initial-key-loading facility, and stored en route, under auditable controls that can account for the location of every components at every point.

b) PCI F2, CAS F2.a: Procedures are in place to transfer accountability for the device from the manufacturer to the initial-key-loading facility.

988 If no explicit information is available from the ST the evaluator will need to make a determination of the necessary measures. In cases where the developer's measures are considered less than what is necessary, a clear justification should be provided for the assessment, based on a potential exploitable vulnerability.

989 The following types of security measures are considered by the evaluator when examining the documentation:

a) physical, for example physical access controls used to prevent unauthorised access to the TOE development environment (during normal working hours and at other times);

b) procedural, for example covering:

- granting of access to the development environment or to specific parts of the environment such as development machines

- revocation of access rights when a person leaves the development team

- transfer of protected material within and out of the development environment and between different development sites in accordance with defined acceptance procedures

- admitting and escorting visitors to the development environment

- roles and responsibilities in ensuring the continued application of security measures, and the detection of security breaches.

c) personnel, for example any controls or checks made to establish the trustworthiness of new development staff;

d) other security measures, for example the logical protections on any development machines.

990 The development security documentation should identify the locations at which development occurs, and describe the aspects of development performed, along with the security measures applied at each location and for transports between different locations. For example, development could occur at multiple facilities within a single building, multiple buildings at the same site, or at multiple sites. Transports of parts of the TOE or the unfinished TOE between different development sites are to be covered by Development security (ALC_DVS), whereas the transport of the finished TOE to the consumer is dealt with in Delivery (ALC_DEL).

991 Development includes the production of the TOE.

ALC_DVS.2.2C ***The development security documentation shall justify that the security measures provide the necessary level of protection to maintain the confidentiality and integrity of the TOE.***

ALC_DVS.2-2 The evaluator ***shall examine*** the development security documentation to determine that an appropriate justification is given why the security measures provide the necessary level of protection to maintain the confidentiality and integrity of the TOE..

992 Since attacks on the TOE or its related information are assumed in different design and production stages, measures and procedures need to have an appropriate level necessary to prevent those attacks or to make them more difficult.

993 Since this level depends on the overall attack potential claimed for the TOE (cf. the Vulnerability analysis (AVA_POI) component chosen), the development security documentation should justify the necessary level of protec-

tion to maintain the confidentiality and integrity of the TOE. This level has to be achieved by the security measures applied.

994 The concept of protection measures should be consistent, and the justification should include an analysis of how the measures are mutually supportive. All aspects of development and production on all the different sites with all roles involved up to delivery of the TOE should be analysed.

995 Justification may include an analysis of potential vulnerabilities taking the applied security measures into account.

996 There may be a convincing argument showing that e.g.

- The technical measures and mechanisms of the developer's infrastructure are sufficient for keeping the appropriate security level (e.g. cryptographic mechanisms as well as physical protection mechanisms, properties of the CM system (cf. ALC_CMC.4-5));
- The system containing the implementation representation of the TOE (including concerning guidance documents) provides effective protection against logical attacks e.g. by “Trojan” code or viruses. It might be adequate, if the implementation representation is kept on an isolated system where only the software necessary to maintain it is installed and where no additional software is installed afterwards.
- Data brought into this system need to be carefully considered to prevent the installation of hidden functionality onto the system. The effectiveness of these measures need to be tested, e.g. by independently trying to get access to the machine, install some additional executable (program, macro etc.) or get some information out of the machine using logical attacks.
- The appropriate organisational (procedural and personal) measures are unconditionally enforced.

ALC_DVS.2-3 The evaluator ***shall examine*** the development confidentiality and integrity policies in order to determine the sufficiency of the security measures employed.

997 The evaluator should examine whether the following is included in the policies:

a) what information relating to the TOE development needs to be kept confidential, and which members of the development staff are allowed to access such material;

b) what material must be protected from unauthorised modification in order to preserve the integrity of the TOE, and which members of the development staff are allowed to modify such material.

998 The evaluator should determine that these policies are described in the development security documentation, that the security measures employed are consistent with the policies, and that they are complete.

999 It should be noted that configuration management procedures will help protect the integrity of the TOE and the evaluator should avoid overlap with the work-units conducted for the CM capabilities (ALC_CMC). For example, the CM documentation may describe the security procedures necessary for controlling the roles or individuals who should have access to the development environment and who may modify the TOE.

1000 Whereas the CM capabilities (ALC_CMC) requirements are fixed, those for the Development security (ALC_DVS), mandating only necessary measures, are dependent on the nature of the TOE, and on information that may be provided in the ST. For example, the ST may identify a security objective for the development environment that requires the TOE to be developed by staff that has security clearance. The evaluators would then determine that such a policy had been applied under this sub-activity.

5.3.1.4 Action ALC_DVS.2.2E

ALC_DVS.2-4 The evaluator ***shall examine*** the development security documentation and associated evidence to determine that the security measures are being applied.

1001 This work unit requires the evaluator to determine that the security measures described in the development security documentation are being followed, such that the integrity of the TOE and the confidentiality of associated documentation is being adequately protected. For example, this could be determined by examination of the documentary evidence provided. Documentary evidence should be supplemented by visiting the development environment. A visit to the development environment will allow the evaluator to:

a) observe the application of security measures (e.g. physical measures);

b) examine documentary evidence of application of procedures;

c) interview development staff to check awareness of the development security policies and procedures, and their responsibilities.

1002 A development site visit is a useful means of gaining confidence in the measures being used. Any decision not to make such a visit should be determined in consultation with the evaluation authority.

1003 For guidance on site visits see [CEM A.4], Site Visits.

1004 **A site visit on the final assembly stage of the TOE might help gaining confidence. The evaluator shall confirm that the security measures are being applied by examination of the developer's documentation and evidences. The security measures involving the final assembly and the Initial Key Loading facilities shall be checked during a site visit (CAS E9).**

Both visits (at Initial Key Loading and at final assembly) shall be performed if those two stages are not simultaneous.

6 Class AVA: Vulnerability assessment

6.1 Vulnerability analysis (AVA_POI)

1005 **As stated in the introduction of this document, the acronym POI designates the Target of Evaluation (TOE) of the [CC POI PP]. The levels of AVA_POI requirements are designed to apply to different components of the TOE, attacked at different levels by attackers possessing distinct attack potential. Hence, the TOE, or POI, is divised between five subsets of components (MSR, PED Middle TSF, Middle TSF, Core TSF and Core TSF keys), defined in the [CC POI PP], and the vulnerability analysis sketched at four hierarchical levels of attack required from the attacker.**

1006 **These four levels of attack are represented in the following AVA_POI requirements:**

- **AVA_POI.1 applied to Magnetic Stripe Reader, considering an attacker with POI-Basic attack potential**
- **AVA_POI.2, applied to PED Middle TSF and Middle TSF, considering an attacker with POI-Low attack potential**
- **AVA_POI.3 applied to CoreTSF, considering an attacker with POI-Moderate attack potential**
- **AVA_POI.4 applied to Core TSF keys, considering an attacker with POI-High attack potential**

1007 **In what follows, the term POI is used as a synonim to TOE. Therefore, when "POI or POI components" is used in the description of action elements, we refer to the TOE or TOE subset which is targetted at by the respective action element. This terminology has been chosen to enable the vulnerability analysis of a whole POI at one AVA_POI level. Indeed in some evaluations, it may be the choice of the developper to evaluate the whole POI, for instance, at the AVA_POI.4 level. Thus the principle that if a POI component is referred to in two or more AVA_POI requirements then the more demanding requirement shall apply.**

6.1.1 Evaluation of sub-activity (AVA_POI.1)

6.1.1.1 Objectives

1008 The objective of this sub-activity is to determine whether **the POI or POI components**, in **their** operational environment, **have** vulnerabilities exploitable by attackers possessing **an attack potential of POI-Basic.**

1009 **The POI or POI components in this sub-activity include the Magnetic Stripe Reader as defined in the [CC POI PP]. This sub-activity is applicable only to PED-ONLY and POI-COMPREHENSIVE configurations,**

since there is no Magnetic Stripe Reader component in POI-OPTION configuration.

6.1.1.2 Input

1010 The evaluation evidence for this sub-activity is:

a) the ST;

b) the functional specification;

c) the **POI or POI components** design;

d) the security architecture description;

e) the guidance documentation;

f) the **POI or POI components** suitable for testing;

g) information publicly available to support the identification of possible potential vulnerabilities;

1011 Further evidence is,

a) the Magnetic Stripe Reader hardware implementation representation, in the configurations of the POI: PED ONLY and POI COMPREHENSIVE;

b) a mapping of SFRs to the implementation representation of the Magnetic Stripe Reader hardware.

1012 The remaining implicit evaluation evidence for this sub-activity depends on the components that have been included in the assurance package. The evidence provided for each component is to be used as input in this sub-activity.

1013 Other input for this sub-activity is:

a) current information regarding public domain potential vulnerabilities and attacks (e.g. from an evaluation authority)

6.1.1.3 Application notes

1014 The evaluator should consider performing additional tests as a result of potential vulnerabilities encountered during other parts of the evaluation.

1015 The evaluator should use the hardware implementation representation as a guide to penetration testing.

6.1.1.4 Action AVA_POI.1.1E

AVA_POI.1.1C ***The POI or POI components shall be suitable for testing.***

AVA_POI.1-1 The evaluator ***shall examine*** the **POI or POI components** to determine that the test configuration is consistent with the configuration under evaluation as specified in the ST.

