Document history

Version 0.9.2, August 18th, 2016

Federal Office for Information Security
Post Box 20 03 63
D-53133 Bonn
Phone: +49 22899 9582-0
E-Mail: eid@bsi.bund.de
Internet: https://www.bsi.bund.de

Table of Contents

Document history ... 2

1 Introduction ... 5

2 PP Module Introduction ... 6

2.1 PP Module Identification ... 6

2.2 Base PPs ... 6

2.3 TOE Overview ... 6

2.3.1 TOE Definition, Operational Usage, and Difference to Base PP ... 6

2.3.2 TOE Major Security Features for Operational Use ... 7

2.3.3 TOE Type ... 7

2.3.4 TOE Life Cycle ... 7

2.3.5 Non-TOE Hardware/Software/Firmware ... 7

2.4 Consistency Rationale with base PP ... 7

2.4.1 TOE Type ... 7

2.4.2 Security Problem Definition (SPD) ... 7

2.4.3 Security Objectives and Security Functional Requirements ... 8

2.4.4 Conclusion ... 10

3 Conformance Claims ... 11

3.1 CC Conformance Claim ... 11

3.2 Conformance Statement ... 11

4 Security Problem Definition ... 12

4.1 Introduction ... 12

4.1.1 Assets ... 12

4.1.2 Subjects ... 13

4.2 Threats ... 13

4.3 Organizational Security Policies ... 13

4.4 Assumptions ... 14

5 Security Objectives ... 15

5.1 Security Objectives for the TOE ... 15

5.2 Security Objectives for the Environment ... 15

5.2.1 Security Objectives for the Development and Production Environment ... 15

5.2.2 Security Objectives for the Operational Environment ... 15

5.3 Security Objective Rationale ... 16

6 Extended Components Definition ... 18

7 Security Requirements ... 19

7.1 Security Functional Requirements ... 19

7.1.1 Class FCS ... 19

7.1.2 Class FIA ... 22

7.1.3 Class FDP ... 23

7.1.4 Class FAU ... 27

7.1.5 Class FMT ... 28

7.1.6 Class FPT ... 29

7.1.7 Class FTP Trusted Path/Channels ... 31

7.2 Security Requirements Rationale ... 31

7.2.1 Security Functional Requirements Rationale ... 31
7.2.2 Rationale for SFR's Dependencies ... 33
7.2.3 Security Assurance Requirements Rationale ... 33
7.2.4 Security Requirements – Internal Consistency ... 33
8 PP-Configuration ... 34
8.1 Reference ... 34
8.2 Components Statement ... 34
8.3 Conformance Statement ... 34
8.4 Conformity to Security Assurance Requirements ... 34
Glossary and Abbreviations ... 35
Glossary ... 35
Abbreviations ... 35
Reference Documentation ... 37

Tables

Table 1: Security Objective Rationale ... 16
Table 2: Coverage of Security Objectives for the TOE by SFRs ... 32

1 Introduction

This document consists of two parts:

- Chapter 8 defines the certified *Common Criteria PP-Configuration Machine Readable Electronic Documents - Optionales Nachladen (Optional Post-Emission Updates) [MR.ED-ON-PP]*, identified with BSI-CC-PP-0090-2016. This PP-Configuration is based on [MR.ED2.0], and one PP-module (see below).
- Chapters 2-7 define the *Common Criteria PP-Module Machine Readable Electronic Documents - Optionales Nachladen (Optional Post-Emission Updates) [MR.ED-ON-PP-Module]*. Note that according to [CC-Mod], *"The evaluation of a PP-module alone is meaningless. A PP-module has to be evaluated as part of a PP-configuration"*. Hence, the registration id BSI-CC-PP-0090-2016 identifies the certified configuration, whereas the module is identified by its title and version. Also, whereas of course other PP-Configurations than BSI-CC-PP-0090-2016 can be built using this module, such configurations require separate certifications.

2 PP Module Introduction

This section provides document management and overview information required to register the protection profile module (PP module) and to enable a potential user of the module to determine, whether it is of interest.

2.1 PP Module Identification

Title:	Common Criteria PP Module Machine Readable Electronic Documents - Optionales Nachladen (Optional Post-Emission Updates) [MR.ED-ON-PP-Module]
Editor/Sponsor:	Bundesamt für Sicherheit in der Informationstechnik (BSI)
CC Version:	3.1 (Revision 4)
Assurance Level:	Minimum assurance level for this PP is EAL4 augmented.
General Status:	final
Version Number:	Version 0.9.2 as of August 18th, 2016
Registration:	-[1]
Keywords:	electronic document, smart card, update mechanism

2.2 Base PPs

The single base PP for this protection profile module is:

Title:	Common Criteria Protection Profile Machine-Readable Electronic Documents based on BSI TR-03110 for Official Use [MR.ED-PP]
Registration:	BSI-CC-PP-0087-V2-2016-MA-01
Version:	Version 2.0.3, July 18th, 2016

2.3 TOE Overview

2.3.1 TOE Definition, Operational Usage, and Difference to Base PP

The definition of the TOE and its operational usage is the same as in [MR.ED2.0] with one exception: Here, the TOE additionally has the ability to update its TOE software during the life-cycle phase *operational use* by an update mechanism. The security of this update mechanism is subject of this PP module.

Note that this PP module is *only* concerned with the update mechanism itself and its security. It is assumed that updates must be authorized by a central entity (e.g. some entity in charge of the security of the electronic document, the document issuer, the manufacturer, the TOE software developer or some other entity) prior application, and supports this process by cryptographic means.

In particular, the updated TOE software is out of scope of this PP module. No assumption is made on the quality and security of the update. To make the point, installing a completely flawed TOE software update that creates new security vulnerabilities or deactivates security mechanisms of the original TOE via the update mechanism would be absolutely valid under the assumptions of this PP module.

1 according to [CC-Mod]: *"The evaluation of a PP-module alone is meaningless. A PP-module has to be evaluated as part of a PP-configuration"*. Hence, this PP module has no registration id. It should be referred to simply by its title and version.

Hence, *after* a software update is installed via the mechanism described in this PP module, the TOE is *not in a certified mode afterwards*. To be in a certified mode, the whole TOE including the *software update that was*
45 *installed* must be re-certified.

2.3.2 TOE Major Security Features for Operational Use

The TOE here has all security features of the TOE defined in [MR.ED2.0]. In addition, it allows to update the TOE software during the life-cycle phase *operational use*.

2.3.3 TOE Type

Except for the update mechanism, the TOE is exactly the same as in [MR.ED2.0].

2.3.4 TOE Life Cycle

The life-cycle is the same as in [MR.ED2.0].

2.3.5 Non-TOE Hardware/Software/Firmware

50 In [MR.ED2.0], several types of terminals are listed. These are used to connect to the TOE and access data on the TOE. Here, one additional type of terminal must be considered, namely an *update terminal*. Such an update terminal is used to read out version information of the TOE software, read update log data, and to install new TOE software on the TOE.

2.4 Consistency Rationale with base PP

This section analyzes consistency of the TOE type, the SPD, and security objectives of the base PP with those
55 of this PP module.

2.4.1 TOE Type

The TOE type of [MR.ED2.0] is

a smartcard programmed according to [TR03110-1] and [TR03110-2]. The smartcard contains multiple applications (at least one).

The TOE type is exactly the same as in this TOE.

2.4.2 Security Problem Definition (SPD)

Threats

60 Compared to the SPD of [MR.ED2.0], this PP module adds the threat **T.FaTSF - Faulty TSF** and **T.UaU Unauthorized Update**.

