**Single Evaluation Report ##Template**

as part of the

Evaluation Technical Report, Part B

**Evaluation of CC Assurance Class** **AVA:**

**Evaluation Assurance Level** **EAL POI**

**Version:**  
Version 2.0

**Date:**  
25th January 2019

**Filename:**  
[SER AVA] Template CertID\_POIModel\_AVA\_yymmdd\_vx.x - after JTEMS comments.docx

**Product:**

**Sponsor:**

**Evaluation Facility:**  
##ITSEF name  
##ITSEF address line 1  
##ITSEF address line 2

**Certification ID:** ##cert ID

**Signatures:** ##

**Author(s):** ##

**QS:** ##

*The following document is a template (Version 2.0, 25th January 2019) for the Single Evaluation Report of the Assurance Class AVA\_POI, Assurance Components AVA\_POI.1 iterated over TSF parts MSR, MiddleTSF, PEDMiddleTSF, IC Card Reader TSF, CoreTSF and CoreTSFKeys according to [POI CEM] and [PP]. For each work unit, the template proposes a framework to be used by the evaluator.*

*The evaluator shall document the evaluation results according to the [CEM] and [POI CEM]. In this template, suggestions for coverage of work units are indicated with red text. Red text shall only be considered as a suggestion for content and the evaluator is entitled to provide different arguments as long as the CC and the CEM are followed.*

*Each work unit ends with the final judgment of the evaluator. Each framework includes the evaluator statement of a judgment which may be used in case of a positive result.*

*Note 1: Framework elements are highlighted cursively or marked with ##. The evaluator shall delete highlighted cursively text in his final report.*

Document Information

History of changes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Date | Approved | Changes | Application Note (Reason for change, effects of change on work units, if applicable which comments of certification body were observed) |
| 1.0 | 22.05.2018 |  | Initial version of template | Template generated. Utilised POI-PP v4.0 including IC Card Reader part and POI attack potential definitions. |
| 2.0 | 25 January 2019 |  | Updated template after JTEMS comments |  |

|  |  |  |
| --- | --- | --- |
| Name | Invariant (edit here) | Output value |
| Filename and size | calculated automatically |  |
| Current version | Version 2.0 |  |
| Date | 25th January 2019 | 25th January 2019 |
| Classification | Company Confidential |  |
| TOE name (long) | ##TOE full | ##TOE full |
| TOE name (short) | ##TOE short | ##TOE short |
| Sponsor (long) | ##sponsor full | ##sponsor full |
| Sponsor (short) | ##sponsor short | ##sponsor short |
| Developer (short) | ##developer short | ##developer short |
| Certification ID | ##cert ID | ##cert ID |
| Certification body (long) | ##certification body incl. address | ##certification body incl. address |
| Certification body (short) | ##cert body short | ##cert body short |

Table of contents

[1 Impact in case of a delta process 7](#_Toc536187727)

[2 Basis of the evaluation and documentation used 8](#_Toc536187728)

[3 Evaluation objective / Dependencies 9](#_Toc536187729)

[4 Requirements for evidence and evaluation 10](#_Toc536187730)

[5 Evaluation results 11](#_Toc536187731)

[5.1 AVA\_POI.1 Vulnerability analysis 12](#_Toc536187732)

[5.1.1 AVA\_POI.1.1E 12](#_Toc536187733)

[5.1.2 AVA\_POI.1.2E 15](#_Toc536187734)

[5.1.3 AVA\_POI.1.3E 17](#_Toc536187735)

[5.1.4 AVA\_POI.1.4E 19](#_Toc536187736)

[5.2 Missing Information 28](#_Toc536187737)

[5.3 Questions to / Conditions on the Developer 28](#_Toc536187738)

[5.4 Necessary Changes/Improvements 28](#_Toc536187739)

[5.5 Effects on other Documents 28](#_Toc536187740)

[6 Annex 29](#_Toc536187741)

[6.1 TOE Description 29](#_Toc536187742)

[6.1.1 Input for AVA 29](#_Toc536187743)

[6.2 TOE vulnerabilities 29](#_Toc536187744)

[6.2.1 Identification of Potential vulnerabilities 30](#_Toc536187745)

[6.2.2 Attack scenarios 43](#_Toc536187746)

[6.3 Penetration Tests 62](#_Toc536187747)

[6.3.1 Evaluator testing effort, testing approach, TOE configuration, depth and results 62](#_Toc536187748)

[6.3.2 Partial HW attacks 63](#_Toc536187749)

[6.3.3 HW penetration tests 63](#_Toc536187750)

[6.3.4 Logical penetration tests 64](#_Toc536187751)

[6.4 Code Review 65](#_Toc536187752)

[6.4.1 CR\_01\_Overview on code structure/quality 66](#_Toc536187753)

[6.4.2 CR\_02\_Code optimization and compiler settings 66](#_Toc536187754)

[6.4.3 CR\_03\_Third party software used securely 66](#_Toc536187755)

[6.4.4 CR\_04\_Underlying security hardware requirements 66](#_Toc536187756)

[6.4.5 CR\_05\_Input/output buffer management – parameters checks 67](#_Toc536187757)

[6.4.6 CR\_06\_24 hours Self-test 67](#_Toc536187758)

[6.4.7 CR\_07\_Response to self-test failure 67](#_Toc536187759)

[6.4.8 CR\_08\_Firmware/Application update and authentication 67](#_Toc536187760)

[6.4.9 CR\_09\_Usage and storage of confidential data 67](#_Toc536187761)

[6.4.10 CR\_10\_PIN processing 67](#_Toc536187762)

[6.4.11 CR\_11\_Verification of password 67](#_Toc536187763)

[6.4.12 CR\_12\_Key management 67](#_Toc536187764)

[6.4.13 CR\_13\_RNG 68](#_Toc536187765)

[6.4.14 CR\_14\_Power interrupt protection 68](#_Toc536187766)

[6.5 Exploitable vulnerabilities and residual vulnerabilities 68](#_Toc536187767)

[6.6 Glossary and list of acronyms 69](#_Toc536187768)

[6.7 Bibliography 69](#_Toc536187769)

# Impact in case of a delta process

*## In case of a delta process the impact resulting from the changes that have been applied to the product have to be discussed in this chapter only. Therefore, the evaluator might use the suitable parts of the Impact Analysis Report.*

*## The differences between the certified and the changed TOE should solely be discussed in this chapter. The remaining resp. following chapters should contain the appropriately marked changes with respect to the previous evaluation process. Furthermore the following chapters should not mention the previous TOE to obtain a consistent description allowing further delta process.*

## The current evaluation process is ##not a delta evaluation process.

# Basis of the evaluation and documentation used

The evaluation basis for the current (TOE) is the version 3.1 of the Common Criteria (see [1], [2] and [3]) and the Common Evaluation Methodology (see [4] and [POI CEM]) in accordance with the Security Target [ST].

TOE identification according to [ST]:

|  |  |
| --- | --- |
| Hardware Version | <HW version> |
| Firmware Version: | <FW version> |
| Guidance documents | <guidance docs> |

The subject of the current report is the assessment of the vulnerabilities of the TOE as required by the Assurance Class AVA. This Assurance Class comprises merely one Assurance Family: AVA\_POI (Vulnerability Analysis), whereby it defines several Assurance Components, being dependent on the evaluation assurance package chosen.

The following additional documents [POI AttackPot] and [POI AttackMeth] were used in the course of this evaluation task.

The Developer Action Elements required for the developer are the following:

AVA\_POI.1.1D and AVA\_POI.1.2D

containing the POI components:

* MSR
* MiddleTSF
* PEDMiddleTSF
* IC Card Reader
* CoreTSF
* CoreTSFKeys

The developer contributions are listed in ETR.

There are no further references to former evaluations of the TOE or to any observation reports.