1016 The **POI or POI components** provided by the developer and identified in the test plan should have the same unique reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST introduction.

1017 It is possible for the ST to specify more than one configuration for evaluation. The **POI or POI components** may comprise a number of distinct hardware and software entities that need to be tested in accordance with the ST. The evaluator verifies that all test configurations are consistent with the ST.

1018 The evaluator should consider the security objectives for the operational environment described in the ST that may apply to the test environment and ensure they are met in the testing environment. There may be some objectives for the operational environment that do not apply to the test environment. For example, an objective about user clearances may not apply; however, an objective about a single point of connection to a network would apply.

1019 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to ensure that these resources are calibrated correctly.

AVA_POI.1-2 The evaluator ***shall examine*** the **POI or POI components** to determine that it has been installed properly and is in a known state.

1020 It is possible for the evaluator to determine the state of the **POI or POI components** in a number of ways. For example, previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will satisfy this work unit if the evaluator still has confidence that the **POI or POI components** being used for testing was installed properly and is in a known state. If this is not the case, then the evaluator should follow the developer's procedures to install and start up the **POI or POI components**, using the supplied guidance only.

1021 If the evaluator has to perform the installation procedures because the **POI or POI components** is in an unknown state, this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.

6.1.1.5 Action AVA_POI.1.2E

AVA_POI.1-3 The evaluator ***shall examine*** sources of information publicly available to identify potential vulnerabilities in the **POI or POI components**.

1022 The evaluator examines the sources of information publicly available to support the identification of possible potential vulnerabilities in the **POI or POI components**. There are many sources of publicly available information **or restricted documents** which the evaluator should consider using items such as those available on the world wide web, including:

a) specialist publications (magazines, books);

b) research papers;

c) the document “Attack Methods to POIs” ([AttackMethPOI]).

1023 The evaluator should not constrain their consideration of publicly available information to the above, but should consider any other relevant information available.

1024 While examining the evidence provided the evaluator will use the information in the public domain to further search for potential vulnerabilities. Where the evaluators have identified areas of concern, the evaluator should consider information publicly available that relate to those areas of concern.

1025 The availability of information that may be readily available to an attacker that helps to identify and facilitate attacks may substantially enhance the attack potential of a given attacker. The accessibility of vulnerability information and sophisticated attack tools on the Internet makes it more likely that this information will be used in attempts to identify potential vulnerabilities in the **POI or POI components** and exploit them. Modern search tools make such information easily available to the evaluator, and the determination of resistance to published potential vulnerabilities and well known generic attacks can be achieved in a cost-effective manner.

1026 The search of the information publicly available should be focused on those sources that refer specifically to the product from which the **POI or POI components** is derived. The extensiveness of this search should consider the following factors: **POI or POI components** type, evaluator experience in this type, expected attack potential and the level of ADV evidence available.

1027 The identification process is iterative, where the identification of one potential vulnerability may lead to identifying another area of concern that requires further investigation.

1028 The evaluator will report what actions were taken to identify potential vulnerabilities in the evidence. However, in this type of search, the evaluator may not be able to describe the steps in identifying potential vulnerabilities before the outset of the examination, as the approach may evolve as a result of findings during the search.

1029 The evaluator will report the evidence examined in completing the search for potential vulnerabilities. This selection of evidence may be derived from those areas of concern identified by the evaluator, linked to the evidence the attacker is assumed to be able to obtain, or according to another rationale provided by the evaluator.

6.1.1.6 Action AVA_POI.1.3E

AVA_POI.1-4 The evaluator ***shall conduct*** a search of ST, guidance documentation, functional specification, **POI or POI components** design and security architecture description evidence to identify possible potential vulnerabilities in the **POI or POI components.**

1030 A search of the evidence should be completed whereby specifications and documentation for the **POI or POI components** are analysed and then potential vulnerabilities in the **POI or POI components** are hypothesised, or speculated. The list of hypothesised potential vulnerabilities is then prioritised on the basis of the estimated probability that a potential vulnerability exists and, assuming an exploitable vulnerability does exist the attack potential required to exploit it, and on the extent of control or compromise it would provide. The prioritised list of potential vulnerabilities is used to direct penetration testing against the **POI or POI components**.

1031 The security architecture description provides the developer vulnerability analysis, as it documents how the TSF protects itself from interference from untrusted subjects and prevents the bypass of security enforcement functionality. Therefore, the evaluator should use this description of the protection of the TSF**, as well as the implementation representation and the mapping of the SFRs to this implementation representation,** as a basis for the search for possible ways to undermine the TSF.

1032 Subject to the SFRs the **POI or POI components** is to meet in the operational environment, the evaluator's independent vulnerability analysis should consider generic potential vulnerabilities under each of the following headings:

a) generic potential vulnerabilities relevant for the type of **POI or POI components** being evaluated, as may be supplied by the evaluation authority;

b) bypassing;

c) tampering;

d) direct attacks;

e) monitoring;

f) misuse.

1033 The security architecture description should be considered in light of each of the above generic potential vulnerabilities. Each potential vulnerability should be considered to search for possible ways in which to defeat the TSF protection and undermine the TSF.

AVA_POI.1-5 The evaluator ***shall record*** in the ETR the identified potential vulnerabilities that are candidates for testing and applicable to the **POI or POI components** in its operational environment.
It is impossible to describe potential vulnerabilities exhaustively because these evolve in time.

1034 It may be identified that no further consideration of the potential vulnerability is required if for example the evaluator identifies that measures in the operational environment, either IT or non-IT, prevent exploitation of the potential vulnerability in that operational environment. For instance, restricting physical access to the **POI or POI components** to authorised users only may effectively render a potential vulnerability to tampering unexploitable.

1035 The evaluator records any reasons for exclusion of potential vulnerabilities from further consideration if the evaluator determines that the potential vulnerability is not applicable in the operational environment. Otherwise the evaluator records the potential vulnerability for further consideration.

1036 A list of potential vulnerabilities applicable to the **POI or POI components** in its operational environment, which can be used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.

6.1.1.7 Action AVA_POI.1.4E

AVA_POI.1-6 The evaluator ***shall devise*** penetration tests, based on the independent search for potential vulnerabilities.

1037 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the **POI or POI components**, in its operational environment, to the potential vulnerabilities identified during the search of the sources of information publicly available. Any current information provided to the evaluator by a third party (e.g. evaluation authority) regarding known potential vulnerabilities will be considered by the evaluator, together with any encountered potential vulnerabilities resulting from the performance of other evaluation activities.

1038 The evaluator is reminded that, as for considering the security architecture description in the search for vulnerabilities (as detailed in AVA_POI.1-4), testing should be performed to confirm the architectural properties. This is likely to require negative tests attempting to disprove the properties of the security architecture. In developing the strategy for penetration testing, the evaluator will ensure that each of the major characteristics of the security architecture description are tested, either in functional testing (as considered in [CEM] section 14 ATE Class) or evaluator penetration testing.

1039 The evaluator will probably find it practical to carry out penetration test using a series of test cases, where each test case will test for a specific potential vulnerability.

1040 The evaluator is not expected to test for potential vulnerabilities (including those in the public domain) beyond those which required **POI-Basic attack potential**. In some cases, however, it will be necessary to carry out a test before the exploitability can be determined. Where, as a result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond **POI-Basic attack potential**, this is reported in the ETR as a residual vulnerability.

1041 Guidance on determining the necessary attack potential to exploit a potential vulnerability can be found in **[AttackPotPOI].**

1042 Potential vulnerabilities hypothesised as exploitable only by attackers possessing **POI-Low, POI-Moderate or POI-High attack potential** do not result in a failure of this evaluator action. Where analysis supports the hypothesis, these need not be considered further as an input to penetration testing. However, such vulnerabilities are reported in the ETR as residual vulnerabilities.

1043 Potential vulnerabilities hypothesised as exploitable by an attacker possessing **an attack potential of POI-Basic** and resulting in a violation of the security objectives should be the highest priority potential vulnerabilities comprising the list used to direct penetration testing against the **POI or POI components**.

1044 Devise of penetration testing shall comprise but is not limited to checking the following property:

a) The Magnetic Stripe Reader detects and responds to tampering.

AVA_POI.1-7 The evaluator ***shall produce*** penetration test documentation for the tests based on the list of potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test documentation shall include:

a) identification of the potential vulnerability the **POI or POI components** are being tested for;

b) instructions to connect and setup all required test equipment as required to conduct the penetration test;

c) instructions to establish all penetration test prerequisite initial conditions;

d) instructions to stimulate the TSF;

e) instructions for observing the behaviour of the TSF;

f) descriptions of all expected results and the necessary analysis to be performed on the observed behaviour for comparison against expected results;

g) instructions to conclude the test and establish the necessary post-test state for the **POI or POI components**.

1045 The evaluator prepares for penetration testing based on the list of potential vulnerabilities identified during the search of the public domain and the analysis of the evaluation evidence.

1046 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond those for which **an attack potential of POI-Basic** is required to effect an attack. However, as a result of evaluation expertise, the evaluator may discover a potential vulnerability that is exploitable only by an attacker with greater than **POI-Basic** attack potential. Such vulnerabilities are to be reported in the ETR as residual vulnerabilities.