The base PP defines several threats that address misuse and direct access of user data (or indirect by weakening TSF), and it specifically includes the life cycle phase *operational use*. There, the threat is considered, it's future implications are estimated, and countered. All this includes the implicit assumption –
65 as with all CC – that this is done at the time of evaluation.

The threat T.FaTSF - Faulty TSF is a meta-threat that concerns *unexpected* technical developments such as e.g. breakthroughs in cryptanalysis, and/or flaws that manifest itself after the TOE enters the life cycle phase *operational use*.

This scenario was not considered in [MR.ED2.0]. Consistency can only be violated if there is a contradiction between the threats of the base PP and those defined here.

The threat T.FaTSF - Faulty TSF considers faulty TSF, where the fault manifests itself unexpectedly within the life-cycle operational use. None of the threats of the base PP consider this scenario. Hence this is an additional threat.

In order to counter the previously introduced threats, security functional requirements need to be introduced, resulting in additional TSF, namely a mechanism to apply software updates. This mechanism itself can be subject to security flaws and has the potential to be misused by an attacker. This is targeted in the threat T.UaU Unauthorized Update.

Hence, these threats consider only scenarios that are a strict superset of those considered in the base PP.

Organizational Security Policies

This PP module adds new organizational security policies **P.Code_Confidentiality**, **P.Secure_Environment**, and **P.Eligible_Terminals_Only**. They define new policies, but these policies only address the newly considered security issues that can arise when a TOE that is fitted with an update functionality enters the life cycle phase *operational use*:

The policy P.Code_Confidentiality ensures that update code is kept confidential by the TOE software developer or electronic document manufacturer. If the TSF has flaws that are fixed by installing the update, then knowledge about the update code can be used to gain information about the flaw that is about to be fixed. Obviously, this policy only addresses the new scenario treated here, and has no potential for conflict with the base PP.

P.Secure_Environment again ensure that the update code is kept confidential while delivering the update to the TOE. As described above, analysis of the update code can be used to derive knowledge about the flaws of the TSF that are fixed by installing the update. Again, this policy only addresses the new scenario treated here, and has no potential for conflict with the base PP.

P.Eligible_Terminals_Only ensures that only terminals with proper authorization can be used to install updates. The base PP contains a similar policies (P.Terminal, P.EAC2_Terminal) that ensures that read/write access to user data of the TOE is granted only to eligible terminals. While logically distinct features, P.Eligible_Terminals_Only can thus be seen as the natural extension of these policies to include also authorization for software updates. As update terminals are not considered in the base PP, there is no potential for conflict.

Assumptions

No new assumptions are added within this PP module.

2.4.3 Security Objectives and Security Functional Requirements

Security Objectives for the TOE

The following new security objectives for the TOE are introduced:

- OT.Update_Mechanism TOE Update Mechanism
- OT.Enc_Sign_Update Encrypted-Then-Signed Update Packages
- OT.Update_Auth Updates only by authenticated Update Terminals
- OT.UPDM_Misuse Prevention of Misuse of the Update Mechanism
- OT.Attack_Detection Detection of Attacks on the TOE using the Update Mechanism
- OT.Key_Secrecy Key Secrecy of Cryptographic Update Keys

As they address the security of the update mechanism, they address only aspects that are outside of the scope of the base PP.

Security Objectives for the (Operational) Environment
The following new security objectives for the environment are introduced:

- OE.Code_Confidentiality
- OE.Secure_Environment
- OE.Eligible_Terminals_Only

These newly introduced objectives for the environment directly target the organizational policies P.Code_Confidentiality, P.Secure_Environment, and P.Eligible_Terminals_Only. Hence, the specific analysis above in the paragraph *Organizational Security Policies* directly applies here.

Security Functional Requirements
The following new security functional requirements are introduced:

- FCS_COP.1/UPD_ITC
- FCS_CKM.1/UPD_ITC
- FCS_COP.1/UPD_DEC
- FCS_CKM.1/UPD_DEC
- FCS_COP.1/UPD_SIG
- FCS_COP.1/UPD_INT
- FCS_CKM.1/UPD_INT
- FCS_CKM.4/UPD

- FIA_AFL.1/UPD
- FIA_UID.1/UPD
- FIA_UAU.1/UPD

- FDP_ACC.1/UPD
- FDP_ACF.1/UPD
- FDP_IFC.1/UPD
- FDP_IFF.1/UPD
- FDP_RIP.1/UPD

- FAU_SAS.1/UPD

- FMT_SMF.1/UPD
- FMT_MTD.1/UPD_SK_PICC
- FMT_MTD.1/UPD_KEY_READ
- FMT_SMR.1/UPD

- FPT_EMS.1/UPD
- FPT_FLS.1/UPD
- FPT_TST.1/UPD
- FTP_ITC.1/UPD
-

All these newly introduced security functional requirements solely address the update functionality. Hence, in general, no inconsistency exists with the security functional requirements of the base PP. The same holds for the two refined SFRs

Potential causes for inconsistency are the next two items:

– In [MR.ED2.0] threats related to tracing the electronic document holder by accessing data on the electronic document are considered and averted. A newly introduced asset here is *version information*.

155 Version information could be potentially misused to perform some kind of tracing of the document holder if a link between a specific version number and a document holder can be established.

– If a TOE software version is known to have security flaws, the version number can be used to identify flawed TOEs.

Both of these issues are addressed however by the SFRs FDP_ACF.1/UPD and FDP_IFF.1/UPD, which
160 precisely determine which define that access to version information and log data, and access to initiate the update procedure is limited to update terminals only. Similar, these SFRs guarantees that update terminals must not have access to user or other TSF data (the exception being one physical terminal that fulfills both the roles of an update terminal and an EAC2 terminal).

Application Note 1: As said, a potential threat (and cause for inconsistency) is that an update terminal is
165 misused to gain information that only a dedicated EAC2 terminal should have access to. Hence, while the technical details of the update mechanism are left open in this PP module, the mechanism and its implementation must provide a comparable (w.r.t. EAC2) resistance against an attacker with high attack potential. A suitable mechanism is for example to directly use the Terminal Authentication 2 protocol, or an adaption of it to authenticate the update terminal.

2.4.4 Conclusion

170 In summary, the TOE considered in this PP module adds update functionality compared to the TOE considered in [MR.ED2.0]. This allows to apply software updates to counter bugs and security flaws that are detected only after the TOE entered the life cycle phase *operational use*.

The main potential contradiction with the TOE considered in [MR.ED2.0] occurs when such an update mechanism is misused by an attacker. However, newly introduced security objectives for the TOE and for the
175 (operational) environment make sure that this is not the case. These security objectives are supported by security functional requirements that can be separated into three groups:

1. SFRs that ensure the secure implementation of an authentication mechanism, such that updates can only be installed by properly authenticated terminals,
2. SFRs that ensure the implementation of a proper cryptographic check prior to installation of an update
180 package, such that only encrypted and signed update packages will be accepted by the TOE, and
3. SFRs that implement basic logging and self-testing, such that the TOE can provide version information to properly authenticated terminals, such that the TOE allows auditing of update logs, and to ensure that the TOE is always in a functionally stable state.

3 Conformance Claims

3.1 CC Conformance Claim

This protection profile module claims conformance to

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

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

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

as follows

Part 2 extended,

Part 3 conformant.