*##Or, in case of a delta evaluation: The evaluator should here refer to the previous certification process and, optionally, give a short description of the main impacting factors.*

# Evaluation objective / Dependencies

The objective of this particular Single Evaluation Report is to find out whether potential vulnerabilities identified during the evaluation of the development and anticipated operation of the TOE or by other methods (e.g. by flaw hypotheses or quantitative or statistical analysis of the security behaviour of the underlying security mechanisms), could allow attackers to violate the SFRs (TSF). Hereby the requirements given by the Common Criteria, [3] are to apply. This report also treats residual vulnerabilities (not violating TSF ), if any.

In detail, the following assurance components are analysed in this report:

**AVA\_POI.1 iterated over the following components**

|  |  |
| --- | --- |
|  |  |
| MSR | Basic POI vulnerability analysis |
| MiddleTSF | Basic POI vulnerability analysis |
| PEDMiddleTSF | Low POI vulnerability analysis |
| IC Card Reader | POI-EnhancedLow vulnerability analysis |
| CoreTSF | Moderate POI Vulnerability Analysis |
| CoreTSFKeys | High POI vulnerability analysis |

According to the section 8.3 of the PP this assurance component implies the following dependencies:

|  |  |
| --- | --- |
| AVA\_POI.1 | ADV\_ARC.1 Security architecture design  ADF\_FSP.2 Security-enforcing Functional Specification  ADV\_TDS.1 Basic design  AGD\_OPE.1 Operational user guidance  AGD\_PRE.1 Preparative procedures |

# Requirements for evidence and evaluation

The evaluation was performed on the basis of the Common Evaluation Methodology [4] and [POI CEM]. The examinations conducted in this report are grouped into work units according to the POI CEM. The following table shows the dependencies between the work units defined by the POI CEM and the Common Criteria assurance elements defined by [3].

An evaluator action element shall be applied to the content and presentation of evidence element. The relevant application instructions are given in the respective work units as shown below:

| No. | evaluator action element (to be applied to content and presentation of evidence elements) | Refinement | related evaluator work units according to [POI CEM] | Verdict |
| --- | --- | --- | --- | --- |
|  | AVA\_POI.1.1E |  |  | **##PASS ##FAIL ##INCONCLUSIVE** |
|  | AVA\_POI.1.1C |  | AVA\_POI.1-1 |  |
|  |  |  | AVA\_POI.1-2 |  |
|  | AVA\_POI.1.2E |  |  | **##PASS ##FAIL ##INCONCLUSIVE** |
|  |  |  | AVA\_POI.1-3 |  |
|  | AVA\_POI.1.3E |  |  | **##PASS ##FAIL ##INCONCLUSIVE** |
|  |  |  | AVA\_POI.1-4 |  |
|  |  |  | AVA\_POI.1-5 |  |
|  | AVA\_POI.1.4E |  |  | **##PASS ##FAIL ##INCONCLUSIVE** |
|  |  |  | AVA\_POI.1-6 |  |
|  |  |  | AVA\_POI.1-7 |  |
|  |  |  | AVA\_POI.1-8 |  |
|  |  |  | AVA\_POI.1-9 |  |
|  |  |  | AVA\_POI.1-10 |  |
|  |  |  | AVA\_POI.1-11 |  |
|  |  |  | AVA\_POI.1-12 |  |

# Evaluation results

**Summary Verdict for the Assurance Class AVA:**  
**##PASS ##FAIL ##INCONCLUSIVE**.  
##If all work units are met: Because all assurance requirements to be examined in this report have a positive evaluation result (PASS), the entire evaluation aspect (assurance class AVA) is assessed with PASS.

##If a work unit is not fulfilled: The TOE does not fulfil all requirements of the assurance component AVA\_POI.1. For further details, please refer to section 5.1below.

## AVA\_POI.1 Vulnerability analysis

**Summary Verdict for the Assurance Component AVA\_POI.1:**  
**##PASS ##FAIL ##INCONCLUSIVE**.  
The **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components** meet all requirements of the assurance component AVA\_POI.1. This result is based on the results provided by the evaluator actions and performed work units below.

### AVA\_POI.1.1E

Evaluator action element:

**AVA\_POI.1.1E:** The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

Content and presentation of evidence elements:

**AVA\_POI.1.1C:** The **Magnetic Stripe Reader component of the POI,** **MiddleTSF’s components,** **PEDMiddleTSF’s components, IC Card Reader components,** **CoreTSF’s components** and **CoreTSFKeys** **components** shall be suitable for testing.

Work units:

**[AVA\_POI.1-1]** The evaluator ***shall examine*** the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader com-ponents, CoreTSF’s components and CoreTSFKeys components** to determine that the test configuration is consistent with the configuration under evaluation as specified in the ST.

The current work unit deals with the question whether the component of the POI under testing is configured in accordance with the ST.

Summary:

The evaluator should examine the TOE to be used for testing and verify that the configuration is consistent with the [ST].

Analysis:

The evaluator should first claim how many TOEs are provided for penetration testing. For each received sample, the evaluator should retrieve the HW and FW versions by following the procedures described in [SEC POLICY] document.

The TOE identification steps performed by the evaluator are described in the following tables. The same steps were performed on all samples, leading to analogous results.

|  |  |  |  |
| --- | --- | --- | --- |
| VER\_01 TOE FW version check | | | |
| **Purpose** | | Check that the Firmware version of the TOE is consistent with the [ST] | |
| **Pre-requisites** | | [ST] and a running TOE. | |
| **Reference** | | [ST] | |
| **Steps** | | **Expected result** | **Actual Result** |
| 1 | Power the TOE on. | TOE Turns on | As expected |
| 2 | #describe how to retrieve the FW version | #... | #... |
| 3 | Compare the claimed FW version in the ST and the retrieved FW version from the tested TOE. | The FW version of the tested TOE is consistent with the FW version in the ST.  The firmware versions claimed in the [ST]:  #List the claimed FW version here  The TOE is expected to display firmware versions compatible with the ones claimed in the [ST]:  #List the tested TOE FW version here. | #... |
| **Result** | | #PASS or #FAIL | |
| **Test executed by** | | #LAB\_NAME (Month/Year) | |

|  |  |  |  |
| --- | --- | --- | --- |
| VER\_02 TOE HW version check | | | |
| **Purpose** | | Check that the hardware version of the TOE is consistent with the [ST] | |
| **Pre-requisites** | | [ST] and a TOE. | |
| **Reference** | | [ST] | |
| **Steps** | | **Expected result** | **Actual Result** |
| 1 | Visually inspect the TOE and retrieve the HW version from the label.  ## describe where is the label. | The HW version of the tested TOE is consistent with the HW version in the ST.    [ST] claims HW versions:  #List the claimed HW version here  The HW version of the tested TOE configuration:  #List the tested TOE HW version here. | #... |
| **Result** | | #PASS or #FAIL | |
| Test executed by | | #LAB\_NAME (Month/Year) | |

The evaluator shall refer to the evaluation results for the ALC\_CMC sub-activities, cf. single evaluation report [##]. There, the following TOE references are managed: ## …

Then the evaluator referred to the operational environment as stated in the ST, sec. ## and found the following conditions being applicable to the test environment: ##. The evaluator checked for fulfilment of these conditions within the current test environment.

Or##: Then the evaluator referred to the operational environment as stated in the ST, sec. ## and did not find any special conditions which could be applicable to the test environment.

Assessment and Verdict:

The evaluator’s analysis showed the following:

* ##not all configurations of the components of the POI declared being under evaluation in the ST are also being tested; the configuration ##ST\_conf\_3 was not tested due to the reason that ##
* the unique reference of the components of the POI under testing is ##not commensurate with the configuration of the component of the POI being under evaluation as stated in the ST and with the component of the POI reference as managed by the CM.

Hence, the current work unit is fulfilled (pass) or is not fulfilled (fail).

**[AVA\_POI.1-2]** The evaluator ***shall examine*** the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components** to determine that it has been installed properly and is in a known state.