1047 With an understanding of the potential vulnerability, the evaluator determines the most feasible way to test for the **POI or POI components'** susceptibility. Specifically the evaluator considers:

a) the TSFI or other **POI** interface that will be used to stimulate the TSF and observe responses (It is possible that the evaluator will need to use an interface to the **POI** other than the TSFI to demonstrate properties of the TSF such as those described in the security architecture description (as required by ADV_ARC). It should the noted, that although these **POI** interfaces provide a means of testing the TSF properties, they are not the subject of the test.);

b) initial conditions that will need to exist for the test (i.e. any particular objects or subjects that will need to exist and security attributes they will need to have);

c) special test equipment that will be required to either stimulate a TSFI or make observations of a TSFI (although it is unlikely that specialist equipment would be required to exploit a potential vulnerability assuming a Basic attack potential);

d) whether theoretical analysis should replace physical testing, particularly relevant where the results of an initial test can be extrapolated to demonstrate that repeated attempts of an attack are likely to succeed after a given number of attempts.

1048 The evaluator will probably find it practical to carry out penetration testing using a series of test cases, where each test case will test for a specific potential vulnerability.

1049 The intent of specifying this level of detail in the test documentation is to allow another evaluator to repeat the tests and obtain an equivalent result.

AVA_POI.1-8 The evaluator ***shall conduct*** penetration testing.

1050 The evaluator uses the penetration test documentation resulting from work unit AVA_POI.1-6 as a basis for executing penetration tests on the **POI or POI components**, but this does not preclude the evaluator from performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests as a result of information learnt during penetration testing that, if performed by the evaluator, are to be recorded in the penetration test documentation. Such tests may be required to follow up unexpected results or observations, or to investigate potential vulnerabilities suggested to the evaluator during the pre-planned testing.

1051 Should penetration testing show that a hypothesised potential vulnerability does not exist, then the evaluator should determine whether or not the evaluator's own analysis was incorrect, or if evaluation deliverables are incorrect or incomplete.

1052 The evaluator is not expected to test for potential vulnerabilities (including those in the public domain) beyond those which required **an attack potential of POI-Basic**. In some cases, however, it will be necessary to carry out a test before the exploitability can be determined. Where, as a result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond basic attack potential, this is reported in the ETR as a residual vulnerability.

AVA_POI.1-9 The evaluator ***shall record*** the actual results of the penetration tests.

1053 While some specific details of the actual test results may be different from those expected (e.g. time and date fields in an audit record) the overall result should be identical. Any unexpected test results should be investigated. The impact on the evaluation should be stated and justified.

AVA_POI.1-10 The evaluator ***shall report*** in the ETR the evaluator penetration testing effort, outlining the testing approach, configuration, depth and results.

1054 The penetration testing information reported in the ETR allows the evaluator to convey the overall penetration testing approach and effort expended on this sub-activity. The intent of providing this information is to give a meaningful overview of the evaluator's penetration testing effort. It is not intended that the information regarding penetration testing in the ETR be an exact reproduction of specific test steps or results of individual penetration tests. The intention is to provide enough detail to allow other evaluators and evaluation authorities to gain some insight about the penetration testing approach chosen, amount of penetration testing performed, **POI or POI components** test configurations, and the overall results of the penetration testing activity.

1055 Information that would typically be found in the ETR section regarding evaluator penetration testing efforts is:

a) **POI or POI components** test configurations. The particular configurations of the **POI or POI components** that were penetration tested;

b) TSFI penetration tested. A brief listing of the TSFI and other **POI** interfaces that were the focus of the penetration testing;

c) Verdict for the sub-activity. The overall judgement on the results of penetration testing.

1056 This list is by no means exhaustive and is only intended to provide some context as to the type of information that should be present in the ETR concerning the penetration testing the evaluator performed during the evaluation.

AVA_POI.1-11 The evaluator ***shall examine*** the results of all penetration testing to determine that the **POI or POI components**, in its operational environment, is resistant to an attacker possessing **an attack potential POI-Basic**.

1057 If the results reveal that the **POI or POI components**, in its operational environment, has vulnerabilities exploitable by an attacker possessing less than a **POI-Basic** attack potential, then this evaluator action fails.

1058 The guidance in B.4 should be used to determine the attack potential required to exploit a particular vulnerability and whether it can therefore be exploited in the intended environment. It may not be necessary for the attack potential to be calculated in every instance, only if there is some doubt as to whether or not the vulnerability can be exploited by an attacker possessing an attack potential less than **POI-Basic**.

AVA_POI.1-12 The evaluator ***shall report*** in the ETR all exploitable vulnerabilities and residual vulnerabilities, detailing for each:

a) its source (e.g. CEM activity being undertaken when it was conceived, known to the evaluator, read in a publication);

b) the SFR(s) not met;

c) a description;

d) whether it is exploitable in its operational environment or not (i.e. exploitable or residual).

e) the amount of time, level of expertise, level of knowledge of the **POI or POI components**, level of opportunity and the equipment required to perform the identified vulnerabilities, and the corresponding values using the tables 3 and 4 of Annex [AttackPotPOI]

6.1.2 Evaluation of sub-activity (AVA_POI.2)

6.1.2.1 Objectives

1059 The objective of this sub-activity is to determine whether **the POI or POI components**, in **their** operational environment, **have** vulnerabilities exploitable by attackers possessing **an attack potential of POI-Low.**

1060 **Depending on the configuration, the POI or POI components in this sub-activity include either the PED Middle TSF, in case of PED ONLY configuration, either the PED Middle TSF plus the Middle TSF in case of POI COMPREHENSIVE or POI OPTION configurations, as defined in the [CC POI PP].**

6.1.2.2 Input

1061 The evaluation evidence for this sub-activity is:

a) the ST;

b) the functional specification;

c) the **POI or POI components** design;

d) the security architecture description;

e) the guidance documentation;

f) the **POI or POI components** suitable for testing;

g) information publicly available to support the identification of possible potential vulnerabilities.

1062 Further evidence is,

a) the PED Middle TSF hardware and software implementation representation, in the three possible configurations of the POI (PED ONLY, POI COMPREHENSIVE or POI OPTION);

b) a mapping of SFRs to the implementation representation of the PED Middle TSF hardware and software.

1063 The remaining implicit evaluation evidence for this sub-activity depends on the components that have been included in the assurance package. The evidence provided for each component is to be used as input in this sub-activity.

1064 Other input for this sub-activity is:

a) current information regarding public domain potential vulnerabilities and attacks (e.g. from an evaluation authority)

6.1.2.3 Application notes

1065 The evaluator should consider performing additional tests as a result of potential vulnerabilities encountered during other parts of the evaluation.

1066 The evaluator should use the implementation representation as a guide to penetration testing.

6.1.2.4 Action AVA_POI.2.1E

AVA_POI.2.1C ***The POI or POI components shall be suitable for testing.***

AVA_POI.2-1 The evaluator ***shall examine*** the **POI or POI components** to determine that the test configuration is consistent with the configuration under evaluation as specified in the ST.

1067 The **POI or POI components** provided by the developer and identified in the test plan should have the same unique reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST introduction.

1068 It is possible for the ST to specify more than one configuration for evaluation. The **POI or POI components** may comprise a number of distinct hardware and software entities that need to be tested in accordance with the ST. The evaluator verifies that all test configurations are consistent with the ST.

1069 The evaluator should consider the security objectives for the operational environment described in the ST that may apply to the test environment and ensure they are met in the testing environment. There may be some objectives for the operational environment that do not apply to the test environment. For example, an objective about user clearances may not apply; however, an objective about a single point of connection to a network would apply.

1070 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to ensure that these resources are calibrated correctly.

AVA_POI.2-2 The evaluator ***shall examine*** the **POI or POI components** to determine that it has been installed properly and is in a known state.

1071 It is possible for the evaluator to determine the state of the **POI or POI components** in a number of ways. For example, previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) will satisfy this work unit if the evaluator still has confidence that the **POI or POI components** being used for testing was installed properly and is in a known state. If this is not the case, then the evaluator should follow the developer's procedures to install and start up the **POI or POI components**, using the supplied guidance only.

1072 If the evaluator has to perform the installation procedures because the **POI or POI components** is in an unknown state, this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.

6.1.2.5 Action AVA_POI.2.2E

AVA_POI.2-3 The evaluator ***shall examine*** sources of information publicly available to identify potential vulnerabilities in the **POI or POI components**.

1073 The evaluator examines the sources of information publicly available to support the identification of possible potential vulnerabilities in the **POI or POI components**. There are many sources of publicly available information **or restricted documents** which the evaluator should consider using items such as those available on the world wide web, including:

b) specialist publications (magazines, books);

c) research papers;

d) the JTEMS document “Attack Methods to POIs” ([AttackMeth-POI]).

1074 The evaluator should not constrain their consideration of publicly available information to the above, but should consider any other relevant information available.

1075 While examining the evidence provided the evaluator will use the information in the public domain to further search for potential vulnerabilities.

Where the evaluators have identified areas of concern, the evaluator should consider information publicly available that relate to those areas of concern.

