Electronic Purse Protection Profile

Emission Date	:	04 February 2010
Ref./Version	:	SFPMEI-CC-PP-EP/1.5
Number of pages	:	67 (including 6 header pages)

Table of contents

List of figures

List of tables

1 Protection Profile Introduction

1.1 Protection Profile Identification

Title:	Protection Profile - Electronic Purse
Version number:	1.5
Date:	04 February 2010
Sponsors:	BMS - SFPMEI
Technical editor:	Trusted Labs S.A.S, 5 rue du Bailliage 78000 Versailles
CC version:	3.1 revision 3

1.2 Context

The *Electronic Purse Protection Profile*, sponsored by BMS (Billettique Monétique Services) and SFPMEI (Société Financière du Porte-Monnaie Électronique Interbancaire) is the Electronic Purse (EP) counterpart of the *Intersector Electronic Purse and Purchase Device Protection Profile* [PP/0101], updated to comply with Common Criteria v3.1 revision 3 and to cover the most recent state of the art in terms of security of an Electronic Money (EM) System. The Security Access Module (SAM) counterpart of [PP/0101] is addressed by the *SAM for EM System Protection Profile* [PP SAM]. These new Protection Profiles supersede the Protection Profile [PP/0101].

1.3 Moneo System

The Moneo system is a payment system (Electronic Money system or EM system) intended for low value off line payment transactions. The global functioning of the EM system is based on three cycles:

- a first one consisting in EM creation,
- a second one consisting in payment of goods or services,
- a third one consisting in EM extinguishment.

EM is the counterpart of funds received by the EM issuer. It is defined by the identity of the EM issuer, the currency denomination and the EM amount. The EP that receives amounts in several transactions may aggregate them into a single EM amount. Conversely, the EM amount stored in an EP may be broken up and dispensed in several transactions. The aggregation may also be performed by the SAM.

These cycles can be summarized as follows:

- the purse holder gives funds to the EM issuer who loads the EP with an equivalent amount of EM (**load and quickload transactions**),
- the purse holder asks the merchant for a service and transfers EM from his EP to the SAM (**EM Payment transaction, corresponding to a debit operation for the EP and a credit transaction for the SAM**),
- the merchant asks the EM issuer for credit on its bank account in exchange for EM (**collect transaction**)

During each transaction (load, quickload, EM payment, collect), EM circulates within a closed loop system. Only the EM issuer is authorised to create or extinguish EM. Furthermore, the EM credited on one side (EP or SAM) should be always equal to the EM debited on the other side (SAM or EP, respectively).

Figure 1 gives an overview of the Moneo system:

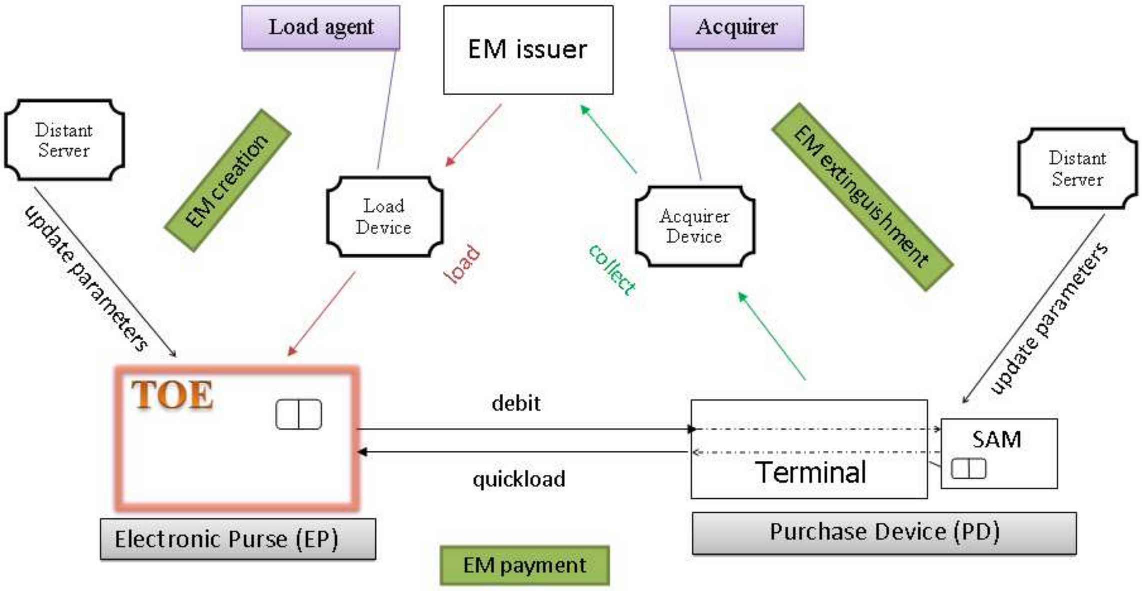

Figure 1: Moneo System

The following sections describe the devices, the actors and the operations involved in the EM system.

1.3.1 Devices

This section describes the devices that play a role in an EM system.

Electronic Purse

An Electronic Purse (EP) is a smartcard or similar device comprising a Security IC, embedded software and data. Its functionalities are similar to traditional purse functionalities with the distinction that it uses Electronic Money (EM) instead of cash money. An EP is used to facilitate payments of low value. The fully operational EP contains various updatable parameters.

Purchase device

A purchase device is a physical device installed at the merchant or a server used to accept payment from an EP in an EM payment transaction. It includes a Secure Access Module (SAM), built on a Security IC. The SAM shall provide the necessary security for the EM payment, the quickload and the collect transactions. It contains various updatable parameters.

Load device

A load device is a server. Its main functionality is to load the EM in the EP in exchange of funds via the EM issuer.

Acquirer device

An acquirer device is a server intended to handle the collect transactions.

Distant Server

A distant server has the responsibility to update the internal parameters of the EP and the SAM.

1.3.2 Actors

EM issuer

The EM issuer guarantees the EM in an EM system. To this end, the EM issuer:

- creates and dispenses EM in exchange for funds received,
- redeems collected EM and extinguishes it.

Purse holder

The purse holder is the person that is in possession of the EP and uses it for EM payment transactions. Purse holders need to protect their EP in the same way as cash.

Merchant

A merchant sells goods or services for which he accepts payment by EP. In order to handle the EM payment transactions, the merchant operates one or more purchase devices in which a SAM stores EM until collect. The merchant is responsible for the operational security of the purchase device he controls.

Load agent

A load agent is a trusted agent of an EM issuer. He executes the load transactions with the EP on behalf of the EM issuer and operates a load device for this purpose. A load agent is responsible for the operational security of its part of the EM system, and must protect the load devices he controls against authorised use. He is also responsible for transferring payment received from the Purse holder to the EM issuer for settlement.

Acquirer

An acquirer is a trusted agent of the EM issuer who is responsible for collecting EM from a SAM concerning EM payment transactions. In order to handle the collect transactions the acquirer operates one or more acquirer devices. The acquirer is responsible for the operational security of its part of the EM system, and must protect the acquirer devices he controls against authorised use. He is also responsible for transferring payment received from the EM issuer to the merchant for settlement.

1.3.3 Transactions

The EM system IT functionalities consist of the following operations:

Load

The EP is credited with an amount of EM created by the EM issuer, via a load agent; the purse holder gives a corresponding amount of funds in turn.

Quickload

This operation is used to load EM into the EP. It is processed offline by the SAM. It may be performed when the purchase amount is greater than the balance of the purse in order to complete with the EM payment.

EM payment (debit/credit operation)

The EP is debited of an EM amount while the SAM is credited with the same amount. The purse holder receives goods or services in return.

Collect

One or several amounts of the EM corresponding to a set of payment transactions stored by a SAM are delivered to the EM issuer via the acquirer device.

Parameters update

Internal EP or SAM parameters are updated by a distant server. Parameters that are addressed are, for instance, the expense limit per transaction and the transaction keys.

1.4 Overview of the TOE

This section describes the Target Of Evaluation (TOE). It addresses the product type and the intended usage of the TOE.

1.4.1 TOE Type

The Target Of Evaluation (TOE) is an Electronic Purse (EP), comprising the IC and all the embedded software necessary to the implementation of the EP functionalities, with the IC certified conformant to [PP-BSI-0035-2007].

This protection profile does not mandate any particular software technology; both native and virtual-machine-based implementations of EP are acceptable, provided they comply with one of the following protection profiles:

- Embedded Software for Smart Secure Devices Protection Profile, Basic and Extended Configurations [ANSSI-CC-PP-2009/02],
- Java Card System Protection Profiles [PP-JCS-Collection]. Note that Java Card System - Open Configuration Protection Profile [PP-JCS-Open] and Java Card System - Closed

Configuration Protection Profile [PP-JCS-Closed], both conformant to CC v3.1 and currently under evaluation, should be preferred to the CC v2.1 profiles defined in [PP-JCS-Collection] once their certificates will be available and whenever they match the characteristics of the TOE.

Conformity to [ANSSI-CC-PP-2009/02] can be claimed for any kind of implementation, native or interpreted (e.g. Java Card). Conformity to [PP-JCS-Collection] ([PP-JCS-Closed] or [PP-JCS-Open]), can be claimed on Java Card electronic purses only. Note that conformity can be reached through a separate evaluation (so that the EP or its underlying platform holds a valid certificate for one of these PPs) or the EP can be evaluated at the same time against this protection profile and against [ANSSI-CC-PP-2009/02] or [PP-JCS-Collection] ([PP-JCS-Closed] or [PP-JCS-Open]). The developer and SFPMEI/BMS shall agree on the PP configuration to be used.

The TOE can provide a contact or a contactless interface with other devices and can be in various final composite product form factors.

Note that the IC may support other applications, either NBA (Non-Banking Application) or EMV payment applications.

There are three types of electronic purses, B1, B2 and B3, described in Table 1. This protection profile addresses electronic purses of type B2 and B3, that is, electronic purses with purse holder authentication capabilities through a Personal Identification Number (PIN). The evaluation of electronic purses of type B1 cannot claim conformity to this PP. Nevertheless, in order to facilitate the use of this PP to build a security target for type B1 purses, all the threats, objectives and requirements that do not apply to type B1 electronic purses are explicitly indicated.

Type B1	Such an EP is called “anonymous EP”. This type of EP coexists with one or more NBA applications or other applications. The EP is not linked to a Purse holder bank account.
Type B2	Such an EP is called “dedicated EP”. It is linked to a Purse holder bank account. Thus, the EP holds an activated PIN. Moreover, quickload transactions are enabled. This type of EP coexists only with one or more NBA applications or other applications.
Type B3	Such an EP is similar to type B2 EP. It is linked to a Purse holder bank account. Thus, a PIN is activated. The PIN is shared by EP and EMV payment applications. Moreover, quickload transactions are enabled. In addition, it integrates an EMV payment application (which is not in the scope of the TOE). This type of EP coexists with one or more NBA applications or other applications.