This protection profile module also claims conformance to

195 • Common Criteria: CCDB-2014-03-001, CC and CEM addenda - Modular PP - Version 1.1, [CC-Mod][2]

The

• Common Methodology for Information Technology Security Evaluation, Evaluation methodology; CCMB-2012-09-004, Version 3.1, Revision 4, September 2012 [CC4]

has to be taken into account.

3.2 Conformance Statement

This PP module inherits the conformance statement of its base PP [MR.ED2.0], which means strict conformance of any ST or PP claiming conformance to a PP configuration based on this PP module.

2 whereas [CC-Mod] is an official part of revision 4 of CC 3.1 revision 4, at the time of evaluation, [CC-Mod] was in the process of being integrated into CC. References within the officially published document [CC-Mod] were still linking to revision 3 of CC 3.1 however. The evaluator considered the changes from CC and CEM version 3.1, revision 3, to CC and CEM version 3.1, revision 4 during evaluation.

4 Security Problem Definition

4.1 Introduction

4.1.1 Assets

4.1.1.1 Primary Assets

200 Primary assets are the same as in [MR.ED2.0].

4.1.1.2 Secondary Assets

In addition to the secondary assets of [MR.ED2.0], the following assets are added:

Secret Cryptographic Update Keys
All cryptographic key material related to the update mechanism; i.e. cryptographic material that is used to establish a secure communication channel with the update terminal, to authenticate an update terminal, to decrypt and verify the authenticity of an update package, and for other update-related cryptographic
205 operations. Note that this term deliberately includes public (in the cryptographic sense) signing keys installed on the TOE for verifying the authenticity of update packages, as well as ephemeral keys.

Meta-Data
Data that contains information about the update, e.g. version information, checksums, information w.r.t. applicability to specific product versions and platforms, etc.

210 *Application Note 2*: Note that, depending on the deployment scenario, some meta-data are security relevant and must be encrypted. Consider for example an identifier, that uniquely identifies a product version. If the update fixes a security flaw, an attacker that obtains the identifier can directly find out whether some product is vulnerable. The precise definition of meta-data and which data are encrypted shall be given by the ST-Writer.

Update Data
215 Unencrypted data that is used to update the TOE software.

Note that we use the term *update data* to denote the unencrypted data. Encrypted update data, appended with optional additional unencrypted meta-data (i.e. version number, TOE product identifier), and signed, is called an *update package*.

Update Log Data
220 Log records that store information about previously applied updates and failed update attempts.

Update Package
Encrypted update data, appended with optional unencrypted meta-data, and signed.

Update Package Verification Status
Security attribute indicating whether the supplied update was successfully verified (and where hence its authenticity and integrity can be assumed) or not, and whether an attempt to verify was made or not.
225 Allowed values are NOT VERIFIED, SUCCESSFULLY VERIFIED and VERIFICATION FAILED.

Version Information
Version information that uniquely identify the version of the TOE software currently installed on the TOE.

4.1.2 Subjects

In addition to the subjects of [MR.ED2.0], the following subject is added here:

Update Terminal
A terminal to read out version information and update log data of the TOE software, and to install updates of the TOE software. Prior executing these functions, the update terminal must authenticate itself towards the TOE.

4.2 Threats

T.FaTSF Faulty TSF

Adverse action: An attacker gains read or write access to user data or TSF data, or manipulates or mitigates the TSF, for example due to:

- software issues that were not detected, not exploitable, or deemed unable to being exploitable at the time of certification, but due to unforeseen advances in technology became a security risk during operational use of the TOE, or
- cryptographic mechanisms that were deemed secure at the time of certification, but due to unforeseen advances in the field of cryptography became a security risk during operational use of the TOE.

Threat agent: having high attack potential, being in possession of one or more legitimate electronic documents

Asset: all data stored on the TOE (esp. the integrity, authenticity and – if applicable – secrecy of the data)

T.UaU Unauthorized Update

Adverse action: An attacker gains read or write access to user data or TSF data, or manipulates or mitigates the TSF by misuse of the update functionality. This threat contains two main aspects:

- the unauthorized installation, which may lead to the use of untimely, outdated or revoked updates.
- the installation of updates that are not authorized and authentic.

Threat agent: having high attack potential, being in possession of one or more legitimate electronic documents

Asset: all data stored on the TOE (esp. the integrity, authenticity and – if applicable – secrecy of the data)

4.3 Organizational Security Policies

P.Code_Confidentiality
Update code packages that are created by the TOE software developer or document manufacturer are kept confidential, are encrypted after development at the site of the electronic document manufacturer, and are delivered to the TOE in encrypted form.

P.Secure_Environment
Update terminals are placed in a secure environment that prevents unauthorized physical access, and are operated by authorized staff only. Authorized staff oversees the complete update procedure.

P.Eligible_Terminals_Only
Update terminals (i.e. terminals with appropriate certificates that are able to install updates) are handed only to those entities where P.Secure_Environment is enforced. In case of a security incident, these update terminals are functionally disabled (through organizational and/or cryptographic means by e.g. withdrawing
260 certificates).

4.4 Assumptions

No additional assumptions to the base PP are made in this PP module.

5 Security Objectives

This chapter describes the security objectives for the TOE and for the TOE environment. The security objectives for the TOE environment are separated into security objectives for the development and production environment, and security objectives for the operational environment.

5.1 Security Objectives for the TOE

OT.Update_Mechanism TOE Update Mechanism
The TSF provides a mechanism to install code-signed updates of the TOE software by authorized staff during operational use.

OT.Enc_Sign_Update Encrypted-then-signed Update Packages
The TOE only installs update packages that are encrypted, integrity-protected and signed by the authority in charge of delivering and installing updates.

OT.Update_Terminal_Auth Updates only by authenticated Update Terminals
The TOE allows only authenticated update terminals to upload an update package to the TOE and to initiate the update procedure. The TOE uses a dedicated cryptographic method (to be defined or referenced by the ST-Writer) to authenticate an update terminal.

OT.Attack_Detection Detection of Attacks on the TOE using the Update Mechanism
The TOE has logging capabilities that track installed updates and failed update attempts. It also limits the amount of faulty (signature verification or decryption fails) update attempts. It allows dedicated terminals to read out the update logs.

OT.Key_Secrecy Key Secrecy of Cryptographic Update Keys
The TOE keeps the cryptographic update keys secret, and is designed such that emissions from the TOE do not allow to read out or gain full or partial information about the keys.

5.2 Security Objectives for the Environment

5.2.1 Security Objectives for the Development and Production Environment

OE.Code_Confidentiality
The operational environment must ensure that the TOE software developer or document manufacturer keeps update code packages confidential, encrypts them after development at the site of the developer/manufacturer, and delivers them to the TOE in encrypted form.

5.2.2 Security Objectives for the Operational Environment

OE.Secure_Environment
The operational environment must ensure that update terminals are placed in a secure environment that prevents unauthorized physical access, and are operated by authorized staff only. The operational environment must also ensure through e.g. organizational policies and procedures, that authorized staff oversees the complete update procedure.

OE.Eligible_Terminals_Only
The operational environment must also ensure by e.g. organizational procedures, supported by cryptographic means, that only those entities that have policies in place that guarantee

OE.Secure_Environment, are supplied with update terminals. Moreover the operational environment guarantees that update terminals can be functionally deactivated if these policies are no longer in place or not enforced at the entities. This can be implemented for example by the issuance of certificates for update
285 terminals together with a public key infrastructure.