1076 The availability of information that may be readily available to an attacker that helps to identify and facilitate attacks may substantially enhance the attack potential of a given attacker. The accessibility of vulnerability information and sophisticated attack tools on the Internet makes it more likely that this information will be used in attempts to identify potential vulnerabilities in the **POI or POI components** and exploit them. Modern search tools make such information easily available to the evaluator, and the determination of resistance to published potential vulnerabilities and well known generic attacks can be achieved in a cost-effective manner.

1077 The search of the information publicly available should be focused on those sources that refer specifically to the product from which the **POI or POI components** is derived. The extensiveness of this search should consider the following factors: **POI or POI components** type, evaluator experience in this type, expected attack potential and the level of ADV evidence available.

1078 The identification process is iterative, where the identification of one potential vulnerability may lead to identifying another area of concern that requires further investigation.

1079 The evaluator will report what actions were taken to identify potential vulnerabilities in the evidence. However, in this type of search, the evaluator may not be able to describe the steps in identifying potential vulnerabilities before the outset of the examination, as the approach may evolve as a result of findings during the search.

1080 The evaluator will report the evidence examined in completing the search for potential vulnerabilities. This selection of evidence may be derived from those areas of concern identified by the evaluator, linked to the evidence the attacker is assumed to be able to obtain, or according to another rationale provided by the evaluator.

6.1.2.6 Action AVA_POI.2.3E

AVA_POI.2-4 The evaluator ***shall conduct*** a search of ST, guidance documentation, functional specification, **POI or POI components** design and security architecture description evidence to identify possible potential vulnerabilities in the **POI or POI components**.

1081 A search of the evidence should be completed whereby specifications and documentation for the **POI or POI components** are analysed and then potential vulnerabilities in the **POI or POI components** are hypothesised, or speculated. The list of hypothesised potential vulnerabilities is then prioritised on the basis of the estimated probability that a potential vulnerability exists and, assuming an exploitable vulnerability does exist the attack potential required to exploit it, and on the extent of control or compromise it would provide. The prioritised list of potential vulnerabilities is used to direct penetration testing against the **POI or POI components**.

1082 The security architecture description provides the developer vulnerability analysis, as it documents how the TSF protects itself from interference from untrusted subjects and prevents the bypass of security enforcement functionality. Therefore, the evaluator should use this description of the protection of the TSF, **as well as the implementation representation and the mapping of the SFRs to this implementation representation,** as a basis for the search for possible ways to undermine the TSF.

1083 Subject to the SFRs the **POI or POI components** is to meet in the operational environment, the evaluator's independent vulnerability analysis should consider generic potential vulnerabilities under each of the following headings:

a) generic potential vulnerabilities relevant for the type of **POI or POI components** being evaluated, as may be supplied by the evaluation authority;

b) bypassing;

c) tampering;

d) direct attacks;

e) monitoring;

f) misuse.

1084 The security architecture description should be considered in light of each of the above generic potential vulnerabilities. Each potential vulnerability should be considered to search for possible ways in which to defeat the TSF protection and undermine the TSF.

AVA_POI.2-5 The evaluator ***shall record*** in the ETR the identified potential vulnerabilities that are candidates for testing and applicable to the **POI or POI components** in its operational environment.

1085 It may be identified that no further consideration of the potential vulnerability is required if for example the evaluator identifies that measures in the operational environment, either IT or non-IT, prevent exploitation of the potential vulnerability in that operational environment. For instance, restricting physical access to the **POI or POI components** to authorised users only may effectively render a potential vulnerability to tampering unexploitable.

1086 The evaluator records any reasons for exclusion of potential vulnerabilities from further consideration if the evaluator determines that the potential vulnerability is not applicable in the operational environment. Otherwise the evaluator records the potential vulnerability for further consideration.

1087 A list of potential vulnerabilities applicable to the **POI or POI components** in its operational environment, which can be used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.

6.1.2.7 Action AVA_POI.2.4E

AVA_POI.2-6 The evaluator ***shall devise*** penetration tests, based on the independent search for potential vulnerabilities.

1088 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the **POI or POI components**, in its operational environment, to the potential vulnerabilities identified during the search of the sources of information publicly available. Any current information provided to the evaluator by a third party (e.g. evaluation authority) regarding known potential vulnerabilities will be considered by the evaluator, together with any encountered potential vulnerabilities resulting from the performance of other evaluation activities.

1089 The evaluator is reminded that, as for considering the security architecture description in the search for vulnerabilities (as detailed in AVA_POI.2-4), testing should be performed to confirm the architectural properties. This is likely to require negative tests attempting to disprove the properties of the security architecture. In developing the strategy for penetration testing, the evaluator will ensure that each of the major characteristics of the security architecture description are tested, either in functional testing (as considered in [CEM] section 14 ATE Class) or evaluator penetration testing.

1090 The evaluator will probably find it practical to carry out penetration test using a series of test cases, where each test case will test for a specific potential vulnerability.

1091 The evaluator is not expected to test for potential vulnerabilities (including those in the public domain) beyond those which required **a POI-Low attack potential**. In some cases, however, it will be necessary to carry out a test before the exploitability can be determined. Where, as a result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond **a POI-Low attack potential**, this is reported in the ETR as a residual vulnerability.

1092 Guidance on determining the necessary attack potential to exploit a potential vulnerability can be found in **[AttackPotPOI]**.

1093 Potential vulnerabilities hypothesised as exploitable only by attackers possessing **POI-Moderate or POI-High attack potential** do not result in a failure of this evaluator action. Where analysis supports the hypothesis, these need not be considered further as an input to penetration testing. However, such vulnerabilities are reported in the ETR as residual vulnerabilities.

1094 Potential vulnerabilities hypothesised as exploitable by an attacker possessing **an attack potential of POI-Low** and resulting in a violation of the security objectives should be the highest priority potential vulnerabilities comprising the list used to direct penetration testing against the **POI or POI components**.

1095 **Devise of penetration testing shall comprise but is not limited to checking the following properties:**

a) **PED prompts are fully under the PED Middle TSF control.**

b) **The POI uses tamper detection and response mechanisms to ensure that POI components in the PED Middle TSF (e.g. the PED display, the PED keypad and the IC Card Reader) become immediately inoperable and erase any secret information which may be stored in the PED Middle TSF components in case of tampering detection.**

c) **The Middle TSF components, if present, e.g. in a POI OPTION or POI COMPREHENSIVE configuration, ensure the integrity protection of POI management and payment data using cryptographic means at the external communication lines.**

d) **The Middle TSF components, if present, e.g. in a POI OPTION or POI COMPREHENSIVE configuration, ensure the authenticity and integrity of administration (e.g. downloading updates) of POI management and transaction processing software and keys, including appropriate cryptographic means.**

1096 **Typically, the Middle TSF comprise user I/Os components and their correct management has to be ensured by the TOE.**

AVA_POI.2-7 The evaluator ***shall produce*** penetration test documentation for the tests based on the list of potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test documentation shall include:

a) identification of the potential vulnerability the **POI or POI components** are being tested for;

b) instructions to connect and setup all required test equipment as required to conduct the penetration test;

c) instructions to establish all penetration test prerequisite initial conditions;

d) instructions to stimulate the TSF;

e) instructions for observing the behaviour of the TSF;

f) descriptions of all expected results and the necessary analysis to be performed on the observed behaviour for comparison against expected results;

g) instructions to conclude the test and establish the necessary post-test state for the **POI or POI components**.

1097 The evaluator prepares for penetration testing based on the list of potential vulnerabilities identified during the search of the public domain and the analysis of the evaluation evidence.

1098 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond those for which **an attack potential of POI-Low** is required to effect an attack. However, as a result of evaluation expertise, the evaluator may discover a potential vulnerability that is exploitable only by an attacker with greater than **POI-Low** attack potential. Such vulnerabilities are to be reported in the ETR as residual vulnerabilities.

1099 With an understanding of the potential vulnerability, the evaluator determines the most feasible way to test for the **POI or POI components**' susceptibility. Specifically the evaluator considers:

a) the TSFI or other **POI** interface that will be used to stimulate the TSF and observe responses (It is possible that the evaluator will need to use an interface to the **POI** other than the TSFI to demonstrate properties of the TSF such as those described in the security architecture description (as required by ADV_ARC). It should the noted, that although these **POI** interfaces provide a means of testing the TSF properties, they are not the subject of the test.);

b) initial conditions that will need to exist for the test (i.e. any particular objects or subjects that will need to exist and security attributes they will need to have);

c) special test equipment that will be required to either stimulate a TSFI or make observations of a TSFI;

d) whether theoretical analysis should replace physical testing, particularly relevant where the results of an initial test can be extrapolated to demonstrate that repeated attempts of an attack are likely to succeed after a given number of attempts.

1100 The evaluator will probably find it practical to carry out penetration testing using a series of test cases, where each test case will test for a specific potential vulnerability.

1101 The intent of specifying this level of detail in the test documentation is to allow another evaluator to repeat the tests and obtain an equivalent result.

AVA_POI.2-8 The evaluator ***shall conduct*** penetration testing.

1102 The evaluator uses the penetration test documentation resulting from work unit AVA_POI.2-6 as a basis for executing penetration tests on the **POI or POI components**, but this does not preclude the evaluator from performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests as a result of information learnt during penetration testing that, if performed by the evaluator, are to be recorded in the penetration test documentation. Such tests may be required to follow up unexpected results or observations, or to investigate potential vulnerabilities suggested to the evaluator during the pre-planned testing.