Table 1: Types of electronic purses

1.4.2 TOE Usage

The TOE is able to:

- Store its amount of EM which defines the balance (BAL and BAL’) of the EP,
- Indicate the available amount of EM,
- Debit EM via **debit** operations,

- Credit EM via load and quickload transactions,
- Update parameters.

1.4.3 Security features of the TOE

The primary functionality of the TOE is to allow the purse holder to make EM payment in a simple, secure and fast way.

The main characteristic of the TOE is that it provides offline EM payment transactions capabilities thus requiring security mechanisms to prevent from fraud:

- Integrity protection of EM during load, quickload and debit operations,
- Integrity and confidentiality protection of cryptographic keys when used or stored,
- Mutual authentication between the TOE and the SAM during quickload and debit operations,
- Mutual authentication between the TOE and the load device during load transactions,
- Authentication of the Purse holder (EP type B2 and type B3) during load and quickload transactions.

An EP contains an end of life date. However, the expiration of this date does not trigger any security mechanism to block the use of the EP. The management of expired EP is achieved through specific flow controls built into the Moneo system itself (either in the terminal or in the load device). The EP does not change its internal state when it expires, the possibility to use it being handled by other components of the Moneo system.

1.4.4 TOE life cycle

The TOE life cycle follows the life cycle described in [PP-BSI-0035-2007]. It is divided into seven distinct phases:

- Phase 1 "Embedded software development" that concerns the development of the IC dedicated software as well as the embedded software for EP purposes,
- Phase 2 "IC design",
- Phase 3 "IC manufacturing",
- Phase 4 "IC packaging",
- Phase 5 "Composite product integration" that covers the composite product finishing process, preparation and shipping to the EP personalization line,
- Phase 6 "Personalisation", where the EP administration data and purse holder's data are loaded into the EP's memory, following the personalization instructions manual,
- Phase 7 "Usage stage" that corresponds to the operational phase of the TOE.

The TOE can be delivered at the end of phases 3, 4, 5 or 6. All the phases before TOE delivery are evaluated according to the EAL4+ requirements. The TOE protects itself in Phase 7. Phases between TOE delivery and Phase 7 are handled according to standard banking best practices. Personalization activities, for example, are conducted using standard secure protocols such as SCP02. Security requirements are defined by BMS concerning theses phases and are agreed upon by means of dedicated contracts with each entity performing them. The requirements are enforced by conducting periodic security audits.

1.4.5 Protection Profile Usage

The requirements presented in this PP define the minimum security rules to which a Security Target (ST) of an EP shall conform, but are in no way exhaustive.

It remains indeed possible to add functionalities or also refer to another PP. However, any modifications to this PP are restricted by the rules defined by the conformance as set forth in Section 1.5.

1.5 Common Criteria Conformance

1.5.1 Conformance claim to CC

This Protection Profile is CC Part 2 conformant [CC2] and CC Part 3 [CC3] conformant.

1.5.2 Conformance claim to a package

The minimum assurance level for the evaluation of an electronic purse against this PP is EAL4 augmented with AVA_VAN.5 and ALC_DVS.2 components.

1.5.3 Conformance claim of the PP

This PP does not claim conformance to any another PP.

1.5.4 Conformance claim to the PP

The conformance to this PP, required for the Security Targets and Protection Profiles claiming conformance to it, is **strict**, as defined in CC Part 1 [CC1].

2 Security problem definition

2.1 Assets

In the following, the description of each asset provides the type of protection required for each asset.

As the main objective of the EM system is to preserve the EM flow, the assets of the EP are the following ones:

2.1.1 User data

Electronic Money (EM)

For the EP, Electronic Money (EM) is an electronic substitute of funds received by the EM issuer. It is defined by the amount (BAL and BAL') stored in the Electronic Purse:

- BAL: The balance of the EP, loaded by load transactions,
- BAL': The balance of the EP, loaded by quickload transactions.

Protection: integrity.

EP identification and validity data (IV DATA)

The EP identification and validity data includes the Primary Account Number, which is a unique sequence of numbers assigned to the EP used for identification purposes, and validity data that allows to detect EP end-of-life.

Protection: integrity.

2.1.2 TSF data

EP application code

The application code embedded in the EP.

Protection: integrity

Flow traceability data

Flow traceability data stands for information on the last transactions stored in log files.

Protection: integrity

Keys

The EP secret keys used for authentication purposes.

Protection: integrity and confidentiality.

Application note:

Keys hold information like:

- a key identifier (ID),
- a key version number,
- a counter of failed access.

If too many failures occur when using a key, this one is blocked and cannot be used anymore.

PIN

The Personal Identification Number allows the authentication of the Purse holder to the TOE.

Protection: integrity and confidentiality.

Application note:

The PIN is required for a load or a quickload transaction in EP of type B2 and B3. EP of type B1 does not hold this asset.

PIN try counter

The PIN try counter manages the number of PIN authentication failed attempts. When it reaches the limit of failed attempts, the PIN verification mechanism is blocked.

Protection: integrity.

Application note:

This asset also includes the PIN try counter limit. The PIN is required for a load or a quickload transaction in EP of type B2 and B3. EP of type B1 does not hold this asset.

Static counters

Static counters cover the configuration (parameters) of the EP:

- the maximum amount of EM stored into the EP,
- the EM maximum amount of debit operations,
- the quickload transaction amount,
- the maximum number of consecutive quickload transactions.

Protection: integrity.

Sequence counters

The sequence counters count the successive transactions:

- load transactions,
- quickload transactions,
- debit operations.

Protection: integrity.

State of the EP

The state of the EP stores information about the EP internal states during its usage phase.

Protection: integrity.

Application note:

The behavior of the EP is modeled using a state machine. A state machine is composed of states defining authorised operations and transitions from one state to another. Transitions are usually triggered by direct or indirect activation of the device inputs (for instance the receipt of a transaction).

2.2 Users

The users of the TOE are the entities who interact with the EP through its physical or logical external interfaces.

2.3 Threats

The threats described here after apply to the usage phase of the TOE. The threat agent or "attacker" is a hostile user that is physically and logically outside the TOE. The threat agent wishes to abuse the assets of the TOE by functional, physical or logical attacks or any combination of them.

2.3.1 COUNTERFEITING

The creation of fake transactions by falsification of assets in order to create or lose EM.

T.COUNTERFEITING_DEBIT

Counterfeiting of a debit operation in order to debit the EP with an EM greater or lesser than the EM amount specified in the transaction; it leads to unauthorized EM creation or loss.

Related asset: EM.

T.COUNTERFEITING_LOAD

Counterfeiting of a load transaction in order to credit the EP without any financial counterpart; it leads to unauthorized EM creation.

Related asset: EM.

T.COUNTERFEITING_QUICKLOAD

Counterfeiting of a quickload transaction in order to credit the EP with an EM greater or lesser than the EM amount specified in the transaction; it leads to unauthorized EM creation or loss.

Related asset: EM.

Application note:

This threat does not apply to EP of type B1.

T.COUNTERFEITING_UPDATE

Counterfeiting of a parameters update transaction in order to change the keys values or the static counters values.

Related assets: Keys, Static counters.

Application note:

The update of counters is performed using a load transaction and the keys during parameter update transaction.

2.3.2 DISCLOSURE

Unauthorised disclosure of assets.

T.DISCLOSURE_KEYS

Unauthorised access to the secret keys.

Related assets: Keys.

T.DISCLOSURE_PIN

Unauthorised access to the PIN value.

Related asset: PIN.

Application note:

An attacker discloses the value of the PIN stored in the EP in order, for instance, to illegitimately authenticate himself subsequently as the Purse holder during a load or a quickload transaction. This threat does not apply to EP of type B1.

2.3.3 LOSS OF INTEGRITY

Unauthorised modification of assets.

T.INTEG_CODE

Unauthorised modification of the TOE code: an attacker modifies the code in order to bypass the security policy of the EP.

Related asset: EP application code.

T.INTEG_EM

Unauthorised modification of stored EM: An attacker modifies the amount of EM stored in the EP (BAL and BAL') in order to increase or decrease the amount.

Related asset: EM.

T.INTEG_FLOW_TRA_DATA

Unauthorised modification of stored flow traceability data: an attacker modifies the log of the last transactions in order to hide potentially malicious operations performed on the EP.

Related asset: Flow traceability data.

T.INTEG_KEYS

Unauthorised modification of stored keys: an attacker modifies the value of the secret keys and associated attributes stored in the EP in order to input a known key.

Related asset: Keys.

T.INTEG_IV_DATA

Unauthorised modification of stored EP identification and validity data: an attacker modifies the value of the EP identification and validity data stored in the EP in order to input another one.

Related asset: EP identification and validity data.

T.INTEG_PIN

Unauthorised modification of stored PIN: an attacker modifies the value of the PIN stored in the EP in order to input a known PIN.

Related asset: PIN.

Application note:

This threat does not apply to EP of type B1.

T.INTEG_SEQ_COUNT

Unauthorised modification of stored sequence counters: an attacker modifies the value of sequence counters in order to force the EP accepting counterfeited or replayed transactions.

Related asset: Sequence counters.

T.INTEG_SM

Unauthorised modification of the State Machine: an attacker modifies or deletes information that defines the current state of the EP in order, for instance, to bypass a secure state.

Related asset: State of the EP.

T.INTEG_STATIC_COUNT

Unauthorised modification of stored static counters: an attacker modifies the value of static counters which define the configuration of the EP in order to bypass controls or limitations enforced by the EM system.

Related asset: Static counters.

T.INTEG_TRY_COUNT

Unauthorised modification of stored PIN try counter: an attacker modifies the value of the PIN try counter stored in the EP in order to change the maximum number of PIN authentication failures thus finally retrieving the PIN.

Related assets: PIN try counter, PIN.

Application note:

This threat does not apply to EP of type B1.

2.3.4 REPLAY

Replay of a previous transaction or the last transaction. Such a replay can be performed immediately after the first sending of the transaction or later.

T.REPLAY_DEBIT

Replay of a debit: an EP is debited several times via a previous complete sequence of debit operations; it leads to unauthorized EM loss.

Related asset: EM.

T.REPLAY_LOAD

Replay of a load transaction: an EP is loaded several times via a previous complete sequence of load operations; it leads to unauthorized EM creation.

Related asset: EM.

T.REPLAY_QUICKLOAD

Replay of a quickload transaction: an EP is loaded several times via a previous complete sequence of quickload operations; it leads to unauthorized EM creation.

Related asset: EM.

Application note:

This threat does not apply to EP of type B1.

T.REPLAY_UPDATE

Replay of a parameters update transaction: an EP is updated several times via a previous complete sequence of parameters update operations; it leads to fraudulent changes of parameters stored in the EP (keys and static counters).

