Profile Editor – User Manual

| | |
|---|---|
| Document Title: | Profile Editor Tool - User Guide |
| Document Purpose: | Technical Documentation for the Profile Editor Tool |
| Emission Date: | 23/01/2014 |
| Version: | 4.0.20142301 |
| Author: | |
| Page Count: | 66 |

# Contents

# List of Figures

# List of Tables

# 1 Introduction & Scope

The document presents the overview and user-guide for the Profile Editor Tool. This document should be delivered with the demonstration version of the tool. The document will cover all of the technical details in the tool.

# 2 Tool Requirements

This section will cover the tool requirements, and assist in helping the user understand the prerequisites that we will need before launching the tool.

## 2.1 Java

The tool will require Java 1.6 or above to run. If you're currently not running any version of Java, it can be downloaded from `http://java.com/en/download/`. Otherwise make sure you update Java to the correct version. It is advisable that the tool is run in Java 6.

# 3 Launching the Tool

## 3.1 Windows

To launch the tool, double-click on the *ProfileEditor.exe* application, as seen in Figure 1. No installation is required.

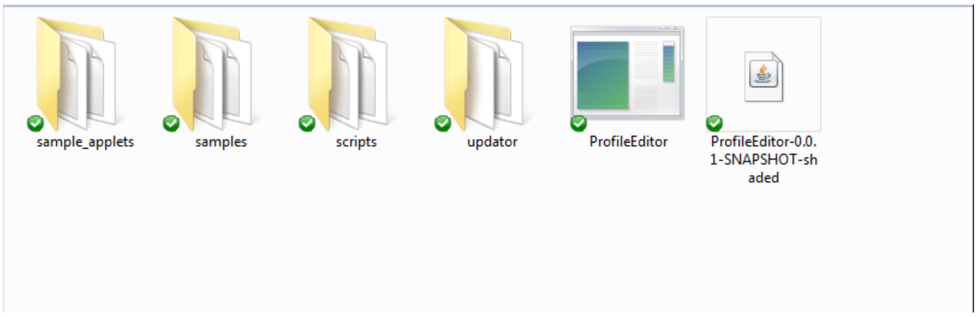


Figure 1: Windows Example Package

## 3.2 Linux

To launch the tool, run the jar file seen in Figure 2. No installation is required.

To run the jar file, the user will have to navigate to the terminal. Once inside the terminal, the user must navigate to the folder where the jar file is held. Next, the user must run the jar file. For the example shown in Figure 2, the command to run the jar file would be as follows:

*java -jar 'Home/Downloads/ProfileEditor20121030/ProfileEditor0.0.1-SNAPSHOT-shaded.jar'*

## 3.3 Apple Mac

Some users may find they have problems with the tool after double-clicking the Profile Editor jar file. To fix this error, please follow the instructions shown in Linux section 3.2.

# 4 Setting Up Your Environment

Once you've loaded the Profile Editor tool you should be presented with the window in Figure 3. This will be your starting point for each personalisation.

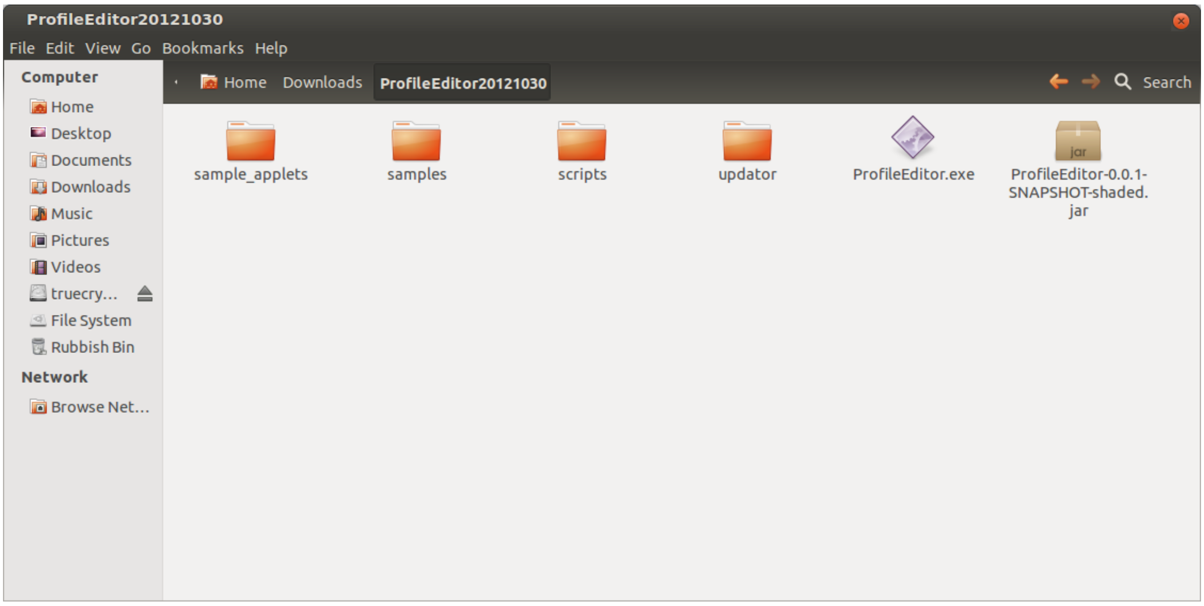


Figure 2: Linux Example Package

## 4.1 Select Card Reader

The card reader selector can be accessed by clicking on the *Select Card Reader* menu in the top menu.

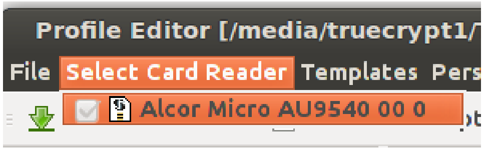


Figure 3: List of Available Card Readers

This menu will contain all the card readers installed on your PC. To select the reader of your choice just click on its name.

Obviously, you cannot send data to a card reader that doesn't have a card in it. In order to aid the process of selecting viable card readers an image is displayed showing whether or not a card is currently inside that card reader.

If you start the tool with 2 or more card readers connected then you will have to select the card reader you wish to use. However, if you start the tool with only 1 card reader connected then the tool will automatically select this reader for you.

# 5 Modules

In this section we'll explain all of the different functionality that each module contains. Each module handles different parts of the personalisation process.

Here is the list of all of the modules that the Profile Editor supports:

- Pre-Personalisation,
- File System Editor,
- Network Configuration,
- Simba Security Configuration,
- WIB Loading,
- S@T Loading,
- Applet Loading,
- SimbaTalk Applet Loading,
- Create Perso Image,
- Dynamic Data,
- Post-Perso.

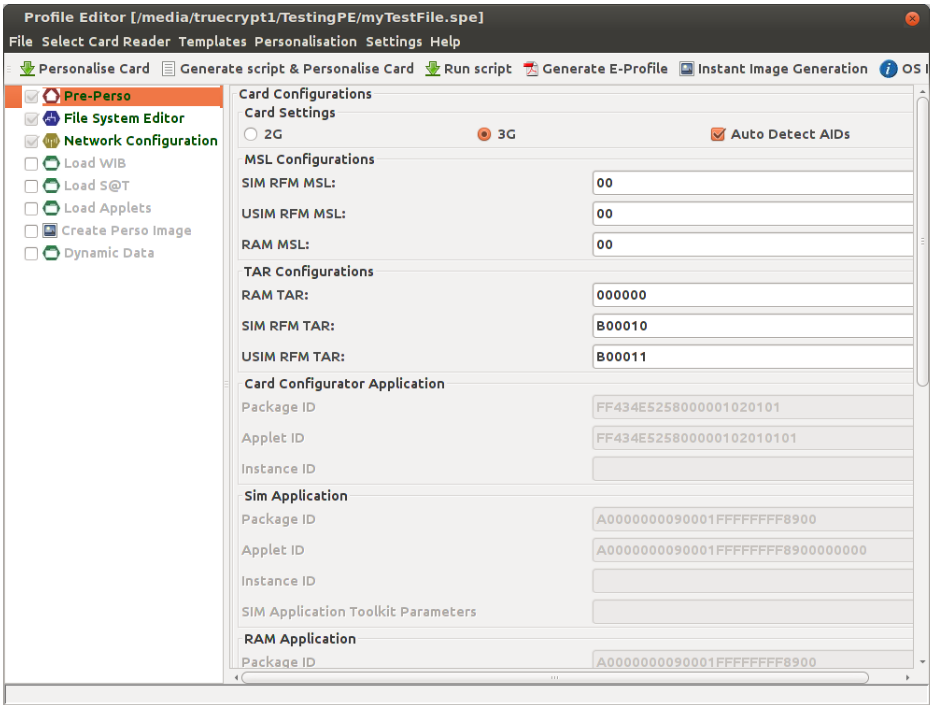
Each of these modules will be explained in more detail under their own sub-section.

Most of the modules can be run separately from one another. However, there are four modules which are linked for personalisation, these modules are Pre-Personalisation, File System Editor, Network Configuration, and Simba Security Configuration. These modules will be turned on/off when just one of the modules is selected/deselected. The modules can be turned on and off depending on the users requirements. This is done by ticking the required modules, found in the tree on the left hand side of Figure 4. Ticked module will be included during personalisation.

## 5.1 Pre-Personalisation

To access the Pre-Personalisation module, first click on the *Pre-Perso* item in the tree that's located at the left hand-side of the tool. This will display the window in Figure 4. The user should select either 2G or 3G within the *Card Settings* panel depending on which profile they are going to create. The user should also select the OS that they require. This menu can be seen in Figure 5. The current Operating Systems which are supported are:

- Default Java
- Simba Native



Profile Editor [/media/truecrypt1/TestingPE/myTestFile.spe]

Card Configurations

Card Settings

○ 2G ● 3G ☑ Auto Detect AIDs

MSL Configurations

SIM RFM MSL: 00

USIM RFM MSL: 00

RAM MSL: 00

TAR Configurations

RAM TAR: 000000

SIM RFM TAR: B00010

USIM RFM TAR: B00011

Card Configurator Application

Package ID: FF434E5258000001020101

Applet ID: FF434E525800000102010101

Instance ID:

Sim Application

Package ID: A0000000090001FFFFFFFF8900

Applet ID: A0000000090001FFFFFFFF8900000000

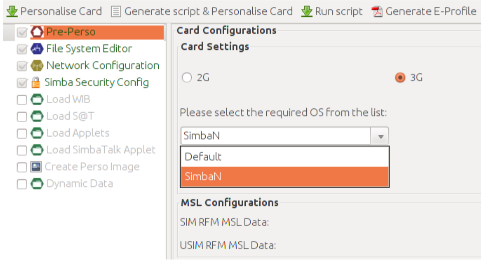
Instance ID:

SIM Application Toolkit Parameters:

RAM Application

Package ID: A0000000090001FFFFFFFF8900

Figure 4: Tool Landing and Pre-Perso View



Card Configurations

Card Settings

○ 2G ● 3G

Please select the required OS from the list:

SimbaN

Default

SimbaN

MSL Configurations

SIM RFM MSL Data:

USIM RFM MSL Data:

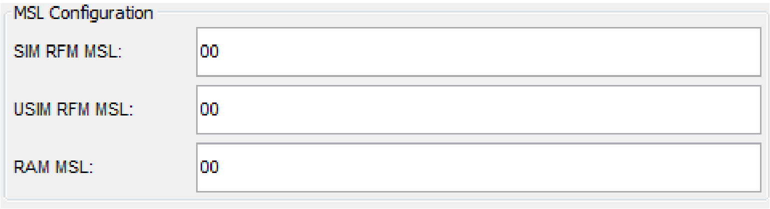
Figure 5: Select OS Menu

### 5.1.1 MSL Configuration

In this section you're able to set the Minimum Security Level (MSL) used by:

- SIM RFM Application,
- USIM RFM Application,

- And RAM MSL Application.



MSL Configuration

SIM RFM MSL: 00

USIM RFM MSL: 00

RAM MSL: 00

Figure 6: MSL Configuration Panel

#### 5.1.1.1 Configure the MSL

Figure 7 shows how to configure the MSL for the security level you require.

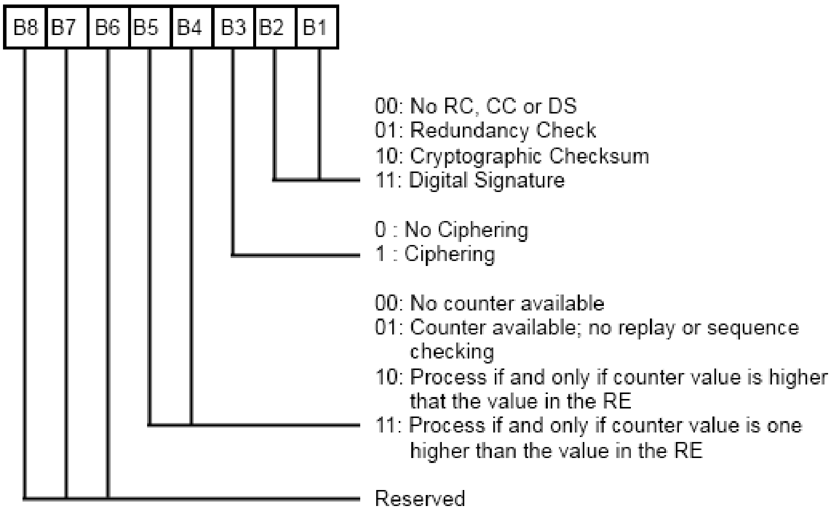

B8
B7
B6
B5
B4
B3
B2
B1
00: No RC, CC or DS
01: Redundancy Check
10: Cryptographic Checksum
11: Digital Signature
0 : No Ciphering
1 : Ciphering
00: No counter available
01: Counter available; no replay or sequence checking
10: Process if and only if counter value is higher that the value in the RE
11: Process if and only if counter value is one higher than the value in the RE
Reserved

Figure 7: How To Configure The MSL

#### 5.1.1.2 Reset MSL Values to Default

To reset the MSL values, right click on the *Pre-Perso* menu item. Clicking this item will allow you to select to reset the MSL values to either the old mask values, the new mask values, or the Simba mask values. By default, the values are set to the new mask values. The user can also reset just the MSL values by selecting the *Reset to Default MSL Values* option from the right click menu. This menu item can be seen in Figure 8.

### 5.1.2 AID Pre-Personalisation

This section allows you to edit the Application Identifiers (AIDs) for the following options:

- Card Configurator Application,
- SIM Application,
- RAM Application,
- USIM Application,
- SIM RFM Application,
- USIM RFM Application.

These values differ from one Operating System (OS) to another.

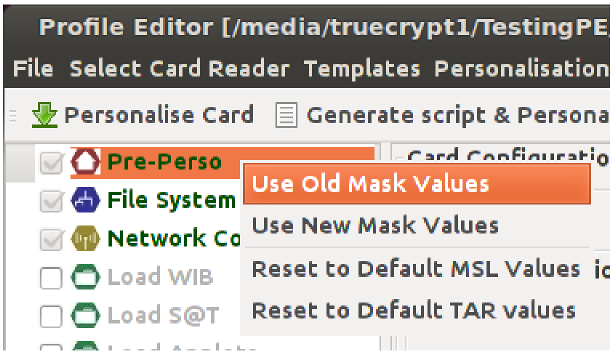

Profile Editor [/media/truecrypt1/TestingPE
File Select Card Reader Templates Personalisation
Personalise Card
Generate script & Persona
☑ Pre-Perso
☑ File System
☑ Network Co
□ Load WIB
□ Load S@T
Card Configuratio
Use Old Mask Values
Use New Mask Values
Reset to Default MSL Values
Reset to Default TAR values

Figure 8: Pre-Perso Right Click Options

#### 5.1.2.1 Reset AID Values to Default

To reset the AID values, right click on the *Pre-Perso* menu item. Clicking this item will allow you to select to reset the AID values to either the old mask values, the new mask values, or the Simba mask values. The user can also reset just the AID values by selecting the *Reset to Default AID Values* option from the right click menu. This menu item can be seen in Figure 8

### 5.1.3 AID Values for Old, New, and Simba Mask

The default AID values for the old, new, and Simba masks are shown in Table 1.

### 5.1.4 Auto-Detect AID Values

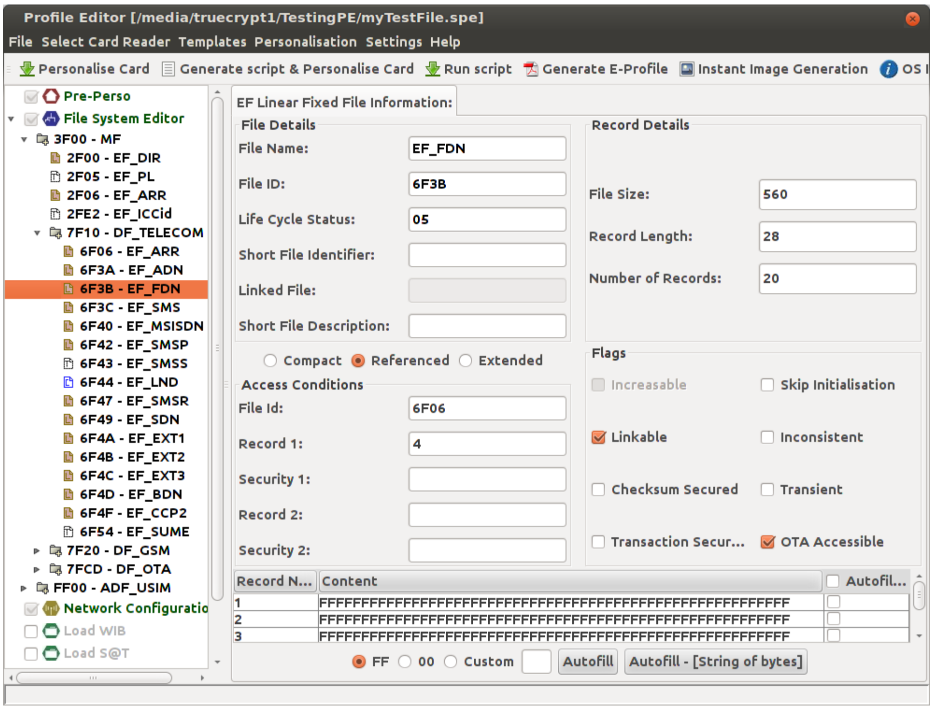
The user has the option to auto-detect the AID values which should be set for the specific card being used. This option is turned on by default. To turn this option off, simply toggle the 'Auto Detect AIDs' radio button at the top of the 'Pre-Perso' Module, as shown in Figure 4.

Table 1: AID Values for Old, New, and Simba Masks

| AID Values | Old Mask | New Mask | Simba Mask |
|---|---|---|---|
| Card Configurator Application | | | |
| Package ID | *FF434E5258000001020101* | *FF434E525810400401* | |
| Applet ID | *FF434E525800000102010101* | *FF434E52581040040100000000000000* | |
| Instance ID | | | |
| SIM Application | | | |
| Package ID | *A0000000090001FFFFFFFF8900* | *A0000000090001FFFFFFFF8900* | *53696D62614E40* |
| Applet ID | *A0000000090001FFFFFFFF8900000000* | *A0000000090001FFFFFFFF8900000000* | *53696D62614E40* |
| Instance ID | | *A0000000090001FFFFFFFF8900000000* | *53696D62614E2E53494D* |
| RAM Application | | | |
| Package ID | *FF434E52581040040202* | *A0000000090001FFFFFFFF8900* | |
| Applet ID | *FF434E52581040040202* | *A0000000090001FFFFFFFF89B00010* | |
| Instance ID | *FF434E525801010106010189000000* | *A0000000090001FFFFFFFF8900000001* | |
| Toolkit Parameters | | *02* | |
| USIM Application | | | |
| Package ID | *A0000000871002FF49FFFF8900* | *A0000000871002FF49FFFF8900* | *53696D62614E43* |
| Applet ID | *A0000000871002FF49FFFF89040B0000* | *A0000000871002FF49FFFF89040B0000* | *53696D62614E43* |
| Instance ID | | | |
| SIM RFM Application | | | |
| Package ID | | | 53696D62614E41 |
| Applet ID | *A0000000090001FFFFFFFF89B00010* | *A0000000090001FFFFFFFF89B00010* | *53696D62614E41* |
| Instance ID | *FF434E525801010105010189012345* | *A0000000090001FFFFFFFF89B0001001* | *53696D62614E2E52464D* |
| Toolkit Parameters | | *01* | |
| Install SIM RFM | *On* | *On* | *On* |
| USIM RFM Application | | | |
| Package ID | *FF434E52581040040203* | *FF434E52581040040203* | 53696D62614E41 |
| Applet ID | *FF434E52581040040203000000000000* | *FF434E52581040040203000000000000* | *53696D62614E41* |
| Instance ID | *FF434E525801010105010189B00010* | *FF434E525810400402030000B0001100* | *53696D62614E2E5552464D* |
| Toolkit Parameters | | *A0000000871002FF49FFFF89040B00FF* | |

## 5.2 File System Editor

To access the File System Editor module, first click on the *File System Editor* item in the tree that's located on the left hand-side of the tool. Figure 9 shows an example file system during development.