1103 Should penetration testing show that a hypothesised potential vulnerability does not exist, then the evaluator should determine whether or not the evaluator's own analysis was incorrect, or if evaluation deliverables are incorrect or incomplete.

1104 The evaluator is not expected to test for potential vulnerabilities (including those in the public domain) beyond those which required **an attack potential POI-Low**. In some cases, however, it will be necessary to carry out a test before the exploitability can be determined. Where, as a result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond basic attack potential, this is reported in the ETR as a residual vulnerability.

AVA_POI.2-9 The evaluator ***shall record*** the actual results of the penetration tests.

1105 While some specific details of the actual test results may be different from those expected (e.g. time and date fields in an audit record) the overall result should be identical. Any unexpected test results should be investigated. The impact on the evaluation should be stated and justified.

AVA_POI.2-10 The evaluator ***shall report*** in the ETR the evaluator penetration testing effort, outlining the testing approach, configuration, depth and results.

1106 The penetration testing information reported in the ETR allows the evaluator to convey the overall penetration testing approach and effort expended on this sub-activity. The intent of providing this information is to give a meaningful overview of the evaluator's penetration testing effort. It is not intended that the information regarding penetration testing in the ETR be an exact reproduction of specific test steps or results of individual penetration tests. The intention is to provide enough detail to allow other evaluators and evaluation authorities to gain some insight about the penetration testing approach chosen, amount of penetration testing performed, **POI or POI components** test configurations, and the overall results of the penetration testing activity.

1107 Information that would typically be found in the ETR section regarding evaluator penetration testing efforts is:

a) **POI or POI components** test configurations. The particular configurations of the **POI or POI components** that were penetration tested;

b) TSFI penetration tested. A brief listing of the TSFI and other **POI** interfaces that were the focus of the penetration testing;

c) Verdict for the sub-activity. The overall judgement on the results of penetration testing.

1108 This list is by no means exhaustive and is only intended to provide some context as to the type of information that should be present in the ETR concerning the penetration testing the evaluator performed during the evaluation.

AVA_POI.2-11 The evaluator ***shall examine*** the results of all penetration testing to determine that the **POI or POI components**, in its operational environment, is resistant to an attacker possessing **an attack potential POI-Low**.

1109 If the results reveal that the **POI or POI components**, in its operational environment, has vulnerabilities exploitable by an attacker possessing less than a **POI-Low** attack potential, then this evaluator action fails.

1110 The guidance in B.4 should be used to determine the attack potential required to exploit a particular vulnerability and whether it can therefore be exploited in the intended environment. It may not be necessary for the attack potential to be calculated in every instance, only if there is some doubt as to whether or not the vulnerability can be exploited by an attacker possessing an attack potential less than **POI-Low**.

AVA_POI.2-12 The evaluator ***shall report*** in the ETR all exploitable vulnerabilities and residual vulnerabilities, detailing for each:

a) its source (e.g. CEM activity being undertaken when it was conceived, known to the evaluator, read in a publication);

b) the SFR(s) not met;

c) a description;

d) whether it is exploitable in its operational environment or not (i.e. exploitable or residual).

e) the amount of time, level of expertise, level of knowledge of the **POI or POI components**, level of opportunity and the equipment required to perform the identified vulnerabilities, and the corresponding values using the tables 3 and 4 of Annex [AttackPotPOI]

6.1.3 Evaluation of sub-activity (AVA_POI.3)

6.1.3.1 Objectives

1111 The objective of this sub-activity is to determine whether **the POI or POI components**, in **their** operational environment, **have** vulnerabilities exploitable by attackers possessing **an attack potential of POI-Moderate.**

1112 **The POI or POI components in this sub-activity include at least the Core TSF as defined in the [CC POI PP].**

6.1.3.2 Input

1113 The evaluation evidence for this sub-activity is:

a) the ST;

b) the functional specification;

c) the **POI or POI components** design;

d) the security architecture description;

e) the guidance documentation;

f) the **POI or POI components** suitable for testing;

g) information publicly available to support the identification of possible potential vulnerabilities.

1114 **Further evidence is,**

a) the Core TSF's components hardware and software implementation representation in the three possible configurations of the POI (PED ONLY, POI COMPREHENSIVE, POI OPTION) and the considered attack potential;

b) **a mapping of SFRs to the implementation representation of the Core TSF's components hardware and software**.

1115 The remaining implicit evaluation evidence for this sub-activity depends on the components that have been included in the assurance package. The evidence provided for each component is to be used as input in this sub-activity.

1116 Other input for this sub-activity is:

c) current information regarding public domain potential vulnerabilities and attacks (e.g. from an evaluation authority)

6.1.3.3 Application notes

1117 The evaluator should consider performing additional tests as a result of potential vulnerabilities encountered during other parts of the evaluation.

1118 **The evaluator should use the implementation representation as a guide to penetration testing.**

6.1.3.4 Action AVA_POI.3.1E

AVA_POI.3.1C ***The POI or POI components shall be suitable for testing.***

AVA_POI.3-1 The evaluator ***shall examine*** the **POI or POI components** to determine that the test configuration is consistent with the configuration under evaluation as specified in the ST.

1119 The **POI or POI components** provided by the developer and identified in the test plan should have the same unique reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST introduction.

1120 It is possible for the ST to specify more than one configuration for evaluation. The **POI or POI components** may comprise a number of distinct hardware and software entities that need to be tested in accordance with the ST. The evaluator verifies that all test configurations are consistent with the ST.

1121 The evaluator should consider the security objectives for the operational environment described in the ST that may apply to the test environment and ensure they are met in the testing environment. There may be some objectives for the operational environment that do not apply to the test environment. For example, an objective about user clearances may not apply; however, an objective about a single point of connection to a network would apply.

1122 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to ensure that these resources are calibrated correctly.

AVA_POI.3-2 The evaluator ***shall examine*** the **POI or POI components** to determine that it has been installed properly and is in a known state.

1123 It is possible for the evaluator to determine the state of the **POI or POI components** in a number of ways. For example, previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will satisfy this work unit if the evaluator still has confidence that the **POI or POI components** being used for testing was installed properly and is in a known state. If this is not the case, then the evaluator should follow the developer's procedures to install and start up the **POI or POI components**, using the supplied guidance only.

1124 If the evaluator has to perform the installation procedures because the **POI or POI components** is in an unknown state, this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.

6.1.3.5 Action AVA_POI.3.2E

AVA_POI.3-3 The evaluator ***shall examine*** sources of information publicly available to identify potential vulnerabilities in the **POI or POI components**.

1125 The evaluator examines the sources of information publicly available to support the identification of possible potential vulnerabilities in the **POI or POI components**. There are many sources of publicly available information **or restricted documents** which the evaluator should consider using items such as those available on the world wide web, including:

a) specialist publications (magazines, books);

b) research papers;

c) "Attack Methods to POIs" ([AttackMethPOI]).

1126 The evaluator should not constrain their consideration of publicly available information to the above, but should consider any other relevant information available.

1127 While examining the evidence provided the evaluator will use the information in the public domain to further search for potential vulnerabilities. Where the evaluators have identified areas of concern, the evaluator should consider information publicly available that relate to those areas of concern.

1128 The availability of information that may be readily available to an attacker that helps to identify and facilitate attacks may substantially enhance the attack potential of a given attacker. The accessibility of vulnerability information and sophisticated attack tools on the Internet makes it more likely that this information will be used in attempts to identify potential vulnerabilities in the **POI or POI components** and exploit them. Modern search tools make such information easily available to the evaluator, and the determination of resistance to published potential vulnerabilities and well known generic attacks can be achieved in a cost-effective manner.

1129 The search of the information publicly available should be focused on those sources that refer specifically to the product from which the **POI or POI components** is derived. The extensiveness of this search should consider the following factors: **POI or POI components** type, evaluator experience in this type, expected attack potential and the level of ADV evidence available.

1130 The identification process is iterative, where the identification of one potential vulnerability may lead to identifying another area of concern that requires further investigation.

1131 The evaluator will report what actions were taken to identify potential vulnerabilities in the evidence. However, in this type of search, the evaluator may not be able to describe the steps in identifying potential vulnerabilities before the outset of the examination, as the approach may evolve as a result of findings during the search.

1132 The evaluator will report the evidence examined in completing the search for potential vulnerabilities. This selection of evidence may be derived from those areas of concern identified by the evaluator, linked to the evidence the attacker is assumed to be able to obtain, or according to another rationale provided by the evaluator.

6.1.3.6 Action AVA_POI.3.3E

AVA_POI.3-4 The evaluator ***shall conduct*** a search of ST, guidance documentation, functional specification, **POI or POI components]** design and security architecture description evidence to identify possible potential vulnerabilities in the **POI or POI components**.

1133 A search of the evidence should be completed whereby specifications and documentation for the **POI or POI components** are analysed and then potential vulnerabilities in the **POI or POI components** are hypothesised, or speculated. The list of hypothesised potential vulnerabilities is then prioritised on the basis of the estimated probability that a potential vulnerability exists and, assuming an exploitable vulnerability does exist the attack potential required to exploit it, and on the extent of control or compromise it would provide. The prioritised list of potential vulnerabilities is used to direct penetration testing against the **POI or POI components**.