Related assets: keys, Static counters.

2.3.5 REPUDIATION

Repudiation of transactions or part of transactions by EP system actors.

T.REPUD_LOAD

The EP performs a load transaction without the Purse holder authentication. Thus, it can lead to a repudiation of those transactions by the Purse Holder.

Related assets: EM, PIN.

T.REPUD_QUICKLOAD

The EP performs a quickload transaction without the Purse holder authentication. Thus, it can lead to a repudiation of those transactions by the Purse Holder.

Related assets: EM, PIN

Application note:

This threat does not apply to EP of type B1.

2.3.6 STEALING

Stealing of the TOE.

T.STEALING

An attacker steals the EP of the legitimate Purse holder in order to use it for his own purpose.

Application note:

In this threat, the aim of the attacker is to use the EM stored in the EP to perform EM payment. He can also try to illegitimately perform load or quickload transactions on behalf of the real Purse holder.

2.3.7 FAILURE

A failure occurring during a transaction leads to a non-secure state inducing vulnerabilities in the EP.

T.FAIL_BYPASS

The attacker may disrupt the EP during a transaction in order, for instance, to bypass a control.

Related assets: EP application code, EM, State of the EP, Static counters, Sequence counters, PIN try counter.

Application note:

This attack is typically realised by a fault injection method.

T.FAIL_TEARING

The attacker may force the EP into a non stable state by stopping or disrupting the execution of the application instance.

Related assets: EM, static counters, sequence counters, State of the EP, PIN try counter.

Application note:

An attack path is tearing the EP out of the reader while a transaction is being performed.

2.4 Organisational Security Policies

The following organisational security policies are mandatory for the TOE:

OSP.AGGREGATE

During a load transaction, respectively a quickload transaction, the EP is able to aggregate the amounts of loaded EM to the amount stored in the EP, BAL respectively BAL'. The result is a new balance BAL, respectively BAL' with a value equivalent to the sum of all the amounts.

Application note:

The quickload transaction part of this OSP does not apply to EP of type B1.

OSP.BREAK

During a debit operation, the EP is able to break an amount of loaded EM into several smaller amounts of EMs; the result is a new collection of EM amounts whose total value is equivalent to the initial amount.

OSP.DEBIT_BEFORE_CREDIT

Debit from the EP always precedes credit of the SAM during a payment transaction.

OSP.MANAGEMENT OF SECRETS

Management of secret data (e.g. generation, storage, distribution, destruction, loading into the TOE of cryptographic private keys, symmetric keys, user authentication data) performed outside the product on behalf of the TOE Manufacturer shall comply with security organisational policies that enforce integrity and confidentiality of these data. Secret data shared with the user of the product shall be exchanged through trusted channels that protect the data against unauthorised disclosure and modification and allow to detect potential security violations.

OSP.PH_BEHAV

The Purse holder shall keep the EP as a real purse with coins and bank notes and she/he shall not loan it, especially to untrusted persons.

2.5 Assumptions

The following assumption concerns the TOE operational environment, after the TOE delivery.

A.PROTECTION_AFTER_TOE_DELIVERY

The TOE is assumed to be protected by the environment after delivery (may range from Phase 3 to 6) and before entering the final usage phase (Phase 7 of the life cycle). It is assumed that the persons manipulating the TOE in the operational environment follow the TOE guides (user and administrator guidance of the product, installation documentation and personalization guide). It is also assumed that the persons responsible for the application of the procedures contained in the guides, and the persons involved in delivery and protection of the product have the required skills and are aware of the security issues.

Note: The TOE certificate is valid only when the guides are applied. For instance, for pre-personalization or personalization guides, only the described set-up configurations or personalization profiles are covered by the certificate; any divergence would not be covered by the certificate.

3 Security Objectives

3.1 Security Objectives for the TOE

The main security objective for the TOE is to ensure EM flow preservation. The TOE shall implement state of art technology to achieve the following derived security objectives.

O.AGGREGATE

During a load transaction, respectively a quickload transaction, the EP shall aggregate the amounts of the loaded EM to the amount stored in the EP, BAL respectively BAL'. The result is a new balance BAL, respectively BAL' with a value equivalent to the sum of all the amounts.

Application note:

The quickload transaction part of this objective does not apply to EP of type B1.

O.AUTH

The EP shall enforce mutual authentication with external devices (load device, SAM and distant server)prior any transaction.

O.BREAK

During a debit operation, the EP shall break an amount of loaded EM into several smaller amounts of EMs; the result is a new collection of EM amounts whose total value is equivalent to the initial amount.

O.CONF_DATA

The EP shall prevent unauthorised disclosure of confidential TSF data: the PIN and the Keys.

Application note:

The PIN is not relevant to EP of type B1.

O.EM

The EP shall prevent unauthorised creation or loss of EM.

O.INTEG_DATA

The EP shall prevent unauthorised modification of user and TSF data.

O.LIMIT

The EP behavior shall be limited by maximum values defined in the static counters.

O.OPERATE

The EP shall ensure the continued correct operation of its security functions especially in case of abnormal transactions and unexpected interruption.

O.PURSE_HOLDER_AUTH

The EP shall authenticate the Purse holder to the TOE for load and quickload transactions.

Application note:

This authentication is performed using a PIN based authentication. The quickload part of this objective does not apply to EP of type B1.

O.RECORD

The EP shall record the last transactions to support effective security management.

O.REPLAY

The EP shall detect and reject replayed transactions.

O.TAMPER

The EP shall prevent physical tampering of its security critical parts.

3.2 Security objectives for the Operational Environment

OE.DEBIT_BEFORE_CREDIT

Debit must always precede credit during EM payment transaction.

OE.MANAGEMENT_OF_SECRETS

The secret User or TSF data managed outside the TOE shall be protected against unauthorised disclosure and modification.

OE.PROTECTION_AFTER_TOE_DELIVERY

Procedures and controlled environment shall ensure protection of the TOE and related information after delivery. Procedures shall ensure that people involved in TOE delivery and protection have the required skills. The persons using the TOE in the operational environment shall apply the product guides (user and administrator guidance of the product, installation documentation and personalization guide).

OE.TOE-USAGE

The EM issuer shall communicate to the Purse holder the rules dealing with the use of the EP. Especially it must inform the user that:

- he must keep EP the same way he does for a real purse,
- he must not divulgate his PIN to anyone.

The Purse holder shall enforce these rules.

3.3 Security Objectives Rationale

3.3.1 Threats

3.3.1.1 COUNTERFEITING

T.COUNTERFEITING_DEBIT This threat is countered by:

- O.CONF_DATA and O.INTEG_DATA that prevent the unauthorized disclosure or modification of data,
- O.AUTH that requires the authentication of both the TOE and the external device before performing any transaction,
- O.EM which ensures EM flow preservation so that fraudulent creation or loss of EM in the EP using a debit operation is not possible,
- O.RECORD which ensures that the TOE records necessary events and data (flow traceability data) in order to be presented again as an element of evidence of the real transaction.

T.COUNTERFEITING_LOAD This threat is countered by:

- O.CONF_DATA and O.INTEG_DATA that prevent the unauthorized disclosure or modification of data,
- O.AUTH that requires the authentication of both the TOE and the external device before performing any transaction,
- O.EM which ensures EM flow preservation so that fraudulent creation of EM in the EP using a load transaction is not possible,
- O.PURSE_HOLDER_AUTH which ensures the attacker cannot perform load transaction since he does not have the PIN of the TOE,
- O.RECORD which ensures that the TOE records necessary events and data (flow traceability data) in order to be presented again as an element of evidence of the real transaction,
- OE.TOE-USAGE which ensures the TOE issuer provides to the user the rules to securely use its TOE and especially the Purse holder must not provide its PIN to anyone.

T.COUNTERFEITING_QUICKLOAD This threat is countered by:

- O.CONF_DATA and O.INTEG_DATA that prevent the unauthorized disclosure or modification of data,
- O.AUTH that requires the authentication of both the TOE and the external device before performing any transaction,
- O.EM which ensures EM flow preservation so that fraudulent creation or loss of EM in the EP using a quickload transaction is not possible,
- O.PURSE_HOLDER_AUTH which ensures the attacker cannot perform load transactions since he does not have the PIN of the TOE,
- O.RECORD which ensures that the TOE records necessary events and data (flow traceability data) in order to be presented again as an element of evidence of the real transaction,

- OE.TOE-USAGE which ensures the TOE issuer provides to the user the rules to securely use its TOE and especially the Purse holder must not provide its PIN to anyone.

T.COUNTERFEITING_UPDATE This threat is countered by:

- O.CONF_DATA and O.INTEG_DATA that prevent the unauthorized disclosure or modification of data,
- O.AUTH that requires the authentication of both the TOE and the external device before performing any transaction.

3.3.1.2 DISCLOSURE

T.DISCLOSURE_KEYS This threat is countered by:

- O.CONF_DATA which prevents from illegal disclosure of the security assets of the TOE,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered in order to allow the disclosure of the assets.

T.DISCLOSURE_PIN This threat is countered by:

- O.CONF_DATA which prevents from illegal disclosure of the security assets of the TOE,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered in order to allow the disclosure of the assets,
- OE.TOE-USAGE which ensures the TOE issuer provides to the user the rules to securely use his EP and especially the Purse holder must not provide his PIN to anyone.

3.3.1.3 LOSS OF INTEGRITY

T.INTEG_CODE This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_EM This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.EM which ensures EM flow preservation so that fraudulent creation of EM in the EP is not possible,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_FLOW_TRA_DATA This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.RECORD which ensures that the TOE records necessary events and data (flow traceability data) in order to be presented again as an element of evidence of the real transaction,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_KEYS This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_IV_DATA This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_PIN This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_SEQ_COUNT This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_SM This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_STATIC_COUNT This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

T.INTEG_TRY_COUNT This threat is countered by:

- O.INTEG_DATA which prevents from illegal modification of the security assets of the TOE,
- O.OPERATE which ensures the correct operation of the related transaction,
- O.TAMPER which ensures security information and especially security assets of the TOE cannot be physically tampered.

3.3.1.4 REPLAY

T.REPLAY_DEBIT This threat is countered by:

- O.EM which ensures EM flow preservation so that fraudulent loss of EM in the EP using a debit operation is not possible,
- O.REPLAY which ensures the EP will operate in a continuous secure state in case of replayed debit operation; the replayed transaction will be detected and rejected by the EP.

T.REPLAY_LOAD This threat is countered by:

- O.EM which ensures EM flow preservation so that fraudulent creation of EM in the EP using a load transaction is not possible,
- O.REPLAY which ensures the EP will operate in a continuous secure state in case of replayed load transaction; the replayed transaction will be detected and rejected by the EP.