The current work unit deals with the description of the security domains maintained by the TSF. These security domains shall be described in the security architecture description.

Summary:

The evaluator should perform the acceptance procedures of the TOE as described in the [SEC POLICY] document (see also AGD assurance class) in order to confirm that the received TOE has been properly installed and that it is in a known state. .

Analysis:

First of all, the evaluator should perform the acceptance procedure steps listed in the [SEC POLICY] document. These acceptance procedure steps are intended to instruct merchants on how to identify the device (both HW and FW) and on how to determine whether any tampering has occurred.

Then the evaluator should list all the received test TOEs here.

Assessment and Verdict:

Based on the analysis above the evaluator confirms (##or disproves) that the component of the POI being tested is exactly commensurate with the component of the POI as it is intended in the ST (‘known state’). He encountered no (##or the following) difficulties during this activity: ##

Hence, the current work unit is fulfilled (pass) or is not fulfilled (fail).

**Verdict for AVA\_POI.1.1E:**  
**##PASS ##FAIL ##INCONCLUSIVE**  
The evaluator confirms (##or disproves) that the information provided in the analysed documentation meet all requirements for content and presentation of evidence.

### AVA\_POI.1.2E

Evaluator action element:

AVA\_POI.1.2E The evaluator *shall perform* a search of public domain sources to identify potential vulnerabilities in the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components**

Work units:

**[AVA\_POI.1-3]** The evaluator ***shall examine*** sources of information publicly available to identify potential vulnerabilities in the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components**.

The current work unit focuses on what actions were taken to identify potential vulnerabilities. A list of potential vulnerabilities applicable to the component of the POI is expected in the context of the work unit AVA\_POI.1-5.

Summary:

The evaluator searched for publicly available sources of information that would allow identification of potential vulnerabilities of the **Magnetic Stripe Reader component,** **MiddleTSF’s components,** **PEDMiddleTSF’s components, IC Card Reader components,** **CoreTSF’s components** and **CoreTSFKeys** **components** of the TOE. In particular, the evaluator:

* searched the web for known vulnerabilities related to the implementation of the **Magnetic Stripe Reader component,** **MiddleTSF’s components,** **PEDMiddleTSF’s components, IC Card Reader components,** **CoreTSF’s components** and **CoreTSFKeys** **components** of the TOE.
* used document [POI AttackMeth] as suggested by the [PP]. In the [POI AttackMeth] document, a list of attack scenarios that are relevant for POI is presented. The evaluator studied each attack scenario and assessed whether it could be exploited through any vulnerability of the TOE. A summary of this analysis is presented in section 6.2.1.

Analysis:

The public vulnerabilities are analysed in section 6.2.1.

The evaluator should list the summary of the public vulnerability analysis of the Magnetic Stripe Reader component, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components of the TOE.

Assessment and Verdict:

Based on the analysis above the evaluator confirms (##or disproves) that he was able to identify potential vulnerabilities in the components of the POI using information available.

Hence, the current work unit is **fulfilled** (pass) or is **not fulfilled** (fail).

**Verdict for AVA\_POI.1.2E:**  
**##PASS ##FAIL ##INCONCLUSIVE**   
The evaluator confirms (##or disproves) that the information provided in the analysed documentation meet all requirements for content and presentation of evidence.

### AVA\_POI.1.3E

Evaluator action element:

AVA\_POI.1.3E The evaluator *shall perform* an independent vulnerability analysis of the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components** using the guidance documentation, functional specification, design and security architecture description **as well as the available implementation representation and the mapping of SFRs to the implementation representation** to identify potential vulnerabilities.

Work units:

**[AVA\_POI.1-4]** The evaluator ***shall conduct*** a search of ST, guidance documentation, functional specification, **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components** design and security architecture description evidence to identify possible potential vulnerabilities in the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components**.

The current work unit focuses on what actions were taken to identify potential vulnerabilities. A final list of potential vulnerabilities applicable to the component of the POI is expected to be provided in the context of work unit AVA\_POI.1-5, where the results of examination of the publicly available sources (work unit AVA\_POI.1-3) and of the current work unit (AVA\_POI.1-3) will be merged.

Summary:

The evaluator performed all sub-activities of the following assurance classes

* ASE
* ADV
* AGD

For each of these assurance classes the evaluator identified raw vulnerabilities (i.e possible potential vulnerabilities) for different TOE components. Such raw vulnerabilities have been collected in section 6.2.1 of this document.

Analysis:

The raw vulnerabilities that the evaluator identified while performing previous evaluation activities are listed in section 6.2.1 for each of the TSF components. All raw vulnerabilities are analysed. After analysing each raw vulnerability is either mitigated or determined as a potential vulnerability. If it is a potential vulnerability, the evaluator shall design a penetration test for it and determine the attack potential after testing.

Assessment and Verdict:

Based on the analysis above the evaluator confirms (##or disproves) that he was able to identify potential vulnerabilities in the components of the POI using components of the POI specific information available.

Hence, the current work unit is fulfilled (pass) or is not fulfilled (fail).

**[AVA\_POI.1-5** The evaluator ***shall record*** in the ETR the identified potential vulnerabilities that are candidates for testing and applicable to the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components** in its operational environment.

The current work unit deals with creating the final list of potential vulnerabilities identified in the component of the POI which may be exploitable in its operational environment.

Summary:

The raw vulnerabilities that were identified in the previous work unit are discussed by the evaluator as candidates for testing in the operational environment of the TOE.

If additional observations sufficiently mitigate or eliminate a raw vulnerability, this is no further considered. Such additional observations of the evaluator might originate from:

* knowledge of the TOE gained by the evaluator through document review,
* inspection of the TOE implementation
* results of other tests/code review.

In contrast, if a raw vulnerability is not eliminated, it is labelled as ‘potential vulnerability’ and it is recorded in section 6.2.1 as a candidate for testing.

Analysis:

**Section 6.2.1** in this SER document reports an analysis of the evaluator on the identified raw vulnerabilities and indicates whether they are eliminated or will become potential vulnerabilities to be recorded as candidates for penetration tests.

Potential vulnerabilities that have been judged applicable to the operational environment of the TOE will be further investigated by the evaluator through penetration testing (see section 6.2.2 of this document and see section 6.3 of this document for description of the performed penetration test).

Assessment and Verdict:

The evaluator confirms (##or disproves) that he was able to create a list of potential vulnerabilities applicable to the component of the POI in its operational environment.

Hence, the current work unit is **fulfilled** (pass) or is **not fulfilled** (fail).

**Verdict for AVA\_POI.1.3E:**  
**##PASS ##FAIL ##INCONCLUSIVE**   
The evaluator confirms (##or disproves) that the information provided in the analysed documentation meet all requirements for content and presentation of evidence.

### AVA\_POI.1.4E

Evaluator action element:

AVA\_POI.1.4E The evaluator *shall conduct* penetration testing, based on the identified potential vulnerabilities, to determine that the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components** is resistant to attacks performed by an attacker possessing **attack potential equal or higher than** **components assigned attack potential with a minimum attack potential for the exploitation phase of a value defined in [POI AttackPot]:**

|  |  |
| --- | --- |
| **MSR** | **POI-Basic** |
| **MiddleTSF** | **POI-Basic** |
| **PEDMiddleTSF** | **POI-Low** |
| **IC Card Reader** | **POI-EnhancedLow** |
| **CoreTSF** | **POI-Moderate** |
| **CoreTSFKeys** | **POI-High** |

*Refinement from [PP]:*

*In particular, for SFR-supporting features related to Open Protocols, the following holds:*

*PCIG2: In particular the evaluator shall exploit public-knowledge vulnerabilities on all SFR-supporting TSFIs of the following types: Link Layer Protocols, IP Protocols, Security Protocols, IP Services. Exploitation methods shall include at least replay of messages and exploitation of insecure exception handling.*

Work units:

**[AVA\_POI.1-6]** The evaluator ***shall devise*** penetration tests, based on the independent search for potential vulnerabilities.