Profile Editor [/media/truecrypt1/TestingPE/myTestFile.spe]
File Select Card Reader Templates Personalisation Settings Help
Personalise Card
Generate script & Personalise Card
Run script
Generate E-Profile
Instant Image Generation
OS I

- Pre-Perso
- File System Editor
  - 3F00 - MF
    - 2F00 - EF_DIR
    - 2F05 - EF_PL
    - 2F06 - EF_ARR
    - 2FE2 - EF_ICCid
    - 7F10 - DF_TELECOM
      - 6F06 - EF_ARR
      - 6F3A - EF_ADN
      - 6F3B - EF_FDN
      - 6F3C - EF_SMS
      - 6F40 - EF_MSISDN
      - 6F42 - EF_SMSP
      - 6F43 - EF_SMSS
      - 6F44 - EF_LND
      - 6F47 - EF_SMSR
      - 6F49 - EF_SDN
      - 6F4A - EF_EXT1
      - 6F4B - EF_EXT2
      - 6F4C - EF_EXT3
      - 6F4D - EF_BDN
      - 6F4F - EF_CCP2
      - 6F54 - EF_SUME
    - 7F20 - DF_GSM
    - 7FCD - DF_OTA
  - FF00 - ADF_USIM
- Network Configuratio
- □ Load WIB
- □ Load S@T

EF Linear Fixed File Information:

File Details
File Name: EF_FDN
File ID: 6F3B
Life Cycle Status: 05
Short File Identifier:
Linked File:
Short File Description:

○ Compact ● Referenced ○ Extended

Access Conditions
File Id: 6F06
Record 1: 4
Security 1:
Record 2:
Security 2:

Record Details
File Size: 560
Record Length: 28
Number of Records: 20

Flags
□ Increasable
□ Skip Initialisation
☑ Linkable
□ Inconsistent
□ Checksum Secured
□ Transient
□ Transaction Secur...
☑ OTA Accessible

| Record N... | Content | □ Autofil... |
|---|---|---|
| 1 | FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF | □ |
| 2 | FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF | □ |
| 3 | FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF | □ |

● FF ○ 00 ○ Custom

Figure 9: File System View

This module will help you to design and implement your SIM and USIM file systems.

### 5.2.1 File Types

This section contains a quick description of the files that this tool supports.

**5.2.1.1 Dedicated Files (DF File)** A dedicated file acts similar to a folder. They contain a collection of files, and can contain other dedicated files. This allows you to sort files into groups and give each group its own dedicated file. By doing this, the file system will be sorted into a parental hierarchy.

**5.2.1.2 Elementary Files (EF File)** Elementary files are the actual data files that can't contain other files. They come in three forms; Transparent Files, Cyclic Files, and Linear Fixed Files.

**5.2.1.2.1 Transparent Files** A transparent file consists of a sequence of bytes of a fixed length. The structure can be seen in Figure 10.

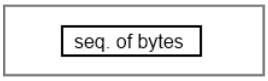


Figure 10: Transparent File Structure

**5.2.1.2.2 Linear Fixed Files** A linear fixed file contains an array of fixed length sequences of bytes. Records can be accessed by using the absolute record number. They can also be accessed by using the NEXT or PREVIOUS command. The structure can be seen in Figure 11.

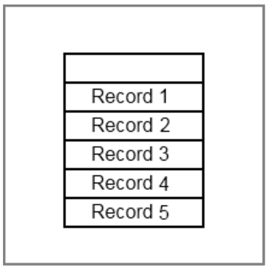


Figure 11: Linear Fixed File Structure

**5.2.1.2.3 Cyclic Files** A cyclic file is used to store records in chronological order. When all records have been used for storage, then the next storage of data shall overwrite the oldest information.

A cyclic file consists of a fixed number of records of the same length. When the file points to the last record, the next record to be accessed will be the first record. The structure can be seen in Figure 12.

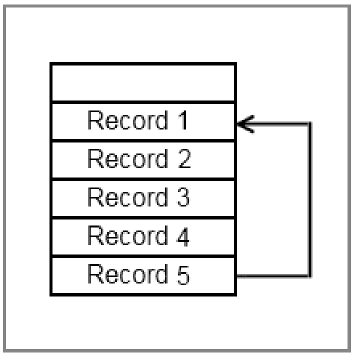


Figure 12: Cyclic File Structure

### 5.2.2 Add/Remove Files

The functions for adding and removing file are the same for SIM and USIM, but the below examples are based on the SIM file system.

**5.2.2.1 Adding a Master File or an Application Dedicated File** To add a new Master File (MF) or a new Application Dedicated File (ADF) file to this module is simple. Right-click the module in the tree located on the left hand-side. This should pop-up the menu, as you can see in Figure 13.

To add a new MF file, click on the *Add new MF* or to add a new ADF click on *Add new ADF*. This will add the MF or ADF in a new node under this module item in the tree. See Figure 14 for an example.

**5.2.2.2 Adding a Dedicated, Transparent, Linear Fixed or Cyclic File** Adding a new DF or EF file is simple. Just right-click on the node and this will pop-up the menu shown in Figure 15. From this menu select the file type you want to add. To add a new file under a DF just right-click that DF and the file you select will be added under the selected DF.

Files are added as default with an ID of '0000'. Should this file ID already exist, the user will be prompted to enter a different file ID. This ID must be 2 bytes long and contain only hex values.

#### 5.2.2.3 Removing a File

Removing a file from the file system is just as easy as adding a file. To delete a file just right-click on it and this should pop-up the menu shown in Figure 16. From this menu select *Delete* and this will remove that file from the files system. This can also be done by selecting the file and pressing the 'Delete' key.

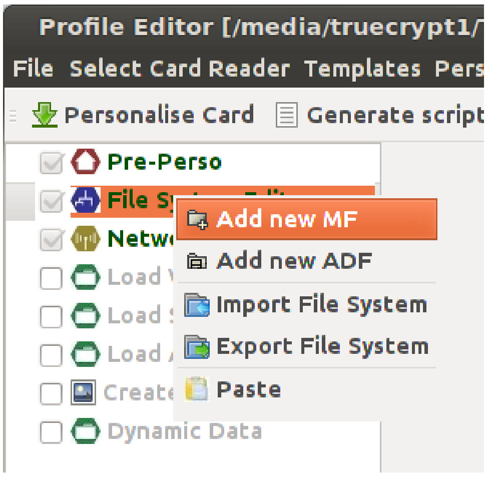


Figure 13: Adding an MF or an ADF

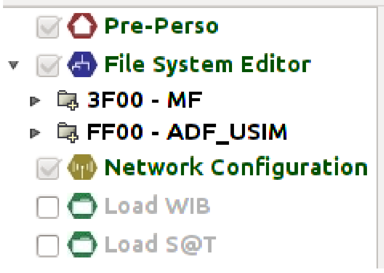


Figure 14: Example MF and ADF

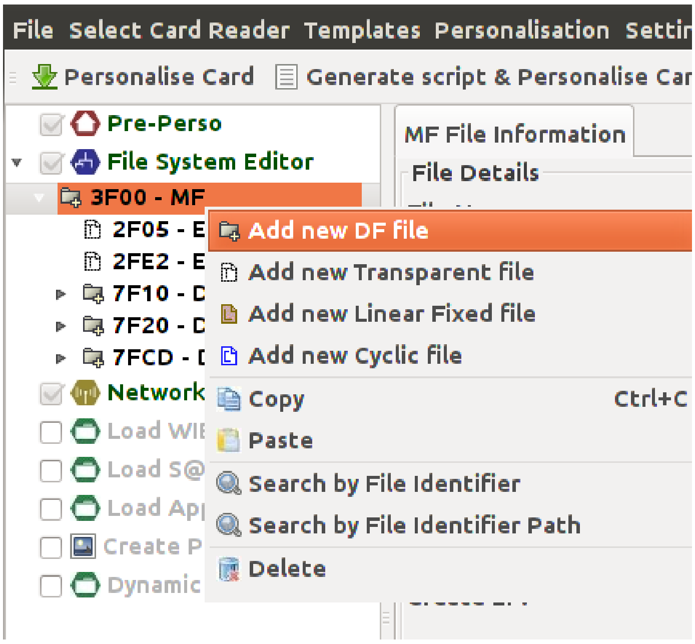


Figure 15: Adding a New File

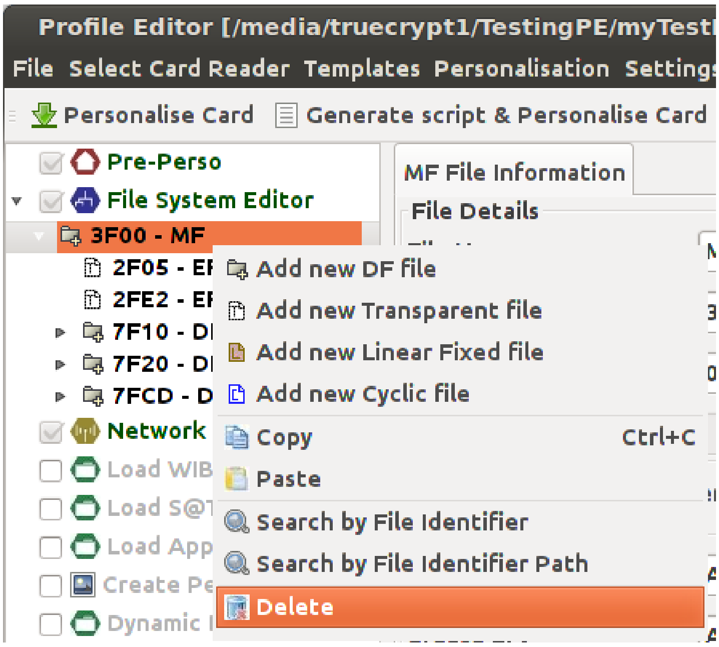


Figure 16: Deleting a File

### 5.2.3 Copying and Pasting a File

To copy a file from the file system, the user must first right-click on the required file. This will bring up the pop-up as seen in Figure 17. Clicking the 'Copy' option will copy that file. This can also be done by selecting the required file and pressing “Ctrl + C”.

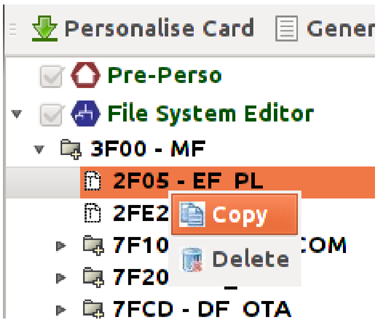


Figure 17: Copying a New File

To paste the copied file into a DF, simply right-click the DF and select the 'Paste' option, as seen in Figure 18. This will paste the file into the selected DF. This can also be done by selecting the destination DF and pressing “Ctrl + V”. If the pasted file is in the same location as the copied file, then the user will be to change the file ID so that it is unique.

Copy and pasting a file will work over multiple Profile Editor instances. Therefore it is possible to copy a file from one profile to another if the user has both windows open.

### 5.2.4 Adding Templates

The process of adding each file one by one is a bit monotonous. Therefore to speed up the process we’ve implemented templates. Templates are pre-defined file systems that the user can use as a framework for their own file systems. Files can be added or removed from the template at the users command.

The Profile Editor has a template for the following file systems:

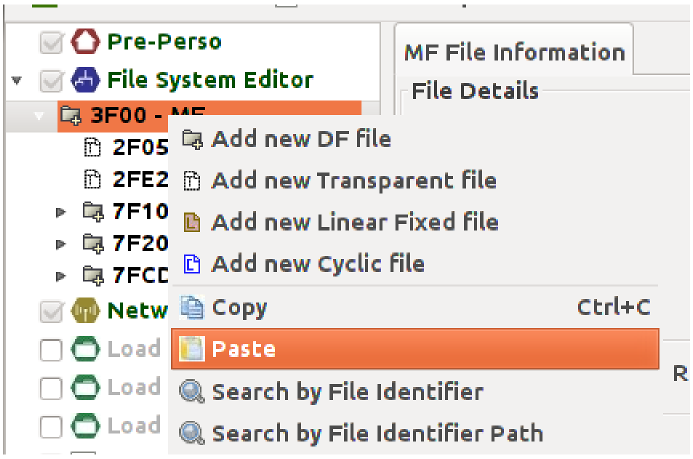


Figure 18: Pasting a New File

- SIM,
- USIM - This template includes a SIM FS as well because of the linked files.
- Simba SIM,
- Simba USIM - This template includes a Simba SIM FS as well because of the linked files.
- WIB - This is only available in the Default Java OS.
- S@T - This is only available in the Default Java OS.

Adding a template to the File System Editor is easy. Navigate to the top menu and select the *Templates* menu item. Select the template you want to add. This menu can be seen in Figure 19. The template file system will be added under the *File System Editor* node in the tree on the left hand-side.

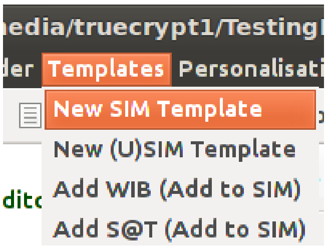


Figure 19: Adding a Template

### 5.2.5 Editing a Files Content and Structure

After creating a new file you'll be able to edit its structure and content. To view the file details just click on its node in the left-hand tree. Figure 20 shows the file details for a DF file.

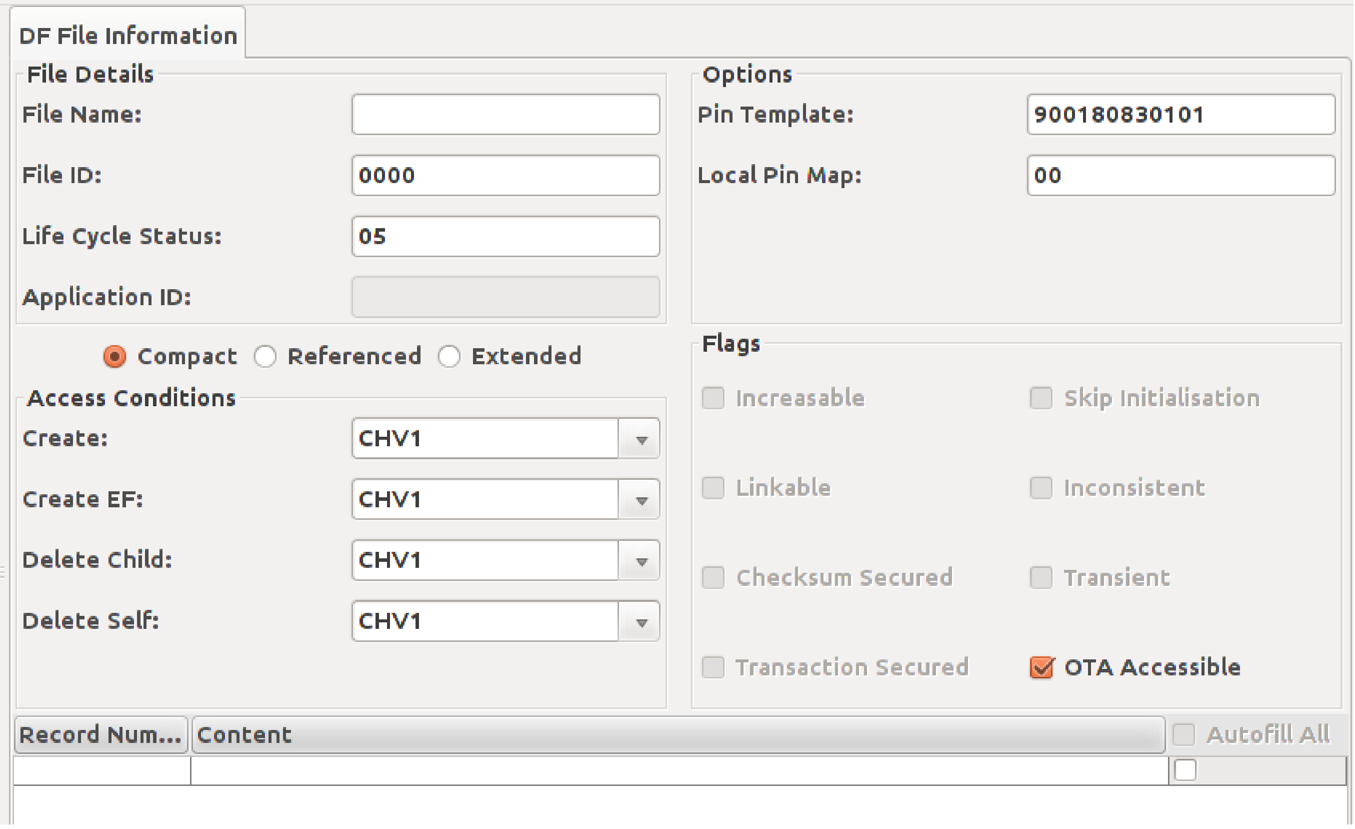
#### 5.2.5.1 File Details

*File Details* are different for DF and Elementary File (EF) files, Table 2 explains the differences.

**File Name**

The *File Name* attribute is a label that helps the user identify the file quicker in the file system list. This value isn't personalised onto the card at any stage.

One useful function we've added to assist you is the automatic File Name filler. When the *File ID* is changed and it corresponds to a file ID that is defined in the *3GPP TS 11.11 specification*, the tool will use the file name defined in that specification. The file's full path should correspond to the specification as well.



DF File Information

File Details

| | |
|---|---|
| File Name: | |
| File ID: | 0000 |
| Life Cycle Status: | 05 |
| Application ID: | |

● Compact ○ Referenced ○ Extended

Access Conditions

| | |
|---|---|
| Create: | CHV1 |
| Create EF: | CHV1 |
| Delete Child: | CHV1 |
| Delete Self: | CHV1 |

Options

| | |
|---|---|
| Pin Template: | 900180830101 |
| Local Pin Map: | 00 |

Flags

□ Increasable □ Skip Initialisation
□ Linkable □ Inconsistent
□ Checksum Secured □ Transient
□ Transaction Secured ☑ OTA Accessible

| Record Num... | Content | □ Autofill All |
|---|---|---|
| | | □ |

Figure 20: File Details For a DF

Table 2: Differences of File Details Between DF and EF

| Attribute | DF | EF |
|---|---|---|
| File Name | File label. | File label. |
| File ID | Hex value, 2 bytes. | Hex value, 2 bytes. |
| Life Cycle Status - LCS | Hex value, 1 byte. | Hex value, 1 byte. |
| Application ID | Enabled only for ADF file. | Disabled for all EFs. |
| Short File Identifier - SFI | Disabled for all DFs. | Hex value, 2 bytes. |
| Linked/Mapped File | Disabled for all DFs. | Only enabled for USIM EF files. |
| Short File Description | Disabled for all Dfs | Enabled for all EFs |

**File ID**

The *File ID* must always be a 2 byte value. If the value entered isn't 2 bytes then the text will indicate an error by formatting the text colour to red. If the value is invalid then it will not be saved.

**Life Cycle Status**

The *Life Cycle Status* must always be a byte. If the value entered isn't a byte then the text will indicate an error by formatting the text colour to red. If the value is invalid then it will not be saved.

**Application ID**

The*Application ID* is only enabled for an ADF file. This is the Instance for the USIM DF. This will only be enabled when you select the ADF node in the tree.

**Short File Identifier**

The *Short File Identifier* must always be a byte. If the value entered isn't a byte then the text will indicate an error by formatting the text colour to red. If the value is invalid then it will not be saved.

**Linked/Mapped File**

The *Linked/Mapped File* attribute is only enabled for USIM files. This value should contain a full path to the SIM file that you want to link with the USIM file.
E.g. If you want to link FF00->6F07 to 3F00->7F20->6F07, then the *Linked/Mapped File* value should be *3F007F206F07*.

A link will only be valid if the SIM file exists and that both files are of the same type. If the value that is entered is invalid then it will be indicated by formatting the text colour to red.

**Short File Description**
This text-box enables the user to write a short description of the file. This is helpful to use to write down notes as you are creating a profile.

**Temporary File**
This option is only available when the Simba OS is selected. If this check-box is ticked then the file will not be guaranteed over sessions. Therefore the contents of the temporary file will be lost after a reset.

**5.2.5.2 Record Details** *Record Details* are different for DF and EF files. Table 3 explains the differences.

Table 3: Differences of Record Details Between DF and EF

| Attribute | DF | EF |
|---|---|---|
| Pin Template | Enabled | Hidden for all EFs. |
| Local Pin Map | PIN Status Template. | Hidden for all EFs. |
| File Size | Hidden for all DFs. | Enabled for Transparent File only. |
| Record Length | Hidden for all DFs. | Enabled for Linear Fixed and Cyclic File only. |
| Number of Records | Hidden for all DFs. | Enabled for Linear Fixed and Cyclic File only. |

**Pin Template**
The *Pin Template* attribute is only available for DF files.

**Local Pin Map**
The *Local Pin Map* attribute is only available for DF files. This value allows you to set the PIN Status Template.

**File Size**
The *File Size* attribute is only available for transparent files. Changing this value will change the file data length.

**Record Length**
The *Record Length* attribute is only available for a Linear Fixed and a Cyclic File.

**Number of Records**
The *Number of Records* attribute is only available for a Linear Fixed and a Cyclic File.

**5.2.5.3 Access Conditions** There are three types of Access Conditions (AC) that the user can choose between. These are Compact, Referenced, and Extended. All of the available AC can be found in Appendix B, section 10

**5.2.5.3.1 Compact AC** Compact AC gives you the ability to change the AC for four commands; Read, Update, Activate, and Deactivate. These control the ability of the user to access that file. For example, if these AC for the command read are met, then the user will be able to read the data in that file.

**5.2.5.3.2 Referenced AC** A file which uses Referenced AC uses another file to set its AC. The referenced AC can be decoded by selecting the small information symbol next to the record number field. This will bring up a dialog box which tells you the PIN values of the access conditions.

**5.2.5.3.3 Extended/Expanded AC** Extended/Expanded AC works the same as Compact AC. However, they allow the setting of more than one AC for each command.

**5.2.5.4 Flags** Numerous flags can be set which each have different effects on the selected file. The flags which can be set are described in this section.

**5.2.5.4.1 OTA accessible** If the OTA accessible flag is not set then the file will not be accessible over-the-air, e.g. via RFM. This is only available for the Default Java OS.

**5.2.5.4.2 RFM Read** If the RFM Read flag is not set then the file will not be readable over-the-air, e.g. via RFM. This is only available for the Simba OS.

**5.2.5.4.3 RFM Update** If the RFM Update flag is not set then the file will not be update-able over-the-air, e.g. via RFM. This is only available for the Simba OS.

**5.2.5.4.4 Applet Read** If the Applet Read flag is not set then the applets will not be readable over-the-air, e.g. via RFM. This is only available for the Simba OS.

**5.2.5.4.5 Applet Update** If the Applet Update flag is not set then the file will not be update-able over-the-air, e.g. via RFM. This is only available for the Simba OS.

**5.2.5.4.6 Readable/Updatable When Deactivated** If the Readable/Updatable When Deactivated flag is not set then the file will not be read or update a file which has been deactivated. This is only available for the Simba OS.

**5.2.5.4.7 Transaction secured** The transaction secured flag is only available for EFs. If this flag is set then updates to the file's data will be guarded by the JavaCard TM transaction mechanism.