T.REPLAY_QUICKLOAD This threat is countered by:

- O.EM which ensures EM flow preservation so that fraudulent creation of EM in the EP using a quickload transaction is not possible,
- O.REPLAY which ensures the EP will operate in a continuous secure state in case of replayed quickload transaction; the replayed transaction will be detected and rejected by the EP.

T.REPLAY_UPDATE This threat is countered by:

- O.REPLAY which ensures the EP will operate in a continuous secure state in case of replayed parameters update transaction; the replayed transaction will be detected and rejected by the EP.

3.3.1.5 REPUDIATION

T.REPUD_LOAD This threat is countered by:

- O.AUTH that requires the authentication of both the TOE and the external device before performing any transaction,
- O.PURSE_HOLDER_AUTH which ensures the attacker can only perform load transaction if he has been previously authenticated,
- OE.TOE-USAGE which ensures the TOE issuer provides to the user the rules to securely use his EP and especially the Purse holder must not provide his PIN to anyone.

Thus, if the PIN has been entered, kept secure and an authenticated communication has been used, the transaction cannot be repudiated.

T.REPUD_QUICKLOAD This threat is countered by:

- O.AUTH that requires the authentication of both the TOE and the external device before performing any transaction,
- O.PURSE_HOLDER_AUTH which ensures the attacker can only perform quickload transaction if he has been previously authenticated to the TOE,
- OE.TOE-USAGE which ensures the TOE issuer provides to the user the rules to securely use his EP and especially the Purse holder must not provide his PIN to anyone.

Thus, if the PIN has been entered, kept secure and the TOE secret keys have been used, the transaction cannot be repudiated.

3.3.1.6 STEALING

T.STEALING This threat is countered by:

- O.LIMIT which avoids the storage of important amount of EM in the EP and ensures that if the TOE is stolen the attacker can only retrieve a limited amount of EM,
- O.PURSE_HOLDER_AUTH which ensures the attacker cannot perform load or quickload transactions since he does not have the PIN of the TOE,
- OE.TOE-USAGE which ensures the TOE issuer provides to the user the rules to securely use its TOE.

3.3.1.7 FAILURE

T.FAIL_BYPASS This threat is countered by:

- O.TAMPER which ensures the TOE cannot be disrupted using physical interfaces.

T.FAIL_TEARING This threat is countered by:

- O.OPERATE which ensures the correct operation even in case of stress or abnormal transaction such as tearing,
- O.TAMPER which ensures the TOE cannot be tampered in order to put the TOE in such an unstable state.

3.3.2 Organisational Security Policies

OSP.AGGREGATE This OSP is directly covered by O.AGGREGATE.

OSP.BREAK This OSP is directly covered by O.BREAK.

OSP.DEBIT_BEFORE_CREDIT This OSP is directly covered by OE.DEBIT_BEFORE_CREDIT.

OSP.MANAGEMENT OF SECRETS OE.MANAGEMENT_OF_SECRETS directly covers the organisational security policy.

OSP.PH_BEHAV This OSP is covered by:

- OE.TOE-USAGE which ensure that the Purse holder is aware of the security rules related to the TOE.

3.3.3 Assumptions

A.PROTECTION_AFTER_TOE_DELIVERY OE.PROTECTION_AFTER_TOE_DELIVERY directly covers the assumption.

3.3.4 SPD and Security Objectives

Threats	Security Objectives	Rationale
T.COUNTERFEITING_DEBIT	O.EM, O.RECORD, O.AUTH, O.CONF_DATA, O.INTEG_DATA	Section 3.3.1
T.COUNTERFEITING_LOAD	O.EM, OE.TOE-USAGE, O.AUTH, O.RECORD, O.PURSE_HOLDER_AUTH, O.CONF_DATA, O.INTEG_DATA	Section 3.3.1
T.COUNTERFEITING_QUICKLOAD	O.EM, OE.TOE-USAGE, O.AUTH, O.PURSE_HOLDER_AUTH, O.RECORD, O.CONF_DATA, O.INTEG_DATA	Section 3.3.1
T.COUNTERFEITING_UPDATE	O.AUTH, O.CONF_DATA, O.INTEG_DATA	Section 3.3.1
T.DISCLOSURE_KEYS	O.CONF_DATA, O.TAMPER	Section 3.3.1
T.DISCLOSURE_PIN	O.CONF_DATA, O.TAMPER, OE.TOE-USAGE	Section 3.3.1
T.INTEG_CODE	O.INTEG_DATA, O.TAMPER, O.OPERATE	Section 3.3.1
T.INTEG_EM	O.TAMPER, O.INTEG_DATA, O.EM, O.OPERATE	Section 3.3.1
T.INTEG_FLOW_TRA_DATA	O.INTEG_DATA, O.TAMPER, O.OPERATE, O.RECORD	Section 3.3.1
T.INTEG_KEYS	O.INTEG_DATA, O.TAMPER, O.OPERATE	Section 3.3.1
T.INTEG_IV_DATA	O.INTEG_DATA, O.TAMPER, O.OPERATE	Section 3.3.1
T.INTEG_PIN	O.INTEG_DATA, O.TAMPER, O.OPERATE	Section 3.3.1
T.INTEG_SEQ_COUNT	O.INTEG_DATA, O.TAMPER, O.OPERATE	Section 3.3.1
T.INTEG_SM	O.INTEG_DATA, O.TAMPER, O.OPERATE	Section 3.3.1
T.INTEG_STATIC_COUNT	O.INTEG_DATA, O.TAMPER, O.OPERATE	Section 3.3.1
T.INTEG_TRY_COUNT	O.INTEG_DATA, O.TAMPER, O.OPERATE	Section 3.3.1
T.REPLAY_DEBIT	O.EM, O.REPLAY	Section 3.3.1

Threats	Security Objectives	Rationale
T.REPLAY_LOAD	O.EM, O.REPLAY	Section 3.3.1
T.REPLAY_QUICKLOAD	O.REPLAY, O.EM	Section 3.3.1
T.REPLAY_UPDATE	O.REPLAY	Section 3.3.1
T.REPUD_LOAD	OE.TOE-USAGE, O.AUTH, O.PURSE_HOLDER_AUTH	Section 3.3.1
T.REPUD_QUICKLOAD	OE.TOE-USAGE, O.AUTH, O.PURSE_HOLDER_AUTH	Section 3.3.1
T.STEALING	OE.TOE-USAGE, O.LIMIT, O.PURSE_HOLDER_AUTH	Section 3.3.1
T.FAIL_BYPASS	O.TAMPER	Section 3.3.1
T.FAIL_TEARING	O.OPERATE, O.TAMPER	Section 3.3.1

Table 2 Threats and Security Objectives - Coverage

Security Objectives	Threats
O.AGGREGATE	
O.AUTH	T.COUNTERFEITING_DEBIT, T.COUNTERFEITING_LOAD, T.COUNTERFEITING_QUICKLOAD, T.COUNTERFEITING_UPDATE, T.REPUD_LOAD, T.REPUD_QUICKLOAD
O.BREAK	
O.CONF_DATA	T.COUNTERFEITING_DEBIT, T.COUNTERFEITING_LOAD, T.COUNTERFEITING_QUICKLOAD, T.COUNTERFEITING_UPDATE, T.DISCLOSURE_KEYS, T.DISCLOSURE_PIN
O.EM	T.COUNTERFEITING_DEBIT, T.COUNTERFEITING_LOAD, T.COUNTERFEITING_QUICKLOAD, T.INTEG_EM, T.REPLAY_DEBIT, T.REPLAY_LOAD, T.REPLAY_QUICKLOAD
O.INTEG_DATA	T.COUNTERFEITING_DEBIT, T.COUNTERFEITING_LOAD, T.COUNTERFEITING_QUICKLOAD, T.COUNTERFEITING_UPDATE, T.INTEG_CODE, T.INTEG_EM, T.INTEG_FLOW_TRA_DATA, T.INTEG_KEYS, T.INTEG_IV_DATA, T.INTEG_PIN, T.INTEG_SEQ_COUNT, T.INTEG_SM, T.INTEG_STATIC_COUNT, T.INTEG_TRY_COUNT
O.LIMIT	T.STEALING
O.OPERATE	T.INTEG_CODE, T.INTEG_EM, T.INTEG_FLOW_TRA_DATA, T.INTEG_KEYS, T.INTEG_IV_DATA, T.INTEG_PIN, T.INTEG_SEQ_COUNT, T.INTEG_SM, T.INTEG_STATIC_COUNT, T.INTEG_TRY_COUNT, T.FAIL_TEARING

Security Objectives	Threats
O.PURSE_HOLDER_AUTH	T.COUNTERFEITING_LOAD, T.COUNTERFEITING_QUICKLOAD, T.REPUD_LOAD, T.REPUD_QUICKLOAD, T.STEALING
O.RECORD	T.COUNTERFEITING_DEBIT, T.COUNTERFEITING_LOAD, T.COUNTERFEITING_QUICKLOAD, T.INTEG_FLOW_TRA_DATA
O.REPLAY	T.REPLAY_DEBIT, T.REPLAY_LOAD, T.REPLAY_QUICKLOAD, T.REPLAY_UPDATE
O.TAMPER	T.DISCLOSURE_KEYS, T.DISCLOSURE_PIN, T.INTEG_CODE, T.INTEG_EM, T.INTEG_FLOW_TRA_DATA, T.INTEG_KEYS, T.INTEG_IV_DATA, T.INTEG_PIN, T.INTEG_SEQ_COUNT, T.INTEG_SM, T.INTEG_STATIC_COUNT, T.INTEG_TRY_COUNT, T.FAIL_BYPASS, T.FAIL_TEARING
OE.DEBIT_BEFORE_CREDIT	
OE.MANAGEMENT_OF_SECRETS	
OE.PROTECTION_AFTER_TOE_DELIVERY	
OE.TOE-USAGE	T.COUNTERFEITING_LOAD, T.COUNTERFEITING_QUICKLOAD, T.DISCLOSURE_PIN, T.REPUD_LOAD, T.REPUD_QUICKLOAD, T.STEALING

Table 3 Security Objectives and Threats - Coverage

Organisational Security Policies	Security Objectives	Rationale
OSP.AGGREGATE	O.AGGREGATE	Section 3.3.2
OSP.BREAK	O.BREAK	Section 3.3.2
OSP.DEBIT_BEFORE_CREDIT	OE.DEBIT_BEFORE_CREDIT	Section 3.3.2
OSP.MANAGEMENT OF SECRETS	OE.MANAGEMENT_OF_SECRETS	Section 3.3.2
OSP.PH_BEHAV	OE.TOE-USAGE	Section 3.3.2

Table 4 OSPs and Security Objectives - Coverage

Security Objectives	Organisational Security Policies
O.AGGREGATE	OSP.AGGREGATE
O.AUTH	
O.BREAK	OSP.BREAK
O.CONF_DATA	
O.EM	
O.INTEG_DATA	
O.LIMIT	
O.OPERATE	
O.PURSE_HOLDER_AUTH	
O.RECORD	
O.REPLAY	
O.TAMPER	
OE.DEBIT_BEFORE_CREDIT	OSP.DEBIT_BEFORE_CREDIT
OE.MANAGEMENT_OF_SECRETS	OSP.MANAGEMENT OF SECRETS
OE.PROTECTION_AFTER_TOE_DELIVERY	
OE.TOE-USAGE	OSP.PH_BEHAV

Table 5 Security Objectives and OSPs - Coverage

Assumptions	Security objectives for the Operational Environment	Rationale
A.PROTECTION_AFTER_TOE_DELIVERY	OE.PROTECTION_AFTER_TOE_DELIVERY	Section 3.3.3

Table 6 Assumptions and Security Objectives for the Operational Environment - Coverage

Security objectives for the Operational Environment	Assumptions
OE.DEBIT_BEFORE_CREDIT	
OE.MANAGEMENT_OF_SECRETS	
OE.PROTECTION_AFTER_TOE_DELIVERY	A.PROTECTION_AFTER_TOE_DELIVERY
OE.TOE-USAGE	

Table 7 Security Objectives for the Operational Environment and Assumptions - Coverage

4 Security Functional Requirements

4.1 Security Functional Requirements

The Security Target author shall instantiate all the operations in the following Security Functional Requirements according to Moneo Electronic Purse specifications.