The current work unit deals rather with a ‘mental’ preparation of the evaluator for penetration testing. Together with the next work unit they represent a common context while reflecting different abstraction levels: the current work unit deals with a ‘high level’ outlining sensible attack scenarios planned to be performed as penetration tests, whilst the next one – with producing the related penetration test documentation including the relevant test cases and other necessary conditions for conducting penetration tests.

Due to these deliberations, the activity ‘devising penetration tests’ means – in the current context – ‘inventing appropriate attack scenarios for penetration tests’.

Summary:

In view of the potential vulnerabilities identified in section 6.2.1 of this document, the evaluator performed penetration tests on basis of attack scenarios sourced from [POI AttackMeth] and/or devised by the evaluator (see section 6.2.2 in this document). The penetration tests are described in detail in section 6.3.

Analysis:

Section 6.2.2 of this document reports an overview of all attack scenarios that were considered by the evaluator. Such attack scenarios include:

* All attacks listed in document [POI AttackMeth]
* Additional attack scenarios devised by the evaluator on basis of his knowledge of the TOE.

For each attack scenario in section 6.2.2, the evaluator indicates:

* The targeted TSF component(s);
* The feasibility of the attack (in relation to identified vulnerabilities);
* A link to the performed penetration test (if applicable);
* A verdict.

Assessment and Verdict:

The evaluator confirms (##or disproves) that he was able to create a list of attack scenarios for penetration tests being sensible in components of the POI’s operational environment.

Hence, the current work unit is **fulfilled** (pass) or is **not fulfilled** (fail).

**[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 vulnerabilities the Magnetic Stripe Reader, the MiddleTSF, the PEDMiddleTSF, the IC Card Reader, the CoreTSF and the CoreTSFKeys are being tested for;

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

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 tests and establish the necessary post-test states for the Magnetic Stripe Reader, the MiddleTSF, the PEDMiddle TSF, the IC Card Reader, the CoreTSF and the CoreTSFKeys.

The current work unit deals with producing test documentation (specification) for the penetration tests.

Summary:

The evaluator provides in section 6.3 and subsections of the this document information on:

* the goal of each performed penetration test,
* the necessary pre-requisites
* a step-by-step description of the test approach.
* Indication of expected results
* Actual results
* Test verdict.

Analysis:

The actual results of the test include:

* evidence of the execution of the penetration test and
* a calculation of the attack potential on basis of the time and efforts that are required to perform the attack.

By comparison of the actual results with the expected results, the evaluator expresses a verdict on the test (#PASS or # FAIL).

Assessment and Verdict:

The evaluator confirms (##or disproves) that he was able to create penetration test documentation in sufficient detail to enable the tests to be repeatable.

Hence, the current work unit is **fulfilled** (pass) or is **not fulfilled** (fail).

**[AVA\_POI.1-8]** The evaluator ***shall conduct*** penetration testing.

The current work unit deals with practical conducting penetration testing.

Summary:

The evaluator conducted penetration testing and reported them in the section 6.3 of this document.

Analysis:

See section 6.3 of this document.

Assessment and Verdict:

The evaluator confirms (##or disproves) that he was able to conduct penetration testing according to his test documentation.

Hence, the current work unit is **fulfilled** (pass) or is **not fulfilled** (fail).

**[AVA\_POI.1-9]** The evaluator ***shall record*** the actual results of the penetration tests.

The current work unit deals with recording penetration test results.

Summary:

The evaluator recorded the actual results of the penetration tests and reported them in the section 6.3 of this document.

Analysis:

See section 6.3 of this document.

Assessment and Verdict:

The evaluator confirms (##or disproves) that he was able to record the actual results of the penetration tests.

No deviations were found between the actual results of the penetration tests and the expected test results, which would not explained satisfactory.

##Or: There are some deviations between the actual results of the penetration tests and the expected test results having not been explained satisfactory.

Hence, the current work unit is **fulfilled** (pass) or is **not fulfilled** (fail).

**[AVA\_POI.1-10]** The evaluator ***shall report*** in the ETR the evaluator penetration testing effort, outlining the testing approach, configuration, depth and results.

The current work unit deals with reporting the penetration testing conducted.

Instead of reporting penetration testing effort, outlining the testing approach, configuration, depth and results in the ETR summary, the evaluator reports them in section 6.3.1 of this document.

Summary:

The evaluator reported in section 6.3.1 of the this document an overview of the evaluator penetration testing effort, outlining the testing approach, configuration, depth and results.

Analysis:

In section 6.3.1 of this document, the evaluator a brief overview and summary of the overall testing approach and effort of the evaluation lab in performing hardware and logical penetration tests on the TOE. Note that in the context of the [PP], code review is considered to be part of the activities of AVA\_POI, hence will be included in this section.

Section 6.3.1 of this document explicitly indicates:

* TOE configuration for evaluator penetration tests;
* Test environment (with indication of all SW and HW tools used by the evaluator to perform penetration tests/code review);
* Outline of the testing approach illustrating on what basis the evaluator considers the performed penetration tests as sufficient;
* Overall test results (in terms of exploitable and residual vulnerabilities, explained in further details in section 6.5 of this document).

Assessment and Verdict:

The evaluator confirms (##or disproves) that he was able to report his penetration testing effort.

Hence, the current work unit is **fulfilled** (pass) or is **not fulfilled** (fail).

**[AVA\_POI.1-11]** The evaluator ***shall examine*** the results of all penetration testing to determine that the **Magnetic Stripe Reader component of the POI, MiddleTSF’s components, PEDMiddleTSF’s components, IC Card Reader components, CoreTSF’s components and CoreTSFKeys components**, in its operational environment, are resistant to an attacker possessing an attack potential as follows:

MSR POI-Basic

MiddleTSF POI-Basic

PEDMiddleTSF POI-Low

IC Card Reader POI-EnhancedLow

CoreTSF POI-Moderate

CoreTSFKeys POI-High

The current work unit deals with assessment whether the components of the POI are resistant as claimed in the [ST].

*The current work unit is the only one where the evaluator has to assess the results of the penetration testing concerning their impact on the vulnerability assessment.* The violation of the SFRs is the criteria for decision, whether an identified potential vulnerability is relevant for the components of the POI.   
Please note that there might be SFRs (e.g. FCS\_COP.1) requiring certain security functionality, but not stating which property (confidentiality, integrity, etc.) of an asset is protected by them. In such a case it is necessary also to refer to the security objectives related to this SFR.   
Addressing security objectives might also be necessary for the assessment of sufficiency of security mechanisms covered by ADV\_ARC.

Summary:

The evaluator performed the attacks on the MSR component, the MiddleTSF component, the PEDMiddleTSF component, the ICCR component, the CoreTSF component and the CoreTSFKeys of the TOE on basis of the identified vulnerabilities (see test details in this document in section 6.3). All attacks (##OR NOT ALL ATTACKS) were either un-successful in penetrating the TOE or (when they succeeded) the attack potential calculation exceeded the minimum required.

Analysis:

This indicates that:

The implementation of the MSR component is (#OR IS NOT) resistant against attackers with POI-Basic attack potential

The implementation of the MiddleTSF is(#OR IS NOT) resistant against attackers with POI-Basic attack potential

The implementation of the PEDMiddleTSF is(#OR IS NOT) resistant against attackers with POI-Low attack potential

The implementation of the IC Card Reader component is(#OR IS NOT) resistant against attackers with POI-EnhancedLow attack potential

The implementation of the CoreTSF is(#OR IS NOT) resistant against attackers with POI-Moderate attack potential

The implementation of the CoreTSFKeys is(#OR IS NOT) resistant against attackers with POI-High attack potential

Assessment and Verdict:

The MSR component is(#OR IS NOT) resistant against attackers with POI-Basic attack potential.