**5.2.5.4.8 Transient** The transient flag is only available for EFs. If this flag is set then the file's data will be allocated in clear-on-reset RAM.

**5.2.5.4.9 Checksum secured** The checksum secured flag is only available for EFs. If this flag is set then the file's data will be protected by a checksum, which is calculated during updates and verified during reads.

**5.2.5.4.10 Inconsistent** Displays whether the file is inconsistent. This is an internal flag, which must not be set from outside world.

**5.2.5.4.11 Linkable** The linkable flag is only available for EFs. If this flag is not set then the file will not allow any links to itself, e.g. it will be not shareable.

**5.2.5.4.12 Skip initialization** The skip initialization flag is only available for EFs. If this flag is set then the file's data will not be explicitly filled with 0xFF bytes during file creation. This can considerably speed up the personalisation process if the file is to be fully updated at a later time.

**5.2.5.4.13 Increasable** The increasable flag is only available for Cyclic EFs. If this flag is not set then there will be no way to run the INCREASE command on this file in 2G mode; 3G INCREASE ignores this flag.

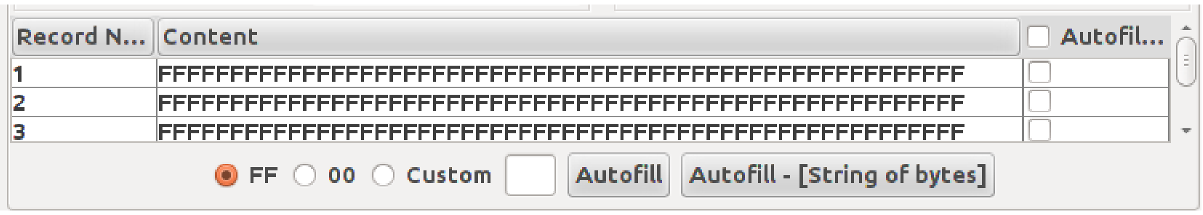
**5.2.5.5 File Content** The content of the file is a sequence of bytes with the length limits set in the File Size for Transparent files, or the Record Length for Cyclic and Linear Fixed files.

The amount of data is controlled by the number of records. To add more data into the file, simply increase the number of records. The newly added data will take the form of the last data entered into the content table. To remove data, simply reduce the number of records.

There are numerous functions to help the user to easily create data. One such function is the padding/cutting of data. If the data entered does not meet the required length, then the data is padded or cut automatically until it does meet the requirements.

Also, if the user accidentally copy and pastes in non-hex characters, then they will be automatically removed from the entered data. When the user moves away from the data then it will be cut or padded as required.

Another function the user can take advantage of is the auto-fill option. This can be seen in Figure 21. If the user wants to fill in the data using repeated bytes, then it is quicker for them to use the auto-fill function instead of manually entering the data. To do this, the user should either select the *Autofill All* check-box or select the check-boxes for the rows which they require to be auto-filled. Once the required check-boxes have been checked, the user can choose the byte they want to fill the data with. The options for this byte are "FF", "00", or a custom byte the user enters. Once the user has selected the relevant byte, clicking the *Autofill* button will fill in the data as required. The user can also choose to autofill the columns with a specified string of bytes. To do this the user can click the *Autofill – [String of bytes]* button as seen in Figure 21. This will bring up an input box where the user can specify the string of bytes they wish to use. Pressing OK will fill the selected columns with the entered data.



| Record N... | Content | □ Autofil... |
|---|---|---|
| 1 | FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF | □ |
| 2 | FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF | □ |
| 3 | FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF | □ |

● FF ○ 00 ○ Custom

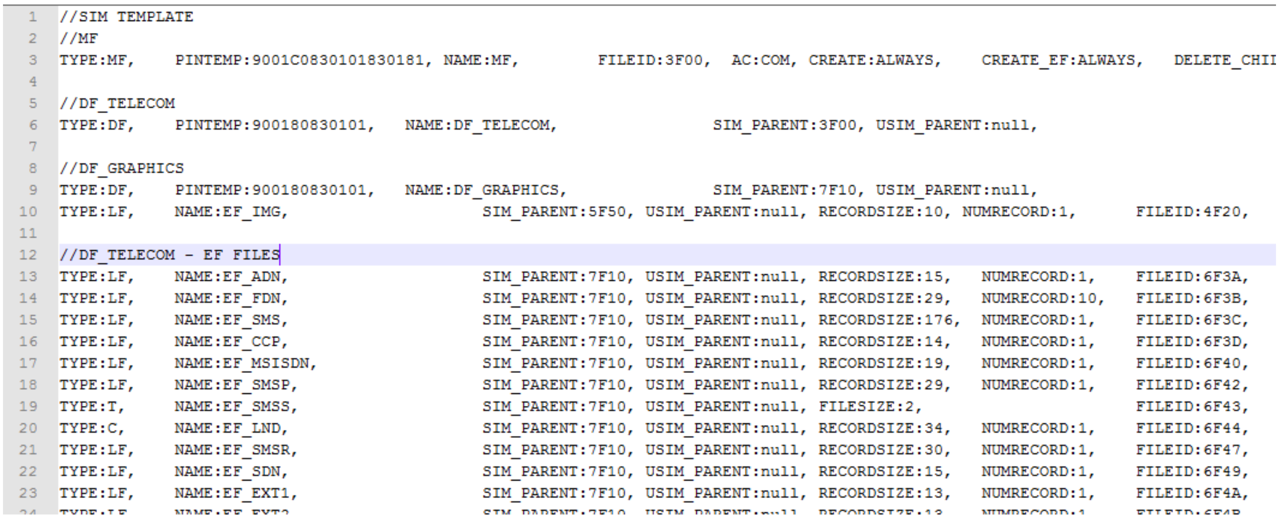
Figure 21: Example Content in a File Along With Autofill Options

### 5.2.6 Import/Export File System for External Editing

#### 5.2.6.1 How To Use This Feature

The Import and Export functionality is a useful feature that can be used by an experienced user to develop a file system quickly and externally from the Profile Editor.

The first thing you'll need to do is insert a template into the Profile Editor. Once your selected template is loaded it can be exported into a text file for external editing. This file can be edited by any conventional text editor.



```
//SIM TEMPLATE
//MF
TYPE:MF,    PINTEMP:9001C0830101830181, NAME:MF,        FILEID:3F00,  AC:COM, CREATE:ALWAYS,    CREATE_EF:ALWAYS,   DELETE_CHII

//DF_TELECOM
TYPE:DF,    PINTEMP:900180830101,   NAME:DF_TELECOM,                SIM_PARENT:3F00, USIM_PARENT:null,

//DF_GRAPHICS
TYPE:DF,    PINTEMP:900180830101,   NAME:DF_GRAPHICS,               SIM_PARENT:7F10, USIM_PARENT:null,
TYPE:LF,    NAME:EF_IMG,                     SIM_PARENT:5F50, USIM_PARENT:null, RECORDSIZE:10, NUMRECORD:1,      FILEID:4F20,

//DF_TELECOM - EF FILES
TYPE:LF,    NAME:EF_ADN,                     SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:15,   NUMRECORD:1,    FILEID:6F3A,
TYPE:LF,    NAME:EF_FDN,                     SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:29,   NUMRECORD:10,   FILEID:6F3B,
TYPE:LF,    NAME:EF_SMS,                     SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:176,  NUMRECORD:1,    FILEID:6F3C,
TYPE:LF,    NAME:EF_CCP,                     SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:14,   NUMRECORD:1,    FILEID:6F3D,
TYPE:LF,    NAME:EF_MSISDN,                  SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:19,   NUMRECORD:1,    FILEID:6F40,
TYPE:LF,    NAME:EF_SMSP,                    SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:29,   NUMRECORD:1,    FILEID:6F42,
TYPE:T,     NAME:EF_SMSS,                    SIM_PARENT:7F10, USIM_PARENT:null, FILESIZE:2,                      FILEID:6F43,
TYPE:C,     NAME:EF_LND,                     SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:34,   NUMRECORD:1,    FILEID:6F44,
TYPE:LF,    NAME:EF_SMSR,                    SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:30,   NUMRECORD:1,    FILEID:6F47,
TYPE:LF,    NAME:EF_SDN,                     SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:15,   NUMRECORD:1,    FILEID:6F49,
TYPE:LF,    NAME:EF_EXT1,                    SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:13,   NUMRECORD:1,    FILEID:6F4A,
TYPE:LF,    NAME:EF_EXT2,                    SIM_PARENT:7F10, USIM_PARENT:null, RECORDSIZE:13,   NUMRECORD:1,    FILEID:6F4B
```

Figure 22: Example of an Exported File

The structure of this text file is explained below.

#### 5.2.6.2 Text File Structure

The exported file will have a single line to represent a single file in the file system. Users must remember to define a DF file before defining any files that are included in this DF.

Each file type has a slight difference to all the other types. The below sections will explain these slight differences.

##### 5.2.6.2.1 DF Structure

Table 4 shows the required structure of a Dedicated File within the import/export text-file.

Table 4: Structure of DF

| Mandatory Tags | Validation |
|---|---|
| TYPE | Must be ‘MF’, ‘ADF’ or ‘DF’. |
| PINTEMP | Value must be even in length. |
| NAME | Can be any value, if null is entered then the file name will be blank. |
| FILE ID | Must be 2 bytes and a valid hex number. |
| AC | This tag is explained in more details in section 6.2.6.2.5. |
| LFS | Must be 1 bytes and a valid hex number. |
| LPM | Must be 1 bytes and a valid hex number. |

| Optional Tags | Validation |
|---|---|
| SIM_PARENT / USIM_PARENT | Must be 2 bytes and a valid hex number,<br>must be used if a DF is inside a ‘MF’ or ‘ADF’. |
| FLAGS | This tag is explained in more details in section 6.2.6.2.6. |

**Examples:** 1) Only mandatory tags,

**TYPE:DF,PINTEMP:900180830101,NAME:MF,FILEID:3F00,AC:COM, CREATE:ALWAYS,CREATE_EF:ALWAYS,DELETE_CHILD:ADM1, DELETE_SELF:NEVER, LFS:05,LPM:00**

2) Mandatory tags and optional tags

**TYPE:DF,PINTEMP:900180830101,NAME:DF_TELECOM,SIM_PARENT:3F00, FILEID:7F10,AC:REF,ACFILEID:2F06,ACRECID:1,LFS:05,LPM:00, FLAGS:SKIP_INITIALIZATION|OTA_ACCESSIBLE**

**Note:** Tags and values can be ordered in any sequence. Place the tags in the order that best suits your working style.

#### 5.2.6.2.2 Transparent Structure

Table 5 shows the required structure of a Transparent File within the import/export text-file.

**Examples:** 1) Only mandatory tags,
TYPE:T,NAME:EF_PL,SIM_PARENT:3F00,FILESIZE:10,FILEID:2F05, AC:REF, ACFILEID:2F06, ACRECID:5,LFS: 05

2) Mandatory tags and optional tags
TYPE:T,NAME:EF_PL,SIM_PARENT:3F00,FILESIZE:10,FILEID:2F05, AC:REF, ACFILEID:2F06, ACRECID:5,LFS: 05,SFI:05, FLAGS:OTA_ACCESSIBLE,DATA:0102030405060708090A, FILEDESC:This file is mandatory.

**Note:** Tags and values can be ordered in any sequence. Place the tags in the order that best suits your working style.

#### 5.2.6.2.3 Linear Fixed Structure

Table 6 shows the required structure of a Linear Fixed File within the import/export text-file.

**Examples:** 1) Only mandatory tags,
TYPE:LF,NAME:EF_ADN,SIM_PARENT:7F10, RECORDSIZE:10,NUMRECORD:250, FILEID:6F3A,AC:REF,ACFILEID:6F06, ACRECID:3,LFS:05

2) Mandatory tags and optional tags
TYPE:LF,NAME:EF_ADN,SIM_PARENT:7F10, RECORDSIZE:10,NUMRECORD:3,FILEID:6F3A, AC:REF,ACFILEID:6F06, ACRECID:3,LFS:05,SFI:null,FLAGS:OTA_ACCESSIBLE|LINKABLE, DATA:0102030405060708090A, FILEDESC:This file is mandatory.

Table 5: Structure of Transparent File

| Mandatory Tags | Validation |
|---|---|
| TYPE | Must be 'T'. |
| NAME | Can be any value, if null is entered then the file name will be blank. |
| SIM_PARENT / USIM_PARENT | Must be 2 bytes and a valid hex number. |
| FILE ID | Must be 2 bytes and a valid hex number. |
| FILESIZE | Must be a valid number and NOT zero. |
| AC | This tag is explained in more details in section 6.2.6.2.5. |
| LFS | Must be 1 bytes and a valid hex number. |
| **Optional Tags** | **Validation** |
| FLAGS | This tag is explained in more details in section 6.2.6.2.6. |
| DATA | This tag is explained in more details in section 6.2.6.2.7. |
| SFI | Must be 2 bytes and a valid hex number,<br>if it's null then the value will be empty. |
| LINKEDFILE | Must be a valid path to a file, |
| FILEDESC | Value must not include the characters "," or ":".<br>The following sequence of characters is reserved for representing spaces: \s/ |

Table 6: Structure of Linear Fixed File

| Mandatory Tags | Validation |
|---|---|
| TYPE | Must be 'LF'. |
| NAME | Can be any value, if null is entered then the file name will be blank, |
| SIM_PARENT / USIM_PARENT | Must be 2 bytes and a valid hex number, |
| FILE ID | Must be 2 bytes and a valid hex number, |
| RECORDSIZE | Must be a valid number and NOT zero. |
| NUMRECORD | Must be a valid number and NOT zero. |
| AC | This tag is explained in more details in section 6.2.6.2.5. |
| LFS | Must be 1 bytes and a valid hex number, |
| **Optional Tags** | **Validation** |
| FLAGS | This tag is explained in more details in section 6.2.6.2.6. |
| DATA | This tag is explained in more details in section 6.2.6.2.7. |
| SFI | Must be 2 bytes and a valid hex number,<br>if it's null then the value will be empty. |
| LINKEDFILE | Must be a valid path to a file, |
| FILEDESC | Value must not include the characters "," or ":".<br>The following sequence of characters is reserved for representing spaces: \s/ |

3) Mandatory tags and optional tags
TYPE:LF,NAME:EF_ADN,SIM_PARENT:7F10, RECORDSIZE:10,NUMRECORD:3,FILEID:6F3A,AC:REF, ACFILEID:6F06, ACRECID:3,LFS:05,SFI:null,FLAGS:OTA_ACCESSIBLE|LINKABLE, DATA:0102030405060708090A-0102030405060708090B-0102030405060708090C, FILEDESC:This file is mandatory.

**Note:** Tags and values can be ordered in any sequence. Place the tags in the order that best suits your working style.

#### 5.2.6.2.4 Cyclic Structure

Table 7 shows the required structure of a Cyclic File within the import/export text-file.

Table 7: Structure of Cyclic File

| **Mandatory Tags** | **Validation** |
|---|---|
| TYPE | Must be 'C'. |
| NAME | Can be any value, if null is entered then the file name will be blank, |
| SIM_PARENT / USIM_PARENT | Must be 2 bytes and a valid hex number, |
| FILE ID | Must be 2 bytes and a valid hex number, |
| RECORDSIZE | Must be a valid number and NOT zero. |
| NUMRECORD | Must be a valid number and NOT zero. |
| AC | This tag is explained in more details in section 6.2.6.2.5. |
| LFS | Must be 1 bytes and a valid hex number, |
| **Optional Tags** | **Validation** |
| FLAGS | This tag is explained in more details in section 6.2.6.2.6. |
| DATA | This tag is explained in more details in section 6.2.6.2.7. |
| SFI | Must be 2 bytes and a valid hex number,<br>if it's null then the value will be empty. |
| LINKEDFILE | Must be a valid path to a file, |
| FILEDESC | Value must not include the characters "," or ":".<br>The following sequence of characters is reserved for representing spaces: \s/ |

**Examples:** 1) Only mandatory tags,
TYPE:C,NAME:EF_LND,SIM_PARENT:7F10,RECORDSIZE:34,NUMRECORD:1, FILEID:6F44,AC:COM, READ:CHV1,UPDATE:CHV1, ACTIVATE:ADM1,DEACTIVATE:ADM1,INCREASE:NEVER,LFS:05

2) Mandatory tags and optional tags
TYPE:C,NAME:EF_LND,SIM_PARENT:7F10,RECORDSIZE:34,NUMRECORD:1, FILEID:6F44,AC:COM, READ:CHV1,UPDATE:CHV1, ACTIVATE:ADM1,DEACTIVATE:ADM1,INCREASE:NEVER,LFS:05,SFI:null, FLAGS:OTA_ACCESSIBLE|LINKABLE, DATA:0102030405060708090A, FILEDESC:This file is mandatory.

3) Mandatory tags and optional tags
TYPE:C,NAME:EF_LND,SIM_PARENT:7F10,RECORDSIZE:34,NUMRECORD:1, FILEID:6F44,AC:COM, READ:CHV1,UPDATE:CHV1, ACTIVATE:ADM1,DEACTIVATE:ADM1,INCREASE:NEVER,LFS:05,SFI:null, FLAGS:OTA_ACCESSIBLE|LINKABLE,
DATA:0102030405060708090A-0102030405060708090B-0102030405060708090C,
`FILEDESC:This file is mandatory.`

**Note:** Tags and values can be ordered in any sequence. Place the tags in the order that best suits your working style.

#### 5.2.6.2.5 Access Condition Validation

This tags validation is a little bit different. The tag can have multiple types of Access Condition. They will also be handled differently for different file types.

| Type | AC Tag | Mandatory DF Tags | Mandatory Transparent Tags | Mandatory Linear Fixed Tags | Mandatory Cyclic Tags |
|---|---|---|---|---|---|
| Compact | COM | CREATE,<br>CREATE_EF,<br>DELETE_CHILD,<br>DELETE_SELF | READ,<br>UPDATE,<br>ACTIVATE,<br>DEACTIVATE | | READ,<br>UPDATE,<br>ACTIVATE,<br>DEACTIVATE,<br>INCREASE |
| Extended | EXT | | | | |
| Referenced | REF | ACFILEID, ACRECID | | | |

When the AC tag is one of the above values then the mandatory flags for that file will also include the tags corresponding to the file type.

Each of these tags will have to contain valid data. For the Compact and the Extended types the values must be any of the values found in Table 8

Table 8: Valid Compact/Extended Access Conditions

| Valid Access Condition Values | | |
|---|---|---|
| ALWAYS | CHV6 | UNIVERSAL |
| CHV1 | CHV7 | _2PIN1 |
| CHV3 | CHV8 | _2PIN2 |
| NEVER | ADM5 | _2PIN3 |
| ADM1 | ADM6 | _2PIN4 |
| ADM2 | ADM7 | _2PIN5 |
| ADM3 | ADM8 | _2PIN6 |
| ADM4 | ADM9 | _2PIN7 |
| CHV4 | ADM10 | _2PIN8 |
| CHV5 | ADM11 | |

The valid access condition values for the Referenced type are shown in Table 9

Table 9: Valid Referenced Access Conditions

| Valid Access Condition Values | Validation |
|---|---|
| ACFILEID | Must be 2 bytes and a valid hex number. |
| ACRECID | Must be a valid number and NOT zero. |

#### 5.2.6.2.6 Flags Validation

This tags validation is a little bit different; it can handle multiple flags in the same tag. This can be done by separating the values with a |. Table 10 contains all the valid values a flag can have.

#### 5.2.6.2.7 Data Validation

This tag is only available for EF files. It's an optional tag that can be included in a file structure to give a file specific data. If this tag isn't supplied then the default data will be 0xFF padded out to the length of the file. If the file is a Linear Fixed or a Cyclic file then all the records will be set to 0xFF.

**Transparent file:** If the DATA tag is included in the file structure then the value for this tag must correspond to the file size. If this isn't complied with then the import will fail.

**Linear Fixed/Cyclic file:**

Table 10: Valid Flag Values

| **Valid Flag Values** | |
|---|---|
| CHECKSUM_SECURED | OTA_ACCESSIBLE |
| INCONSISTENT | SKIP_INITIALIZATION |
| INCREASABLE | TRANSACTION_SECURED |
| LINKABLE | TRANSIENT |
| EDITABLE_ON_DEACTIVATE | RFM_READ |
| RFM_REST | OTHER_APPLET_READ |
| OTHER_APPLET_REST | TEMPORARY |

If the DATA tag is included in this file structure then the data length will have to correspond to the records size. If all the records in the file have the same value as the first record then just leave one record in the DATA tag, the Profile Editor will fill the remaining records with the same content. If the records are different then all the records data will have to be added into the DATA tag. This can be done by splitting each records with the '-' character.

### 5.2.7 Search the File System

Searching the file system is made easy by Profile Editor. To do this, the user must first right-click on a DF. This will bring up the DF Menu as seen in Figure 15. Here the user has two options to search the file system; *Search by File Identifier* and *Search by File Identifier Path*.

Selecting the *Search by File Identifier* option will allow the user to search for a file ID (i.e. 2FE2). This will then search the file system and will select the chosen file. If more than one file is found then the user is asked to select the required file.

Selecting the *Search by File Identifier Path* option will allow the user to search for a file path (i.e. 3F002FE2). The full file path must be supplied here in order for a file to be found. This will then search the file system and will select the chosen file. If more than one file is found then the user is asked to select the required file.

### 5.2.8 Profile Validation

When creating a profile, it may be useful to compare the current state of the profile to those files in the SIM/USIM templates (ie ETSI/3GPP specification). This process can be carried out via the profile validation tool.

The profile validation tool can be found by right clicking *File System Editor* and selecting *Validate Profile*, as shown in Figure 23. This button will check the current file system against the profile editor templates (currently only Java SIM and USIM), and display the files that are mismatched with the templates in a new window. The user can check the relevant check-boxes of the errors that they wish to edit to comply with the template. The user should then press the *validate* button to change the selected files.