4.1.1 *Authentication*

FIA_UAU.1/Distant_server Timing of authentication

FIA_UAU.1.1/Distant_server The TSF shall allow

- **to transfer the information necessary to perform parameters update transactions from the EP to the distant server,**
- **the distant server to authenticate the EP,**
- **[assignment: list of TSF mediated actions]**

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

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

Refinement:

The user stands for the distant server. User authentication stands for distant server authentication by the EP. Parameters update transactions cannot be performed before the user is authenticated.

FIA_UAU.1/Load_device Timing of authentication

FIA_UAU.1.1/Load_device The TSF shall allow

- **to transfer the information necessary for performing load transactions from the EP to the load device,**
- **the load device to authenticate the EP,**
- **[assignment: list of TSF mediated actions]**

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

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

Refinement:

The "user" stands for the load device. User authentication stands for load device authentication by the EP. Load transactions cannot be performed before user authentication.

FIA_UAU.1/Purseholder Timing of authentication

FIA_UAU.1.1/Purseholder The TSF shall allow

- **to read from the EP the EM amount, the value of the static counters and the flow traceability data,**
- **debit operations by an authenticated SAM,**
- **[assignment: list of TSF mediated actions]**

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

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

Refinement:

The "user" stands for the Purse holder. User authentication stands for Purse holder PIN-based authentication. Load and quickload transactions cannot be performed before user authentication.

Application note:

This functional component instance does not apply to EP of type B1.

FIA_UAU.1/SAM Timing of authentication

FIA_UAU.1.1/SAM The TSF shall allow

- **the interaction between the EP and the SAM in order to exchange the information necessary for performing quickload transactions (including key index and versions to be used for authentication),**
- **the SAM to authenticate the EP,**
- **[assignment: list of TSF mediated actions]**

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

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

Refinement:

The "user" stands for the SAM. User authentication stands for SAM authentication by the EP. Quickload transactions cannot be performed before user authentication.

Application note:

This functional component instance does not apply to EP of type B1.

FIA_UAU.6/Distant_server Re-authenticating

FIA_UAU.6.1/Distant_server The TSF shall re-authenticate the user under the conditions

- **beginning of a parameters update transaction**.

Refinement:

The "user" stands for the distant server.

FIA_UAU.6/Load_device Re-authenticating

FIA_UAU.6.1/Load_device The TSF shall re-authenticate the user under the conditions

- **beginning of a load transaction**.

Refinement:

The "user" stands for the load device.

FIA_UAU.6/Purseholder Re-authenticating

FIA_UAU.6.1/Purseholder The TSF shall re-authenticate the user under the conditions

- **beginning of a load transaction,**
- **beginning of a quickload transaction**.

Refinement:

The "user" stands for the Purse holder.

Application note:

This functional component instance does not apply to EP of type B1.

FIA_UAU.6/SAM Re-authenticating

FIA_UAU.6.1/SAM The TSF shall re-authenticate the user under the conditions

- **beginning of a payment transaction,**
- **beginning of a quickload transaction**.

Refinement:

The user stands for the SAM.

Application note:

This functional component instance does not apply to EP of type B1.

FIA_UAU.3 Unforgeable authentication

FIA_UAU.3.1 The TSF shall **prevent** use of authentication data that has been forged by any user of the TSF.

FIA_UAU.3.2 The TSF shall **prevent** use of authentication data that has been copied from any other user of the TSF.

Refinement:

The "user" stands for the SAM, the load device or the distant server.

FIA_UAU.4 Single-use authentication mechanisms

FIA_UAU.4.1 The TSF shall prevent reuse of authentication data related to**:**

- **the SAM authentication mechanism,**
- **the load device authentication mechanism,**
- **the distant server authentication mechanism,**
- **[assignment: identified authentication mechanism(s)]**.

4.1.2 Storage Integrity

FDP_SDI.1 Stored data integrity monitoring

FDP_SDI.1.1 The TSF shall monitor user data stored in containers controlled by the TSF for **any integrity errors** on all objects, based on the following attributes: **[assignment: user data attributes]**.

4.1.3 Security properties on communications

FCO_NRO.2 Enforced proof of origin

FCO_NRO.2.1 The TSF shall enforce the generation of evidence of origin for transmitted **quickload transactions** at all times.

FCO_NRO.2.2 The TSF shall be able to relate the **[assignment: list of attributes]** of the originator of the information, and the **[assignment: list of information fields]** of the information to which the evidence applies.

FCO_NRO.2.3 The TSF shall provide a capability to verify the evidence of origin of information to **EM issuer, recipient and originator** given **[assignment: limitations on the evidence of origin]**.

Application note:

This functional component instance is only relevant for the type B2 and B3 for the quickload transactions, where the EP enforces proof-of-origin of the transactions. Indeed, quickload transactions are sealed by the EP with load keys. It does not apply to EP of type B1.

FPT_ITC.1 Inter-TSF confidentiality during transmission

FPT_ITC.1.1 The TSF shall protect all TSF data transmitted from the TSF to another trusted IT product from unauthorised disclosure during transmission.

Refinement:

The "trusted IT product" stands for the remote distant server, the "transmission" occurs during the parameters update transaction and the "TSF data" stands for EP parameters being updated (e.g. static counters and transactions keys).

FPT_ITI.1 Inter-TSF detection of modification

FPT_ITI.1.1 The TSF shall provide the capability to detect modification of all TSF data during transmission between the TSF and another trusted IT product within the following metric: **[assignment: a defined modification metric]**.

FPT_ITI.1.2 The TSF shall provide the capability to verify the integrity of all TSF data transmitted between the TSF and another trusted IT product and perform **[assignment: action to be taken]** if modifications are detected.

Refinement:

The "trusted IT product" stands for the remote distant server, the "transmission" occurs during the parameters update transaction and the "TSF data" stands for EP parameters being updated (e.g. static counters and transaction keys).

FPT_RPL.1 Replay detection

FPT_RPL.1.1 The TSF shall detect replay for the following entities:

- **debit transactions,**
- **load transactions,**
- **quickload transactions,**
- **parameters update transactions**.

FPT_RPL.1.2 The TSF shall perform

- **the abort of the transaction in process,**
- **[assignment: list of specific actions]**

when replay is detected.

Application note:

The replay of quickload transactions is not relevant for EP of type B1.

4.1.4 Access control security policy

FDP_ACC.2 Complete access control

FDP_ACC.2.1 The TSF shall enforce the **Assets Security policy** on **[assignment: list of subjects and objects]** and all operations among subjects and objects covered by the SFP.

FDP_ACC.2.2 The TSF shall ensure that all operations between any subject controlled by the TSF and any object controlled by the TSF are covered by an access control SFP.

Refinement:

The "objects" under this security access control policy refer to user and TSF data. The "subjects" are the entities that can access these data through specific "operations".

FDP_ACF.1 Security attribute based access control

FDP_ACF.1.1 The TSF shall enforce the **Assets Security policy** to objects based on the following: **[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]**.

FDP_ACF.1.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: **[assignment: rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects]**.

FDP_ACF.1.3 The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: **[assignment: rules, based on security attributes, that explicitly authorise access of subjects to objects]**.

FDP_ACF.1.4 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]**.

Refinement:

Access control rules shall enforce confidentiality and integrity of objects, depending on their characteristics.

FMT_MSA.1/Assets Management of security attributes

FMT_MSA.1.1/Assets The TSF shall enforce the **Assets Security policy** to restrict the ability to **[selection: change_default, query, modify, delete, [assignment: other operations]]** the security attributes **[assignment: list of security attributes]** to **[assignment: the authorised identified roles]**.

FMT_MSA.3/Assets Static attribute initialisation

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

FMT_MSA.3.2/Assets The TSF shall allow the **[assignment: the authorised identified roles]** to specify alternative initial values to override the default values when an object or information is created.

4.1.5 Flow control security policy

FDP_IFC.1 Subset information flow control

FDP_IFC.1.1 The TSF shall enforce the **Transaction policy** on **[assignment: list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SFP]**.

Refinement:

The "operations" under this security flow control policy refer to the debit, load, quickload, and parameters update transactions. The "subjects" are the entities that play an active role in the execution of those transactions and the "information" is the data transmitted during the transactions.

Application note:

Quickload transactions are not relevant for EP of type B1.

FDP_IFF.1 Simple security attributes

FDP_IFF.1.1 The TSF shall enforce the **Transaction policy** based on the following types of subject and information security attributes: **[assignment: list of subjects and information controlled under the indicated SFP, and for each, the security attributes]**.

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: **[assignment: for each operation, the security attribute-based relationship that must hold between subject and information security attributes]**.

FDP_IFF.1.3 The TSF shall enforce the **[assignment: additional information flow control SFP rules]**.

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

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

Application note:

Rules shall typically address the authentication requirements (Purse holder authentication and mutual authentication between the EP and the SAM or the Load device), the conditions on the amounts that can be loaded on or debited from the EP depending on various maximum limit amounts, and the number of transactions of a given a type allowed by the EP. The instantiation of this requirement shall comply with the rules defined in the Moneo specifications.

FDP_ITC.1 Import of user data without security attributes

FDP_ITC.1.1 The TSF shall enforce the **Transaction policy** when importing user data, controlled under the SFP, from outside of the TOE.

FDP_ITC.1.2 The TSF shall ignore any security attributes associated with the user data when imported from outside the TOE.