The MiddleTSF component is(#OR IS NOT) resistant against attackers with POI-Basic attack potential.

The PEDMiddleTSF component is(#OR IS NOT) resistant against attackers with POI-Low attack potential.

The IC Card Reader component is(#OR IS NOT) resistant against attackers with POI-EnhancedLow attack potential.

The CoreTSF component is(#OR IS NOT) resistant against attackers with POI-Moderate attack potential.

The CoreTSFKeys component is(#OR IS NOT) resistant against attackers with POI-High attack potential.

Hence, the current work unit is fulfilled (pass) ##OR FAIL.

**[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 [POI AttackPot].

The current work unit deals with reporting all exploitable vulnerabilities and residual vulnerabilities found.

Summary:

The evaluator reported in section 6.5 the exploitable and residual potential vulnerabilities.

Analysis:

All the exploitable vulnerabilities and residual vulnerabilities found during the evaluation are reported in section 6.5. the following is reported:

* the source of the potential vulnerability,
* a description of the vulnerability
* the SFR(s) not met
* The attack potential associated to exploitation of the vulnerability.
  + If the calculated attack potential is below the minimum required for the targeted TSF part, the vulnerability is labelled as ‘exploitable’.
  + If the calculated attack potential is above the minimum required for the targeted TSF part, the vulnerability is labeled as ‘residual’.

Assessment and Verdict:

The evaluator identified exploitable vulnerabilities and residual vulnerabilities associated to the MSR component, MiddleTSF components, PEDMiddleTSF components, IC Card Reader components, CoreTSF components and CoreTSFKeys component of the TOE and listed them in section 6.5.

Hence, the current work unit is **fulfilled** (pass) or is **not fulfilled** (fail).

**Verdict for AVA\_POI.1.4E:**  
**##PASS ##FAIL ##INCONCLUSIVE**   
The evaluator confirms (##or disproves) (##or disproves) that the information provided in the analysed documentation meet all requirements for content and presentation of evidence.

## Missing Information

##There is no further information, which the developer/sponsor has to provide.

*##In the case of the verdict ‘inconclusive’, the evaluator is expected to put some issues into the sections ‘Missing Information’ or ‘Questions to and Conditions on the Developer’ of his/her single evaluation report, cf. AIS14.*

## Questions to / Conditions on the Developer

##There are no questions, recommendations to or conditions on the developer.

*##In the case of the verdict ‘inconclusive’, the evaluator is expected to put some issues into the sections ‘Missing Information’ or ‘Questions to and Conditions on the Developer’ of his/her single evaluation report, cf. AIS14.*

## Necessary Changes/Improvements

##There are no changes should be done by the developer.

*##In the case of the verdict ‘fail’, the evaluator is expected to put some issues into the section ‘Necessary Changes/Improvements’ of his/her single evaluation report, cf. AIS14.*

## Effects on other Documents

##There are no effects on other documents.

# Annex

## TOE Description

##After reviewing the evidences of ADV and AGD classes, the evaluator should be able to have a full understanding of the TOE design and TOE implementation. In order to perform the vulnerability analysis, the evaluator should first summarise the TOE design (both HW and FW) in this section.

### Input for AVA

##Based on the TOE description, all the indications for potential vulnerabilities should be reported in the table below. Then later these indications will be further analysed in the Potential Vulnerabilities section (see section 6.2.1).

## TOE vulnerabilities

The evaluator performed a vulnerability analysis that includes:

Searching the public domain for known vulnerabilities of TSF components (this includes search on the web and usage of JIL attack list, as reported in document [POI AttackMeth]); these vulnerabilities are listed as raw vulnerabilities in the section below.

In addition, on basis of the knowledge of the TOE that was gained during documentation review, the evaluator devised additional raw vulnerabilities for each TSF section.

All raw vulnerabilities are then analysed by the evaluator and if they cannot be eliminated by further information gained during the evaluation, they become potential vulnerabilities (see section 6.2.1).

A detailed analysis on the feasibility of all attack scenarios from [POI AttackMeth] document and additional attack scenarios devised by the evaluator can be found in section 6.2.2 of this document.

Section 6.2.2 of this document also includes links to all performed attacks and the evaluator verdict. Actual test results and details on the code review performed by the evaluator are in sections 6.3 and 6.4.

Lastly, as an outcome of the evaluator testing campaign, exploitable and residual vulnerabilities are listed in section 6.5 of this document.