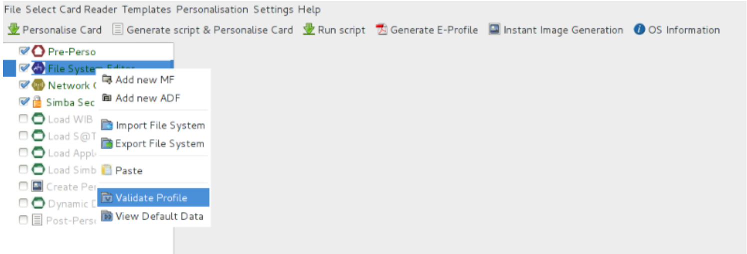
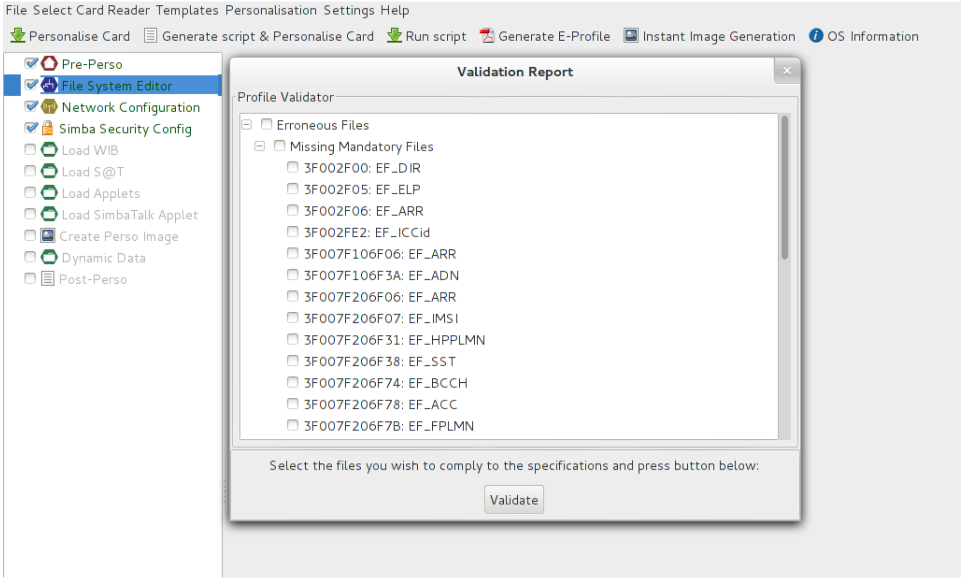


Figure 23: Profile Validator - Profile Validation Selection

Errors are grouped into three main areas. These are; Missing Mandatory Files, Incorrect File Type, and Incorrect Access Conditions.

Missing mandatory files are files which are marked as mandatory in the 3GPP/ETSI specifications. These files are required in order to create a successful personalisation and therefore must be included in the file system. The profile validation window will display the file ID, and name of the missing mandatory file, according to the 3GPP/ETSI specifications, as shown in Figure 24. Selecting and validating these errors will add the template version of these files to the current file system. Should the file be a child of a directory file that does not yet exist, this directory file will be created.



Validation Report

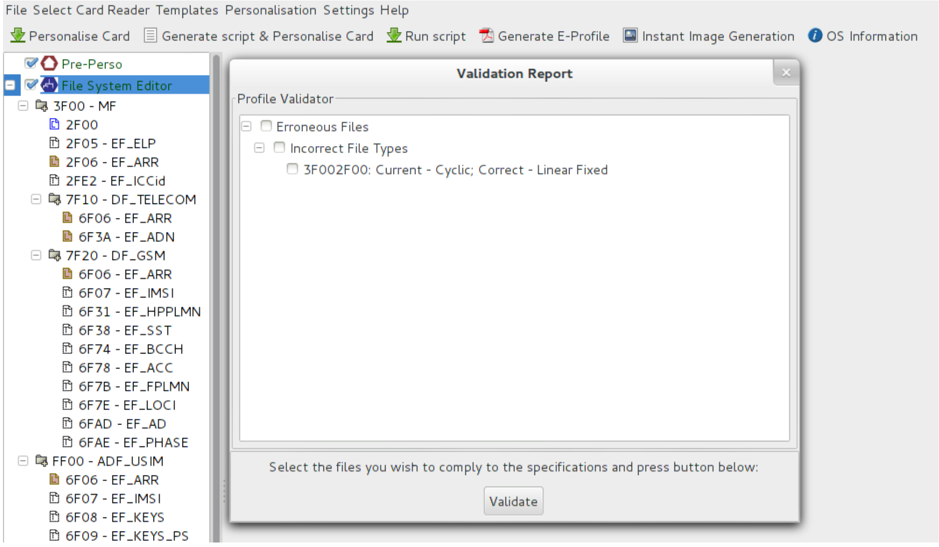
Profile Validator

- □ Erroneous Files
  - □ Missing Mandatory Files
    - □ 3F002F00: EF_DIR
    - □ 3F002F05: EF_ELP
    - □ 3F002F06: EF_ARR
    - □ 3F002FE2: EF_ICCid
    - □ 3F007F106F06: EF_ARR
    - □ 3F007F106F3A: EF_ADN
    - □ 3F007F206F06: EF_ARR
    - □ 3F007F206F07: EF_IMSI
    - □ 3F007F206F31: EF_HPPLMN
    - □ 3F007F206F38: EF_SST
    - □ 3F007F206F74: EF_BCCH
    - □ 3F007F206F78: EF_ACC
    - □ 3F007F206F7B: EF_FPLMN

Select the files you wish to comply to the specifications and press button below:

Figure 24: Profile Validator - Missing Mandatory Files

Incorrect File Type errors are apparent when a file has the ID of a file in the 3GPP/ETSI specifications, but this file is of a different type (e.g. Linear Fixed, Cyclic, or Transparent file types). The profile validation window will display both the current and correct file type for this ID, as seen in Figure 25. Selecting and validating these errors will cause the specified file to be replaced with the relevant file in the template (ie all file content will be edited).



- 3F00 - MF
  - 2F00
  - 2F05 - EF_ELP
  - 2F06 - EF_ARR
  - 2FE2 - EF_ICCid
  - 7F10 - DF_TELECOM
    - 6F06 - EF_ARR
    - 6F3A - EF_ADN
  - 7F20 - DF_GSM
    - 6F06 - EF_ARR
    - 6F07 - EF_IMSI
    - 6F31 - EF_HPPLMN
    - 6F38 - EF_SST
    - 6F74 - EF_BCCH
    - 6F78 - EF_ACC
    - 6F7B - EF_FPLMN
    - 6F7E - EF_LOCI
    - 6FAD - EF_AD
    - 6FAE - EF_PHASE
  - FF00 - ADF_USIM
    - 6F06 - EF_ARR
    - 6F07 - EF_IMSI
    - 6F08 - EF_KEYS
    - 6F09 - EF_KEYS_PS

Validation Report

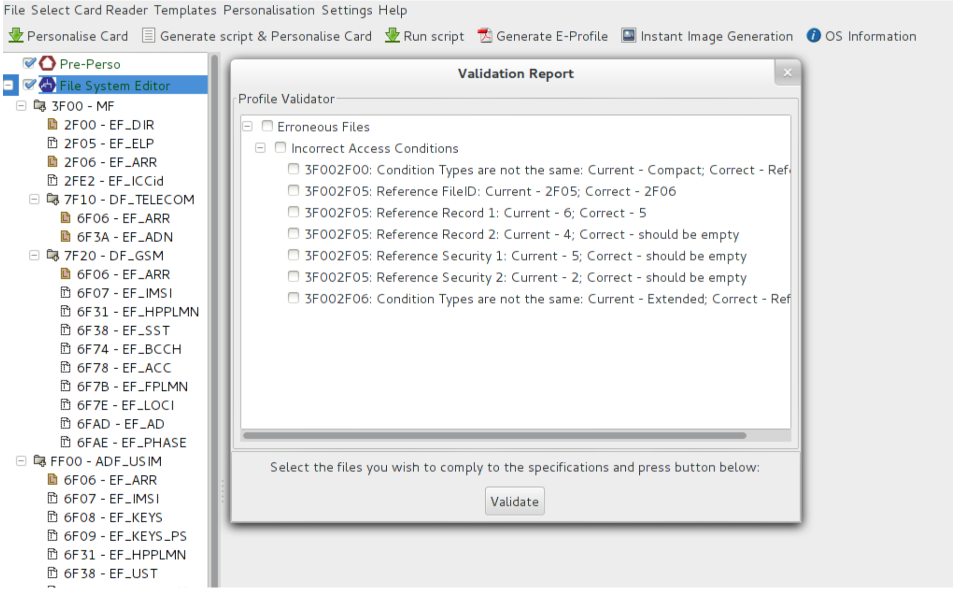
Profile Validator

- □ Erroneous Files
  - □ Incorrect File Types
    - □ 3F002F00: Current - Cyclic; Correct - Linear Fixed

Select the files you wish to comply to the specifications and press button below:

Figure 25: Profile Validator - Incorrect File Types

Incorrect Access Conditions will appear when either the access condition type, or the individual access condition values do not comply with those in the templates. The profile validation window will display the previous and the correct versions of the access conditions so that the user can easily assess whether or not the current version is incorrect, as seen in Figure 26. Selecting and validating these errors will change the individual access conditions that have been selected to those in the templates. Should the access condition type (e.g. compact, referenced, extended) be changed, the new values will automatically be those that are in the 3GPP/ETSI specifications for that file.



□ Erroneous Files
□ Incorrect Access Conditions
□ 3F002F00: Condition Types are not the same: Current - Compact; Correct - Ref
□ 3F002F05: Reference FileID: Current - 2F05; Correct - 2F06
□ 3F002F05: Reference Record 1: Current - 6; Correct - 5
□ 3F002F05: Reference Record 2: Current - 4; Correct - should be empty
□ 3F002F05: Reference Security 1: Current - 5; Correct - should be empty
□ 3F002F05: Reference Security 2: Current - 2; Correct - should be empty
□ 3F002F06: Condition Types are not the same: Current - Extended; Correct - Ref

Select the files you wish to comply to the specifications and press button below:

Figure 26: Profile Validator - Incorrect Access Conditions

### 5.2.9 Viewing Default Data

Profile Editor has the functionality to decode and display the default data in files EF_DIR, EF_EST, EF_OPL, EF_PNN, EF_SPN, EF_SST and EF_UST. This feature can be accessed by selecting the *View Default Data* button in the right click menu of File System Editor Module, as seen in Figure 27.

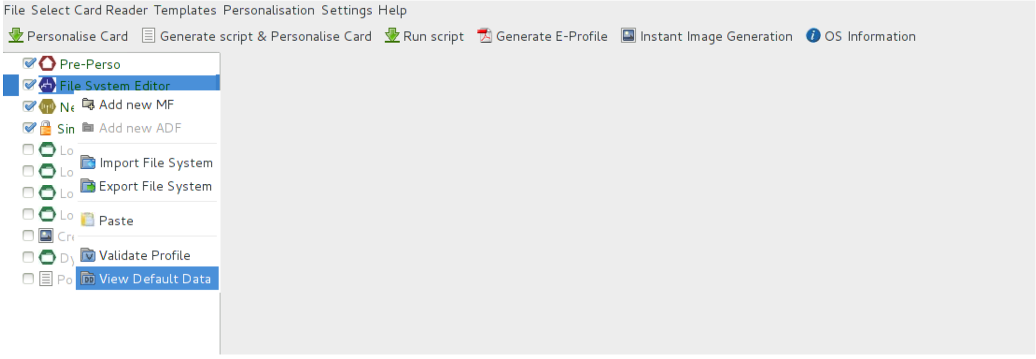
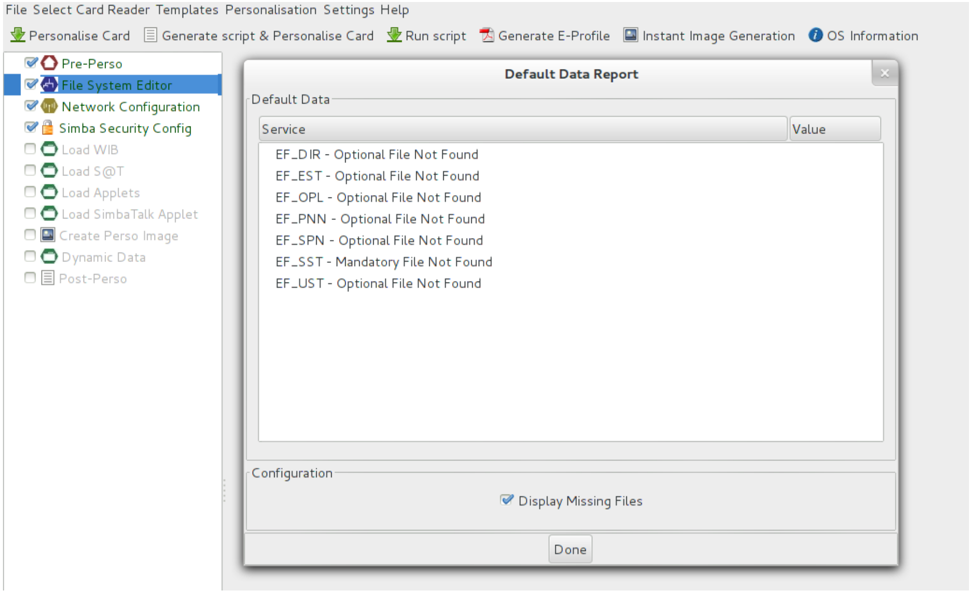


Figure 27: View Default Data

Once selected, a new window will appear with the title *Default Data Report*, as shown in Figure 28. Within this window, different files can be expanded or collapsed to view their data. As well as this, the user has the option to choose whether or not to display missing files (i.e. files not in this profile). This can be done by toggling the *Display Missing Files* check box. Once this is done, any files which are not found in the file system will no longer appear in the Default Data Report.



**Default Data Report**

Default Data

| Service | Value |
|---|---|
| EF_DIR - Optional File Not Found | |
| EF_EST - Optional File Not Found | |
| EF_OPL - Optional File Not Found | |
| EF_PNN - Optional File Not Found | |
| EF_SPN - Optional File Not Found | |
| EF_SST - Mandatory File Not Found | |
| EF_UST - Optional File Not Found | |

Configuration

☑ Display Missing Files

Figure 28: View Default Data - Display Window

## 5.3 Network Configuration

To access the Network Configuration module, first click on the *Network Configuration* item in the tree that's located at the left hand-side of the tool. This will display the window in Figure 29.

### 5.3.1 Network Algorithm

In this section you're able to select the 2G and 3G GSM algorithms. For the 2G GSM algorithm, you can choose between COMP V1, COMP V2, COMP V3, MILENAGE, XOR, CIS-B, MILENAGE V1, and MILENAGE V2. Whereas for the 3G GSM algorithm you can choose between only MILENAGE and XOR. These algorithms are dependent on the OS selected.

### 5.3.2 Milenage Configuration

Here you're able to change the milenage configuration. You can select to use the OP value, which can be edited on this screen. Or you can choose to use the OPC value. If this value is selected then the OPC values must be entered in the dynamic data file which has been loaded into the Dynamic Data Module. If the OPC value is not contained in dynamic data then the file is created with the same default value as listed below for OP.

The default OP value is: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF.

### 5.3.3 SQN Configuration

In this section you're able to reset the data fields for both the SQN Configuration and Milenage Constants Configuration. This is done by selecting the *Reset Defaults* button shown in Figure 29.

Default values for the SQN Configuration are shown in Table 11

Table 11: Default Values for the SQN Configuration

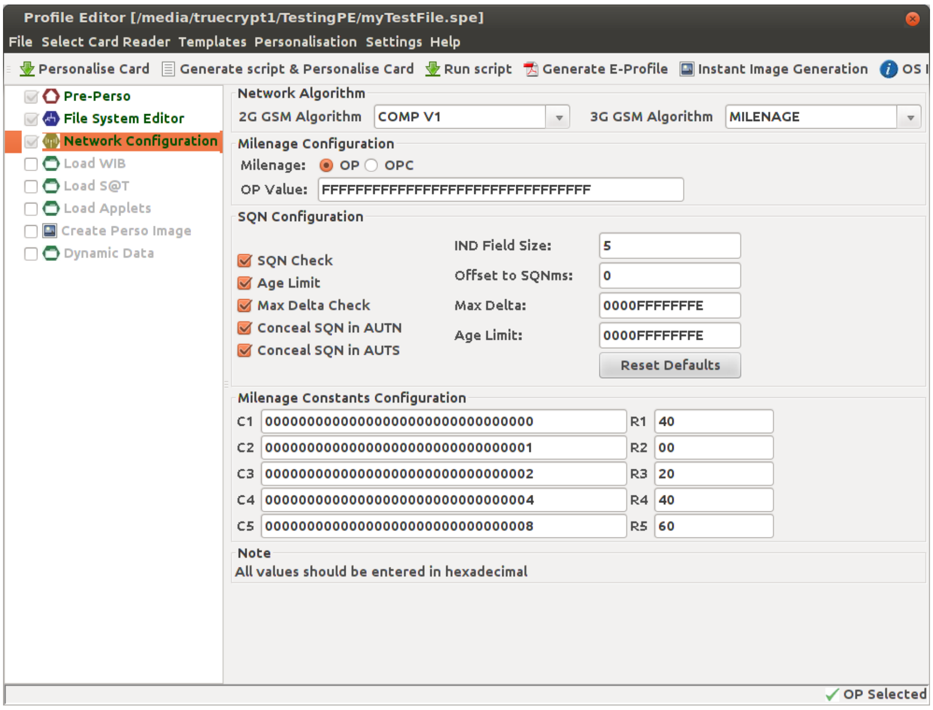
| Field | Value |
|---|---|
| IND Field Size | 5 |
| Offset to SQNms | 0 |
| Max Delta | 000010000000 |
| Age Limit | 000010000000 |

### 5.3.4 Milenage Constants Configuration

Here you're able to set the milenage constants configuration. The default values are shown in Table 12

Table 12: Milenage Constants Configuration

| Field | Value | Field | Value |
|---|---|---|---|
| C1 | 00000000000000000000000000000001 | R1 | 40 |
| C2 | 00000000000000000000000000000001 | R2 | 00 |
| C3 | 00000000000000000000000000000002 | R3 | 20 |
| C4 | 00000000000000000000000000000004 | R4 | 40 |
| C5 | 00000000000000000000000000000008 | R5 | 60 |



Network Algorithm

2G GSM Algorithm: COMP V1

3G GSM Algorithm: MILENAGE

Milenage Configuration

Milenage: ● OP ○ OPC

OP Value: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

SQN Configuration

☑ SQN Check
☑ Age Limit
☑ Max Delta Check
☑ Conceal SQN in AUTN
☑ Conceal SQN in AUTS

IND Field Size: 5

Offset to SQNms: 0

Max Delta: 0000FFFFFFFE

Age Limit: 0000FFFFFFFE

Milenage Constants Configuration

|  |  |  |  |
|---|---|---|---|
| C1 | 00000000000000000000000000000000 | R1 | 40 |
| C2 | 00000000000000000000000000000001 | R2 | 00 |
| C3 | 00000000000000000000000000000002 | R3 | 20 |
| C4 | 00000000000000000000000000000004 | R4 | 40 |
| C5 | 00000000000000000000000000000008 | R5 | 60 |

Note
All values should be entered in hexadecimal

Figure 29: Network Configuration View

## 5.4 Dynamic Data

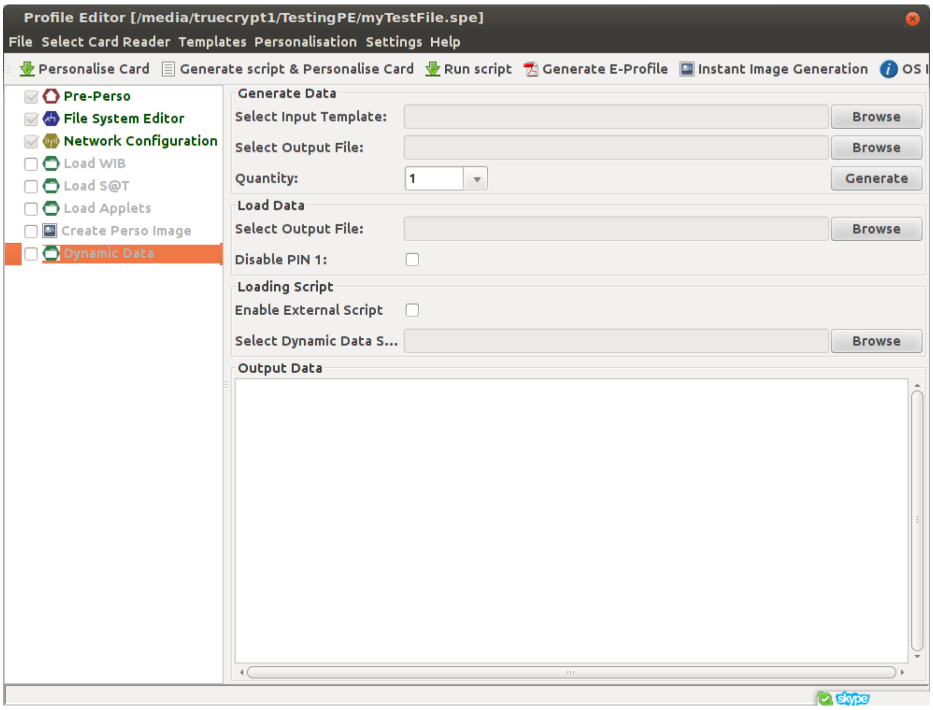
To access the Dynamic Data module, first click on the *Dynamic Data* item in the tree that's located at the left hand-side of the tool. This will display the window in Figure 30.

In order to input data and constraints into the Dynamic Data Module, an XML file of a specific format must be submitted to the Application. Should this format not be adhered to, an error message explaining that the format is invalid will be displayed.

This data is then used to generate new values, as well as save values that the user has specified. Therefore it is very important that the user enters the correct information in the initial XML document, or errors could occur elsewhere in the tool.

### 5.4.1 Generating Data

As can be seen from Figure 30, when the user first loads the interface they have the choice to select an input XML file. This XML file is used to read in constraints on the data to be generated, such as whether to randomise the value based on the length, and format of tag attributes. The user can select this file using the *Browse* button. Should the user not select an XML file, they are notified with an error message box when they try to generate data. The same constraints apply to the *Select Output File* field.