Justification: Each of these security objectives on the environment directly addresses one of the organizational security policies P.Code_Confidentiality, P.Secure_Environment, and P.Eligible_Terminals_Only. Hence, these security objectives for the environment do

- neither mitigate a threat of the base PP that was addressed by security objectives of the base PP,
290 – nor do they fulfill any organizational security policy of the base PP that was meant to be addressed by security objectives of the TOE of the base PP.

Note in particular that OE.Eligible_Terminals_Only requires a general issuance and revocation mechanism for update terminals and leaves the specific implementation open, whereas OE.Terminal_Authentication of the base PP specifically addresses certificates for EAC2 terminals.

5.3 Security Objective Rationale

Tracings

295 Table 1 provides an overview of the security objectives' coverage. According to [CC1], the tracing between security objectives and the security problem definition must ensure that 1) *each security objective traces to at least one threat, OSP and assumption*, 2) *each threat, OSP and assumption has at least one security objective tracing to it*, and 3) *the tracing is correct* (i.e. the main point being that security objectives for the TOE do not trace back to assumptions).

	OT.Update_Mechanism	OT.Enc_Sign_Update	OT.Update_Terminal_Auth	OT.Attack_Detection	OT.Key_Secrecy	OE.Code_Confidentiality	OE.Secure_Environment	OE.Eligible_Terminals_Only
T.FaTSF	x			x	x			
T.UaU		x	x					
P.Code_Confidentiality						x		
P.Secure_Environment							x	
P.Eligible_Terminals_Only								x

Table 1: Security Objective Rationale

Justifications

300 The threat **T.FaTSF** addresses attacks on the TOE and TSF by an attacker exploiting flaws of the TOE software implementation that manifest themselves after the TOE enters the phase operational usage. This threat is countered by the TOE offering a secure update mechanism; in particular:

- The security objective OT.Update_Mechanism counters this threat by ensuring that the TOE has the ability to update the TOE software in a secure manner.

305 – The security objective OT.Attack_Detection ensures that the TOE is able to detect multiple failed update attempts and can take action upon that detection.

– The security objective OT.Key_Secrecy makes sure that the required cryptographic key material for the update mechanism cannot be accessed or reconstructed by a malicious attacker.

The threat **T.UaU** addresses attacks on the TOE and TSF by an attacker installing unauthorized and potential
310 harmful updates:

– The security objective OT.Enc_Sign_Update ensures that only signed and encrypted updates are installed by the TOE, and that during the transmission to the TOE, a protocol based on encrypt-then-MAC is used.

– The security objective OT.Update_Terminal_Auth ensures that only authenticated update terminals are able to read version information, upload update packages on the TOE, and initiate the update procedure.

315 The organizational security policies **P.Code_Confidentiality**, **P.Secure_Environment**, and **P.Eligible_Terminals_Only**, address the confidentiality of the code, the way the update procedure must be carried out, and precise control over which terminals are allowed to carry out the update procedure. Each of these policies are enforced through security objectives for the environment of the TOE, namely OE.Code_Confidentiality, OE.Secure_Environment, and OE.Eligible_Terminals_Only.

6 Extended Components Definition

320 This PP module uses the following extended components:

– FAU_SAS.1 from the family FAU_SAS from [PACEPP]

– FPT_EMS.1 from the family FPT_EMS from [PACEPP]

For precise definitions we refer to [PACEPP].

7 Security Requirements

This part defines detailed security requirements that shall be satisfied by the TOE. The statement of TOE
325 security requirements shall define the *functional* and *assurance* security requirements that the TOE must satisfy in order to meet the security objectives for the TOE.

Common Criteria allows several operations to be performed on security requirements on the component level: *refinement*, *selection*, *assignment* and *iteration,* cf. sec. 8.1 of [CC1]. Each of these operations is used in this PP.

330 The **refinement** operation is used to add detail to a requirement, and thus further restricts a requirement. Refinements of security requirements are denoted in such a way that added words are in **bold text** and removed words are ~~crossed out~~.

The **selection** operation is used to select one or more options provided by CC in stating a requirement. Selections that have been made by the PP author are denoted as underlined text. Selections to be filled in by
335 the ST author appear in square brackets with an indication that a selection has to be made, [selection:], and are *italicized*.

The **assignment** operation is used to assign a specific value to an unspecified parameter, such as the length of a password. Assignments that have been made by the PP author are denoted as underlined text. Assignments to be filled in by the ST author appear in square brackets with an indication that an assignment
340 has to be made [assignment:], and are *italicized*. In some cases the assignment made by the PP authors defines a selection to be performed by the ST author, or the assignment is narrowed down. Then this text is underlined and italicized *like this*.

The **iteration** operation is used when a component is repeated with varying operations. Iteration is denoted by showing a slash "/", and the iteration indicator after the component identifier. For the sake of better
345 readability, the iteration operation may also be applied to a non-repeated single component in order to indicate that such component belongs to a certain functional cluster. In such a case, the iteration operation is applied to only one single component.

7.1 Security Functional Requirements

According to [CC-Mod], no SFRs from the base PP are listed unless they are required to fulfill new needs. Interpreted/refined elements are considered new.

7.1.1 Class FCS

FCS_COP.1/UPD_ITC Cryptographic Operation – Inter Trusted Channel

Hierarchical to:

350 No other components.

Dependencies:

[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
fulfilled by FCS_CKM.1/UPD_ITC
FCS_CKM.4 Cryptographic key destruction
355 fulfilled by FCS_CKM.4/UPD

FCS_COP.1.1/UPD_ITC

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

360 *Application Note 3*: FCS_COP.1/UPD shall be used by the ST-Writer for the cryptographic operations needed for communication via a trusted channel as required by FTP_ITC.1/UPD.

FCS_CKM.1/UPD_ITC Cryptographic Key Generation

Hierarchical to:
No other components.

Dependencies:
[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
fulfilled by FCS_COP.1/UPD_ITC
365 FCS_CKM.4 Cryptographic key destruction
fulfilled by FCS_CKM.4/UPD

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

FCS_COP.1/UPD_DEC Cryptographic Operation – Decryption of Update Packages

Hierarchical to:
No other components.

Dependencies:
[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
fulfilled by FCS_CKM.1/UPD_DEC
375 FCS_CKM.4 Cryptographic key destruction
fulfilled by FCS_CKM.4/UPD

FCS_COP.1.1/UPD_DEC
The TSF shall perform decryption of update packages[3] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic*
380 *key sizes*] that meet the following: [assignment: *list of standards*].

FCS_CKM.1/UPD_DEC Cryptographic Key Generation

Hierarchical to:
No other components.

Dependencies:
[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
fulfilled by FCS_COP.1/UPD_DEC
FCS_CKM.4 Cryptographic key destruction
385 fulfilled by FCS_CKM.4/UPD

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

FCS_COP.1/UPD_SIG Cryptographic Operation – Signature Verification of Update Packages

Hierarchical to:
390 No other components.

3 [assignment: *list of cryptographic operations*]

Dependencies:

[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
not fulfilled but justified: The TOE uses security attributes (keys) that have been defined during the personalization and are fixed over the whole life time of the TOE. No import or generation of these security attributes is necessary here.

FCS_CKM.4 Cryptographic key destruction
not fulfilled but justified: The TOE uses security attributes (keys) that have been defined during the personalization and are fixed over the whole life time of the TOE. Key destruction implies not being able to verify digital signatures from then on, and hence, is not applicable here.

FCS_COP.1.1/UPD_SIG

The TSF shall perform digital signature verification[4] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].

FCS_COP.1/UPD_INT Cryptographic Operation – Integrity Verification of Update Package

Hierarchical to:

No other components.

