

PED SW development guideline

Version: 1.2.0

CONTENTS:

VERSION HISTORY 2

1 SCOPE..... 3

2 SECURITY ADMINISTRATOR..... 3

3 SOFTWARE DEVELOPMENT 3

4 SOFTWARE REVIEW..... 3

5 SOFTWARE BINDING..... 4

 5.1 PURPOSE..... 4

 5.2 GENERAL..... 4

 5.3 A RECOMMENDED PROCEDURE 5

REFERENCES 6

GLOSSARY 6

Version history

Date	Version	Description	Issued/revised by
2003-09-25	1.0.0	Adapted to EMVTERMINAL style. Some important SW issues added. Impact of Visa PED Evaluation.	Hans Olof Andersson
2003-10-02	1.1.0	Approved.	SÄKO-I
2005-12-12	1.2.0	New definition of SA. New references. Reviewer used in place of auditor.	Hans Olof Andersson

1 Scope

This document defines the process of software development and binding and gives some guidelines regarding how to implement procedures for the final binding and freezing of a software version.

The guidelines given here are mainly aiming to be valid for security applications which are dealing with PIN and cryptographic keys, thus typically the security software of a PIN entry device, PED. They may, however, be applied to all situations where strict control and management of software versions and load entities are required.

What is said here about load entities shall be applicable to situations where software is loaded as one single module as well as when software may be loaded in several parts.

2 Security administrator

One important role in the process of issuing new PED (or secure device) software is the security administrator, SA, appointed by an acquiring bank. The SA will be responsible for the acceptance of the PED software. Thus the SA has to make sure that a new PED software release has been properly reviewed and frozen.

The procedures proposed below (or similar procedures) may, partly or entirely, be delegated to the terminal vendor, TV, or another trusted third party.

3 Software development

The software (firmware) of a secure device has to be developed with a special concern. It is obvious that some mistakes in software development may jeopardise the security, regardless of the hardware security of the device. The good manners for security software development come with experience. Thus the need for software review may depend on the skills of the development organisation.

Here are a few important aspects:

- storage and use of keys or other secret values (e.g. PIN) must be controlled in such a way that a key never may be exposed intentionally or unintentionally,
- if there is an application besides the bank security application, there must be full separation between those two,
- quality assurance processes have to be implemented to minimise the risk of a developer leaving, intentionally or unintentionally, a malicious function in the software

4 Software review

Software review means a procedure in which an SA (or its representative), based on some relevant documentation, approves the release of a software version. Some general guidelines for this are given in ref[sacaudit]. A certification procedure may be anything from the SA accepting the terminal vendor's documented claim that a new version has to be issued, up to a review of the complete source code as a part of the procedure. The purpose of the software review process is to grant that only reviewed and thus certified software versions are taken into operation, and that they also are fully revisable at some later moment.

Please note that the PCI PED Evaluation, ref[pcipedreq], is applicable to every new version of a PED even if it is caused by only a minor change in the PED software.

5 Software binding

5.1 Purpose

The overall purpose of the software binding is to create an unchangeable load entity and a corresponding reference copy of all modules and other items constituting a software version. The total process shall grant that only frozen software versions are put into operation and that there is a revisable archive of each such software version.

By software binding we thus mean a process where a reviewer is able to certify that the source code used to build a load entity also was copied to a reference library. Also, it shall be possible for a reviewer, to grant the integrity of the load entity. For this purpose a digital signing procedure shall be used.

5.2 General

We assume the following environment:

- There is a target system for the software, typically a PIN Entry Device, PED. This PED must have a digital signature checking capability as a part of the software loading process.
- There is a development environment where it is possible to create the source code of a software project.
- There is a build environment where it is possible to compile and link this software to build a load entity.
- There is a signing environment where it is possible to create a digital signature (certificate) for the load entity.

All these aspects have to be considered by the vendor of a PED (or any other equipment for which this document is applicable). Tools and routines shall be possible to follow and supervise by a reviewer (normally appointed by the security administrator).

Fig. 1 shows the principle of a typical binding process. In fig. 1 it is assumed that a PKA based method is used for the signing. Note that the tasks of the SA, as defined in fig. 1 may be delegated to the TV or to another representative.