FDP_ITC.1.3 The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: **[assignment: additional importation control rules]**.

Application note:

"User data" stands for user and TSF data entering the EP during load, quickload or debit transactions.

FMT_MSA.1/Transaction Management of security attributes

FMT_MSA.1.1/Transaction The TSF shall enforce the **Transaction policy** to restrict the ability to **[selection: change_default, query, modify, delete, [assignment: other operations]]** the security attributes **[assignment: list of security attributes]** to **[assignment: the authorised identified roles]**.

FMT_MSA.3/Transaction Static attribute initialisation

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

FMT_MSA.3.2/Transaction The TSF shall allow the **[assignment: the authorised identified roles]** to specify alternative initial values to override the default values when an object or information is created.

4.1.6 Audit

FAU_GEN.1 Audit data generation

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

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

b) All auditable events for the **not specified** level of audit; and

c) **the following auditable events:**

- o **last load and quickload transactions,**

- **last debit operations,**
- **[assignment: other specifically defined auditable events]**.

Refinement:

The audit functions are active all the time, hence item a) Start-up and shutdown of the TOE is not relevant.

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

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

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, **[assignment: other audit relevant information]**.

Application note:

For each type auditable event, the Security Target author shall define the maximum number of information stored.

Date and time of the event are determined by the terminal.

Quickload transactions are not relevant for EP of type B1.

FAU_SAR.1 Audit review

FAU_SAR.1.1 The TSF shall provide **[assignment: authorised users]** with the capability to read **[assignment: list of audit information]** from the audit records.

FAU_SAR.1.2 The TSF shall provide the audit records in a manner suitable for the user to interpret the information.

FAU_STG.1 Protected audit trail storage

FAU_STG.1.1 The TSF shall protect the stored audit records in the audit trail from unauthorised deletion.

FAU_STG.1.2 The TSF shall be able to **detect** unauthorised modifications to the stored audit records in the audit trail.

4.1.7 Fail safe

FPT_RCV.4 Function recovery

FPT_RCV.4.1 The TSF shall ensure that **debit, load, quickload and parameters update transactions** have the property that the function either completes successfully, or for the indicated failure scenarios, recovers to a consistent and secure state.

Application note:

Quickload transactions are not relevant for EP of type B1.

4.1.8 Cryptography and random generation

FCS_COP.1 Cryptographic operation

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

Refinement:

The "list of cryptographic operations" shall include all the operations that support the mutual authentication of EP and external IT products (especially the SAM, the load device and the distant server).

FIA_SOS.2 TSF Generation of secrets

FIA_SOS.2.1 The TSF shall provide a mechanism to generate secrets that meet **[assignment: random numbers quality metric]**.

FIA_SOS.2.2 The TSF shall be able to enforce the use of TSF generated secrets for **[assignment: list of TSF functions]**.

Refinement:

"Secrets" stand for random values.

Application note:

The "quality metric" shall meet national schemes requirements (e.g. [CRYPTO] in France).

4.1.9 Platform and IC protection

FPT_PHP.2 Notification of physical attack

FPT_PHP.2.1 The TSF shall provide unambiguous detection of physical tampering that might compromise the TSF.

FPT_PHP.2.2 The TSF shall provide the capability to determine whether physical tampering with the TSF's devices or TSF's elements has occurred.

FPT_PHP.2.3 For **[assignment: list of TSF devices/elements for which active detection is required]**, the TSF shall monitor the devices and elements and notify **[assignment: a designated user or role]** when physical tampering with the TSF's devices or TSF's elements has occurred.

Application note:

The TSF shall rely on its Integrated Circuit certified against [PP-BSI-0035-2007] to detect physical attacks. The Security Target author shall explain in the TOE Summary Specification how the IC and the embedded software cooperate to implement this requirement.

FPT_PHP.3 Resistance to physical attack

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

Application note:

The TSF shall rely on its Integrated Circuit certified against [PP-BSI-0035-2007] to resist to physical tampering scenarios. The Security Target author shall explain in the TOE Summary Specification how the IC and the Embedded Software cooperate to implement this requirement.

4.2 Security Assurance Requirements

The security assurance requirement level is EAL4 augmented with ALC_DVS.2 and AVA_VAN.5.

4.3 Security Requirements Rationale

4.3.1 Objectives

4.3.1.1 Security Objectives for the TOE

O.AGGREGATE This objective is covered by:

- FDP_IFF.1, FDP_IFC.1, FDP_ITC.1, FMT_MSA.3/Transaction and FMT_MSA.1/Transaction which ensure the TOE is able to aggregate the EM amounts to its global overall amount (BAL respectively BAL') when the TOE processes a load, respectively a quickload transaction.

O.AUTH This objective is covered by:

- FIA_UAU.1/SAM, FIA_UAU.1/Load_device, FIA_UAU.1/Distant_server which require the authentication of the corresponding external device to the TOE to perform a transaction,
- FIA_UAU.3, which prevents against use of forged authentication data,
- FIA_UAU.4 which prevents against reuse of authentication data,
- FIA_UAU.6/SAM, FIA_UAU.6/Load_device, FIA_UAU.6/Distant_server which require the re-authentication of the corresponding external device to the TOE each time a transaction with an external device needs to be (re)initiated,
- FIA_SOS.2 which ensures the TOE can generate random value to perform authentication processes.

O.BREAK This objective is covered by:

- FDP_IFF.1, FDP_IFC.1, FDP_ITC.1, FMT_MSA.3/Transaction and FMT_MSA.1/Transaction which defines the information flow control policy within the TOE for the protection of the security assets of the TOE, in particular the rules to apply for the case of any transaction. This includes the TOE is able to break an amount of loaded EM into several smaller amounts of EMs.

O.CONF_DATA This objective is covered by:

- FPT_ITC.1 which ensures the confidentiality of the TSF data during transmission for the case of the parameters update transactions,
- FDP_ACC.2, FDP_ACF.1, FMT_MSA.3/Assets and FMT_MSA.1/Assets which ensures that security assets cannot be retrieved without passing by a secure access control,
- FPT_PHP.2 and FPT_PHP.3 that address physical protection of confidential data,
- FPT_RCV.4 which ensures that the TOE data cannot be disclosed and that it is impossible to put the TOE in an inconsistent and unstable state allowing to retrieve the security assets.

O.EM This objective is directly covered by:

- FCO_NRO.2 which ensures the TSF can prove the validity of loaded EM for the case of quickload transactions.
- FDP_IFC.1, FDP_IFF.1, FDP_ITC.1, FMT_MSA.3/Transaction and FMT_MSA.1/Transaction which enforce an information flow control policy applicable to user data and TSF data,
- FPT_RCV.4 which ensures that a transaction is performed completely or is aborted and that a secure state is preserved,
- FPT_RPL.1 which ensures that load, quickload, debit and parameters update transactions are protected against replay; the TSF can detect it and react by aborting the transaction in process,
- FCS_COP.1 which ensures that efficient cryptographic means are used to protect EM on the TOE and during transactions.

This specific objective is also indirectly covered by all the SFRs related to the other objectives. Indeed, these objectives ensure the correct and secure operation of the TOE which guarantees that it is not possible to create or loss EM (i.e. modify) from outside or even inside the TOE.

O.INTEG_DATA This objective is covered by:

- FDP_SDI.1 which ensures that user data stored in the TOE are monitored against any integrity error,
- FPT_ITI.1 which ensures the integrity of the TSF data during transmission for the case of the parameters update transactions,
- FDP_ACC.2, FDP_ACF.1, FMT_MSA.3/Assets and FMT_MSA.1/Assets which ensure that security assets cannot be modified without passing by a secure access control,
- FPT_PHP.2 and FPT_PHP.3 that address physical protection of integer data,
- FPT_RCV.4 which ensures that the TOE data cannot be modified and put in an inconsistent and unstable state which could alter the integrity of the TOE security assets,
- FAU_STG.1 which protects the audit record stored in the TOE against unauthorised deletion and detects any attack against this security asset.

O.LIMIT This objective is covered by:

- FDP_IFF.1, FDP_IFC.1, FDP_ITC.1, FMT_MSA.3/Transaction and FMT_MSA.1/Transaction which define the access control policy within the TOE for the protection of the security assets of the TOE, in particular the rules to apply for the case of any transaction. This includes the EM limited by the value of a maximum amount when the TOE processes a load or a quickload transaction.

O.OPERATE This objective is covered by:

- FPT_RCV.4 which ensures that the TOE cannot enter in an unstable and inconsistent state, even due to a failure during a transaction. Indeed, such an unstable state could lead to incorrect behaviour of the TOE.

O.PURSE_HOLDER_AUTH This objective is covered by:

- FIA_UAU.1/Purseholder which ensures the authentication of the Purse holder,
- FIA_UAU.6/Purseholder which ensures that authentication of the Purse holder is required each time a load or a quickload transaction needs to be (re)initiated.

O.RECORD This objective is covered by:

- FAU_GEN.1 which requires the generation of an audit record of the last performed transactions,
- FAU_SAR.1 which allows the capability to read this audit record.

O.REPLAY This objective is covered by:

- FPT_RPL.1 which ensures that load, quickload, debit and parameters update transactions are protected against replay; the TSF can detect it and react by aborting the transaction in process,
- FIA_SOS.2 which ensures the TOE can generate random value to enforce the protection against replay attacks.

O.TAMPER This objective is covered by:

- FPT_PHP.2 which requires detection of the physical tampering,

- FPT_PHP.3 which requires the protection against physical tampering.