### Identification of Potential vulnerabilities

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Raw Vulnerabilities | | Analysis | | | Potential Vulnerabilities | |
| AVA\_POI.1/MSR | | | | | | | |
| Vulnerabilities from publicly available sources of information | The MSR component of the TOE might be vulnerable to the attack on the MSR from [POI AttackMeth] document. | | To avoid redundancies, the evaluator will analyse all the vulnerabilities related to attacks from the [POI AttackMeth] in the next section (section 6.2.2). | | | See section 6.2.2 | |
| ##The evaluator shall perform public vulnerability search of the MSR component of the TOE and identify raw vulnerabilities/weaknesses | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario (AS.##).  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none | |
| ## add more if needed | | ##add if needed | | | ##add if needed | |
| Vulnerabilities on basis of ST, TOE design documentation, functional specification, security architecture, guidance documentation and implementation representation | ##The evaluator shall perform an independent vulnerability analysis of the MSR component of the TOE and identify raw vulnerabilities/weaknesses. | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none | |
| ## add more if needed | | ##add if needed | | | ##add if needed | |
| AVA\_POI.1/Middle TSF | | | | | | | |
| Vulnerabilities from publicly available sources of information | The Middle TSF component of the TOE might be vulnerable to the attacks from [POI AttackMeth] document. | | To avoid redundancies, the evaluator will analyse all the vulnerabilities related to attacks from the [POI AttackMeth] in the next section (section 6.2.2). | | | See section 6.2.2 | |
| ##The evaluator shall perform public vulnerability search of the MiddleTSF component of the TOE and identify raw vulnerabilities/weaknesses.  Particularly for SFR-supporting features related to Open Protocols, the evaluator should exploit the related public vulnerabilities on all SFR-supporting TSFIs. | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none | |
| ## add more if needed | | ##add if needed | | | ##add if needed | |
| Vulnerabilities on basis of ST, TOE design documentation, functional specification, security architecture, guidance documentation and implementation representation | ##The evaluator shall perform an independent vulnerability analysis of the MiddleTSF component of the TOE and identify raw vulnerabilities/weaknesses. | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none | |
| ## add more if needed | | ##add if needed | | | ##add if needed | |
| AVA\_POI.1/PED Middle TSF | | | | | | | |
| Vulnerabilities from publicly available sources of information | The PED Middle TSF component of the TOE might be vulnerable to the attacks from [POI AttackMeth] document. | | To avoid redundancies, the evaluator will analyse all the vulnerabilities related to attacks from the [POI AttackMeth] in the next section (section 6.2.2). | | | See section 6.2.2 | |
| ##The evaluator shall perform public vulnerability search of the PED MiddleTSF component of the TOE and identify raw vulnerabilities/weaknesses.  Particularly for SFR-supporting features related to Open Protocols, the evaluator should exploit the related public vulnerabilities on all SFR-supporting TSFIs. | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none | |
| ## add more if needed | | ##add if needed | | | ##add if needed | |
| Vulnerabilities on basis of ST, TOE design documentation, functional specification, security architecture, guidance documentation and implementation representation | ##The evaluator shall perform an independent vulnerability analysis of the PED MiddleTSF component of the TOE and identify raw vulnerabilities/weaknesses. | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none | |
| ## add more if needed | | ##add if needed | | | ##add if needed | |
| **AVA\_POI.1/IC Card Reader** | | | | | | | |
| Vulnerabilities from publicly available sources of information | The IC Card Reader TSF component of the TOE might be vulnerable to the attacks from [POI AttackMeth] document. | | To avoid redundancies, the evaluator will analyse all the vulnerabilities related to attacks from the [POI AttackMeth] in the next section (section 6.2.2). | | | See section 6.2.2 | |
| ##The evaluator shall perform public vulnerability search of the IC Card Reader component of the TOE and identify raw vulnerabilities/weaknesses.  Particularly for SFR-supporting features related to Open Protocols, the evaluator should exploit the related public vulnerabilities on all SFR-supporting TSFIs. | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none | |
| ## add more if needed | | ##add if needed | | | ##add if needed | |
| Vulnerabilities on basis of ST, TOE design documentation, functional specification, security architecture, guidance documentation and implementation representation | ##The evaluator shall perform an independent vulnerability analysis of the IC Card Reader component of the TOE and identify raw vulnerabilities/weaknesses. | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none | |
| ## add more if needed | | ##add if needed | | | ##add if needed | |
| AVA\_POI.1/Core TSF | | | | | | | |
| Vulnerabilities from publicly available sources of information | The Core TSF component of the TOE might be vulnerable to the attacks from [POI AttackMeth] document. | | To avoid redundancies, the evaluator will analyse all the vulnerabilities related to attacks from the [POI AttackMeth] in the next section (section 6.2.2). | | | See section 6.2.2 | |
| **Raw vulnerabilities of the CoreTSF component related to flawed code design/implementation are listed below (these raw vulnerabilities are sourced from [ANNEX 4]):** | | |  | | |  |
| **Buffer overflow** | Buffer overflowmight allow an attacker to overwrite the address of the next executable function with the specific memory address, which means an attacker is able to execute other stored functions in the data segment. Furthermore an attacker might bypass the authentication and install malicious software on the TOE via stack buffer overflow. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_02\_Code optimization and compiler settings.**  **🡪 CR\_05\_Input/Output buffer management – parameter check.**  See section 6.4 | | | None |
| **Undefined behaviour (C language)**  Undefined behaviour is unpredictable and might lead to software bugs. | In computer programming, undefined behaviour (UB) is the result of executing computer code whose behaviour is not prescribed by the language specification to which the code adheres, for the current state of the program (e.g. memory). The undefined behaviour in the language might lead to vulnerabilities. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_01\_Overview on code structure/quality**  See section 6.4 | | | None |
| **Aliasing** | Aliasing means that it allows the same piece of memory to be referenced in different ways. This implementation is a common source of ambiguity problem. It might lead an attacker to fool the system and compromise the security of the TOE. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_01\_Overview on code structure/quality**  See section 6.4 | | | None |
| **Poor exception handling** | Poor exception handling might leak the internal error message which contains the information such as stack traces and database dump to an attacker. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_01\_Overview on code structure/quality**  See section 6.4 | | | None |
| **Problematic construct** | Improper pointer usage might lead the TOE to crashes or lead to buffer overflow problems. The attacker might use this feature to compromise the security of the TOE. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_01\_Overview on code structure/quality**  See section 6.4 | | | None |
| **Third party security dependency** | If the TOE relies on any third party software but these software are not used according to the security guide, it might lead to security issues. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_03\_Third party software used securely**  See section 6.4 | | | None |
| If the device relies on the security features provided by the underlying hardware, but these security features are not activated, then an attack or is able to play with the device without the deserved security protection. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_04\_Underlying security hardware requirements**  See section 6.4 | | | None |
| **Compiler settings** | **Compiler optimization** | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_02\_Code optimization and compiler setting**  See section 6.4 | | | None |
| **Secure flags**  If the secure flags are not properly set when compiling, it might lead the TOE exploitable to some public vulnerabilities. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_02\_Code optimization and compiler setting**  See section 6.4 | | | None |
| **Unintended libraries or source code**  If the firmware of the TOE contains any unintended libraries or programming data, it might lead an attacker to attack the TOE by using these resources. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_02\_Code optimization and compiler setting**  See section 6.4 | | | None |
| **Dead code**  The dead code in the firmware leaves the capability of using them together with other exploits like buffer overflows. It might lead the attack or to manipulate the TOE to do something wired without installing any malicious code on it. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_02\_Code optimization and compiler setting**  See section 6.4 | | | None |
| **Minimal configuration** | If the TOE contains unnecessary functions and capabilities, it might increase the probability of penetrating into the TOE. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_02\_Code optimization and compiler setting**  See section 6.4 | | | None |
| **Power interruption** | If sensitive data to be stored in non-volatile memory areas is written in a way being dependent from a power interrupt, an attacker may be able to interfere the writing via power interrupt. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_14\_Power interrupt protection**  See section 6.4 | | | None |
| If error counters are decreased after operation, it lead the attacker be able to cut of the power after the operation but before the error counters down. Thus the attacker is able to try as many times as possible without triggering the counters down. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_14\_Power interrupt protection**  See section 6.4 | | | None |
| If the usage counters are increased after the operation, it might lead an attacker be able to cut of the power after the operation but before the counters up. Thus the attacker is able to operate as many times as possible without triggering the counters up. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_14\_Power interrupt protection**  See section 6.4 | | | None |
| **Good design/poor implementation**  If the TOE is well designed but not well implemented, then this might vitally compromise the security of the TOE. Thus the evaluator considers it is important to spot check the source code | Self-test flow not properly executed. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_06\_24 hours self-test**  See section 6.4 | | | None |
| Failure response to self-test | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_07\_Response to self-test failure**  See section 6.4 | | | None |
| Firmware/application updates authentication | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_08\_Firmware/application update and authentication**  See section 6.4 | | | None |
| * PIN entry * PIN formatting and PIN encryption | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_10\_PIN processing**  See section 6.4 | | | None |
| PAN protection | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_09\_Usage and storage of confidential data**  See section 6.4 | | | None |
| Verification of password | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_11\_Verification of password**  See section 6.4 | | | None |
| Vulnerabilities on basis of ST, TOE design documentation, functional specification, security architecture, guidance documentation and implementation representation | ##The evaluator shall perform an independent vulnerability analysis of the CoreTSF component of the TOE and identify raw vulnerabilities/weaknesses. | | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none |
| **AVA\_POI.1/Core TSF keys** | | | | | | | |
| Vulnerabilities from publicly available sources of information | The CoreTSFKeys portion of the TOE might be vulnerable to attacks described in the [POI AttackMeth] document. | | To avoid redundancies, the evaluator will analyse all the vulnerabilities related to attacks from the [POI AttackMeth] in the next section (section 6.2.2). | | | See section 6.2.2 | |
| **Raw vulnerabilities of the CoreTSF component related to flawed code design/implementation are listed below (these raw vulnerabilities are sourced from [ANNEX 4]):** | | | |  | |  |
| **Good design/poor implementation**  If the TOE is well designed but not well implemented, then this might vitally compromise the security of the TOE. Thus the evaluator considers it is important to spot check the source code. | Random number generator could be improperly implemented/used. | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_13\_RNG**  See section 6.4 | | | None |
| * Key generation * Duplicated key injection * Key usage * Key storage * Key deletion | | Via code inspection, the evaluator verified that this raw vulnerability is not applicable (#or removed or mitigated).  **🡪 CR\_12\_Key management**  See section 6.4 | | | None |
| Vulnerabilities on basis of ST, TOE design documentation, functional specification, security architecture, guidance documentation and implementation representation | ##The evaluator shall perform an independent vulnerability analysis of the CoreTSFKeys component of the TOE and identify raw vulnerabilities/weaknesses. | | | ##Put here the evaluator analysis on the raw vulnerability. Is the vulnerability applicable? Is it countered? Is it mitigated?  If countered by TOE implementation/guidance or proven inapplicable by ATE tests, the raw vulnerability is considered solved and not further discussed by the evaluator.  Otherwise it is listed as potential vulnerability in the column on the right and the evaluator must devise an attack scenario that might exploit the vulnerability. | | | ##In order to test whether the potential vulnerability (if any) can be exploited, the evaluator must identify an attack scenario.  Insert here the unique identifier to the attack scenario.  Such attack scenario will then be described and addressed in section 6.2.2  Or  #none |