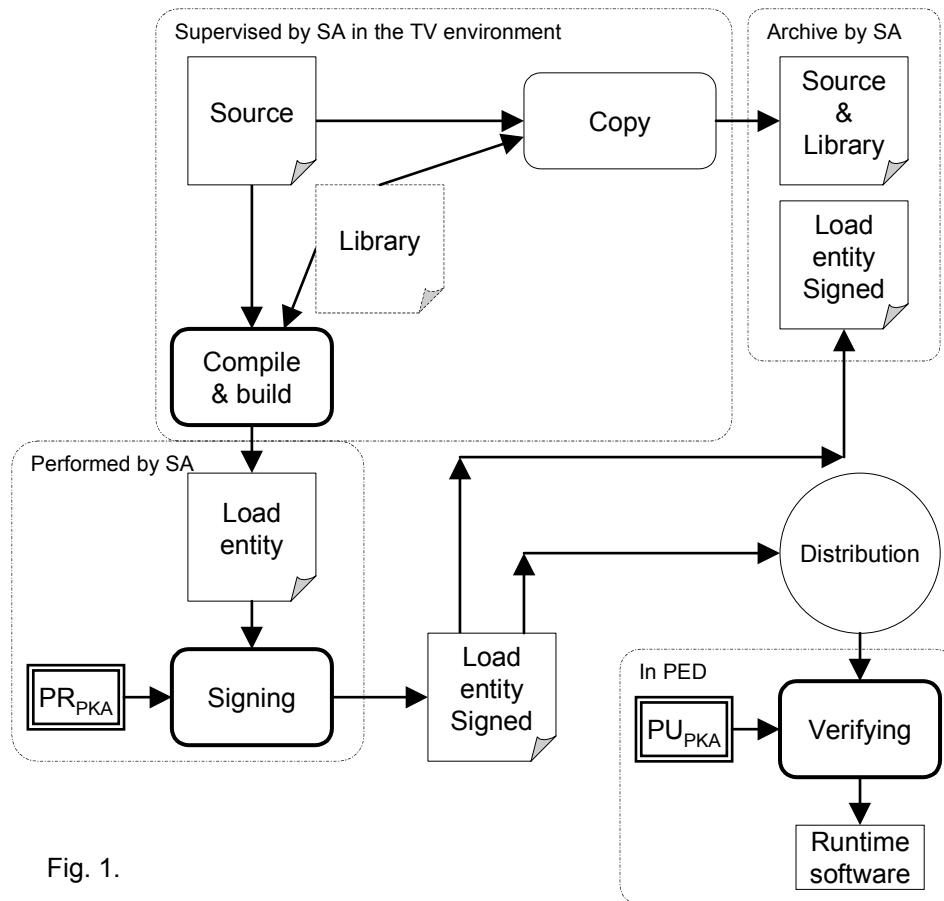


Fig. 1.

The binding (sometimes also called freezing) is the total process leading to a situation where the controlling party (reviewer) has in its possession a copy of all files involved in the binding process and as a minimum the source code and the signed load entity.

In order for the binding to have a value from a revision point of view it is important that the load entity can be verified by the device during the load operation, and that the cryptographic key used for this can be controlled by the reviewer.

This puts up a few requirements that the PED vendor, TV, will have to fulfil regarding the procedures and equipment used for the development and finalising of a software load entity.

- If it is the first release for a new product a PED approval shall be requested. Such a request is put forward to PNC SAC as defined in ref[sacaudit] and will be processed according to that document.
- If it is a new release of an already approved product it is the responsibility of the signing party to request an approval from PNC SAC. If there already is an approval as defined by ref[pcipedreq] this shall be reported and will be considered in the PNC SAC approval process.
- For the approval process the TV shall, on request, hand over to the reviewer source code and other documents necessary to make a software review possible.
- All procedures involved in a software binding session, shall be well documented and revisable.
- The representative of the TV shall be able to explain each step taken during the binding session.
- Supporting software used shall be easy to use and have clear and simple user interfaces.
- Equipment used shall be installed in such a way that a reviewer can be convinced that the appointed equipment is the only equipment involved in the binding process.
- A digital signing procedure shall be available. In this, it shall be possible for the reviewer to generate or supply a signature key of its own choice.

5.3 A recommended procedure

The actual implementation of the binding process will be considered during the (initial) product

approval process. Still it is the SA that has the final decision regarding the suitability of the actual implementation of the binding process. The SA may adopt these guidelines to its experience or needs, which may mean adding rules to suit the operations of the SA in question.

The binding procedure shall take place in a room

- where practically all steps involved in the binding process can take place,
- which is equipped for this purpose,
- where no other activity shall be going on during the specific binding session.

The equipment involved shall

- consist of as few units as possible, if possible all steps of a binding session shall be performed on a single unit,
- be as much "standard" as possible, the reviewer shall be able to feel familiar with (recognise) the equipment used.

The 2 clauses above are applicable to hardware as well as software.

The software environment shall be configured so that

- it is easy to understand which support programs that are used and what they do,
- file paths shall be defined openly and in a manner that is easy to understand and follow.

The signing process shall

- preferably use PKA methods,
- allow for the SA (or reviewer) to generate or supply its own signing key.

The loading process shall

- verify the signature of the load entity and only allow the software to become operational when the signature verifies OK,
- prevent "resets" that may make it possible for the loaded device to verify load entities not signed by the proper SA.

The SA is responsible that the archive is maintained for the time deemed necessary, as a suggestion 1 to 3 years after a software version were taken out of operation. This archive process shall with a reasonable certainty grant that a software version may be revised in detail at a later moment, e.g. when a situation of doubt regarding software "quality" has occurred.

In a situation when archived information is taken out for revision, it is important to follow procedures that fulfil not only revision aspects but also the possible legal aspects of such a revision.

References

[pcipedreq]	PCI: "POS PED Security Requirements Manual".
[sacaudit]	PNC SAC: "Security audit guideline".
[sacreq]	PNC SAC: "Security Requirements for an EFTPOS Terminal".

Glossary

archive	A space (and corresponding procedures) where printed or recorded material can be stored in a safe way.
binding	A process establishing a link between the source code and the load entity of a software project.
certification	A process to verify a software version and approve that it is taken into operations.
EFTPOS	Electronic Funds Transfer at the Point of Sale.
load entity	The loadable (and executable) representation of a software version.
PED	PIN Entry Device. General for a unit used to enter a card holder's PIN.
PIN	Personal Identification Number.
PKA	Public Key Algorithm. General for a public key algorithm, practically RSA.
PNC SAC	Security committee of the Swedish card issuers branch organisations.

PR _{PKA}	General for a private PKA-key.
PU _{PKA}	General for a public PKA-key.
review	A process where something is looked at in order to verify its suitability for a certain situation.
reviewer	One or more persons appointed by the SA to validate the binding process.
revision	For this document a more thorough review of something.
RSA	Rivest, Shamir, Adleman. The most common public key algorithm.
SA	Security administrator. Responsible for security aspects on the use of PIN for bank cards. Normally appointed by the acquiring bank. The role of the SA is defined in ref[säkoreq].
signature	For the purpose of this guideline, a digital check sum created using cryptographic methods, with the purpose to grant the authenticity of an entity.
TV	Terminal Vendor. Someone supplying a device to which these guidelines apply.