4.3.2 Rationale tables of Security Objectives and SFRs

Security Objectives	Security Functional Requirements	Rationale
O.AGGREGATE	FDP_ITC.1, FDP_IFC.1, FDP_IFF.1, FMT_MSA.3/Transaction, FMT_MSA.1/Transaction	Section 4.3.1
O.AUTH	FIA_UAU.1/SAM, FIA_UAU.3, FIA_UAU.6/SAM, FIA_SOS.2, FIA_UAU.1/Load_device, FIA_UAU.1/Distant_server, FIA_UAU.4, FIA_UAU.6/Load_device, FIA_UAU.6/Distant_server	Section 4.3.1
O.BREAK	FDP_ITC.1, FDP_IFC.1, FDP_IFF.1, FMT_MSA.3/Transaction, FMT_MSA.1/Transaction	Section 4.3.1
O.CONF_DATA	FPT_RCV.4, FPT_PHP.2, FPT_PHP.3, FDP_ACC.2, FDP_ACF.1, FPT_ITC.1, FMT_MSA.3/Assets, FMT_MSA.1/Assets	Section 4.3.1
O.EM	FPT_RCV.4, FCS_COP.1, FCO_NRO.2, FDP_ITC.1, FPT_RPL.1, FDP_IFC.1, FDP_IFF.1, FMT_MSA.3/Transaction, FMT_MSA.1/Transaction	Section 4.3.1
O.INTEG_DATA	FDP_SDI.1, FPT_RCV.4, FPT_PHP.2, FPT_PHP.3, FAU_STG.1, FDP_ACC.2, FDP_ACF.1, FPT_ITI.1, FMT_MSA.3/Assets, FMT_MSA.1/Assets	Section 4.3.1
O.LIMIT	FDP_ITC.1, FDP_IFC.1, FDP_IFF.1, FMT_MSA.3/Transaction, FMT_MSA.1/Transaction	Section 4.3.1
O.OPERATE	FPT_RCV.4	Section 4.3.1
O.PURSE_HOLDER_AUTH	FIA_UAU.1/Purseholder, FIA_UAU.6/Purseholder	Section 4.3.1
O.RECORD	FAU_GEN.1, FAU_SAR.1	Section 4.3.1
O.REPLAY	FPT_RPL.1, FIA_SOS.2	Section 4.3.1
O.TAMPER	FPT_PHP.2, FPT_PHP.3	Section 4.3.1

Table 8 Security Objectives and SFRs - Coverage

Security Functional Requirements	Security Objectives
FIA_UAU.1/Distant_server	O.AUTH
FIA_UAU.1/Load_device	O.AUTH
FIA_UAU.1/Purseholder	O.PURSE_HOLDER_AUTH
FIA_UAU.1/SAM	O.AUTH
FIA_UAU.6/Distant_server	O.AUTH
FIA_UAU.6/Load_device	O.AUTH
FIA_UAU.6/Purseholder	O.PURSE_HOLDER_AUTH
FIA_UAU.6/SAM	O.AUTH
FIA_UAU.3	O.AUTH
FIA_UAU.4	O.AUTH
FDP_SDI.1	O.INTEG_DATA
FCO_NRO.2	O.EM
FPT_ITC.1	O.CONF_DATA
FPT_ITI.1	O.INTEG_DATA
FPT_RPL.1	O.EM, O.REPLAY
FDP_ACC.2	O.CONF_DATA, O.INTEG_DATA
FDP_ACF.1	O.CONF_DATA, O.INTEG_DATA
FMT_MSA.1/Assets	O.CONF_DATA, O.INTEG_DATA
FMT_MSA.3/Assets	O.CONF_DATA, O.INTEG_DATA
FDP_IFC.1	O.AGGREGATE, O.BREAK, O.EM, O.LIMIT
FDP_IFF.1	O.AGGREGATE, O.BREAK, O.EM, O.LIMIT
FDP_ITC.1	O.AGGREGATE, O.BREAK, O.EM, O.LIMIT
FMT_MSA.1/Transaction	O.AGGREGATE, O.BREAK, O.EM, O.LIMIT
FMT_MSA.3/Transaction	O.AGGREGATE, O.BREAK, O.EM, O.LIMIT
FAU_GEN.1	O.RECORD
FAU_SAR.1	O.RECORD
FAU_STG.1	O.INTEG_DATA
FPT_RCV.4	O.CONF_DATA, O.EM, O.INTEG_DATA, O.OPERATE
FCS_COP.1	O.EM
FIA_SOS.2	O.AUTH, O.REPLAY
FPT_PHP.2	O.CONF_DATA, O.INTEG_DATA, O.TAMPER

Security Functional Requirements	Security Objectives
FPT_PHP.3	O.CONF_DATA, O.INTEG_DATA, O.TAMPER

Table 9 SFRs and Security Objectives

4.3.3 Dependencies

4.3.3.1 SFRs dependencies

Requirements	CC Dependencies	Satisfied Dependencies
FIA_UAU.1/Distant_server	(FIA_UID.1)	
FIA_UAU.1/Load_device	(FIA_UID.1)	
FIA_UAU.1/Purseholder	(FIA_UID.1)	
FIA_UAU.1/SAM	(FIA_UID.1)	
FIA_UAU.6/Distant_server	No dependencies	
FIA_UAU.6/Load_device	No dependencies	
FIA_UAU.6/Purseholder	No dependencies	
FIA_UAU.6/SAM	No dependencies	
FIA_UAU.3	No dependencies	
FIA_UAU.4	No dependencies	
FDP_SDI.1	No dependencies	
FCO_NRO.2	(FIA_UID.1)	
FPT_ITC.1	No dependencies	
FPT_ITI.1	No dependencies	
FPT_RPL.1	No dependencies	
FDP_ACC.2	(FDP_ACF.1)	FDP_ACF.1
FDP_ACF.1	(FDP_ACC.1) and (FMT_MSA.3)	FDP_ACC.2, FMT_MSA.3/Assets
FMT_MSA.1/Assets	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_ACC.2
FMT_MSA.3/Assets	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/Assets
FDP_IFC.1	(FDP_IFF.1)	FDP_IFF.1
FDP_IFF.1	(FDP_IFC.1) and (FMT_MSA.3)	FDP_IFC.1, FMT_MSA.3/Transaction
FDP_ITC.1	(FDP_ACC.1 or FDP_IFC.1) and (FMT_MSA.3)	FDP_IFC.1, FMT_MSA.3/Transaction
FMT_MSA.1/Transaction	(FDP_ACC.1 or FDP_IFC.1) and (FMT_SMF.1) and (FMT_SMR.1)	FDP_IFC.1
FMT_MSA.3/Transaction	(FMT_MSA.1) and (FMT_SMR.1)	FMT_MSA.1/Transaction
FAU_GEN.1	(FPT_STM.1)	
FAU_SAR.1	(FAU_GEN.1)	FAU_GEN.1

Requirements	CC Dependencies	Satisfied Dependencies
FAU_STG.1	(FAU_GEN.1)	FAU_GEN.1
FPT_RCV.4	No dependencies	
FCS_COP.1	(FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2) and (FCS_CKM.4)	
FIA_SOS.2	No dependencies	
FPT_PHP.2	(FMT_MOF.1)	
FPT_PHP.3	No dependencies	

Table 10 SFRs dependencies

Rationale for the exclusion of dependencies

The dependency FIA_UID.1 of FIA_UAU.1/Distant_server is unsupported. The dependency with FIA_UID.1 is not relevant: there is no identification to the TOE. It is always the TOE which identifies itself to other devices.

The dependency FIA_UID.1 of FIA_UAU.1/Load_device is unsupported. The dependency with FIA_UID.1 is not relevant: there is no identification to the TOE. It is always the TOE which identifies itself to other devices.

The dependency FIA_UID.1 of FIA_UAU.1/Purseholder is unsupported. The dependency with FIA_UID.1 is not relevant for the case of the EP. Indeed, the identification of the Purse holder is implicit. The identification function itself is done by the presentation of the physical device.

The dependency FIA_UID.1 of FIA_UAU.1/SAM is unsupported. The dependency with FIA_UID.1 is not relevant: there is no identification to the TOE. It is always the TOE which identifies itself to other devices.

The dependency FIA_UID.1 of FCO_NRO.2 is unsupported. The dependency with FIA_UID.1 is not relevant. Indeed the identification of the Purse holder is implicit. The identification function itself is done by the presentation of the physical device.

The dependency FMT_SMF.1 of FMT_MSA.1/Assets is unsupported. This PP does not mandate any specific security management function. The ST author shall add FMT_SMF if necessary (for instance, if the standard CC operations for the management of security attributes are not enough or appropriate).

The dependency FMT_SMR.1 of FMT_MSA.1/Assets is unsupported. This PP does not mandate any specific security role. The ST author shall add FMT_SMR if necessary. Otherwise the autorised identified role in FMT_MSA.1.1/Assets shall be filled in with "none".

The dependency FMT_SMR.1 of FMT_MSA.3/Assets is unsupported. This PP does not mandate any specific security role. The ST author shall add FMT_SMR if necessary. Otherwise FMT_MSA.3.2 shall be filled in with "none".

The dependency FMT_SMF.1 of FMT_MSA.1/Transaction is unsupported. This PP does not mandate any specific security management function. The ST author shall add FMT_SMF if necessary (for instance, if the standard CC operations for the management of security attributes are not enough or appropriate).

The dependency FMT_SMR.1 of FMT_MSA.1/Transaction is unsupported. This PP does not mandate any specific security role. The ST author shall add FMT_SMR if necessary. Otherwise the "autorised identified role" in FMT_MSA.1.1/Transaction shall be filled in with "none".

The dependency FMT_SMR.1 of FMT_MSA.3/Transaction is unsupported. This PP does not mandate any specific security role. The ST author shall add FMT_SMR if necessary. Otherwise FMT_MSA.3.2 shall be filled in with "none".

The dependency FPT_STM.1 of FAU_GEN.1 is unsupported. The dependency with FPT_STM.1 is not relevant to the TOE: correctness of time is of no use for the TOE objectives.

The dependency FCS_CKM.1 or FDP_ITC.1 or FDP_ITC.2 of FCS_COP.1 is unsupported. The dependency with FCS_CKM.1 is not satisfied because the TOE does not generate keys. The dependency with FDP_ITC.1 or FDP_ITC.2 is not satisfied because this PP does not mandate any specific key importation method. The ST author is allowed to extend the scope of the TOE and add requirements on these issues. Nevertheless, keys are particular TSF data that shall be protected against disclosure and modification, as specified in FPT_ITC.1 and FPT_ITI.1.

The dependency FCS_CKM.4 of FCS_COP.1 is unsupported. The dependency with FCS_CKM.4 "Cryptographic key destruction" is discarded since key destruction is out of the scope of the TOE.

The dependency FMT_MOF.1 of FPT_PHP.2 is unsupported. The dependency with FMT_MOF.1 is not relevant: during operational use of the TOE, the behaviour of security functions could not be changed.