Profile Editor [/media/truecrypt1/TestingPE/myTestFile.spe]

Generate Data

Select Input Template:

Select Output File:

Quantity: 1

Load Data

Select Output File:

Disable PIN 1: □

Loading Script

Enable External Script □

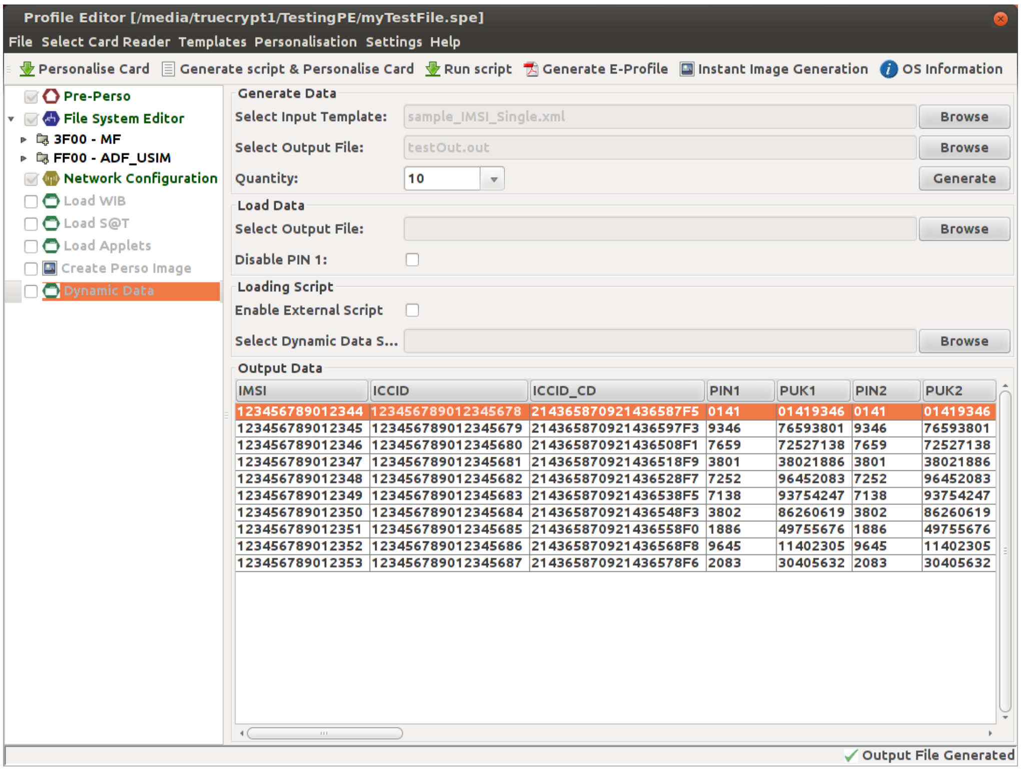
Select Dynamic Data S...

Figure 30: Dynamic Data View

The quantity field allows the user to specify how many data sets they wish to create. The user is limited to generating twenty data sets at any one time. To change the quantity, simply select the combo-box and choose the required number. The user can disable PIN1 by selecting the 'Disable PIN1' check-box. Pressing the 'Generate Data' button then reads in the input XML file, performs all of the constraint checks, and generates the amount of data sets specified in the quantity field. These data sets are displayed in a dynamic table in the bottom panel, as shown in Figure 31. The output order in this table is specified by the output tag read in from the input XML file, this is explained in detail in Appendix A, section 9.

### 5.4.2 Loading Data

The *Load Data* pane allows the user to load in a .OUT file that has been previously generated. To do this, press the *Browse* button in the *Load Data* panel, browse for the file and select it. Once 'open' is pressed, the



Generate Data

Select Input Template: sample_IMSI_Single.xml

Select Output File: testOut.out

Quantity: 10

Load Data

Select Output File:

Disable PIN 1: □

Loading Script

Enable External Script □

Select Dynamic Data S...

Output Data

| IMSI | ICCID | ICCID_CD | PIN1 | PUK1 | PIN2 | PUK2 |
|---|---|---|---|---|---|---|
| 123456789012344 | 123456789012345678 | 214365870921436587F5 | 0141 | 01419346 | 0141 | 01419346 |
| 123456789012345 | 123456789012345679 | 214365870921436597F3 | 9346 | 76593801 | 9346 | 76593801 |
| 123456789012346 | 123456789012345680 | 214365870921436508F1 | 7659 | 72527138 | 7659 | 72527138 |
| 123456789012347 | 123456789012345681 | 214365870921436518F9 | 3801 | 38021886 | 3801 | 38021886 |
| 123456789012348 | 123456789012345682 | 214365870921436528F7 | 7252 | 96452083 | 7252 | 96452083 |
| 123456789012349 | 123456789012345683 | 214365870921436538F5 | 7138 | 93754247 | 7138 | 93754247 |
| 123456789012350 | 123456789012345684 | 214365870921436548F3 | 3802 | 86260619 | 3802 | 86260619 |
| 123456789012351 | 123456789012345685 | 214365870921436558F0 | 1886 | 49755676 | 1886 | 49755676 |
| 123456789012352 | 123456789012345686 | 214365870921436568F8 | 9645 | 11402305 | 9645 | 11402305 |
| 123456789012353 | 123456789012345687 | 214365870921436578F6 | 2083 | 30405632 | 2083 | 30405632 |

✓ Output File Generated

Figure 31: Dynamic Data Loaded Into Tool

table is loaded into the *Output Data* pane. This allows the viewing of data that has been generated, and also the selection of the data to send to the card.

In order to add an OPC Value into Dynamic Data and therefore Network Configuration, the OPC value must be manually added to the .OUT file for each row in the .OUT file.

### 5.4.3 Generating Scripts

Once data is generated or loaded the user can select a row which they would like to send to the SIM card. First of all, the user must select *Dynamic Data* in the modules tree on the left-hand side of the tool. They must then select data to send by clicking on the row, and pressing 'Generate Scripts & Personalise Card' in the top left of the window. If no row is selected, the module will default to generate scripts for the first row in the table.

### 5.4.4 External Scripts

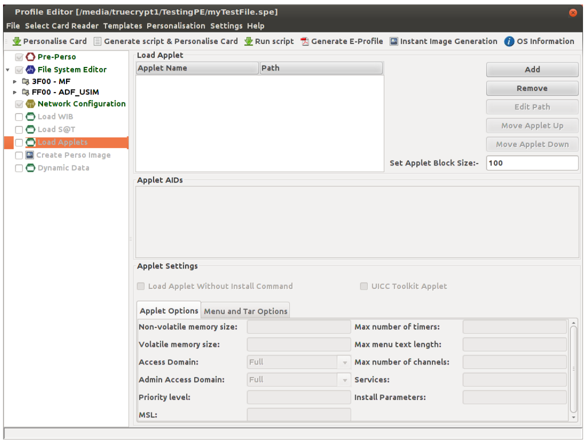
The user has the option to use an external script to set the dynamic data values. To do this, the user must select the *Enable External Script* check-box, and then click the *Browse* button in the *Loading Script* pane. This pane can be seen in the middle of Figure 31. This will bring up a file chooser so that the user can select the script they wish to use. Once this script has been selected, the dynamic data values will be set from the script when the user is personalising the card. The external script is checked for errors upon their load into the tool. If an error is found then the script will not be loaded into the tool to save errors in personalisation.

### 5.4.5 Clearing the Data

The user can clear the data loaded in the Dynamic Data module. By right-clicking the *Dynamic Data* module on the left-hand side of the tool, the user will see the option to *Clear Data*. After this option is selected, all data will be removed from this module.

## 5.5 Applet Loading

To access the Load Applet module, first click on the *Load Applet Module* item in the tree that's located at the left hand-side of the tool. This will display the window in Figure 32.



Profile Editor [/media/truecrypt1/TestingPE/myTestFile.spe]

Set Applet Block Size:- 100

Access Domain: Full

Admin Access Domain: Full

Figure 32: Loading Applet View

### 5.5.1 Adding an Applet

To add an applet ready to be loaded onto a card, you should first click the *Add* button as seen in Figure 32. You should then select the required applet. The applet you selected will then be added into the *Load Applet* list, and will be ready for removing, editing, or adding to the card during personalisation.

It is important to note here that there are two different types of applet that may be added to be personalised. These are both java (Default OS in pre-perso) and native (SimbaN OS in pre-perso). In Profile Editor, Java applets can be loaded onto the card. However, Simba applets must be pre-loaded and Profile Editor will only attempt to install these already loaded applets. As can be seen when the add button is selected, there is a limited amount of pre-supported SimbaN applets available via profile editor. These can be selected with a check box if this OS is selected. Should a custom applet require installing onto SimbaN OS, the *Install Loaded Applet* button shall be used as explained in section 5.5.2.

Once a Java applet (Default OS in per-perso module) has been added, the user can change the path to that applet without having to remove and re-add the applet. To do this, the *Edit Path* button is to be used. This button can be seen on the right-hand side of Figure 32.

### 5.5.2 Adding a Loaded Applet

In certain circumstances, such as for native cards, it may be useful to install an applet that is already loaded onto the card. Profile Editor provides this functionality in the form of the *Install Loaded Applet* button. To add an already loaded applet to be installed during personalisation, press the *Install Loaded Applet* button. This will display another window where the relavent information (Name, Package AID, Applet AID, Instance AID) can be entered. Here, only the Package AID is mandatory to enter.

Once the correct information is entered, the user should press the *OK* button to add the applet to the list. Once added, it can be seen that this applet has no path, and it is not possible to select this applet to be loaded as SAT. These are two features which are not accessible for pre-loaded applets.

### 5.5.3 Removing an Applet

To remove an applet so that it will not be added to the card during personalisation, you should click on the unwanted applet in the *Load Applet* list and then select the *Remove* button as seen in Figure 32. The applet will then be removed from the *Loaded Applets* list and will not be added to the card during personalisation.

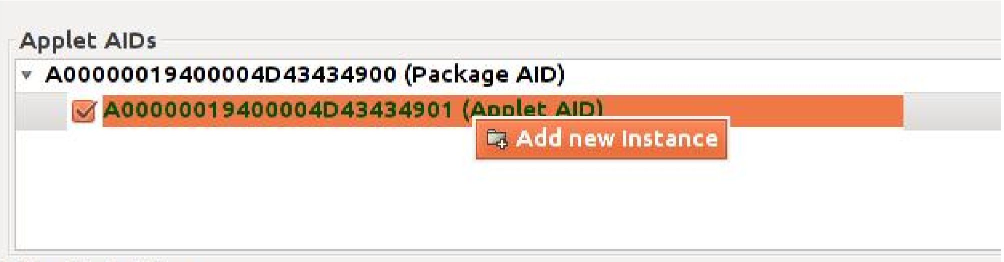
### 5.5.4 Applet AIDs

After adding an applet into the Module, the user will be able to find the Package AID and the Applet AIDs in the *Applet AIDs* panel shown in the middle of Figure 32.

The panel allows the user to add and remove instances. This will be explained in Sections 5.5.4.1 and 5.5.4.2.

This panel also allows the user to select which applets they require to be included in the card personalisation. This is done by selecting the check-boxes by the Applet AID or the Instance AID which are required, as seen in Figure 33. If an applet has a selected instance then this instance will be included during the personalisation.

**5.5.4.1 Adding an Instance** The user can add an instance by right clicking on an applet AID and selecting add instance. This can be seen in Figure 33. This will bring up a message box allowing the user to enter the required Instance AID. This must be a hex value, given in bytes, of length 5-16 bytes.



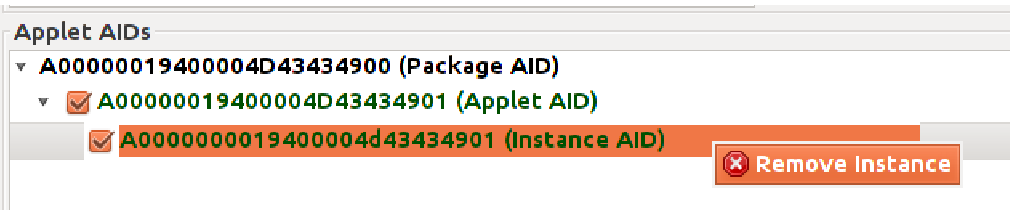
Applet AIDs

A00000019400004D43434900 (Package AID)

☑ A00000019400004D43434901 (Applet AID)

Figure 33: Adding an Instance to an Applet

**5.5.4.2 Removing an Instance** The user can remove an instance by right clicking on an Instance AID and selecting remove instance. This can be seen in Figure 34.



Applet AIDs

A00000019400004D43434900 (Package AID)

☑ A00000019400004D43434901 (Applet AID)

☑ A0000000019400004d43434901 (Instance AID)

Figure 34: Removing an Instance from an Applet

### 5.5.5 Editing an Applet

Applet and Instance settings can be edited by using the forms at the bottom of Figure 32 and Figure 35. These settings will be auto-saved when the user navigates away from the settings panel.

**5.5.5.1 Applet Options**

Table 13 shows what data the options menu tab stores, as well as the default values and required length of these options.

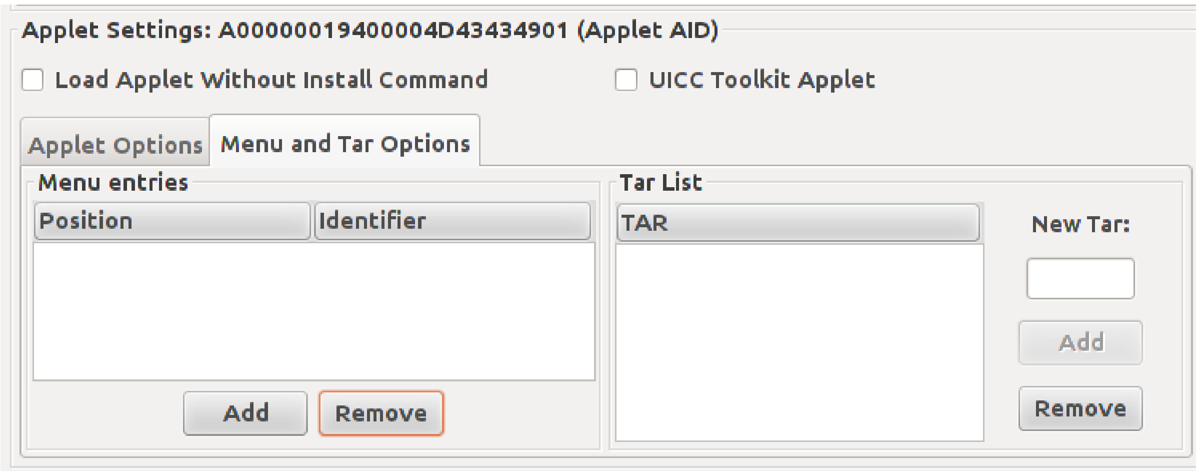
The values in Table 13 are padded to the required length if the entered data is not sufficient.

These settings can be found by clicking the *Applet Options* tab as shown at the bottom of Figure 32.

**5.5.5.2 Applet Menu Item/Tar List**

The menu item/tar list tab stores the menu entries, and the tar list. The menu items are split into two columns, position and identifier. Both of these values must be at least 1 byte long. If incorrect characters are entered in the data, then the user is asked whether they want to manually edit the data or if they want it to be edited automatically.

The default menu item data is shown in Table 14



Applet Settings: A00000019400004D43434901 (Applet AID)

□ Load Applet Without Install Command □ UICC Toolkit Applet

Figure 35: Applet Setting Menu and Tar Options Panel

Table 13: Applet Options Default Values

| Applet Option | Default Data | Required Length (in bytes) |
|---|---|---|
| Non-Volatile Memory Size | 00 | 1 |
| Max Number of Timers | 00 | 1 |
| Volatile Memory Size | 00 | 1 |
| Max Menu Text Length | 16 | 1 |
| Access Domain | FULL | - |
| Max Number of Channels | | 1 |
| Priority Level | 00 | 1 |
| Install Parameters | | |
| MSL | 00 | 1 |

Table 14: Default Applet Menu Values

| Position | Identifier |
|---|---|
| 00 | 00 |

The tar list stores a list of the tars. These must be a sequence of hex characters which are 3 bytes long. The user will not be able to add the new tar until this criteria is met. The default tar list is empty.

These settings can be found by clicking the *Menu and Tar* Options tab as shown at the bottom of Figure 35.

### 5.5.6 Changing the Order of Loaded Applets

It is possible to change the order of the applets you have loaded into this module ready to be added to the card during personalisation. To do this, you need to select an applet, as shown in Figure 36. If the applet selected can be moved, then clicking the applet will enable a button to move the applet up, and a button to move the applet down. These buttons can be seen in the top right hand corner of Figure 36.

### 5.5.7 Changing the Applet Block Size

The user has the option to edit the applet block size. The maximum size of the blocks is 255, with a default value of 100. The text-box to edit this value can be seen on the right-hand side of Figure 36.

### 5.5.8 Export and Import Applet And Settings

The user may want to have the applet settings in a text document. This is made possible by the tool through the *Export Applet and Settings* and the *Import Applet and Settings* buttons. This will produce a text file which you can edit manually. This text file must follow a specified format in order to be imported back into the tool. The following is a list of available tags:

- Package Name
- Package Path
- Package AID
- Package Load As S@T
- Applet Aid
- Applet Load And Install
- Applet Selectable
- Applet Install Params
- Applet Max Menu Text Length
- Applet Max Num Channels
- Applet Max Num Timers
- Applet Msl
- Applet Non Vol Mem Size
- Applet Priority Level
- Applet Vol Mem Size
- Applet Access
- Applet UICC
- Applet Admin Access
- Applet Services

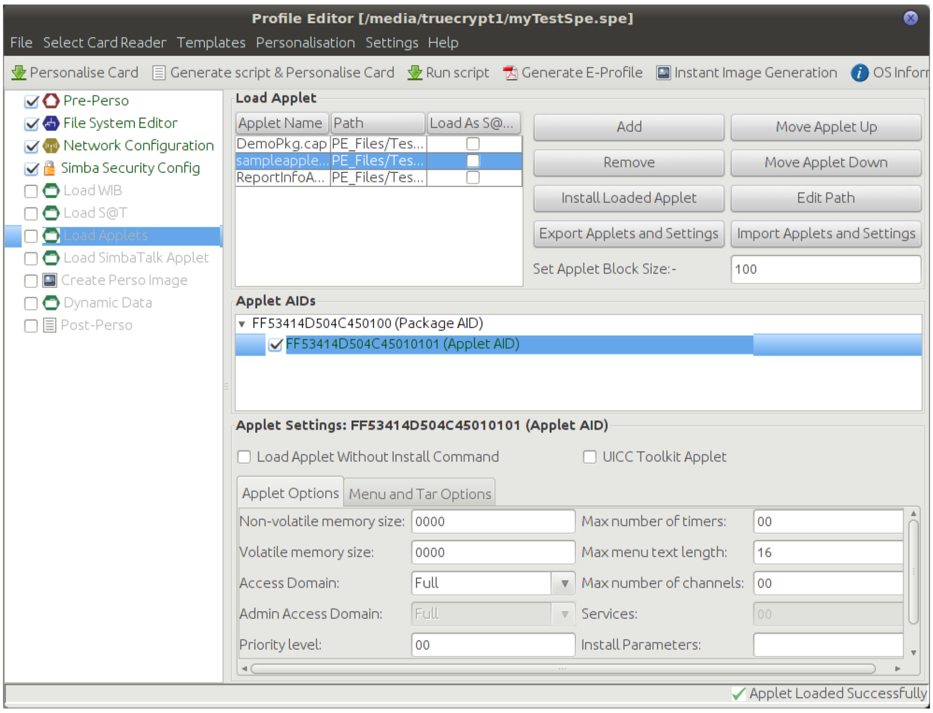
The '@' symbol must be used to specify the end of a line. There must be only one tag per line.

### 5.5.9 Copy and Paste Applet And Settings

To copy an applet, the user has two options. It is possible to right click the required applet and select *copy*. This will copy the applet and its settings to the clipboard. This can also be done by selecting the applet and pressing 'ctrl+C'. Once the applet is copied, it is then possible to right click the applet AID table and selecting the *paste* button. If the applet is valid then it will be pasted into the applet table, and the settings will be updated to correspond to the copied applet. This is particularly useful across multiple instances of Profile Editor.