1134 The security architecture description provides the developer vulnerability analysis, as it documents how the TSF protects itself from interference from untrusted subjects and prevents the bypass of security enforcement functionality. Therefore, the evaluator should use this description of the protection of the TSF, **as well as the implementation representation and the mapping of the SFRs to this implementation representation,** as a basis for the search for possible ways to undermine the TSF.

1135 Subject to the SFRs the **POI or POI components** is to meet in the operational environment, the evaluator's independent vulnerability analysis should consider generic potential vulnerabilities under each of the following headings:

a) generic potential vulnerabilities relevant for the type of **POI or POI components** being evaluated, as may be supplied by the evaluation authority;

b) bypassing;

c) tampering;

d) direct attacks;

e) monitoring;

f) misuse.

1136 The security architecture description should be considered in light of each of the above generic potential vulnerabilities. Each potential vulnerability should be considered to search for possible ways in which to defeat the TSF protection and undermine the TSF.

AVA_POI.3-5 The evaluator ***shall record*** in the ETR the identified potential vulnerabilities that are candidates for testing and applicable to the **POI or POI components** in its operational environment.

1137 It may be identified that no further consideration of the potential vulnerability is required if for example the evaluator identifies that measures in the operational environment, either IT or non-IT, prevent exploitation of the potential vulnerability in that operational environment. For instance, restricting physical access to the **POI or POI components** to authorised users only may effectively render a potential vulnerability to tampering unexploitable.

1138 The evaluator records any reasons for exclusion of potential vulnerabilities from further consideration if the evaluator determines that the potential vulnerability is not applicable in the operational environment. Otherwise the evaluator records the potential vulnerability for further consideration.

1139 A list of potential vulnerabilities applicable to the **POI or POI components** in its operational environment, which can be used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.

6.1.3.7 Action AVA_POI.3.4E

AVA_POI.3-6 The evaluator ***shall devise*** penetration tests, based on the independent search for potential vulnerabilities.

1140 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the **POI or POI components**, in its operational environment, to the potential vulnerabilities identified during the search of the sources of information publicly available. Any current information provided to the evaluator by a third party (e.g. evaluation authority) regarding known potential vulnerabilities will be considered by the evaluator, together with any encountered potential vulnerabilities resulting from the performance of other evaluation activities.

1141 The evaluator is reminded that, as for considering the security architecture description in the search for vulnerabilities (as detailed in AVA_POI.3-4), testing should be performed to confirm the architectural properties. This is likely to require negative tests attempting to disprove the properties of the security architecture. In developing the strategy for penetration testing, the evaluator will ensure that each of the major characteristics of the security architecture description are tested, either in functional testing (as considered in [CEM] section 14 ATE Class) or evaluator penetration testing.

1142 The evaluator will probably find it practical to carry out penetration test using a series of test cases, where each test case will test for a specific potential vulnerability.

1143 The evaluator is not expected to test for potential vulnerabilities (including those in the public domain) beyond those which required **a POI-Moderate attack potential**. In some cases, however, it will be necessary to carry out a test before the exploitability can be determined. Where, as a result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond **a POI-Moderate attack potential**, this is reported in the ETR as a residual vulnerability.

1144 Guidance on determining the necessary attack potential to exploit a potential vulnerability can be found in **[AttackPotPOI]**.

1145 Potential vulnerabilities hypothesised as exploitable only by attackers possessing **a higher attack potential than POI-Moderate** do not result in a failure of this evaluator action. Where analysis supports the hypothesis, these need not be considered further as an input to penetration testing. However, such vulnerabilities are reported in the ETR as residual vulnerabilities.

1146 Potential vulnerabilities hypothesised as exploitable by an attacker possessing **an attack potential of POI-Moderate** and resulting in a violation of the security objectives should be the highest priority potential vulnerabilities comprising the list used to direct penetration testing against the **POI or POI components**.

1147 **Devise of penetration testing for the Core TSF shall comprise but is not limited to checking the following properties:**

a) **PIN entry must be performed without exposure of the PIN digits (e. g. any array related to PIN entry displays only non significant symbols PCI B5) via the PED Keypad which is part of the Core TSF.**

AVA_POI.3-7 The evaluator ***shall produce*** penetration test documentation for the tests based on the list of potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test documentation shall include:

a) identification of the potential vulnerability the **POI or POI components** are being tested for;

b) instructions to connect and setup all required test equipment as required to conduct the penetration test;

c) instructions to establish all penetration test prerequisite initial conditions;

d) instructions to stimulate the TSF;

e) instructions for observing the behaviour of the TSF;

f) descriptions of all expected results and the necessary analysis to be performed on the observed behaviour for comparison against expected results;

g) instructions to conclude the test and establish the necessary post-test state for the **POI or POI components**.

1148 The evaluator prepares for penetration testing based on the list of potential vulnerabilities identified during the search of the public domain and the analysis of the evaluation evidence.

1149 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond those for which **an attack potential of POI-Moderate** is required to effect an attack. However, as a result of evaluation expertise, the evaluator may discover a potential vulnerability that is exploitable only by an attacker with greater than **POI-Moderate** attack potential. Such vulnerabilities are to be reported in the ETR as residual vulnerabilities.

1150 With an understanding of the potential vulnerability, the evaluator determines the most feasible way to test for the **POI or POI components**' susceptibility. Specifically the evaluator considers:

a) the TSFI or other **POI** interface that will be used to stimulate the TSF and observe responses (It is possible that the evaluator will need to use an interface to the **POI** other than the TSFI to demonstrate properties of the TSF such as those described in the security architecture description (as required by ADV_ARC). It should the noted, that although these **POI** interfaces provide a means of testing the TSF properties, they are not the subject of the test.);

b) initial conditions that will need to exist for the test (i.e. any particular objects or subjects that will need to exist and security attributes they will need to have);

c) special test equipment that will be required to either stimulate a TSFI or make observations of a TSFI;

d) whether theoretical analysis should replace physical testing, particularly relevant where the results of an initial test can be extrapolated to demonstrate that repeated attempts of an attack are likely to succeed after a given number of attempts.

1151 The evaluator will probably find it practical to carry out penetration testing using a series of test cases, where each test case will test for a specific potential vulnerability.

1152 The intent of specifying this level of detail in the test documentation is to allow another evaluator to repeat the tests and obtain an equivalent result.

AVA_POI.3-8 The evaluator ***shall conduct*** penetration testing.

1153 The evaluator uses the penetration test documentation resulting from work unit AVA_POI.3-6 as a basis for executing penetration tests on the **POI or POI components**, but this does not preclude the evaluator from performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests as a result of information learnt during penetration testing that, if performed by the evaluator, are to be recorded in the penetration test documentation. Such tests may be required to follow up unexpected results or observations, or to investigate potential vulnerabilities suggested to the evaluator during the pre-planned testing.

1154 Should penetration testing show that a hypothesised potential vulnerability does not exist, then the evaluator should determine whether or not the evaluator's own analysis was incorrect, or if evaluation deliverables are incorrect or incomplete.

1155 The evaluator is not expected to test for potential vulnerabilities (including those in the public domain) beyond those which required **an attack potential POI-Moderate**. In some cases, however, it will be necessary to carry out a test before the exploitability can be determined. Where, as a result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond basic attack potential, this is reported in the ETR as a residual vulnerability.

AVA_POI.3-9 The evaluator ***shall record*** the actual results of the penetration tests.

1156 While some specific details of the actual test results may be different from those expected (e.g. time and date fields in an audit record) the overall result should be identical. Any unexpected test results should be investigated. The impact on the evaluation should be stated and justified.

AVA_POI.3-10 The evaluator ***shall report*** in the ETR the evaluator penetration testing effort, outlining the testing approach, configuration, depth and results.

1157 The penetration testing information reported in the ETR allows the evaluator to convey the overall penetration testing approach and effort expended on this sub-activity. The intent of providing this information is to give a meaningful overview of the evaluator's penetration testing effort. It is not intended that the information regarding penetration testing in the ETR be an exact reproduction of specific test steps or results of individual penetration tests. The intention is to provide enough detail to allow other evaluators and evaluation authorities to gain some insight about the penetration testing approach chosen, amount of penetration testing performed, **POI or POI components** test configurations, and the overall results of the penetration testing activity.

1158 Information that would typically be found in the ETR section regarding evaluator penetration testing efforts is:

a) **POI or POI components** test configurations. The particular configurations of the **POI or POI components** that were penetration tested;

b) TSFI penetration tested. A brief listing of the TSFI and other **POI** interfaces that were the focus of the penetration testing;

c) Verdict for the sub-activity. The overall judgement on the results of penetration testing.

1159 This list is by no means exhaustive and is only intended to provide some context as to the type of information that should be present in the ETR concerning the penetration testing the evaluator performed during the evaluation.

AVA_POI.3-11 The evaluator ***shall examine*** the results of all penetration testing to determine that the **POI or POI components**, in its operational environment, is resistant to an attacker possessing **an attack potential POI-Moderate**.

1160 If the results reveal that the **POI or POI components**, in its operational environment, has vulnerabilities exploitable by an attacker possessing less than **a POI-Moderate attack potential**, then this evaluator action fails.