Table 1 Summary of TOE vulnerabilities collected from publicly available sources of information, from review of TOE documentation and from inspection of TOE implementation

### Attack scenarios

In this table, the evaluator identifies the test plan to be executed in response to the vulnerability analysis conducted in the previous section.

| Attack Scenario | Targeted TSF component and minimum attack potential | Analysis on Feasibility of Attack | Penetration Tests Performed by the Evaluator | Verdict |
| --- | --- | --- | --- | --- |
| **From [POI AttackMeth]** | | | | |
| **AS.1: Insert PIN Disclosing Bug on Flex PCB** | IC Card Reader TSF  Requires POI-EnhancedLow attack potential (minimum 20 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.2: Wire Hook Attack** | IC Card Reader TSF  Requires POI-EnhancedLow attack potential (minimum 20 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.3: Attack Redundant Keys** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.4: Monitoring IC Card Supply** | IC Card Reader TSF Requires POI-EnhancedLow attack potential (minimum 20 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.5: Monitoring Keyboard Scan Signal on Power Supply** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.6: Determine Keys by Side Channel Analysis** | CoreTSFkeys  Requires POI-High attack potential (minimum 35 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.7: Removal sensor deactivation** | No minimum attack potential is indicated as this is only a partial attack | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.8: Case Switch Deactivation** | No minimum attack potential is indicated as this is only a partial attack | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.9: Attack PCB Switch** | No minimum attack potential is indicated as this is only a partial attack | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.10: Penetrate printed grids** | No minimum attack potential is indicated as this is only a partial attack | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.11: Penetrate Potted Module** | No minimum attack potential is indicated as this is only a partial attack. | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.12: Probing into Secure Modules** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.13: Wire/finger/PCB routing Probing of Single Chip BGA** | CoreTSFkeys  Requires POI-High attack potential (minimum 35 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.14: Chip Level Attacks** | CoreTSFkeys  Requires POI-High attack potential (minimum 35 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.15: Physical Reading of Non Volatile Memories Content** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.16: Perturbation Attacks on Lock Bits** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.17: Perturbation Attack on Software** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.18: DFA Caused by Transient Perturbation** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.19: Behavior Analysis Attacks Caused by Transient Perturbations** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.20: Environmental Perturbation Attacks** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.21: Data reminisce of Volatile Memories Content** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.22: Cut Key Block** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.23: Rip Out Rubber Keys** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.24: Monitoring Keyboard Sound** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.25: Monitoring Electromagnetic Emanation** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.26: Microwave Scanning** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.27: Monitoring Keyboard Emanation** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.28: Attack on True Random Number Generator (TRNG)** | CoreTSFkeys  Requires POI-High attack potential (minimum 35 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.29: Attack on Pseudo Random Number Generator (PRNG)** | CoreTSFkeys  Requires POI-High attack potential (minimum 35 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.30: Off-device Attacks**  **-Information gathering** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **Off-device Attacks**  **- Editing commands** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **Off-device Attacks**  **- Direct protocols Attacks** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **Off-device Attacks**  **- Man-in-the-middle** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **Off-device Attacks**  **- Replay attacks** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **Off-device Attacks**  **- Buffer overflow** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.31: On-device Attacks**   * **Secure Operating System:**   **Direct memory accesses** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **On-device Attacks**   * **Secure Operating System:**   **Illegal access to buffers** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **On-device Attacks**   * **Hypervisor:**   **Attack on memory de-allocation** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **On-device Attacks**   * **Hypervisor:**   **Attack the interruption management** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **On-device Attacks**   * **Hypervisor:**   **Hidden communication channel** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **On-device Attacks**   * **Virtual machine:**   **Ill-formed code** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **On-device Attacks**   * **Virtual machine:**   **Buffer/Stack overflow** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **On-device Attacks**   * **Virtual machine:**   **Illegal use of features (API methods)** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.32: Limit Key Encryption Key search by Check Value** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **As.33: Weakly Padded PIN Blocks** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.34: Exhaustive PIN Search on Secondary PIN Related Functions** | CoreTSF  Requires POI-Moderate attack potential (minimum 26 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.35: Clear Keypad Entry and Display Modification** | PEDMiddleTSF  Requires POI-Low attack potential (minimum 18 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **AS.36: Attack Magnetic Head Using Doors or Covers** | MSR  Requires POI-Basic attack potential (minimum 16 points) | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| **Additional evaluator attack scenarios** | | | | |
| **AS.37:**  #additional attack scenario devised by the evaluator in an attempt at exploiting a potential vulnerability identified in section 6.2.1 | ##add brief description of attack scenario, indicate the targeted TSF component and the minimum attack potential. | ##The evaluator must determine whether this attack scenario is applicable to the TOE in its operational environment. For this assessment, the evaluator may also estimate the attack potential. If this exceeds the minimum required, the attack is deemed not feasible and there is no need to execute a penetration test on the attack scenario.  Otherwise, the evaluator will perform a suitable penetration test (reference will be in next column). | ##none  or  The evaluator includes here a reference to relevant subsections of section 6.3, where the test steps are described and test results are shown. | #Pass/fail |
| #add more if needed | #add more if needed | #add more if needed | #add more if needed | #add more if needed |

Table 2 Attack table

## Penetration Tests

### Evaluator testing effort, testing approach, TOE configuration, depth and results

As part of work units AVA\_POI.1-10 for MSR, MiddleTSF, PEDMiddleTSF, IC Card Reader, CoreTSF and CoreTSFKey, the evaluator provides in this section a brief overview and summary of the overall testing approach and effort of the evaluation lab in performing hardware and logical penetration tests on the TOE. Note that in the context of the [PP], code review is considered to be part of the activities of AVA\_POI, hence will be included in this section.

#### TOE configuration for Evaluator tests

#the evaluator should perform TOE configuration check of the tested samples and report here.

#### Test environment

#The evaluator should list all the test tools and test instructions on how to prepare the testing environment.

#### Evaluator testing approach, efforts and results

The evaluator performed hardware and logical penetration tests and code review in order to ensure that:

* All TSFIs of the TOE have been covered by tests. Note that when sufficient assurance on the secure behaviour of a TSFI was gained through ATE tests performed by either the vendor or the evaluator, no further penetration tests have been performed by the evaluator;
* All SFRs of the TOE have been covered by tests (either ATE tests, penetration tests or code review);
* All vulnerabilities that were identified in the course of previous evaluation activities have been discussed and tested when judged necessary;
* the source code of the device is resistant against the identified vulnerabilities.

In section 6.3.3 and in section 6.3.4, for each hardware and logical penetration test the evaluator reports:

* Test goal
* Test pre-requisites (TOE configuration, necessary HW and/or SW tools)
* Step-by-step description of the test
* Expected result of each step
* Actual results observed by the evaluator for each step (including photographic evidence when appropriate);
* Overall test verdict
* Date of the test.

##if all the tests passed: All tests passed (no exploitable vulnerabilities were found), however a number of residual vulnerabilities have been identified by the evaluator and are listed in section 6.5.

##If any tests failed: Not All tests passed. The exploitable vulnerabilities and a number of residual vulnerabilities identified by the evaluator are listed in section 6.5.

### Partial HW attacks

This section details individual steps that the evaluator performed in order to physically access the TOE. These steps are considered as ‘partial’ attacks as they do not allow an attacker to compromise any sensitive asset of the TOE unless included in a full attack path.

These partial attacks are referred to during execution of the penetration hardware attacks (see section 6.3.3).

##The evaluator should report all the partial hardware attacks here.