### 5.5.10 Loading an Applet as S@T

Java applets that are to be loaded and installed onto the card via Profile Editor can be loaded onto the card as SAT applets. To toggle this, check and un-check the check box in the applet table for the required applets row. This functionality is not available for applets that are already loaded onto the card, such as Simba applets or applets added via the *Install Loaded Applets* button.



| Applet Name | Path | Load As S@... |
|---|---|---|
| DemoPkg.cap | PE_Files/Tes... | □ |
| sampleapple... | PE_Files/Tes... | □ |
| ReportInfoA... | PE_Files/Tes... | □ |

Set Applet Block Size:- 100

▾ FF53414D504C450100 (Package AID)
☑ FF53414D504C45010101 (Applet AID)

Applet Settings: FF53414D504C45010101 (Applet AID)

□ Load Applet Without Install Command □ UICC Toolkit Applet

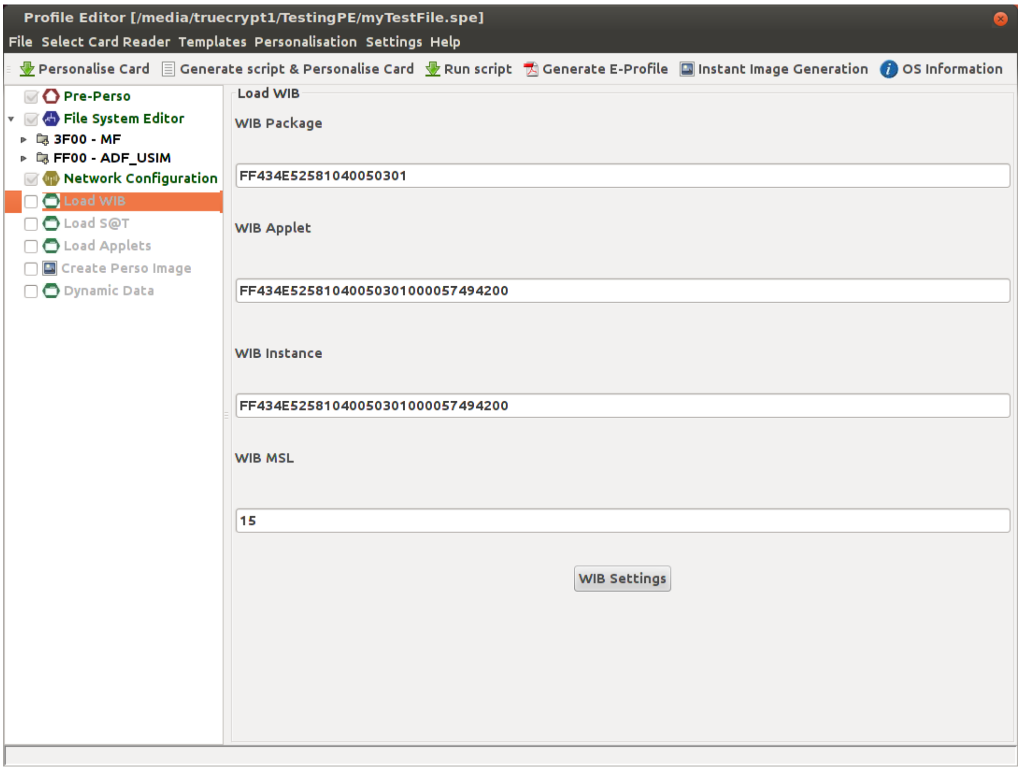
| | | | |
|---|---|---|---|
| Non-volatile memory size: | 0000 | Max number of timers: | 00 |
| Volatile memory size: | 0000 | Max menu text length: | 16 |
| Access Domain: | Full | Max number of channels: | 00 |
| Admin Access Domain: | Full | Services: | 00 |
| Priority level: | 00 | Install Parameters: | |

✓ Applet Loaded Successfully

Figure 36: Multiple Applets Loaded Into The Tool

## 5.6 WIB Loading

To access the Loading WIB module, first click on the *Loading WIB Module* item in the tree that's located at the left hand-side of the tool. This will bring up the window as seen in Figure 37.



Load WIB

WIB Package

FF434E52581040050301

WIB Applet

FF434E5258104005030100005749420O

WIB Instance

FF434E5258104005030100005749420O

WIB MSL

15

Figure 37: Loading WIB View

### 5.6.1 WIB

Once in the Loading WIB Module, you can see and edit the WIB values. Table 15 shows the default WIB data, and the required field length.

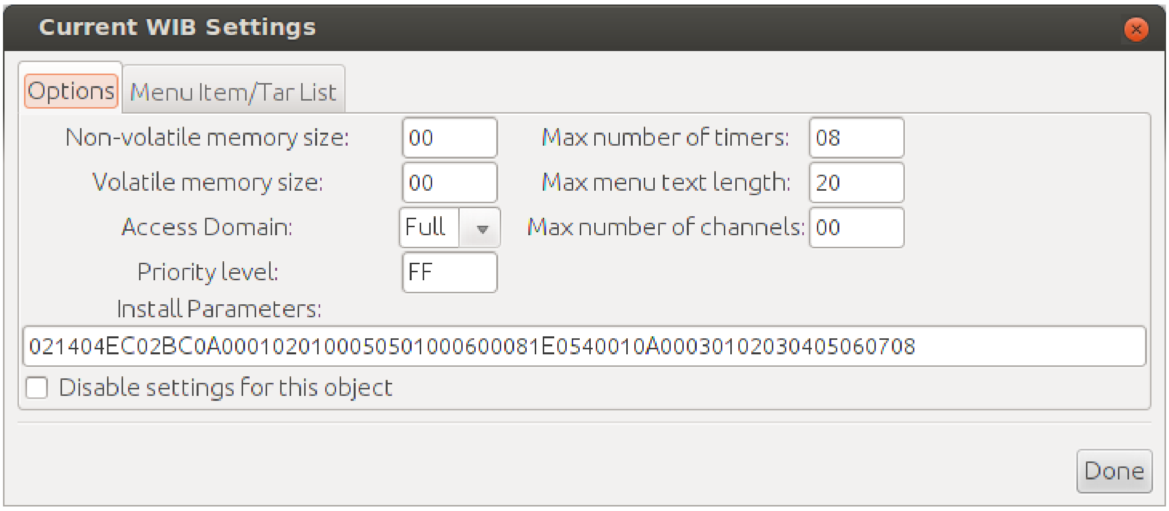
Table 15: Default WIB Values

| WIB Setting: | Default Value | Required Length (In Bytes) |
|---|---|---|
| WIB Package | FF434E52581040050301 | 10 |
| WIB Applet | FF434E52581040050301000057494200 | 16 |
| WIB Instance | FF434E52581040050301000057494200 | 16 |
| WIB MSL | 15 | 1 |

If the required length is not met, then the entered data will be padded or cut until it meets the requirements. If data is entered which is not in hexadecimal format or which has a space in, then these illegal values are removed from the data and the data is then cut or padded to the correct length.

### 5.6.2 WIB Settings

Further WIB Settings can be found by clicking the *WIB Settings* button shown in Figure 37. The pop-up shown in Figure 38 will then be visible. This window holds the options menu, as well as the menu item/tar list.



Non-volatile memory size: 00
Volatile memory size: 00
Access Domain: Full
Priority level: FF
Max number of timers: 08
Max menu text length: 20
Max number of channels: 00
Install Parameters:
021404EC02BC0A0001020100050501000600081E0540010A00030102030405060708
□ Disable settings for this object

Figure 38: WIB Options

#### 5.6.2.1 WIB Options

Table 16 shows what data the options menu tab stores, as well as the default values and required length of these options:

Table 16: Default WIB Option Values

| WIB Option: | Default Data | Required Length (In Bytes) |
|---|---|---|
| Non-Volatile Memory Size | 00 | 1 |
| Max Number of Timers | 08 | 1 |
| Volatile Memory Size | 00 | 1 |
| Max Menu Text Length | 20 | 1 |
| Access Domain | FULL | - |
| Max Number of Channels | 00 | 1 |
| Priority Level | FF | 1 |
| Install Parameters | 021404EC02BC0A00010201000505010006<br>00081E0540010A00030102030405060708 | - |

The values in Table 16 are also cut or padded to the required length if the entered data is not sufficient.

#### 5.6.2.2 WIB Menu Item/Tar List

The menu item/tar list tab stores the menu entries, and the Tar list. The menu items are split into two columns, position and identifier. Both of these values must be at least 1 byte long. If incorrect characters are entered in the data, then the user is asked whether they want to manually edit the data or if they want it to be edited automatically. The user cannot move away from this screen unless the data entered is a Hex character of minimum length 1 byte.

The default menu item data is shown in Table 17

The tar list stores a list of the tars. These must be a sequence of hex characters which are 3 bytes long. The user will not be able to add the new tar until this criteria is met.

The default tar list is shown in Table 18

### 5.6.3 Resetting WIB Values to Default

It is easily possible to reset all of the WIB settings mentioned in this section to their default values. To do this, navigate to the Loading WIB Module and right-click on this module header. A pop-up menu will now appear

Table 17: Default WIB Menu Values

| Position | Identifier |
|---|---|
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |

Table 18: Default WIB Tar Values

| Tar |
|---|
| BFFF01 |
| BFFF02 |
| BFFF03 |
| 57002F |

with the option *Reset WIB Configuration*, which when clicked will reset all values immediately. This can be seen in Figure 39.

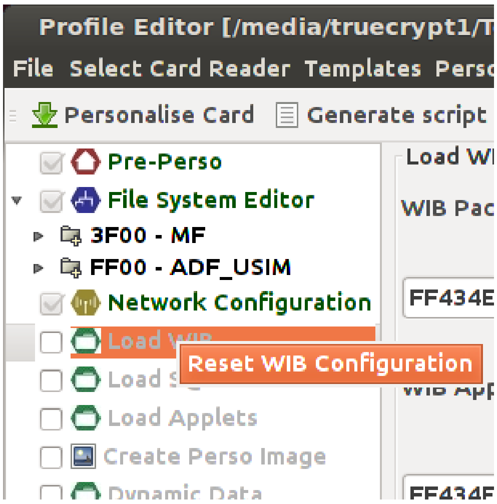


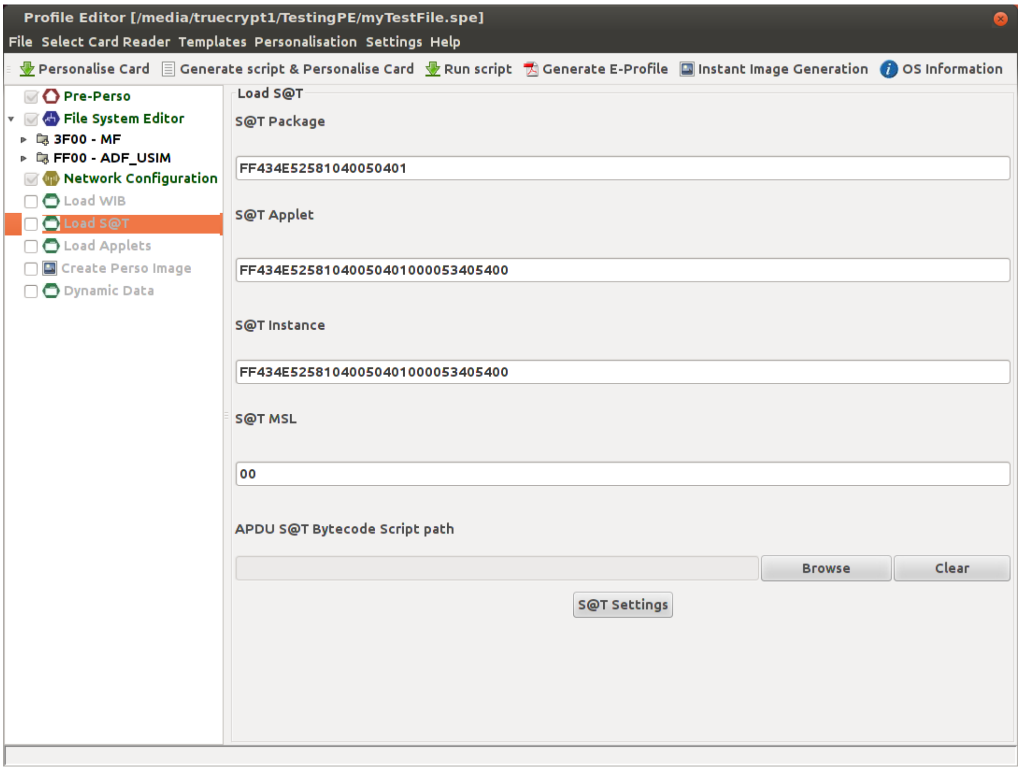
Figure 39: Loading WIB Right Click Options

### 5.6.4 Using the WIB Template

A WIB template has been provided to assist the user with their WIB profile creation. This can be found by clicking the *Templates* menu within the top menu bar. The user will then see the option to *Add WIB (Add to SIM). By selecting this option the WIB template will be added to the users file system.*

## 5.7 S@T Loading

To access the Loading S@T module, first click on the *Load S@T Module* item in the tree that's located at the left hand-side of the tool. This will bring up the window as seen in Figure 40.



Load S@T

S@T Package

FF434E52581040050401

S@T Applet

FF434E52581040050401000053405400

S@T Instance

FF434E52581040050401000053405400

S@T MSL

00

APDU S@T Bytecode Script path

Figure 40: Loading SAT View

### 5.7.1 S@T

Once in the Loading S@T Module, you can see and edit the S@T values. A table showing the default S@T data, and the required field length is shown in Table 19

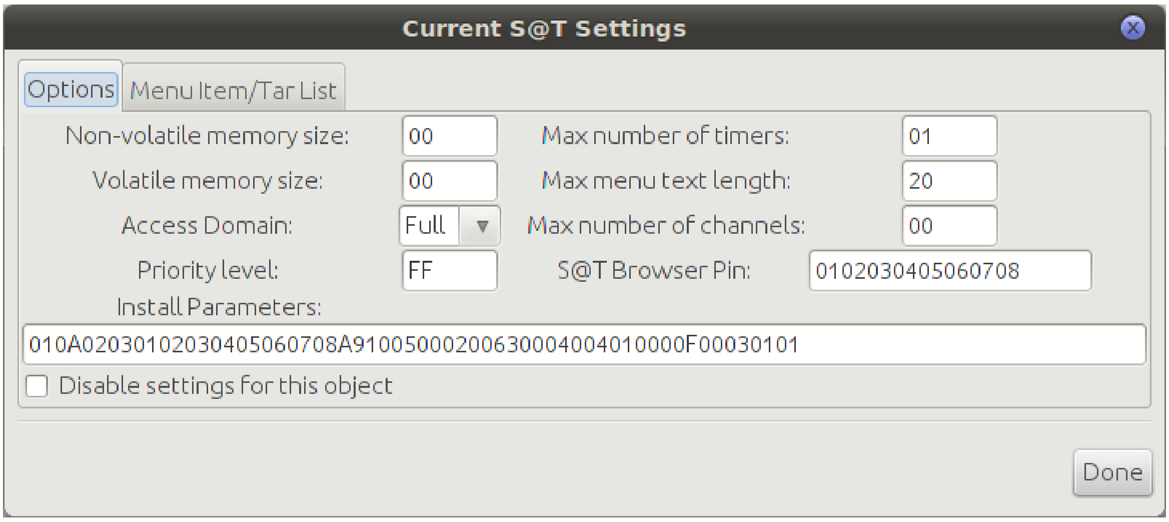
Table 19: Default S@T Values

| S@T Setting: | Default Value | Required Length (In Bytes) |
|---|---|---|
| S@T Package | FF434E52581040050401 | 10 |
| S@T Applet | FF434E52581040050401000053405400 | 16 |
| S@T Instance | FF434E52581040050401000053405400 | 16 |
| S@T MSL | 00 | 1 |

If the required length is not met, then the entered data will be padded or cut until it meets the requirements. If data is entered which is not in hexadecimal format or which has a space in, then these illegal values are removed from the data and the data is then cut or padded to the correct length.

### 5.7.2 S@T Settings

Further S@T Settings can be found by clicking the S@T *Settings* button shown in Figure 40. The pop-up shown in Figure 41 will then be visible. This window holds the options menu, as well as the menu item/tar list.



Non-volatile memory size: 00
Volatile memory size: 00
Access Domain: Full
Priority level: FF
Max number of timers: 01
Max menu text length: 20
Max number of channels: 00
S@T Browser Pin: 0102030405060708
Install Parameters:
010A02030102030405060708A91005000200630004004010000F00030101
□ Disable settings for this object

Figure 41: Loading SAT Settings Panel

#### 5.7.2.1 S@T Options

The below table shows what data the options menu tab stores, as well as the default values and required length of these options:

Table 20: Default S@T Option Values

| S@T Option: | Default Data | Required Length (In Bytes) |
|---|---|---|
| Non-Volatile Memory Size | 00 | 1 |
| Max Number of Timers | 20 | 1 |
| Volatile Memory Size | 00 | 1 |
| Max Menu Text Length | 20 | 1 |
| Access Domain | FULL | - |
| Max Number of Channels | 00 | 1 |
| Priority Level | FF | 1 |
| S@T Browser Pin | 0102030405060708 | 8 |
| Install Parameters | 010A02030102030405060708A9100<br>5000200630004004010000F00030101 | - |

The values shown in Table 20 are also cut or padded to the required length if the entered data is not sufficient.

#### 5.7.2.2 S@T Menu Item/Tar List

The menu item/tar list tab stores the menu entries, and the tar list. The menu items are split into two columns, position and identifier. Both of these values must be at least 1 byte long. If incorrect characters are entered in the data, then the user is asked whether they want to manually edit the data or if they want it to be edited automatically. The user cannot move away from this screen unless the data entered is a hexadecimal character of minimum length 1 byte.

The default menu item data is shown in Table 21

The tar list stores a list of the tars. These must be a sequence of hex characters which are 3 bytes long. The user will not be able to add the new tar until this criteria is met.

The default tar list is shown in Table 22

Table 21: Default S@T Menu Values

| Position | Identifier |
|---|---|
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |
| 00 | 00 |

Table 22: Default S@T Tar Values

| Tar |
|---|
| 534054 |
| 505348 |

### 5.7.3 Resetting S@T Values to Default

It is easily possible to reset all of the S@T settings mentioned in this section to their default values. To do this, navigate to the Loading S@T Module and right-click on this module header. A pop-up menu will now appear with the option *Reset S@T Configuration,* which when clicked will reset all values immediately. This can be seen in Figure 42.

### 5.7.4 Using the S@T Template

A S@T template has been provided to assist the user with their S@T profile creation. This can be found by clicking the *Templates* menu within the top menu bar. The user will then see the option to *Add S@T (Add to SIM).* By selecting this option the S@T template will be added to the users file system.

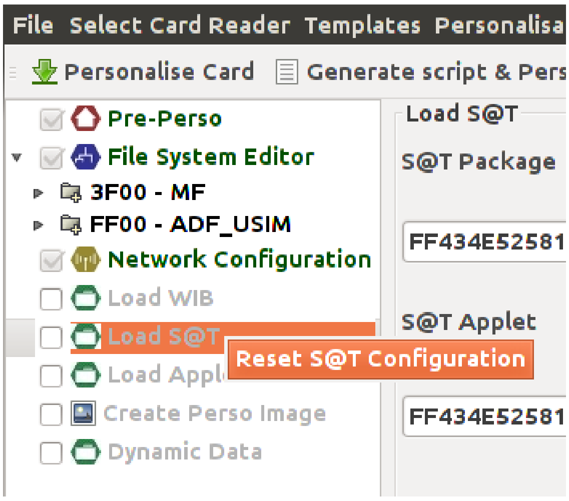


Figure 42: Loading S@T Right Click Options

## 5.8 Create Perso Image

To create a perso image when you personalise a card, you need to first navigate to the *Create Perso Image* item in the tree that's located at the left hand-side of the tool. If you select this then there will be a tick next to the *Create Perso Image* item in the tree. This means that it will be included when the card is next personalised. If there is no tick here, that means that a personalised image will not be created when the card is next personalised.

Once you have selected this module, you must select a path to the required output file. You do this by clicking the *Browse* button as seen in Figure 43. This will bring up a pop-up window where you can select the chosen output file. If no path is selected then Create Perso Image will not be included in the personalisation.

### 5.8.1 Instant Perso Image

You have the option to instantly create a perso image. To do this, you have to navigate to the *Instant Image Generation* button, as seen in Figure 44. This will create an image of the personalisation already loaded onto the inserted card.

### 5.8.2 Clearing the Data

The user can clear the data loaded in the *Create Perso Image* module. By right-clicking the *Create Perso Image* module on the left-hand side of the tool, the user will see the option to *Clear Image Path*. After this option is selected, all data will be removed from this module.

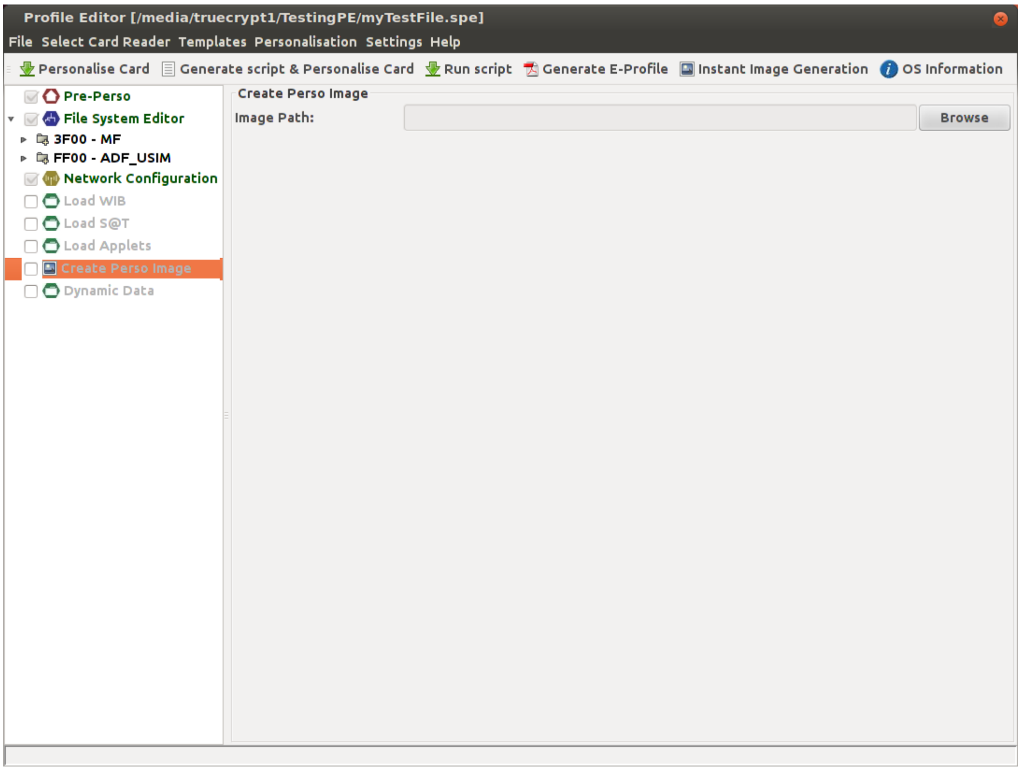


Figure 43: Create Perso Image View

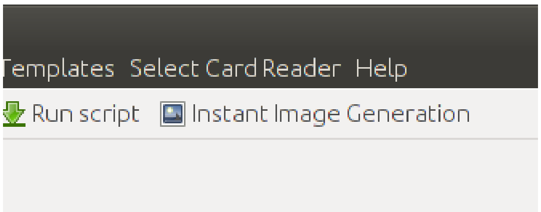


Figure 44: Instant Image Generation Button

## 5.9 Load SimbaTalk Applet

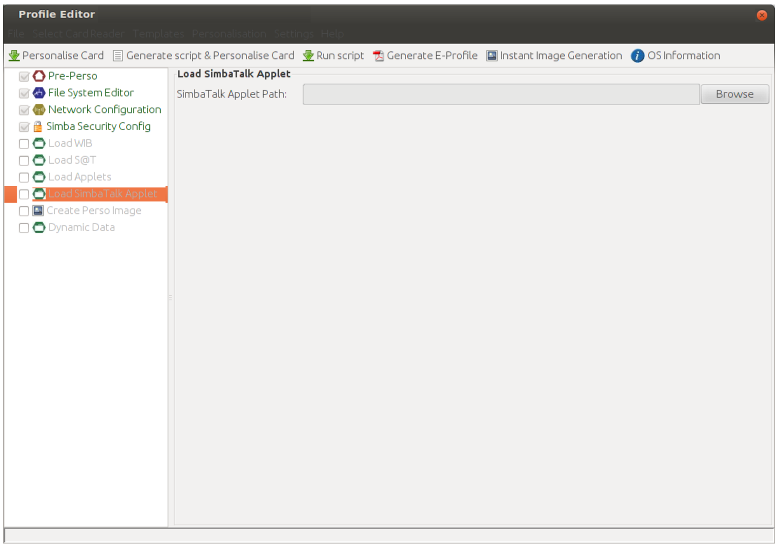
The *Load SimbaTalk Applet* module is only available when the Simba OS is selected.