Dependencies:

[FDP_ITC.1 Import of user data without security attributes, or FDP_ITC.2 Import of user data with security attributes, or FCS_CKM.1 Cryptographic key generation]
fulfilled by FCS_CKM.1/UPD_INT

FCS_CKM.4 Cryptographic key destruction
fulfilled by FCS_CKM.4/UPD

FCS_COP.1.1/UPD_INT

The TSF shall perform integrity verification of update packages[5] in accordance with a specified cryptographic algorithm [assignment: *cryptographic algorithm*] and cryptographic key sizes [assignment: *cryptographic key sizes*] that meet the following: [assignment: *list of standards*].

Application Note 4: Integrity verification of packages is intended to be used by the ST-Writer for a hash function (keyed or unkeyed) with which the TOE checks the integrity of received update packages prior to decryption.

FCS_CKM.1/UPD_INT Cryptographic Key Generation

Hierarchical to:

No other components.

Dependencies:

[FCS_CKM.2 Cryptographic key distribution, or FCS_COP.1 Cryptographic operation]
fulfilled by FCS_COP.1/UPD_INT

FCS_CKM.4 Cryptographic key destruction
fulfilled by FCS_CKM.4/UPD

FCS_CKM.1.1/UPD_INT

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

Application Note 5: This SFR is intended for the key generation in case a keyed hash function is used for FCS_COP.1/UPD_INT. In case of an unkeyed hash function, the integrity is solely implied by digital signature verification. Hence in this case, 'none' can be assigned by the ST-Writer in the above SFR.

4 [assignment: *list of cryptographic operations*]
5 [assignment: *list of cryptographic operations*]

FCS_CKM.4/UPD Cryptographic Key Destruction

Hierarchical to:
No other components.

Dependencies:
[FDP_ITC.1 Import of user data without security attributes, or
FDP_ITC.2 Import of user data with security attributes, or
FCS_CKM.1 Cryptographic key generation]: fulfilled by FCS_CKM.1/UPD_INT, FCS_CKM.1/UPD_DEC and FCS_CKM.1/UPD_ITC

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

7.1.2 Class FIA

FIA_AFL.1/UPD Update Package Verification Failure Handling

Hierarchical to:
No other components.

Dependencies:
FIA_UAU.1 Timing of authentication:
fulfilled by FIA_UAU.1/UPD

FIA_AFL.1.1/UPD
The TSF shall detect when [assignment: *positive integer number*][6] unsuccessful ~~authentication~~ **update attempts** occur~~s~~ related to [assignment: *list of ~~authentication~~ events of the update procedure*][7].

FIA_AFL.1.2/UPD
When the defined number of unsuccessful ~~authentication~~ **update attempts** has been met[8], the TSF shall [assignment: *list of actions*].

Application Note 6: The above SFR is slightly refined here by replacing 'authentication' with 'update'. Also the second assignment is made more precise. An update attempt includes authentication of the update terminal to the TOE. But when a properly authenticated terminal sends an update package that is not authentic or whose integrity cannot be validated, this is still a failed update attempt, and the TOE must handle it according to the above SFR. Hence this refinement is stricter than the original SFR definition.

FIA_UID.1/UPD Timing of Identification

Hierarchical to:
No other components.

Dependencies:
No dependencies.

FIA_UID.1.1/UPD
The TSF shall allow

1) to establish a communication channel,

2) to authenticate an update terminal by [assignment: *cryptographic method*]

6 [selection: *[assignment: positive integer number], an administrator configurable positive integer within [assignment: range of acceptable values]*]
7 [assignment: *list of authentication events*]
8 [selection: *met, surpassed*]

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

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

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

FIA_UAU.1/UPD Timing of Authentication

Hierarchical to:
No other components.

Dependencies:
FIA_UID.1 Timing of Identification
465 fulfilled by FIA_UID.1/UPD

FIA_UAU.1.1/UPD
The TSF shall allow

1) to establish a communication channel,

2) to authenticate an update terminal by [assignment: *cryptographic method*][9]

470 3) [assignment: *list of TSF-mediated actions*]

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

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

7.1.3 Class FDP

FDP_ACC.1/UPD Subset Access Control – Terminal Access

Hierarchical to:
475 No other components.

Dependencies:
FDP_ACF.1 Security attribute based access control: fulfilled by FDP_ACF.1/UPD

FDP_ACC.1.1/UPD
The TSF shall enforce the Update Access Control SFP[10] on

1) Subjects:
   a) terminal,
480 b) update terminal.
2) Objects:
   a) version information identifying the TOE software
   b) update package
   c) update log information
485 3) Operations:
   a) reading out version information,

9 [assignment: *list of TSF-mediated actions*]
10 [assignment: *access control SFP*]

b) reading out log data,

c) uploading an update package on the TOE, or

d) initiating an update procedure[11]

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

Application Note 7: The assignment in FDP_ACC.1.1/UPD may be used in order to extend the subjects and objects needed for additional security functionalities. This can be done by the ST-Writer or in a PP claiming conformance to a PP configuration that uses this PP module.

FDP_ACF.1/UPD Security Attribute based Access Control – Terminal Access

Hierarchical to:

No other components.

Dependencies:

FDP_ACC.1 Subset access control
fulfilled by FDP_ACC.1/UPD
FMT_MSA.3 Static attribute initialization
not fulfilled, but **justified:**
The access control TSF according to FDP_ACF.1/UPD uses security attributes that have been defined during personalization, and that are fixed over the whole life time of the TOE. No management of these security attributes (i.e. SFR FMT_MSA.1 and FMT_MSA.3) is necessary here.

FDP_ACF.1.1/UPD

The TSF shall enforce the Update Access Control SFP[12] to objects based on the following:

1) Subjects:
   a) terminal,
   b) update terminal
2) Objects:
   a) version information identifying the TOE software
   b) update package
   c) update log information
3) Security attributes:
   a) access rights
4) [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*][13].

FDP_ACF.1.2/UPD

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

The authentication level of a terminal must be determined by [assignment: *list of technical specifications of cryptographic procedures*] as required by FIA_UAU.1/UPD. Depending on the authentication level, an authenticated update terminal is allowed one or more of the following:

11 [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*]
12 [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]]
13 [assignment: *list of subjects and objects controlled under the indicated SFP, and for each, the SFP-relevant security attributes, or named groups of SFP-relevant security attributes*]

- read one or more data objects from FDP_ACF.1/UPD
- upload an update package to the TOE and initiate the update procedure.

The precise definition of access rights and how the authentication level is calculated from an authenticated terminal is defined in [assignment: *list of technical specifications of cryptographic procedures*][14].

FDP_ACF.1.3/UPD
The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: none.[15]

FDP_ACF.1.4/UPD
The TSF shall explicitly deny access of subjects to objects based on the following additional rules: [assignment: *rules, based on security attributes, that explicitly deny access of subjects to objects*].

FDP_IFC.1/UPD Subset information flow control

Hierarchical to:
No other components.

Dependencies:
FDP_IFF.1 Simple security attributes, fulfilled by FDP_IFF.1/UPD

FDP_IFC.1.1/UPD:
The TSF shall enforce the Update Flow Control SFP[16] on the following:

1) Subjects:
   a) terminal,
   b) update terminal.
2) information:
   a) update package
   b) update data
   c) meta-data, such as version information
3) operations:
   a) performing an update[17].

FDP_IFF.1/UPD Simple Security Attributes

Hierarchical to:
No other components.