#### ##partial HW attack name

##…

### HW penetration tests

This section includes details on the hardware penetration tests that were performed by the evaluator, and it is organized as follows:

* For each test, goal, pre-requisites, expected results, actual results and verdict are presented.
* Steps performed during the identification and exploitation stages are listed in the ‘Cost calculation’ subsection of each attack,
* When required, the ‘Cost calculation’ subsection includes references to section 6.3.2, where more complete descriptions of individual steps are given.

#### PEN\_HW\_01\_##penetration test name

|  |  |
| --- | --- |
| PEN\_HW\_01\_##penetration test name | |
| **Goal** |  |
| **Pre-requisites** |  |
| **Expected results** |  |
| **Results** |  |
| **Test verdict** | #Pass or #Fail |
| **Test executed by** | #LAB\_NAME (Month/Year) |

##### Cost calculation

#the evaluator should describe the identification phase and the exploitation of the attack, and then calculate the attack potential of the attack.

### Logical penetration tests

This section includes details on the software penetration tests that were performed by the evaluator, and it is organized as follows:

* For each test, goal, pre-requisites, expected results, actual results and verdict are presented.
* When required, further details are provided in the ‘Test details’ subsection of each test.

#### PEN\_FW\_01\_##penetration test name

|  |  |
| --- | --- |
| PEN\_FW\_01\_##penetration test name | |
| **Goal** |  |
| **Pre-requisites** |  |
| **Expected results** |  |
| **Results** |  |
| **Test verdict** | #Pass or #Fail |
| **Test executed by** | #LAB\_NAME (Month/Year) |

##### Test details

#the evaluator should describe the test details here.

## Code Review

As verification checks and in response to the code review guideline from the scheme, the evaluator performed review on the source code of the TOE provided by the vendor. In addition, the evaluator performed code review based on the vulnerability analysis (section 6.2.1) in order to gain sufficient assurance on the secure implementation of the TOE against attack scenarios.

These areas cover the most important and necessary security features that protect the assets of the TOE, including PIN, keys, software. From these areas the evaluator can confirm whether the implementation of the TOE is the same as claimed in documentation and whether the TOE is well protected.

The code review coverage of the requirements from the [ANNEX 4] is shown in Table 3.

|  |  |  |
| --- | --- | --- |
| Scheme requirements in [ANNEX 4] | Covered by the code review performed by the evaluator | verdict |
| No 1 | CR\_01\_Overview on code structure/quality | **#Pass or #Fail** |
| No 2 | CR\_02\_Code optimization and compiler settings | **#Pass or #Fail** |
| No 3 | CR\_03\_Third party software used securely | **#Pass or #Fail** |
| No 4 | CR\_04\_Underlying security hardware requirements | **#Pass or #Fail** |
| No 5 | CR\_06\_24 hours self-test  CR\_07\_Response to self-test failure | **#Pass or #Fail** |
| No 6 | CR\_08\_Firmare/Application update and authentication | **#Pass or #Fail** |
| No 7 | CR\_09\_Usage and storage of confidential data | **#Pass or #Fail** |
| No 8 | CR\_02\_Code optimization and compiler setting | **#Pass or #Fail** |
| No 9 | CR\_05\_Input/output buffer management – parameter checks | **#Pass or #Fail** |
| No 10 | CR\_11\_Verification of password | **#Pass or #Fail** |
| No 11 | CR\_02\_Code optimization and compiler setting  CR\_06\_24 hours self-test  CR\_08\_Firmare/Application update and authentication | **#Pass or #Fail** |
| No 12 | CR\_09\_Usage and storage of confidential data | **#Pass or #Fail** |
| No 13 | CR\_10\_PIN processing | **#Pass or #Fail** |
| No 14 | CR\_10\_PIN processing | **#Pass or #Fail** |
| No 15 | CR\_10\_PIN processing | **#Pass or #Fail** |
| No 16 | CR\_10\_PIN processing | **#Pass or #Fail** |
| No 17 | CR\_13\_RNG | **#Pass or #Fail** |
| No 18 | CR\_12\_Key management  CR\_13\_RNG | **#Pass or #Fail** |
| No 19 | CR\_02\_Code optimization and compiler settings | **#Pass or #Fail** |
| No 20 | Covered in the public vulnerability search in section 6.2.1. | **#Pass or #Fail** |
| No 21 | Covered in the public vulnerability search in section 6.2.1. | **#Pass or #Fail** |
| No 22 | CR\_02\_Code optimization and compiler setting | **#Pass or #Fail** |
| No 23 | CR\_01\_Overview on code structure/quality | **#Pass or #Fail** |
| No 24 | CR\_01\_Overview on code structure/quality | **#Pass or #Fail** |
| No 25 | CR\_05\_Input/output buffer management – parameter checks | **#Pass or #Fail** |
| No 26 | CR\_10\_PIN processing | **#Pass or #Fail** |
| No 27 | CR\_02\_Code optimization and compiler settings | **#Pass or #Fail** |
| No 28 | CR\_01\_Overview on code structure/quality | **#Pass or #Fail** |
| No 29 | CR\_05\_Input/output buffer management – parameter checks | **#Pass or #Fail** |
| No 30 | CR\_01\_Overview on code structure/quality | **#Pass or #Fail** |
| No 31 | CR\_14\_Power interrupt protection | **#Pass or #Fail** |
| No 32 | CR\_14\_Power interrupt protection | **#Pass or #Fail** |
| No 33 | CR\_14\_Power interrupt protection | **#Pass or #Fail** |

Table 3 Code review coverage

### CR\_01\_Overview on code structure/quality

#...

### CR\_02\_Code optimization and compiler settings

#...

### CR\_03\_Third party software used securely

#...

### CR\_04\_Underlying security hardware requirements

#...

### CR\_05\_Input/output buffer management – parameters checks

#...

### CR\_06\_24 hours Self-test

#...

### CR\_07\_Response to self-test failure

#...

### CR\_08\_Firmware/Application update and authentication

#...

### CR\_09\_Usage and storage of confidential data

#...

### CR\_10\_PIN processing

#...

### CR\_11\_Verification of password

#...

### CR\_12\_Key management

#...

#### Key generation

#...

#### Key injection

#...

#### Key storage

#...

### CR\_13\_RNG

#...

### CR\_14\_Power interrupt protection

#...

## Exploitable vulnerabilities and residual vulnerabilities

See [ETR] report.

## Glossary and list of acronyms

All used terms are defined in the Common Criteria standard or in the [PP].

## Bibliography

Criteria and Methodology

[1] ##Common Criteria for Information Technology Security Evaluation, Part 1, Version 3.1, revision 4, September 2012

[2] ##Common Criteria for Information Technology Security Evaluation, Part 2, Version 3.1, revision 4, September 2012

[3] ##Common Criteria for Information Technology Security Evaluation, Part 3, Version 3.1, revision 4, September 2012

[4] ##Common Methodology for Information Technology Security Evaluation, Version 3.1, revision 4, September 2012

[POI CEM] ##Joint Interpretation Library – CEM Refinements for POI Evaluation, Version 1.0, 27th May 2011. *Note: POI evaluations shall rely on the current version of this document at the moment of the evaluation.*

[PP] ##Point of Interaction Protection Profile, Version 4.0, 6th March 2015*.*

[ANNEX 4] ##Common.SECC Source Code Analysis Requirements, Version 0.9, 18th April 2017*.*

[POI AttackPot] ##Joint Interpretation Library / Application of Attack Potential to POIs, Version 1.92, 11th August 2014. *Note: POI evaluations shall rely on the current version of this document at the moment of the evaluation.*

[POI AttackMeth] ## Joint Interpretation Library / Attack Methods for POIs, Version 1.94, February 2015. *Note: POI evaluations shall rely on the current version of this document at the moment of the evaluation.*

Developer Documents

See ETR for full list

Evaluation Reports

[ETR] ##Title ETR, ##Author, Version ##, ##Date

See ETR for full list of evaluation reports.