1161 The guidance in B.4 should be used to determine the attack potential required to exploit a particular vulnerability and whether it can therefore be exploited in the intended environment. It may not be necessary for the attack potential to be calculated in every instance, only if there is some doubt as to whether or not the vulnerability can be exploited by an attacker possessing an attack potential less than **POI-Moderate**.

AVA_POI.3-12 The evaluator ***shall report*** in the ETR all exploitable vulnerabilities and residual vulnerabilities, detailing for each:

a) its source (e.g. CEM activity being undertaken when it was conceived, known to the evaluator, read in a publication);

b) the SFR(s) not met;

c) a description;

d) whether it is exploitable in its operational environment or not (i.e. exploitable or residual).

e) the amount of time, level of expertise, level of knowledge of the **POI or POI components**, level of opportunity and the equipment required to perform the identified vulnerabilities, and the corresponding values using the tables 3 and 4 of Annex [AttackPotPOI].

6.1.4 Evaluation of sub-activity (AVA_POI.4)

6.1.4.1 Objectives

1162 The objective of this sub-activity is to determine whether **the POI or POI components**, in **their** operational environment, **have** vulnerabilities exploitable by attackers possessing **an attack potential of POI-High.**

1163 **The POI or POI components in this sub-activity include at least the Core TSF Keys as defined in the [CC POI PP].**

6.1.4.2 Input

1164 The evaluation evidence for this sub-activity is:

a) the ST;

b) the functional specification;

c) the **POI or POI components** design;

d) the security architecture description;

e) the guidance documentation;

f) the **POI or POI components** suitable for testing;

g) information publicly available to support the identification of possible potential vulnerabilities.

1165 **Further evidence is,**

a) **the Core TSF Keys components hardware and software implementation representation in the three possible configurations of the POI (PED ONLY, POI COMPREHENSIVE, POI OPTION);**

b) **a mapping of SFRs to the implementation representation of the Core TSF Keys components hardware and software.**

1166 The remaining implicit evaluation evidence for this sub-activity depends on the components that have been included in the assurance package. The evidence provided for each component is to be used as input in this sub-activity.

1167 Other input for this sub-activity is:

a) current information regarding public domain potential vulnerabilities and attacks (e.g. from an evaluation authority)

6.1.4.3 Application notes

1168 The evaluator should consider performing additional tests as a result of potential vulnerabilities encountered during other parts of the evaluation.

1169 The evaluator should use the implementation representation as a guide to penetration testing.

6.1.4.4 Action AVA_POI.4.1E

AVA_POI.4.1C ***The POI or POI components shall be suitable for testing.***

AVA_POI.4-1 The evaluator ***shall examine*** the **POI or POI components** to determine that the test configuration is consistent with the configuration under evaluation as specified in the ST.

1170 The **POI or POI components** provided by the developer and identified in the test plan should have the same unique reference as established by the CM capabilities (ALC_CMC) sub-activities and identified in the ST introduction.

1171 It is possible for the ST to specify more than one configuration for evaluation. The **POI or POI components** may comprise a number of distinct hardware and software entities that need to be tested in accordance with the ST. The evaluator verifies that all test configurations are consistent with the ST.

1172 The evaluator should consider the security objectives for the operational environment described in the ST that may apply to the test environment and ensure they are met in the testing environment. There may be some objectives for the operational environment that do not apply to the test environment. For example, an objective about user clearances may not apply; however, an objective about a single point of connection to a network would apply.

1173 If any test resources are used (e.g. meters, analysers) it will be the evaluator's responsibility to ensure that these resources are calibrated correctly.

AVA_POI.4-2 The evaluator ***shall examine*** the **POI or POI components** to determine that it has been installed properly and is in a known state.

1174 It is possible for the evaluator to determine the state of the **POI or POI components** in a number of ways. For example, previous successful completion of the Evaluation of sub-activity (AGD_PRE.1) sub-activity will satisfy this work unit if the evaluator still has confidence that the **POI or POI components** being used for testing was installed properly and is in a known state. If this is not the case, then the evaluator should follow the developer's procedures to install and start up the **POI or POI components**, using the supplied guidance only.

1175 If the evaluator has to perform the installation procedures because the **POI or POI components** is in an unknown state, this work unit when successfully completed could satisfy work unit AGD_PRE.1-3.

6.1.4.5 Action AVA_POI.4.2E

AVA_POI.4-3 The evaluator ***shall examine*** sources of information publicly available to identify potential vulnerabilities in the **POI or POI components**.

1176 The evaluator examines the sources of information publicly available to support the identification of possible potential vulnerabilities in the **POI or POI components**. There are many sources of publicly available information **or restricted documents** which the evaluator should consider using items such as those available on the world wide web, including:

a) specialist publications (magazines, books);

b) research papers;

c) the document “Attack Methods to POIs” ([AttackMethPOI]).

1177 The evaluator should not constrain their consideration of publicly available information to the above, but should consider any other relevant information available.

1178 While examining the evidence provided the evaluator will use the information in the public domain to further search for potential vulnerabilities. Where the evaluators have identified areas of concern, the evaluator should consider information publicly available that relate to those areas of concern.

1179 The availability of information that may be readily available to an attacker that helps to identify and facilitate attacks may substantially enhance the attack potential of a given attacker. The accessibility of vulnerability information and sophisticated attack tools on the Internet makes it more likely that this information will be used in attempts to identify potential vulnerabilities in the **POI or POI components** and exploit them. Modern search tools make such information easily available to the evaluator, and the determination of resistance to published potential vulnerabilities and well known generic attacks can be achieved in a cost-effective manner.

1180 The search of the information publicly available should be focused on those sources that refer specifically to the product from which the **POI or POI components** is derived. The extensiveness of this search should consider the following factors: **POI or POI components** type, evaluator experience in this type, expected attack potential and the level of ADV evidence available.

1181 The identification process is iterative, where the identification of one potential vulnerability may lead to identifying another area of concern that requires further investigation.

1182 The evaluator will report what actions were taken to identify potential vulnerabilities in the evidence. However, in this type of search, the evaluator may not be able to describe the steps in identifying potential vulnerabilities before the outset of the examination, as the approach may evolve as a result of findings during the search.

1183 The evaluator will report the evidence examined in completing the search for potential vulnerabilities. This selection of evidence may be derived from those areas of concern identified by the evaluator, linked to the evidence the attacker is assumed to be able to obtain, or according to another rationale provided by the evaluator.

6.1.4.6 Action AVA_POI.4.3E

AVA_POI.4-4 The evaluator ***shall conduct*** a search of ST, guidance documentation, functional specification, **POI or POI components** design and security architecture description evidence to identify possible potential vulnerabilities in the **POI or POI components**.

1184 A search of the evidence should be completed whereby specifications and documentation for the **POI or POI components** are analysed and then potential vulnerabilities in the **POI or POI components** are hypothesised, or speculated. The list of hypothesised potential vulnerabilities is then prioritised on the basis of the estimated probability that a potential vulnerability exists and, assuming an exploitable vulnerability does exist the attack potential required to exploit it, and on the extent of control or compromise it would provide. The prioritised list of potential vulnerabilities is used to direct penetration testing against the **POI or POI components**.

1185 The security architecture description provides the developer vulnerability analysis, as it documents how the TSF protects itself from interference from untrusted subjects and prevents the bypass of security enforcement functionality. Therefore, the evaluator should use this description of the protection of the TSF, **as well as the implementation representation and the mapping of the SFRs to this implementation representation,** as a basis for the search for possible ways to undermine the TSF.

1186 Subject to the SFRs the **POI or POI components** is to meet in the operational environment, the evaluator's independent vulnerability analysis should consider generic potential vulnerabilities under each of the following headings:

a) generic potential vulnerabilities relevant for the type of **POI or POI components** being evaluated, as may be supplied by the evaluation authority;

b) bypassing;

c) tampering;

d) direct attacks;

e) monitoring;

f) misuse.

1187 The security architecture description should be considered in light of each of the above generic potential vulnerabilities. Each potential vulnerability should be considered to search for possible ways in which to defeat the TSF protection and undermine the TSF.

AVA_POI.4-5 The evaluator ***shall record*** in the ETR the identified potential vulnerabilities that are candidates for testing and applicable to the **POI or POI components** in its operational environment.

1188 It may be identified that no further consideration of the potential vulnerability is required if for example the evaluator identifies that measures in the operational environment, either IT or non-IT, prevent exploitation of the potential vulnerability in that operational environment. For instance, restricting physical access to the **POI or POI components** to authorised users only may effectively render a potential vulnerability to tampering unexploitable.

1189 The evaluator records any reasons for exclusion of potential vulnerabilities from further consideration if the evaluator determines that the potential vulnerability is not applicable in the operational environment. Otherwise the evaluator records the potential vulnerability for further consideration.

1190 A list of potential vulnerabilities applicable to the **POI or POI components** in its operational environment, which can be used as an input into penetration testing activities, shall be reported in the ETR by the evaluators.

6.1.4.7 Action AVA_POI.4.4E

AVA_POI.4-6 The evaluator ***shall devise*** penetration tests, based on the independent search for potential vulnerabilities.

1191 The evaluator prepares for penetration testing as necessary to determine the susceptibility of the **POI or POI components**, in its operational environment, to the potential vulnerabilities identified during the search of the sources of information publicly available. Any current information provided to the evaluator by a third party (e.g. evaluation authority) regarding known potential vulnerabilities will be considered by the evaluator, together with any encountered potential vulnerabilities resulting from the performance of other evaluation activities.