Dependencies:
FDP_IFC.1 Subset information flow control: fulfilled by FDP_IFC.1/UPD

FMT_MSA.3 Static attribute initialization:not fulfilled, but **justified:**
The update control TSF according to FDP_IFF.1/UPD uses security attributes that have been defined during personalization, and that are fixed over the whole life time of the TOE. No management of these security attributes (i.e. SFR FMT_MSA.1 and FMT_MSA.3) is necessary here.

14 [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*]
15 [assignment: *rules, based on security attributes, that explicitly authorize access of subjects to objects*]
16 [assignment: *information flow control SFP*]
17 [assignment: *list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP*]

FDP_IFF.1.1/UPD

The TSF shall enforce the Update Control SFP[18] based on the following types of subject and information security attributes:

1) Subjects:
   a) terminal,
   b) update terminal.
2) information:
   a) update package
   b) update data
   c) meta-data, such as version information
3) security attributes:
   a) update package verification status with the values: NOT VERIFIED (default status), SUCCESSFULLY VERIFIED, and VERIFICATION FAILED[19].

FDP_IFF.1.2/UPD

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

1. The terminal has established a secure channel with the TOE.
2. The TOE shall only accept update packages sent via a secure channel established with an authenticated update terminal[20].

FDP_IFF.1.3/UPD

The TSF shall enforce the following rules in their specific order:

1) The integrity (using the keyed or unkeyed hash function cf. FCS_COP.1/UPD_INT) and authenticity (using the digital signature, cf. FCS_COP.1/UPD_SIG) of the first part of the update package is verified. If the integrity and authenticity are not both validated, abort with VERIFICATION FAILED, and erase all data transferred so far, cf. FDP_RIP.1.
2) The first part of the update package is only decrypted, cf. FCS_COP.1/UPD_DEC, if the integrity and authenticity of the that part has been verified in rule 1. If the decryption fails, abort with VERIFICATION FAILED, and erase all data transferred so far, cf. FDP_RIP.1.
3) If all parts of the update package have been decrypted, continue with rule 4. Otherwise, apply rules 1. and 2. on the remaining parts (replace 'first part' with 'current part' above) until either all parts have been decrypted, or the procedure has been aborted with VERIFICATION FAILED.
4) If additional meta-data is stored in the update package [assignment: list of meta-data contained in the update package or reference to technical specification(s) defining those] is not verified as correct according to [assignment: *technical specification(s) defining correct form and content of meta-data*] the security attribute is set to VERIFICATION FAILED and the update package including all associated data are destroyed, cf. FDP_RIP.1. Correctness w.r.t. the referenced technical specification must not contradict any of the given rules here.
5) Next, the TSF shall verify that:

18 [assignment: *information flow control SFP*]

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

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

a) the version number of the update package must be greater than the version of the installed corresponding software package;

b) the update data are suitable to the specific TOE configuration/platform by checking relevant meta-data (i.e. TOE product identifier, version number etc.).

595 If all conditions in step 5 are verified, the verification status is set to SUCCESSFULLY VERIFIED. Otherwise abort with VERIFICATION FAILED, and erase all data transferred so far, cf. FDP_RIP.1.

Only if the verification status is SUCCESSFULLY VERIFIED, the TOE shall install the update data[21].

FDP_IFF.1.4/UPD
The TSF shall explicitly authorize an information flow based on the following rules: [assignment: *rules,*
600 *based on security attributes, that explicitly authorize information flows*].

FDP_IFF.1.5/UPD
The TSF shall explicitly deny an information flow based on the following rules: [assignment: *rules, based on security attributes, that explicitly deny information flows*].

FDP_RIP.1/UPD Subset Residual Information Protection

Hierarchical to:
No other components.

Dependencies:
605 No dependencies.

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

1) session keys (immediately after closing related communication session),

610 2) all ephemeral keys [assignment: list of ephemeral keys or reference to specification] related to the update mechanism.

3) Update package, decrypted update data and meta-data uploaded to the TOE or generated during the update procedure[22].

4) [assignment: *list of objects*].

615 *Application Note 8*: The functional family FDP_RIP possesses such a general character, so that it is applicable not only to user data (as assumed by the class FDP), but also to TSF-data; in this respect it is similar to the functional family FPT_EMS. Applied to cryptographic keys, FDP_RIP.1 requires a certain quality metric ('any previous information content of a resource is made unavailable') for key destruction in addition to FCS_CKM.4 that merely requires a fact of key destruction according to a method/standard. The ST-Writer
620 should in particular list all relevant ephemeral keys required for the update procedure or reference a technical specification that defines the related protocols and generated ephemeral keys.

7.1.4 Class FAU

FAU_SAS.1/UPD Audit Storage of Update History

Hierarchical to:
No other components.

Dependencies:
No dependencies.

21 [assignment: *additional information flow control SFP rules*]
22 [assignment: *list of objects*]

FAU_SAS.1.1/UPD

625 The TSF shall provide **the TOE update functionality** with the capability to store update log information and version history, namely the following data objects: [assignment: *list of update log information data*][23] in the audit records.

Justification: According to [CC1], a PP author is allowed to refine an SFR to apply to some, but not all subjects. The refinement of this SFR is such an exception, since the TOE update functionality is technically not an au-
630 thorized user. Hence, the refinement is justified.
Note FAU_SAS.1 from [MR.ED2.0] applies as well. The SFR here is a new iteration refining the definition of [CC2] and is only concerned with the TOE update functionality.

7.1.5 Class FMT

FMT_SMF.1/UPD Specification of Management Functions including Updates

Hierarchical to:
No other components.

Dependencies:
No dependencies.

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

1) Updating the TOE software with the mechanism specified in [assignment: *list of technical specification(s) defining an update mechanism*][24].

FMT_MTD.1/UPD_SK_PICC Management of TSF Data – Secret Update Keys

Hierarchical to:
No other components.

Dependencies:
640 FMT_SMF.1 Specification of management functions:
fulfilled by FMT_SMF.1/UPD
FMT_SMR.1 Security roles
fulfilled by FMT_SMR.1/UPD

FMT_MTD.1.1/UPD_SK_PICC
645 The TSF shall restrict the ability to [selection: *create, load*][25] the [selection: *list of, or reference specifying the Secret Cryptographic Update Keys required for the update procedure*][26] to the update key installation agent[27].

FMT_MTD.1/UPD_KEY_READ Management of TSF data – Secret Update Keys

Hierarchical to:
No other components.

Dependencies:
FMT_SMF.1 Specification of management functions
650 fulfilled by FMT_SMF.1/UPD

23 [assignment: *list of audit information*]
24 [assignment: *list of management functions to be provided by the TSF*]
25 [selection: *change_default, query, modify, delete, clear*, [assignment: *other operations*]]
26 [assignment: *list of TSF data*]
27 [assignment: *the authorized identified roles*]

FMT_SMR.1 Security roles
fulfilled by FMT_SMR.1/UPD

FMT_MTD.1.1/UPD_KEY_READ
The TSF shall restrict the ability to read[28] the

1) [assignment: *list of or reference specifying the Secret Cryptographic Update Keys required for the update procedure*]

2) [assignment: *list of TSF data*][29]

to none[30].

FMT_SMR.1/UPD Security roles

Hierarchical to:
No other components.

Dependencies:
FIA_UID.1 Timing of identification:
fulfilled by FIA_UID.1/UPD

FMT_SMR.1.1/UPD
The TSF shall maintain the roles

1) terminal,

2) update terminal

3) update key installation agent

4) [assignment: *the authorized identified roles*][31]

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

7.1.6 Class FPT

FPT_EMS.1 /UPD TOE Emanation

Hierarchical to:
No other components.