4.3.3.2 SARs dependencies

Requirements	CC Dependencies	Satisfied Dependencies
ADV_ARC.1	(ADV_FSP.1) and (ADV_TDS.1)	ADV_FSP.4, ADV_TDS.3
ADV_FSP.4	(ADV_TDS.1)	ADV_TDS.3
ADV_IMP.1	(ADV_TDS.3) and (ALC_TAT.1)	ADV_TDS.3, ALC_TAT.1
ADV_TDS.3	(ADV_FSP.4)	ADV_FSP.4
AGD_OPE.1	(ADV_FSP.1)	ADV_FSP.4
AGD_PRE.1	No dependencies	
ALC_CMC.4	(ALC_CMS.1) and (ALC_DVS.1) and (ALC_LCD.1)	ALC_CMS.4, ALC_DVS.2, ALC_LCD.1
ALC_CMS.4	No dependencies	
ALC_DEL.1	No dependencies	
ALC_DVS.2	No dependencies	
ALC_LCD.1	No dependencies	
ALC_TAT.1	(ADV_IMP.1)	ADV_IMP.1
ASE_CCL.1	(ASE_ECD.1) and (ASE_INT.1) and (ASE_REQ.1)	ASE_ECD.1, ASE_INT.1, ASE_REQ.2
ASE_ECD.1	No dependencies	

Requirements	CC Dependencies	Satisfied Dependencies
ASE_INT.1	No dependencies	
ASE_OBJ.2	(ASE_SPD.1)	ASE_SPD.1
ASE_REQ.2	(ASE_ECD.1) and (ASE_OBJ.2)	ASE_ECD.1, ASE_OBJ.2
ASE_SPD.1	No dependencies	
ASE_TSS.1	(ADV_FSP.1) and (ASE_INT.1) and (ASE_REQ.1)	ADV_FSP.4, ASE_INT.1, ASE_REQ.2
ATE_COV.2	(ADV_FSP.2) and (ATE_FUN.1)	ADV_FSP.4, ATE_FUN.1
ATE_DPT.1	(ADV_ARC.1) and (ADV_TDS.2) and (ATE_FUN.1)	ADV_ARC.1, ADV_TDS.3, ATE_FUN.1
ATE_FUN.1	(ATE_COV.1)	ATE_COV.2
ATE_IND.2	(ADV_FSP.2) and (AGD_OPE.1) and (AGD_PRE.1) and (ATE_COV.1) and (ATE_FUN.1)	ADV_FSP.4, AGD_OPE.1, AGD_PRE.1, ATE_COV.2, ATE_FUN.1
AVA_VAN.5	(ADV_ARC.1) and (ADV_FSP.4) and (ADV_IMP.1) and (ADV_TDS.3) and (AGD_OPE.1) and (AGD_PRE.1) and (ATE_DPT.1)	ADV_ARC.1, ADV_FSP.4, ADV_IMP.1, ADV_TDS.3, AGD_OPE.1, AGD_PRE.1, ATE_DPT.1

Table 11 SARs dependencies

4.3.4 Rationale for the Security Assurance Requirements

EAL4 is required for this type of TOE and product since they are meant to resist against sophisticated attacks. This evaluation assurance level allows a developer to gain maximum assurance from positive security engineering based on good practices. EAL4 represents the highest practical level of assurance expected for a commercial grade product. In order to provide a meaningful level of assurance that the TOE and its embedding product provide an adequate level of defense against such attacks, the evaluators must have access to the low level design and source code. The lowest for which such access is required is EAL4.

4.3.5 ALC_DVS.2 Sufficiency of security measures

Development security is concerned with physical, procedural, personnel and other technical measures that may be used in the development environment to protect the TOE. This assurance component is a higher hierarchical component to EAL4 (only ALC_DVS.1 is included in EAL4). Due to the nature of the TOE, there is a need to justify the sufficiency of these procedures to protect their confidentiality and integrity. ALC_DVS.2 has no dependencies.

4.3.6 AVA_VAN.5 Advanced methodical vulnerability analysis

The TOE must resist to high attack potential. This is due to the fact that the Electronic Purse can be placed in hostile environments, including electronic laboratories. This robustness level is achieved by the assurance requirement AVA_VAN.5. Independent vulnerability analysis is based on highly detailed technical information. The attacker is assumed to be thoroughly familiar with the specific implementation of the TOE. The attacker is presumed to have a high level of technical knowledge. AVA_VAN.5 has dependencies with ADV_ARC.1, ADV_FSP.4, ADV_TDS.3, ADV_IMP.1, AGD_PRE.1, AGD_OPE.1 and ATE_DPT.1. All these dependencies are satisfied by EAL4.

5 Notice

This document has been generated with TL SET version 2.3.6 (for CC3). For more information about the security editor tool of Trusted Labs visit our website at www.trusted-labs.com.

Annexe A Definitions and acronyms

A.1 Definitions

This section provides definitions about terms frequently used in this document. The definition of the Common Criteria related terms is specified in [CC1], § 4.

Acquirer device	An acquirer device is a server intended to handle the collect transactions.
Non Banking Application	Non Banking Application (NBA) is a type of "fidelity" application, which allows the storage and playback of information related to a Purse holder. The type of information stored is freely chosen by the application issuer (merchant, business, etc.) such as reductions, purchase order, coupons, Fidelity points, etc…
Balance	The balance is the amount of EM stored in the EP (in a specific currency). It is increased by load and quickload transactions, and decreased by debit operation: - BAL: The balance of the EP, loaded by load transactions. - BAL': The balance of the EP, loaded by quickload transactions.
BMS	The Billettique Monétique Services company (BMS) groups together the leading: - banking and financial establishments marketing Moneo such as BNP-Paribas, Banque Populaire, Caisse d'Épargne, Crédit Agricole, CIC, HSBC, LCL, Crédit Mutuel, La Banque Postale, Société Générale, - Carriers such as SNCF or RATP - Operators such as France Telecom. BMS is responsible for the design, marketing development and operation of the electronic purse on a multi-application smart card, capitalizing on the technological performance and achievements of French monetics and intermodal tele-cash dispensing.
Collect	One or several amounts of the EM corresponding to a set of payment transactions stored by a SAM are delivered to the EM issuer via acquirer device.
Debit (Payment)	The EP is debited from an amount of EM while the SAM is credited with the same amount of EM. The purse holder receives goods or services in turn. The EM payment transaction is presented as a debit operation when it's only related to the TOE.

Acquirer device	An acquirer device is a server intended to handle the collect transactions.
Electronic Money	Electronic Money is the counter part of the fund received by the EM issuer and stored in the EP or the SAM. It is defined by the balances stored in the Electronic Purse.
EM issuer	The EM issuer guarantees the EM in an EM system. To this end, the EM issuer creates and dispenses EM in exchange for funds received, redeems collected EM and extinguishes it.
Electronic Purse	An EP is an application executed by an OS embedded into an IC. Its functionalities are similar to traditional purse functionalities with the distinction that it uses Electronic Money (EM) instead of cash money. An EP is used to facilitate payments of low value. The fully operational EP contains various parameters that could be updated.
Integrated Circuit	Integrated Circuit is an electronic component designed to perform processing and/or memory functions.
Load Device	A load device is a server. Its main functionality is to load the EM in the EP in exchange of funds via the EM issuer.
Load	The EP is credited with an amount of EM created by the EM issuer, via a load agent; the purse holder gives a corresponding amount of funds in turn.
Merchant	A merchant sells goods or services for which he accepts payment by EP. In order to handle the EM payment transactions, the merchant operates one or more purchase devices in which a SAM stores EM until collect. The merchant is responsible for the operational security of the purchase device he controls.
Purchase Device	A purchase device is a physical device installed at the merchant or a server used to accept payment from an EP in an EM payment transaction. It includes a Secure Access Module (SAM), built on an integrated circuit module. The SAM shall provide the necessary security for the EM payment, the quickload and the collect transactions. It contains various parameters that could be updated.
Purse holder	The purse holder is the person that is in possession of the EP and uses it for EM payment transactions. Purse holders need to protect their EP as if it is cash.

Acquirer device	An acquirer device is a server intended to handle the collect transactions.
Quickload	This operation is used to load EM into the EP. It is processed offline. The SAM processes this operation. It may be performed when the purchase amount is greater than the balance of the purse in order to go on with the EM payment.
SAM	Secure Access Module, typically in an ID-0 plug-in size form factor, placed in a Purchase Device. The SAM identifies the EP, checks the authenticity of the EP, encrypts and protects the exchanges with the EP and the Acquirer Device. It acts as an EM payment transactions certifier.
SFPMEI	Société Financière du Porte-Monnaie Électronique Interbancaire is the EM institution in France. It is approved by the authorities whose objective is to insure consumers of the deposits value made on an electronic purse, regardless of the companies responsible for its operation. The SFPMEI is an EM issuer.

A.2 Acronyms

CC	Common Criteria
ANSSI	Agence Nationale de la Sécurité des Systèmes d'Information
EAL	Evaluation Assurance Level
EM	Electronic Money
EP	Electronic Purse
IC	Integrated Circuit
NBA	Non Banking Application
OS	Operating System
OSP	Organisational Security Policy
PP	Protection Profile
SAM	Secure Access Module
SF	Security Function
TOE	Target Of Evaluation

Annexe B References

[CC1]	Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model. Version 3.1, revision 3, July 2009. CCMB-2009-07-001.
[CC2]	Common Criteria for Information Technology Security Evaluation, Part 2: Security functional requirements. Version 3.1, revision 3, July 2009. CCMB-2009-07-002.
[CC3]	Common Criteria for Information Technology Security Evaluation, Part 3: Security Assurance Requirements. Version 3.1, revision 3, July 2009. CCMB-2009-07-003.
[CEM]	Common Methodology for Information Technology Security Evaluation, Evaluation Methodology. Version 3.1, revision 3, July 2009. CCMB-2009-07-004.
[CRYPTO]	Mécanismes cryptographiques. Règles et recommandations concernant le choix et le dimensionnement des mécanismes cryptographiques. Version 1.11, 24 Octobre 2008, ANSSI. (This version or later applicable one.) Gestion des clés cryptographiques. Règles et recommandations concernant la gestion des clés utilisées dans des mécanismes cryptographiques. Version 1.10, 24 Octobre 2008. ANSSI. (This version or later applicable one.)
[M-CRYPTO]	"Electronic Purse Moneo - Mécanismes cryptographiques", version 2.4 (or later applicable one), SFPMEI.
[ANSSI-CC-PP-2009/02]	Embedded Software for Smart Secure Devices Protection Profile, Basic and Extended configurations, v1.0, 27 November 2009, ANSSI.
[PP-BSI-0035-2007]	Security IC Platform Protection Profile, version 1.0, June 2007. BSI.
[PP-JCS-Collection]	JavaCard System Minimal Configuration Protection Profile version 1.0b, Sun Microsystems, Inc. PP/0303
	JavaCard System Standard 2.1.1 Configuration Protection Profilenversion 1.0b, Sun Microsystems, Inc. PP/0304
	JavaCard System Standard 2.2 Configuration Protection Profile version 1.0b, Sun Microsystems, Inc. PP/0305
	JavaCard System Defensive Configuration Protection Profile version 1.0b, Sun Microsystems, Inc. PP/0306
[PP-JCS-Closed]	Java Card System - Closed configuration Protection Profile, Sun Microsystems, Inc. *Under evaluation at the publication date of this PP.*
[PP-JCS-Open]	Java Card System - Open configuration Protection Profile, Sun Microsystems, Inc. *Under evaluation at the publication date of*

	this PP.
[PP-0101]	Intersector Electronic Purse and Purchase Device (version without Last Purchase Cancellation), version 1.3, March 2001, SFPMEI.
[PP SAM]	SAM for EM system Protection Profile, version 1.5, 4 February 2010, SFPMEI.

Index