1192 The evaluator is reminded that, as for considering the security architecture description in the search for vulnerabilities (as detailed in AVA_POI.4-4), testing should be performed to confirm the architectural properties. This is likely to require negative tests attempting to disprove the properties of the security architecture. In developing the strategy for penetration testing, the evaluator will ensure that each of the major characteristics of the security architecture description are tested, either in functional testing (as considered in [CEM] section 14 ATE Class) or evaluator penetration testing.

1193 The evaluator will probably find it practical to carry out penetration test using a series of test cases, where each test case will test for a specific potential vulnerability.

1194 The evaluator is not expected to test for potential vulnerabilities (including those in the public domain) beyond those which required **a POI-High attack potential**. In some cases, however, it will be necessary to carry out a test before the exploitability can be determined. Where, as a result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond **a POI-High attack potential**, this is reported in the ETR as a residual vulnerability.

1195 Guidance on determining the necessary attack potential to exploit a potential vulnerability can be found in **[AttackPotPOI]**.

1196 Potential vulnerabilities hypothesised as exploitable only by attackers possessing **a higher attack potential than POI-High** do not result in a failure of this evaluator action. Where analysis supports the hypothesis, these need not be considered further as an input to penetration testing. However, such vulnerabilities are reported in the ETR as residual vulnerabilities.

1197 Potential vulnerabilities hypothesised as exploitable by an attacker possessing **an attack potential of POI-High** and resulting in a violation of the security objectives should be the highest priority potential vulnerabilities comprising the list used to direct penetration testing against the **POI or POI components**.

1198 **Devise of penetration testing shall comprise but is not limited to checking the following properties:**

a) **An attacker with an attack potential lower than POI-High shall not be able to recover any PIN security related cryptographic key from Core TSF Keys;**

b) **The POI uses tamper detection and response mechanisms to ensure that POI components in the Core TSF Keys (e.g. the PED Security Module and the IC Card Reader Security Module) become immediately inoperable and erase any secret information in case of tampering detection.**

AVA_POI.4-7 The evaluator ***shall produce*** penetration test documentation for the tests based on the list of potential vulnerabilities in sufficient detail to enable the tests to be repeatable. The test documentation shall include:

a) identification of the potential vulnerability the **POI or POI components are** being tested for;

b) instructions to connect and setup all required test equipment as required to conduct the penetration test;

c) instructions to establish all penetration test prerequisite initial conditions;

d) instructions to stimulate the TSF;

e) instructions for observing the behaviour of the TSF;

f) descriptions of all expected results and the necessary analysis to be performed on the observed behaviour for comparison against expected results;

g) instructions to conclude the test and establish the necessary post-test state for the POI or POI components.

1199 The evaluator prepares for penetration testing based on the list of potential vulnerabilities identified during the search of the public domain and the analysis of the evaluation evidence.

1200 The evaluator is not expected to determine the exploitability for potential vulnerabilities beyond those for which **an attack potential of POI-High** is required to effect an attack. However, as a result of evaluation expertise, the evaluator may discover a potential vulnerability that is exploitable only by an attacker with greater than **POI-High** attack potential. Such vulnerabilities are to be reported in the ETR as residual vulnerabilities.

1201 With an understanding of the potential vulnerability, the evaluator determines the most feasible way to test for the POI or POI components' susceptibility. Specifically the evaluator considers:

a) the TSFI or other **POI** interface that will be used to stimulate the TSF and observe responses (It is possible that the evaluator will need to use an interface to the **POI** other than the TSFI to demonstrate properties of the TSF such as those described in the security architecture description (as required by ADV_ARC). It should the noted, that although these **POI** interfaces provide a means of testing the TSF properties, they are not the subject of the test.);

b) initial conditions that will need to exist for the test (i.e. any particular objects or subjects that will need to exist and security attributes they will need to have);

c) special test equipment that will be required to either stimulate a TSFI or make observations of a TSFI;

d) whether theoretical analysis should replace physical testing, particularly relevant where the results of an initial test can be extrapolated to demonstrate that repeated attempts of an attack are likely to succeed after a given number of attempts.

1202 The evaluator will probably find it practical to carry out penetration testing using a series of test cases, where each test case will test for a specific potential vulnerability.

1203 The intent of specifying this level of detail in the test documentation is to allow another evaluator to repeat the tests and obtain an equivalent result.

AVA_POI.4-8 The evaluator ***shall conduct*** penetration testing.

1204 The evaluator uses the penetration test documentation resulting from work unit AVA_POI.4-6 as a basis for executing penetration tests on the **POI or POI components**, but this does not preclude the evaluator from performing additional ad hoc penetration tests. If required, the evaluator may devise ad hoc tests as a result of information learnt during penetration testing that, if performed by the evaluator, are to be recorded in the penetration test documentation. Such tests may be required to follow up unexpected results or observations, or to investigate potential vulnerabilities suggested to the evaluator during the pre-planned testing.

1205 Should penetration testing show that a hypothesised potential vulnerability does not exist, then the evaluator should determine whether or not the evaluator's own analysis was incorrect, or if evaluation deliverables are incorrect or incomplete.

1206 The evaluator is not expected to test for potential vulnerabilities (including those in the public domain) beyond those which required **an attack potential POI-High**. In some cases, however, it will be necessary to carry out a test before the exploitability can be determined. Where, as a result of evaluation expertise, the evaluator discovers an exploitable vulnerability that is beyond basic attack potential, this is reported in the ETR as a residual vulnerability.

AVA_POI.4-9 The evaluator ***shall record*** the actual results of the penetration tests.

1207 While some specific details of the actual test results may be different from those expected (e.g. time and date fields in an audit record) the overall result should be identical. Any unexpected test results should be investigated. The impact on the evaluation should be stated and justified.

AVA_POI.4-10 The evaluator ***shall report*** in the ETR the evaluator penetration testing effort, outlining the testing approach, configuration, depth and results.

1208 The penetration testing information reported in the ETR allows the evaluator to convey the overall penetration testing approach and effort expended on this sub-activity. The intent of providing this information is to give a meaningful overview of the evaluator's penetration testing effort. It is not intended that the information regarding penetration testing in the ETR be an exact reproduction of specific test steps or results of individual penetration tests. The intention is to provide enough detail to allow other evaluators and evaluation authorities to gain some insight about the penetration testing approach chosen, amount of penetration testing performed, **POI or POI components** test configurations, and the overall results of the penetration testing activity.

1209 Information that would typically be found in the ETR section regarding evaluator penetration testing efforts is:

a) **POI or POI components** test configurations. The particular configurations of the **POI or POI components** that were penetration tested;

b) TSFI penetration tested. A brief listing of the TSFI and other **POI** interfaces that were the focus of the penetration testing;

c) Verdict for the sub-activity. The overall judgement on the results of penetration testing.

1210 This list is by no means exhaustive and is only intended to provide some context as to the type of information that should be present in the ETR concerning the penetration testing the evaluator performed during the evaluation.

AVA_POI.4-11 The evaluator ***shall examine*** the results of all penetration testing to determine that the **POI or POI components**, in its operational environment, is resistant to an attacker possessing **an attack potential POI-High**.

1211 If the results reveal that the **POI or POI components**, in its operational environment, has vulnerabilities exploitable by an attacker possessing less than **a POI-High attack potential**, then this evaluator action fails.

1212 The guidance in B.4 should be used to determine the attack potential required to exploit a particular vulnerability and whether it can therefore be exploited in the intended environment. It may not be necessary for the attack potential to be calculated in every instance, only if there is some doubt as to whether or not the vulnerability can be exploited by an attacker possessing an attack potential less than **POI-High**.

AVA_POI.4-12 The evaluator ***shall report*** in the ETR all exploitable vulnerabilities and residual vulnerabilities, detailing for each:

a) its source (e.g. CEM activity being undertaken when it was conceived, known to the evaluator, read in a publication);

b) the SFR(s) not met;

c) a description;

d) whether it is exploitable in its operational environment or not (i.e. exploitable or residual).

e) the amount of time, level of expertise, level of knowledge of the **POI or POI components**, level of opportunity and the equipment required to perform the identified vulnerabilities, and the corresponding values using the tables 3 and 4 of Annex [AttackPotPOI].

7 References

POI evaluations shall rely on the current version of the following documents, in particular [AttackPotPOI] and [AttackMethPOI] edited by JTEMS and managed by JIWG.

[AttackPotPOI] *Joint Interpretation Library, Application of Attack Potential to POIs*, current approved version.

[AttackMethPOI] Joint Interpretation Library, Attack Methods for POIs, current approved version.

[CAS] Framework of POI Security Requirements, CAS Common Approval Scheme, 27th October 2008, Version Draft 1.0

[CC] *Common Criteria for Information Technology Security Evaluation, Version 3.1*, Revision 3, July 2009.

[CEM] *Common Methodology for Information Technology Security Evaluation (CEM), Version 3.1*, Revision 3, July 2009.

[CC POI PP] Point of Interaction Protection Profile, Version 2.0, 26th November 2010, Common Approval Scheme

8 Glossary of non-CC acronyms

JTEMS	**JIL Terminal Evaluation Methodology Subgroup**
PED	**PIN Entry Device**
PIN	**Personal Identification Number**
POI	**Point of Interaction**