To load a SimbaTalk applet when you personalise a card, you need to first navigate to the *Load SimbaTalk Applet* item in the tree that's located at the left hand-side of the tool. If you select this then there will be a tick next to the *Load SimbaTalk Applet* item in the tree. This means that it will be included when the card is next personalised. If there is no tick here, that means that no SimbaTalk applets will be loaded when the card is next personalised.

Once you have selected this module, you must select a path to the required SimbaTalk applet file. The chosen file should be a *.ldr* file. You do this by clicking the *Browse* button as seen in Figure . This will bring up a pop-up window where you can select the chosen applet file. If no path is selected then Load SimbaTalk Applet will not be included in the personalisation.

### 5.9.1 Clearing the Data

The user can clear the data loaded in the *Load SimbaTalk Applet* module. By right-clicking the *Load SimbaTalk Applet* module on the left-hand side of the tool, the user will see the option to *Clear SimbaTalk Applet Path.* After this option is selected, all data will be removed from this module.



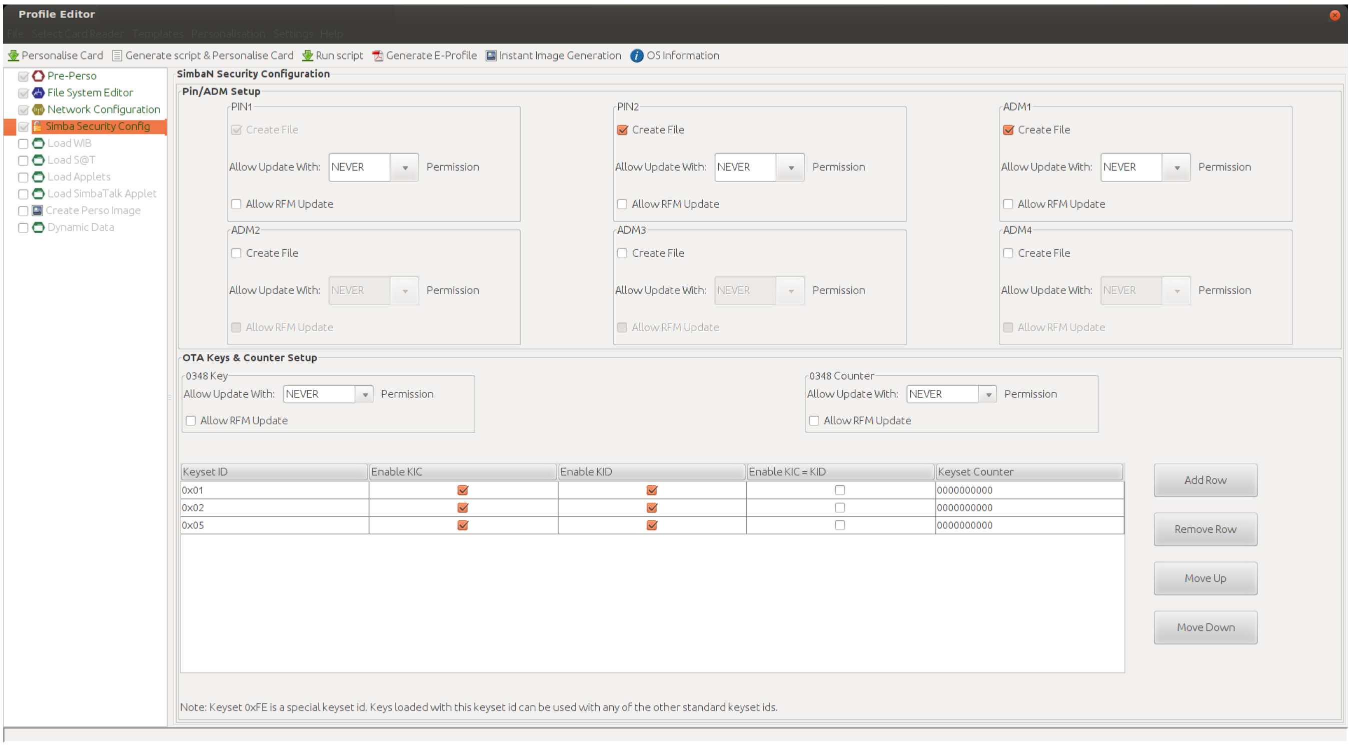
Load SimbaTalk Applet

SimbaTalk Applet Path:

Figure 45: SimbaTalk Applet View

## 5.10 Simba Security Configuration

To access the Simba Security Configuration module, first click on the *Simba Security Configuration* item in the tree that's located at the left hand-side of the tool. This will bring up the window as seen in Figure 46.



SimbaN Security Configuration

Pin/ADM Setup

PIN1
☑ Create File
Allow Update With: NEVER Permission
☐ Allow RFM Update

PIN2
☑ Create File
Allow Update With: NEVER Permission
☐ Allow RFM Update

ADM1
☑ Create File
Allow Update With: NEVER Permission
☐ Allow RFM Update

ADM2
☐ Create File
Allow Update With: NEVER Permission
☐ Allow RFM Update

ADM3
☐ Create File
Allow Update With: NEVER Permission
☐ Allow RFM Update

ADM4
☐ Create File
Allow Update With: NEVER Permission
☐ Allow RFM Update

OTA Keys & Counter Setup

0348 Key
Allow Update With: NEVER Permission
☐ Allow RFM Update

0348 Counter
Allow Update With: NEVER Permission
☐ Allow RFM Update

| Keyset ID | Enable KIC | Enable KID | Enable KIC = KID | Keyset Counter |
|---|---|---|---|---|
| 0x01 | ☑ | ☑ | ☐ | 0000000000 |
| 0x02 | ☑ | ☑ | ☐ | 0000000000 |
| 0x05 | ☑ | ☑ | ☐ | 0000000000 |

Note: Keyset 0xFE is a special keyset id. Keys loaded with this keyset id can be used with any of the other standard keyset ids.

Figure 46: Simba Security Configuration View

### 5.10.1 PIN/ADM Setup

The PIN/ADM Setup panel can be found at the top of the *Simba Security Configuration* module. This stores the 6 available pins and allows the user to change the settings for these pins. The pins are listed below:

- PIN1
- PIN2
- ADM1
- ADM2
- ADM3
- ADM4

The user can select which of these pins to create. If the pins are not selected to be created here then they will not be created during the personalisation. The user can set the access condition for file updating, and can also set whether RFM updates to the file is allowed.

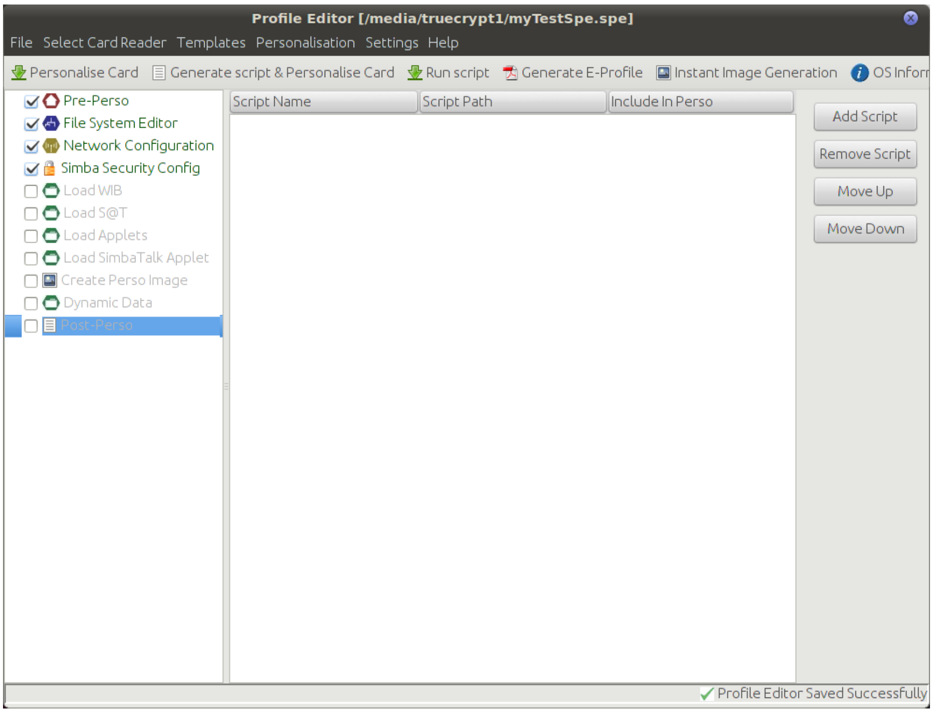
### 5.10.2 OTA Keys & Counter Setup

The OTA Keys and Counter Setup panel can be found at the bottom of the *Simba Security Configuration* module. Here you can change the settings of the 0348 Key and Counter files. These files are automatically created by the tool during personalisation. The user can set the access condition for file updating, and can also set whether RFM updates to the file is allowed.

This panel also includes the keys table. The keys table allows the user to set up the required keys, the tool then automatically creates the key and counter files appropriately. The table has 5 columns; *Keyset ID*, *Enable KIC*, *Enable KID*, *Enable KIC = KID*, *Keyset Counter*.

## 5.10 Post-Perso Module

To access the Post-Perso module, first click on the *Post-Perso* item in the tree that's located at the left hand-side of the tool. This will bring up the window as seen in Figure 47.



✓ Profile Editor Saved Successfully

Figure 47: Post Perso View

To include a script to run after personalisation, this module will need to be checked and the required scripts will need to have the *Include In Perso* check-box in its row checked.

### 5.11.1 Adding a Script

To add a script into the tool, first the user must select the *Add Script* button. This will bring up a chooser where the user must select the required file. This file will then be automatically checked by the tool for errors. If errors are found then the script will not be added into the tool. If the script passes the error checking then it will be loaded into the table.

### 5.11.2 Removing a Script

To remove a script from the tool, the user must first select the applet they wish to remove. Once the applet is selected, the user should press the *Remove Script* button. This will remove the script from the table.

### 5.11.3 Ordering Scripts

The order that the scripts are listed in the table is the order that they will run at personalisation time. The user can change the order of the scripts in the table by selecting either the *Move Up* or *Move Down* buttons.

# 6 Other Features of the Tool

In this section we'll explain all of the different functionality of the tool that is not contained in any of the modules listed in Section 5.

## 6.1 Generate E-Profile.

The user has multiple options if they wish to generate an e-profile using the Profile Editor tool. The first of these options is to use the *Generate E-Profile* button shown in Figure 48. This will create a PDF document showing exactly what is contained within the current profile.

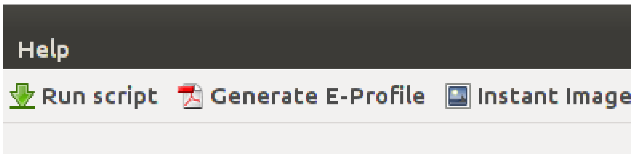
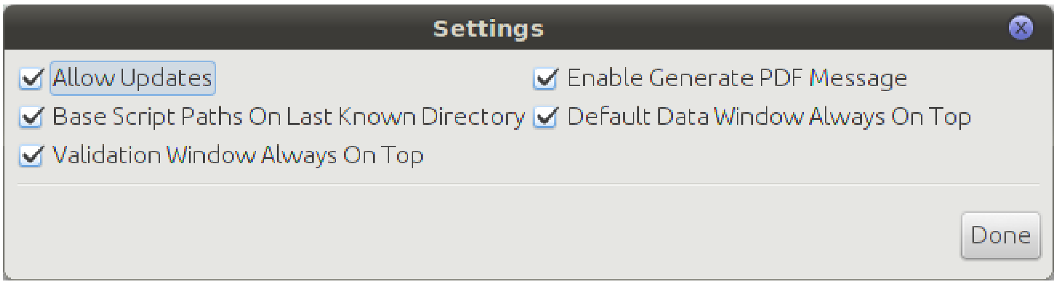


Figure 48: Generate E-Profile Button

Another way to generate an e-profile is to personalise to the card, once the personalisation has finished the user will be asked if they wish to generate a PDF e-profile of the personalised card. If the user selects the yes button then the PDF will be created in the directory chosen by the user.

## 6.2 Updating the Tool

The tool will always check for updates when it is opened unless the user specifies otherwise in the Preferences menu. To change this setting, the user must navigate to the *Settings* menu. They must then select the *Preferences* menu item. This will bring up the menu shown in Figure 49. If the *Allow Updates* check-box is selected then the tool will always check for updates at start-up.


Settings

☑ Allow Updates
☑ Enable Generate PDF Message
☑ Base Script Paths On Last Known Directory
☑ Default Data Window Always On Top
☑ Validation Window Always On Top

Figure 49: Settings Menu Allowing The User To Toggle Update Settings

The user can also manually check for updates from within the tool. To do this the user must navigate to the *Help* menu. From here they must select the *Check For Updates* option as seen in Figure 50.

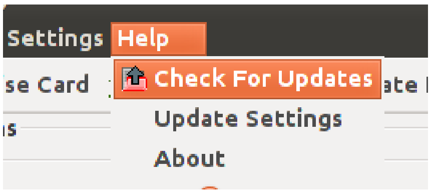
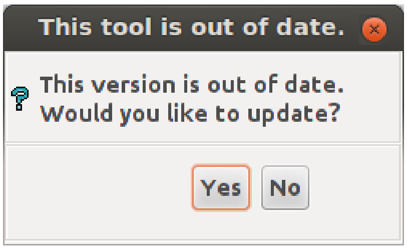


Figure 50: Help Menu

If there is an update found to be available, the user will be given the option of updating to the new version of the tool. Figure 51 shows the pop-up box that the user will see.

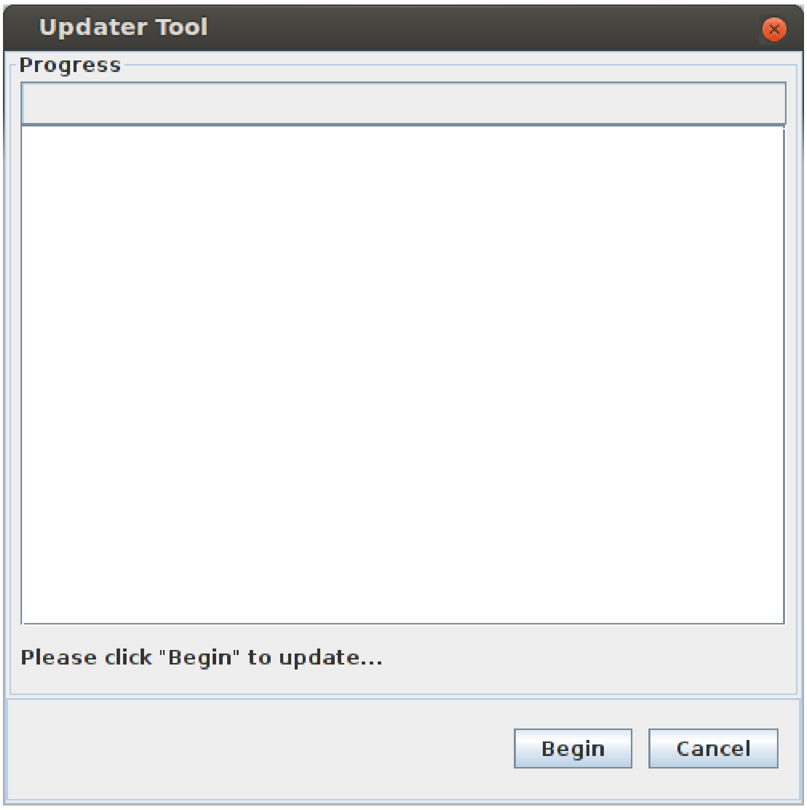


This tool is out of date.

This version is out of date.
Would you like to update?

Figure 51: Tool Out Of Date Pop-Up

If the user chooses to update to the new version of the tool, they will be met by the window shown in Figure 52.



Updater Tool

Progress

Please click "Begin" to update...

Figure 52: Updater Tool View

To start the update, the user must click begin. Choosing cancel will stop the updating process and will launch the old version of the tool. Once the update is complete, the user will be asked whether they wish to restart the tool to use the new version.

## 6.3 Working Paths

The files which are loaded into the tool by the user work on a path that is relative to the spe file upon which they are working. The file paths which should be relative to the spe files location are:

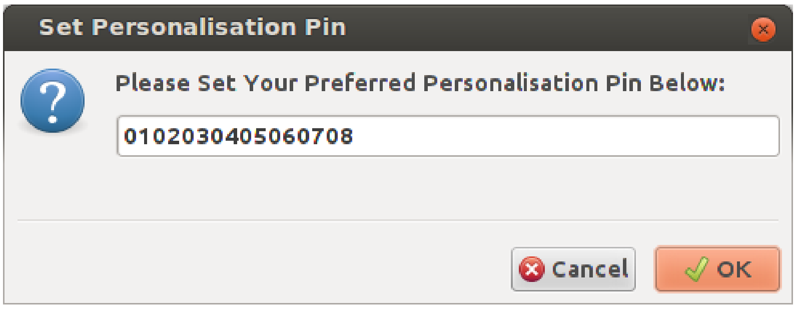
- S@T bytecode script
- Applets path
- Personalised image path

- All dynamic data files

If the tool complains that a file is not loaded in a relative path to the spe file, then please move the files to comply with the working path format. This means that the loaded file should be on a level which is equal or below the spe file.

## 6.4 Personalisation Pin

The personalisation pin can be set in Profile Editor when the Default Java OS is selected. The personalisation pin is used to lock the card for the current session. To set the personalisation pin, the user must first navigate to the *Settings* menu. The user can now select the *Set Personalisation Pin* menu option. This will bring up the window shown in Figure 53. Editing the value in the text-box will change the personalisation pin. This must be a hexadecimal value of length 8 bytes.



Set Personalisation Pin

Please Set Your Preferred Personalisation Pin Below:

0102030405060708

Figure 53: Personalisation Pin View

## 6.5 Security Pin

The security pin can be set in Profile Editor when the Default Java OS is selected. The security pin is used to lock the card so that no recycle commands can be sent to the card without the pin being authorized.

To set the security pin, the user must first navigate to the *Settings* menu. The user can now select *Set Security Pin* menu option. This is shown in Figure 54. The user will now see the security pin editing panel. This is shown in Figure 55. Checking the *Enable Security Pin* check-box will try to verify the security pin next time a command is sent to the card within the tool. To edit the value of the security pin, the text-box within this panel should be edited. This must be a hexadecimal value which is 8 bytes long.

To add the security pin to your card, you need to include the following command in your external script:
*D02400000A FFFF A8B7C6D5E4F3A2B1*
This command will install the security pin onto the card and therefore lock the card prior to pin verification. The section *A8B7C6D5E4F3A2B1* is the default security pin and can be changed to fit the users needs.

## 6.6 OS Init Flags

OS Init Flags can be set onto the card within Profile Editor. To do this, the user must navigate to the *Settings* menu. Here the user will find the *Set OS Init Flags* menu item. Selecting this will bring up the window shown in Figure 56.

The user has an option of which flags to set, these are shown in Table 23.

## 6.7 OS Information

The OS Information screen can be found by selecting the *OS Information* button on the toolbar. This will load the screen as shown in Figure 57.

Within the OS Information tool there are two main actions; *Get OS Information* and *List Installed Applets*.