Dependencies:
No dependencies.

FPT_EMS.1.1/UPD
The TOE shall not emit [*assignment: types of emissions*] in excess of [assignment: *specified limits*] enabling access to

[assignment: *list of or reference specifying the Secret Cryptographic Update Keys used for the update mechanism and other types of TSF data*][32] and [assignment: *list of types of user data*].

28 [selection: *change_default, query, modify, delete, clear*, [assignment: *other operations*]]
29 [assignment: *list of TSF data*]
30 [assignment: *the authorized identified roles*]
31 [assignment: *the authorized identified roles*]
32 [assignment: *list of types of TSF data*]

FPT_EMS.1.2/UPD
The TSF shall ensure any users[33] are unable to use the following interface electronic document's contactless/contact-based interface and circuit contacts[34] to gain access to

680 [assignment: *list of or reference specifying the Secret Cryptographic Update Keys used for the update mechanism and other types of TSF data*][35] and [assignment: *list of types of user data*].

Application Note 9: The TOE shall prevent attacks against the listed secret data where the attack is based on external observable physical phenomena of the TOE. Such attacks may be observable at the interfaces of the TOE, originate from internal operation of the TOE, or be caused by an attacker that varies the physi-
685 cal environment under which the TOE operates. The set of measurable physical phenomena is influenced by the technology employed to implement the smart card. Examples of measurable phenomena include, but are not limited to variations in power consumption, timing of signals, and electromagnetic radiation due to internal operations or data transmissions.
Note that while the security functionality described in FPT_EMS.1 should be taken into account during
690 development of the TOE, associated tests must be carried out as part of the evaluation, and not/not only during product development.
FPT_EMS.1 /UPD is an iteration of the definition of FPT_EMS.1 (defined as an extended component in [MR.ED2.0]). That base PP also contains several other iterations of FPT_EMS.1, such as FPT_EMS.1/EAC1PP, and FPT_EMS.1/EAC2PP. These multiple definitions do not contradict, since one of
695 course must apply a logical 'AND' w.r.t. to all data defined in all FPT_EMS.1/*, i.e. none of any data defined FPT_EMS.1/* must be observable or accessible according to FPT_EMS.1.1 and FPT_EMS.1.2.

FPT_FLS.1/UPD Failure with Preservation of Secure State (Failed Update)

Hierarchical to:
No other components.

Dependencies:
No dependencies.

700 FMT_FLS.1.1/UPD
The TSF shall preserve a secure state when the following types of failures occur:

1) Failure during a transmission of the update package data file
2) Failure detected by TSF according to FPT_TST.1
3) Failure detected after a failed update[36]
705 4) [assignment: *list of types of failures in the TSF*].

Application Note 10: The secure state after a failed update should be achieved by reverting to the previous TOE software version. Nevertheless this capability will have limits, since the atomicity of the software update mechanism can technically only be achieved up to a certain extent.

FPT_TST.1/UPD TSF Testing (after Installation of an Update)

Hierarchical to:
No other components.

Dependencies:
710 No dependencies.

33 [assignment: *type of users*]
34 [assignment: *type of connection*]
35 [assignment: *list of types of TSF data*]
36 [assignment: *list of types of failures in the TSF*]

FMT_TST.1.1/ UPD

The TSF shall run a suite of self tests [*selection: during initial start-up, periodically during normal operation, after a software update, at the request of the authorized user, at the conditions* [*assignment: conditions under which self test should occur]*] to demonstrate the correct operation of the TSF[37].

FPT_TST.1.2/ UPD

The TSF shall provide authorized users with the capability to verify the integrity of the TSF data[38].

FPT_TST.1.3/UPD

The TSF shall provide authorized users with the capability to verify the integrity of stored TSF executable code[39].

7.1.7 Class FTP Trusted Path/Channels

FTP_ITC.1/UPD Inter-TSF trusted Channel

Hierarchical to:

No other components.

Dependencies:

No dependencies.

FTP_ITC.1.1/UPD

The TSF shall provide a communication channel between itself and ~~another trusted IT product~~ **an update terminal** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

FTP_ITC.1.2/UPD

The TSF shall permit ~~another trusted IT product~~ **an update terminal** to initiate communication via the trusted channel.

FTP_ITC.1.3/UPD

The TSF shall ~~initiate~~ **enforce** communication via the trusted channel for any data exchange between the TOE and the update terminal.[40]

7.2 Security Requirements Rationale

7.2.1 Security Functional Requirements Rationale

The following table provides an overview for the coverage of the security functional requirements, and also gives evidence for sufficiency and necessity of the chosen SFRs.

The SFRs FCS_COP.1/UPD_ITC, FCS_CKM.1/UPD_ITC, FCS_COP.1/UPD_DEC, FCS_CKM.1/UPD_DEC, FCS_COP.1/UPD_INT, FCS_CKM.1/UPD_INT, FCS_COP.1/UPD_SIG, FCS_CKM.4/UPD are concerned with cryptographic operations and key generation. They support the objectives OT.Update_Mechanismand OT.Enc_Sign_Update.

FIA_AFL.1/UPD, FIA_UID.1/UPD, FIA_UAU.1/UPD are concerned with identification and authentication towards the TOE. They concern the update mechanism and hence OT.Update_Mechanism, and OT.Update_Terminal_Auth.

FDP_ACC.1/UPD, FDP_ACF.1/UPD, FDP_IFC.1/UPD, FDP_IFF.1/UPD support OT.Update_Mechanism and OT.Update_Terminal_Auth. FDP_RIP.1/UPD supports OT.Update_Mechanism.

37 [selection: *[assignment: parts of TSF], the TSF*]
38 [selection: *[assignment: parts of TSF], TSF data*]
39 [selection: *[assignment: parts of TSF], TSF*]
40 [assignment: *list of functions for which a trusted channel is required*]

FAU_SAS.1/UPD supports OT.Update_Mechanism and OT.Attack_Detection w.r.t. logging.

FMT_SMF.1/UPD, FMT_MTD.1/UPD_SK_PICC, FMT_MTD.1/UPD_KEY_READ, and FMT_SMR.1/UPD are concerned with management functions and data. FMT_SMF.1/UPD supports OT.Update_Mechanism,
745 FMT_MTD.1/UPD_SK_PICC and FMT_MTD.1/UPD_KEY_READ support OT.Enc_Sign_Update, OT.Update_Terminal_Auth and OT.Key_Secrecy, and FMT_SMR.1/UPD supports OT.Enc_Sign_Update, OT.Update_Terminal_Auth.

	OT.Update_Mechanism	OT.Enc_Sign_Update	OT.Update_Terminal_Auth	OT.Attack_Detection	OT.Key_Secrecy
Class FCS					
FCS_COP.1/UPD_ITC	x	x			
FCS_CKM.1/UPD_ITC	x	x			
FCS_COP.1/UPD_DEC	x	x			
FCS_CKM.1/UPD_DEC	x	x			
FCS_COP.1/UPD_INT	x	x			
FCS_CKM.1/UPD_INT	x	x			
FCS_COP.1/UPD_SIG	x	x			
FCS_CKM.4/UPD	x	x			
Class FIA					
FIA_AFL.1/UPD	x		x		
FIA_UID.1/UPD	x		x		
FIA_UAU.1/UPD	x		x		
Class FDP					
FDP_ACC.1/UPD	x		x		
FDP_ACF.1/UPD	x		x		
FDP_IFC.1/UPD	x		x		
FDP_IFF.1/UPD	x		x		
FDP_RIP.1/UPD	x				
Class FAU					
FAU_SAS.1/UPD	x			x	
Class FMT					
FMT_SMF.1/UPD	x				
FMT_MTD.1/UPD_SK_PICC		x	x		x
FMT_MTD.1/UPD_KEY_READ		x	x		x
FMT_SMR.1/UPD		x	x		
Class FPT					
FPT_EMS.1/UPD					x
FPT_FLS.1/UPD				x	
FPT_TST.1/UPD				x	
Class FTP					
FTP_ITC.1/UPD	x		x		

Table 2: Coverage of Security Objectives for the TOE by SFRs

FPT_EMS.1/UPD supports OT.Key_Secrecy, and FPT_FLS.1/UPD and FPT_TST.1/UPD support OT.Attack_Detection.

FTP_ITC.1/UPD supports OT.Update_Mechanism.

7.2.2 Rationale for SFR's Dependencies

750 The dependency analysis for the security functional requirements shows that the basis for mutual support and internal consistency between all defined functional requirements is satisfied. All dependencies between the chosen functional components are analyzed, and non-dissolved dependencies are appropriately explained.

The dependency analysis has directly been made within the description of each SFR in Section 7.1 above. All
755 dependencies being expected by [CC2] and by extended components definition in Chapter 6 are either fulfilled, or their non-fulfillment is justified.

7.2.3 Security Assurance Requirements Rationale

The current assurance package in inherited from the base PP. Hence, the same rationale applies here, and we refer to [MR.ED2.0] for details.

7.2.4 Security Requirements – Internal Consistency

The following part of the security requirements rationale shows that the set of security requirements for the
760 TOE consisting of the security functional requirements (SFRs) and the security assurance requirements (SARs) are internally consistent. The analysis of the TOE´s security requirements with regard to their mutual support and internal consistency demonstrates:

The dependency analysis in Section 7.2.2 for the security functional requirements shows that the basis for internal consistency between all defined functional requirements is satisfied. All dependencies between the
765 chosen functional components are analyzed and non-satisfied dependencies are appropriately justified.

All subjects and objects addressed by more than one SFR are also treated in a consistent way: the SFRs impacting them do not require any contradictory property or behavior of these 'shared' items.

The assurance package EAL4 is a predefined set of internally consistent assurance requirements. The dependency analysis for the sensitive assurance components in Section 7.2.3 shows that the assurance
770 requirements are internally consistent as all (additional) dependencies are satisfied and no inconsistency appears.

Inconsistency between functional and assurance requirements can only arise due to functional-assurance dependencies not being met. As shown in Section 7.2.2 and Section 7.2.3,the chosen assurance components are adequate for the functionality of the TOE. Hence, there are no inconsistencies between the goals of these
775 two groups of security requirements.

8 PP-Configuration

8.1 Reference

This PP-Configuration is identified as:

Title: Machine Readable Electronic Documents - Optionales Nachladen (Optional Post-Emission Updates) [MR.ED-ON-PP],

Version: Version 0.9.2, as of August 18th, 2016

780 Registration: BSI-CC-PP-0090-2016

8.2 Components Statement

This configuration has one single base PP:

Title: Machine-Readable Electronic Documents based on BSI TR-03110 for Official Use [MR.ED-PP]

Version: Version 2.0.3, July 18th, 2016

785 Registration: BSI-CC-PP-0087-V2-MA-01

This configuration consists of the base PP together with the PP-Module

Common Criteria PP-Module Machine Readable Electronic Documents - Optionales Nachladen (Optional Post-Emission Updates) [MR.ED-ON-PP-Module], version 0.9.2, August 18th, 2016, described in Chapters 2-7 of this document.

8.3 Conformance Statement

790 This PP configuration requires *strict* conformance of any ST or PP claiming conformance to this PP.

8.4 Conformity to Security Assurance Requirements

This PP configuration inherits conformity to SAR packages from its base PP [MR.ED2.0]: Assurance package EAL4 augmented with ALC_DVS.2, ATE_DPT.2 and AVA_VAN.5 as defined in [CC3].

Glossary and Abbreviations

Glossary

Ephemeral Key

795 A cryptographic key that is generated each time during a cryptographic procedure (e.g. a key establishment process) and usually used only once or during a single session.

IC Dedicated Software

Software developed and injected into the chip hardware by the IC manufacturer. Such software might support special functionality of the IC hardware and be used, amongst other, for implementing
800 delivery procedures between different entities. The usage of parts of the IC dedicated software might be restricted to certain life phases.

IC Embedded Software

Software embedded in an IC and not being designed by the IC developer. The IC embedded software is designed in the design life phase and embedded into the IC in the manufacturing life phase of the
805 TOE.

Initialization Data

Any data defined by the electronic document manufacturer and injected into the non-volatile memory by the integrated circuit manufacturer. These data are, for instance, used for traceability and for IC identification as IC_Card material (IC identification data).

810 **Personalization**

The process by which data related to the electronic document holder (biographic and biometric data, or key pair(s) for a potential signature application) are stored in and unambiguously, inseparably associated with the electronic document.

Pre-personalization Data

815 Any data that is injected into the non-volatile memory of the TOE by the manufacturer for traceability of the non-personalized electronic document and/or to secure shipment within or between the life cycle phases manufacturing and card issuing.

Secure Messaging

Secure messaging using encryption and message authentication code according to [ISO7816-4].

820 **Update Key Installation Agent**

The entity that installs the cryptographic update key(s) required for the update procedure on the TOE.

Update Procedure

The process of updating the TOE software, i.e. connecting the card to an update terminal, the execution of the update commands and installation of the update package.

825

Abbreviations

CC	Common Criteria
n.a.	Not applicable
OSP	Organizational security policy
830 PCD	Proximity Coupling Device
PICC	Proximity Integrated Circuit Chip
PP	Protection Profile
RF	Radio Frequency

SAR	Security assurance requirements
SFR	Security functional requirement
TOE	Target of Evaluation
TSF	TOE security functionality
TSP	TOE Security Policy (defined by the current document)

Reference Documentation

CC-Mod	Common Criteria: CCDB-2014-03-001, CC and CEM addenda - Modular PP - Version 1.1,
CC1	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and General Model; CCMB-2012-09-001, 3.1, Revision 4
CC2	Common Criteria for Information Technology Security Evaluation, Part 2: Security Functional Components; CCMB-2012-09-002, 3.1, Revision 4
CC3	Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements; CCMB-2012-09-003, 3.1, Revision 4
CC4	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology; CCMB-2012-09-004 , 3.1, Revision 4
ISO7816-4	ISO/IEC: ISO/IEC 7816-4:2013 Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange
MR.ED2.0	BSI: Common Criteria Protection Profile - Machine Readable Electronic Documents based on BSI TR03110, BSI-CC-PP-0087-V2-2016-MA-01
PACEPP	BSI: Common Criteria Protection Profile - Machine Readable Travel Document using Standard Inspection Procedure with PACE (PACE PP), BSI-CC-PP-0068-V2-2011-MA-01
TR03110-1	BSI: TR-03110-1 - Advanced Security Mechanisms for Machine Readable Travel Documents. Part 1 - eMRTDs with BAC/PACEv2 and EACv1, v2.10
TR03110-2	BSI: TR-03110-2 - Advanced Security Mechanisms for Machine Readable Travel Documents. Part 2 - Extended Access Control Version 2 (EACv2), Password Authenticated Connection Establishment (PACE), and Restricted Identification (RI) v2.10

840