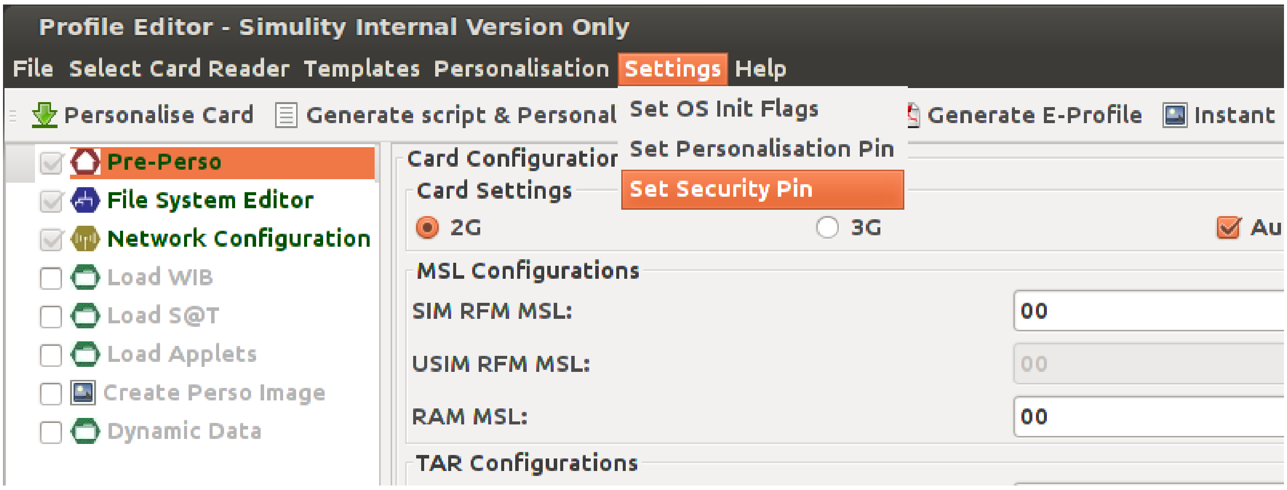
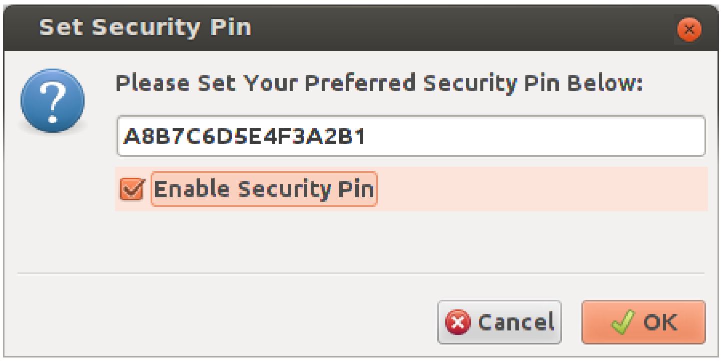


Figure 54: Settings Menu



Set Security Pin

Please Set Your Preferred Security Pin Below:

A8B7C6D5E4F3A2B1

☑ Enable Security Pin

Figure 55: Security Pin View

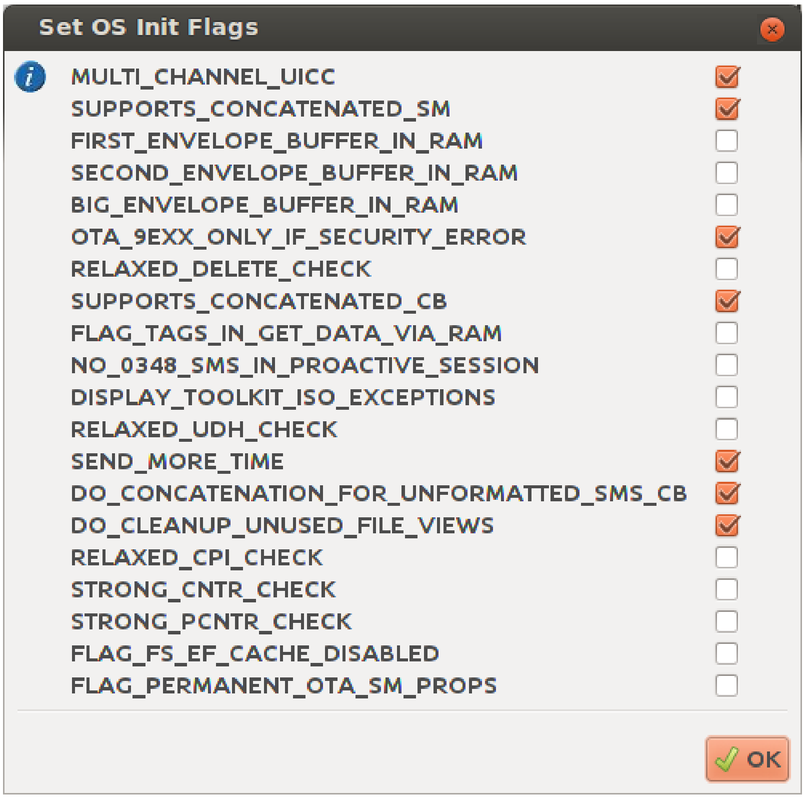
### 6.7.1 Get OS Information

Selecting the *Get OS Information* button will gather multiple values from the card. The following information is supplied to the user:

- Card ATR
- Card Manufacturer
- Card Type
- OS Version
- Mask Details
- Memory Available
- Mask Build Date - Only applicable to Simba OS

### 6.7.2 List Installed Applets

Selecting the *List Installed Applets* button will supply the user with a list of installed applets on the card. These are listed by the Package AID or the Applet AID. To get the list, the user must enter the correct OTA configuration. The user will be prompted to do this when they first select the *List Installed Applets* button. The option is available for the user to save and load their OTA configurations to speed up the process. The user also has the option to edit their current OTA configuration in order to input the correct values.



- MULTI_CHANNEL_UICC ☑
- SUPPORTS_CONCATENATED_SM ☑
- FIRST_ENVELOPE_BUFFER_IN_RAM ☐
- SECOND_ENVELOPE_BUFFER_IN_RAM ☐
- BIG_ENVELOPE_BUFFER_IN_RAM ☐
- OTA_9EXX_ONLY_IF_SECURITY_ERROR ☑
- RELAXED_DELETE_CHECK ☐
- SUPPORTS_CONCATENATED_CB ☑
- FLAG_TAGS_IN_GET_DATA_VIA_RAM ☐
- NO_0348_SMS_IN_PROACTIVE_SESSION ☐
- DISPLAY_TOOLKIT_ISO_EXCEPTIONS ☐
- RELAXED_UDH_CHECK ☐
- SEND_MORE_TIME ☑
- DO_CONCATENATION_FOR_UNFORMATTED_SMS_CB ☑
- DO_CLEANUP_UNUSED_FILE_VIEWS ☑
- RELAXED_CPI_CHECK ☐
- STRONG_CNTR_CHECK ☐
- STRONG_PCNTR_CHECK ☐
- FLAG_FS_EF_CACHE_DISABLED ☐
- FLAG_PERMANENT_OTA_SM_PROPS ☐

Figure 56: OS Init Flags View

The commands send to retrieve this information is shown on the Personalisation Window. This can be viewed by using the *View Trace* button at the bottom of the OS Information screen.

### 6.7.3 Decode Known ATR

Selecting the *Decode Known ATR* button will allow the user to enter an ATR in order to gather the cards information. The user should enter the known ATR into the presented window and select the *OK* button. If the ATR inputted by the user is valid then the card information panel will then be updated.

## 6.8 Preferences

The user has the ability to tailor the Profile Editor tool to better reflect their personal choices. The *Preferences* menu gives the user this flexibility. The *Preferences* menu can be found within the *Settings* menu item on the top toolbar of Profile Editor. This can be seen in Figure 54.

Once the user has selected this menu, they will be presented with the view as shown in Figure 49. Table 24 explains each preference.

## 6.9 Perso Window

### 6.9.1 Clearing The Perso Window

To clear the personalisation window, the user must first navigate to the *Personalisation* menu item at the top of the page. This will bring up the personalisation options. Selecting the *Clear Personalisation Window* menu item, and then selecting *Yes* will clear the personalisation window.

This can also be achieved by right-clicking the personalisation window and selecting the *Clear Personalisation Window* option. Again, the window will only be cleared if the user selects *Yes*.

Table 23: OS Init Flags Which Are Editable Within Profile Editor

| Available OS Init Flags | Default State |
|---|---|
| Multi_Channel_UICC | Enabled |
| Supports_Concatenated_SM | Enabled |
| First_Envelope_Buffer_In_RAM | Disabled |
| Second_Envelope_Buffer_In_RAM | Disabled |
| Big_Envelope_Buffer_In_RAM | Disabled |
| OTA_9EXX_Only_If_Security_Error | Enabled |
| Relaxed_Delete_Check | Disabled |
| Supports_Concatenated_CB | Enabled |
| Flag_Tags_In_Get_Data_Via_RAM | Disabled |
| No_0348_SMS_In_Proactive_Session | Disabled |
| Display_Toolkit_ISO_Exceptions | Disabled |
| Relaxed_UDH_Check | Disabled |
| Send_More_Time | Enabled |
| Do_Concatenation_For_Unformatted_SMS_CB | Enabled |
| Do_Cleanup_Unused_File_Views | Enabled |
| Relaxed_CPI_Check | Disabled |
| Strong_CPI_Check | Disabled |
| Strong_PCNTR_Check | Disabled |
| Flag_FS_EF_Cache_ | Disabled |
| Flag_Permanent_OTA_SM_Props | Disabled |

Table 24: Preferences Explained

| Preference | Meaning |
|---|---|
| Allow Updates | Whether or not the tool checks for updates when it is run. |
| Enable Generate PDF Message | Whether the generate PDF message is displayed at the end of the personalisation |
| Base Script Paths On Last Known Directory | Whether to base the starting path of Run Script and Instant Image on the last know working path. |
| Default Data Window Always On Top | Whether or not the default data window should stay on top. |
| Validation Window Always On Top | Whether or not the validation window should stay on top. |

### 6.9.2 Showing The Perso Window

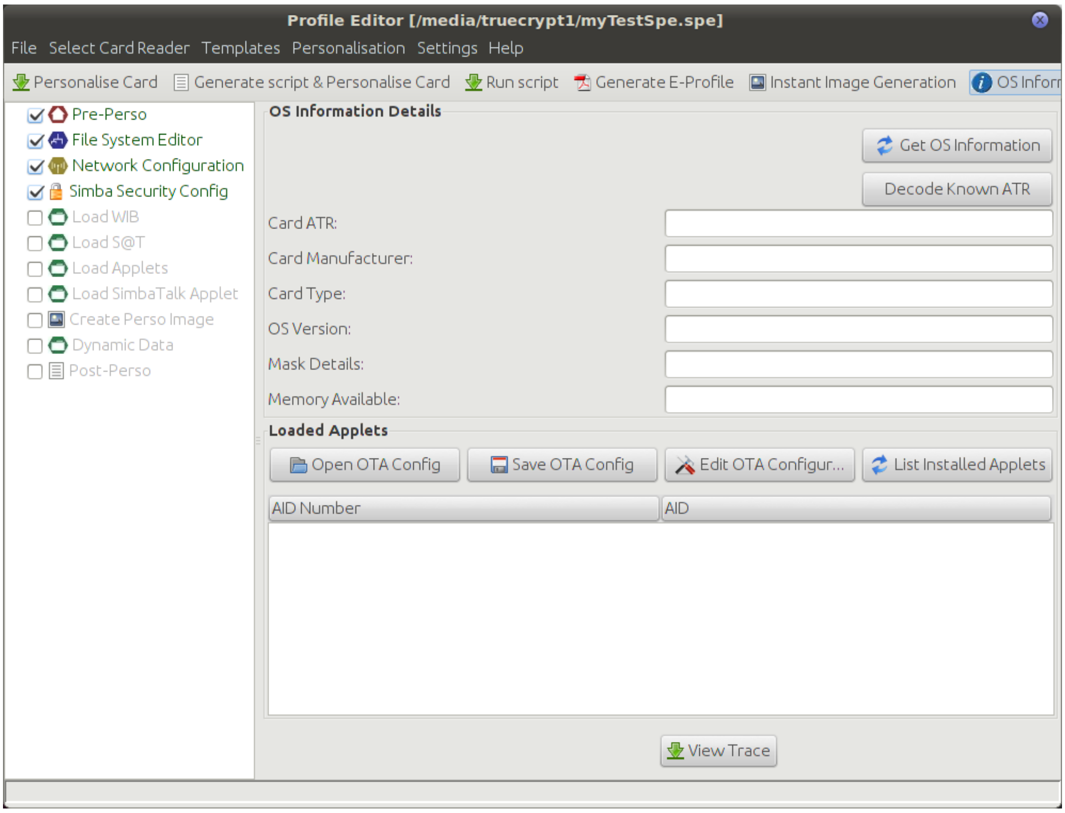
To show the personalisation window, the user must first navigate to the *Personalisation* menu item at the top of the page. This will bring up the personalisation options. Selecting the *Show Personalisation Window* menu item, and then selecting *Yes* will clear the personalisation window.

This can also be achieved by pressing the keyboard combination of ctrl+P.

### 6.9.3 Searching The Perso Window

Searching the personalisation window can be achieved in two ways; Firstly the user could right-click the personalisation window and select the *Find* option. An alternative option for the user is to press the keyboard combination of ctrl+F.

The user can then supply text to search. Clicking *Search* will then show the user the found text, if any is



☑ Pre-Perso
☑ File System Editor
☑ Network Configuration
☑ Simba Security Config
□ Load WIB
□ Load S@T
□ Load Applets
□ Load SimbaTalk Applet
□ Create Perso Image
□ Dynamic Data
□ Post-Perso

Figure 57: OS Information Landing Page

found. The user has the option to make the search case sensitive and to search for whole words only. To move through the found instances of the searched term, the user should click the *Next* or *Previous* buttons.

# 7 Profile Editor Shortcuts

## 7.1 List of Shortcuts

Table 25 shows the keyboard shortcuts which can be used in correspondence with the tool.

Table 25: Keyboard Shortcuts Applicable With Profile Editor

| Shortcut Functionality | Shortcut Command | Where Does This Work? |
|---|---|---|
| Open | Ctrl+O | Everywhere |
| New | Ctrl+N | Everywhere |
| Save | Ctrl+S | Everywhere |
| Save As | Ctrl+Shift+S | Everywhere |
| Copy | Ctrl+C | File System (must select required file) |
| Paste | Ctrl+V | File System (must select destination file) |
| Delete | Del | File System (must select required file) |
| Show Perso Window | Ctrl+P | Everywhere |
| Exit | Alt+F4 | Everywhere |

# 8 Saving and Loading Your Session

## 8.1 Saving Your Session

This section helps you to save your session in order to keep the changes you have made for future use.

To save the session, the *File* menu option should be selected. Next the *Save As* button should be clicked. An output file should then be selected, in which the session values will be stored. The output file should have the .spe extension. To overwrite the currently selected spe file, the user can opt to use the *Save* button. This will save the spe file without asking the user to specify the file. The *Save* and *Save As* options can be seen in Figure 58.

## 8.2 Loading Your Session

In order to use the previously saved session values, you need to load in the file containing them. To do this, firstly the *File* menu option must be selected. Next, the *Open* button should be clicked. The file which holds the saved values should then be selected.

The *Open* menu can be found in Figure 58.

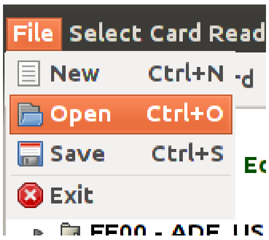


Figure 58: File Menu

# 9 Appendix A

## 9.1 XML Template Description

Below, the format of each element in the XML is explained.

### 9.1.1 Input File

This is the first tag in the XML file, it must encompass all other tags between its own start and close tag.

### 9.1.2 IMSI_Config

This tag includes all imsi values. The first element must be entered in format <imsi_start>IMSIVALUE </imsi_start>, the second element can then be entered as <imsi1_start>IMSIVALUE</imsi1_start> and so on. The IMSI value to be submitted here must be an actual IMSI value, without any swapped nibbles, parity-bits or LEN values appended to the beginning. This value must be only 15 characters long. If this value is not 15 characters long or contains non-hexadecimal characters, the user is informed. A full example is given below.

<imsi_config> <imsi_start>112233445566778</imsi_start> </imsi_config>

### 9.1.3 ICCID_Config

Here, the user enters the start (data row 0) ICCID value in between the start and end tags. The ICCID checkdigit is generated automatically if the user specifies ICCID_CD in the variable output tag. For example:

<iccid_config> <iccid_start>START VALUE</iccid_start> </iccid_config>

### 9.1.4 MSISDN_Config

The user enters the start MSISDN value in between the two tags. For example:

<msisdn> <msisdn_start>START VALUE </msisdn_start> </msisdn>

### 9.1.5 PIN1

There are a range of values that can be submitted for the pin values. First of all, if the user wishes to just enter a specific pin value, they can enter it between the pin1_value tags (or pin 2), and the same for puk. There are also other attributes of the pin tag that the user can take advantage of. The user can specify that they want the program to randomly generate their pin and puk values by specifying the 'random' attribute in the value tags as true or false. Should the user do this, a random value will be generated of the length specified in the respective length tag (there is one length tag for each pin/puk value). Another attribute that the user may wish to use is the 'decimal attribute, allowing the user to choose whether or not to specify the pin value in decimal or hexadecimal format. This is done by editing the 'decimal' attribute in the 'format' tag.

This information also applies for pin2. Only 2 pin/puk's can be entered. A full example is given below:

<pin1> <pin1_value random="TRUE" length="4" decimal="TRUE">0000</pin1_value> <puk1_value random="FALSE" length="8" decimal="TRUE">00000000</puk1_value> </pin1>

### 9.1.6 ADM

The ADM tag can contain 16 different ADM values. Each value can be specified by the user typing the value in-between the 'value' tags, or it can be generated by the user specifying 'random=”TRUE”' instead of 'random=”FALSE”' in the value tag. The value will be generated to the length specified within the same ADM tag it belongs to. A full example is given below:

<adm> <adm1> <adm1_value random="TRUE" length="8" decimal="FALSE"> </adm1_value> </adm1> </adm>

### 9.1.7 KI

The KI value can be generated or specified, it can be encrypted with the key specified in this tag, as long as the user has specified that 'encrypt=”TRUE”'. Different KI values will be generated for each card data, but all

values will be encrypted with the encryption key specified here. To specify your own KI value, you must enter this value between the two 'ki_value' tags. A full example is given below:
<ki> <transport_key encrypt="false">00112233445566FF</transport_key> <ki_value>00112233445566</ki_value> </ki>

### 9.1.8 OTA_Keysets

Each key-set is placed together within a sub-tag inside the ota_keysets tag. Each contains the kic, kid, and kik values for that individual key-set. Individual keys can be set to random by specifying the random attribute to true or false for each individual key. The length for each key set is specified in it's respective key-set tag. This length is measured in bytes, rather than amount of characters. A transport key can also be specified in the key-set tag from the 'transport_key' attribute. Should this attribute be empty, no encryption will take place, otherwise the keys will be encrypted according to this key. Lastly, should you wish all of the key-sets to be the same, or some of them to be the same, you may use the 'copy_keyset' attribute in the key-set tag to specify which key-set you would like this one to be the same as. The first key-set should be set to key-set 1, and not to key-set 0. A full example is given below:
<ota_keysets> <ota_keyset1 length="8" transport_key="00112233445566FF" copy_keyset="2"> <kic1 random="FALSE">1111111111111111</kic1> <kid1 random="TRUE"/> <kik1 random="FALSE" >1111111111111111</kik1> </ota_keyset1> </ota_keysets>

### 9.1.9 Output

The output tag allows the user to specify the values that they want to view/save for when they wish to send there data to card. The format is '<output_data>VAR_OUT:VALUE/VALUE/VALUE</output_data>. A full list of possible values is given here, the name supplied for each value must be specified the same as here.
IMSI / IMSI1 / IMSI2 / ICCID / ICCID_CD / PIN1 / PUK1 / PIN2 / PUK2 / KI / EKI / ADM1 / ADM2 / KIC1 / KIC2 / KIC3 / KIC4 / KIC5 / KIC6 / KIC7 / KIC8 / KIC9 / KIC10 / KIC11 / KIC12 / KIC13 / KIC14 / KIC15 / KIC16 / KID1 / KID2 / KID3 / KID4 / KID5 / KID6 / KID7 / KID8 / KID9 / KID10 / KID11 / KID12 / KID13 / KID14 / KID15 / KID16 / KIK1 / KIK2 / KIK3 / KIK4 / KIK5 / KIK6 / KIK7 / KIK8 / KIK9 / KIK10 / KIK11 / KIK12 / KIK13 / KIK14 / KIK15 / KIK16 /MSISDN / ACC

### 9.1.10 Unsupported Tags

Tags not supported are not read into or recognised by the system. This means that the user cannot input their own tags, if they do this the Dynamic Data Generator will ignore them. Should they also place this tag in the output file, the user will be informed that this value is not supported.

### 9.1.11 Supported Output Vars

The output variable will only accept variables that the software supports. The user cannot input their own variables and tags. Some of these variables are compulsory (ALW) and some are optional (OPT) in the output var tag. The accepted variables are shown in Table 26

### 9.1.12 Compulsory Tags

The following tags must be included in the var output tag:
IMSI, ICCID, ICCID_CD, ACC, PIN1, PUK1, PIN2, PUK2, KI, ADM1.
Without these tags it is impossible to personalise your card with dynamic data.

Table 26: Supported Output Variables

| Output Var | Description | Compulsory |
|---|---|---|
| IMSI# | The IMSI value where '#' specifies which IMSI value they wish to submit. The first IMSI value is 'IMSI' the second is 'IMSI1' and so forth. | ALW |
| ICCID | The ICCID value | ALW |
| ICCID_CD | The ICCID_CD value | ALW |
| PIN# | The PIN value where '#' specifies which PIN value is being referring to. | ALW (at least PIN1) |
| PUK# | The PUK value where '#' specifies which PUK value is being referring to. | ALW (at least PUK1) |
| KI | The KI value | ALW |
| ADM# | The ADM value where '#' specifies which ADM value is being referred to. | ALW (at least adm1) |
| KIC# | The KIC value where '#' specifies which KIC value is being referred to. | OPT |
| KID# | The KID value where '#' specifies which KID value is being referred to. | OPT |
| KIK# | The KIK value where '#' specifies which KIK value is being referred to. | OPT |
| MSISDN | The MSISDN value (Value not determined in XML template) | OPT |
| ACC | The ACC value (Value not determined in XML template) | ALW |

# 10 Appendix B

## 10.1 AC Options

The following options are the AC available for every Compact or Extended/Expanded command:

CHV1, CHV3, CHV4, CHV5, CHV6, CHV7 CHV8. ALWAYS, NEVER. ADM1, ADM2, ADM3, ADM4, ADM5, ADM6, ADM7, ADM8, ADM9, ADM10, ADM11. UNIVERSAL. _2PIN1, _2PIN2, _2PIN3, _2PIN4, _2PIN5, _2PIN6, _2PIN7, _2PIN8.

# 11 Frequently Asked Questions

## 11.1 Pre-Personalisation

### 11.1.1 New Or Old Masks?

**Q: When should I use the new mask values and when should I use the old mask values?**

**A:** New mask values should be used when the user wants to personalise a card which has an OS version which is post-4.2.6.

Whereas old mask values should be used when the user has an OS version which is not post-4.2.6.

### 11.1.2 SIM Or USIM?

**Q: Should I use a SIM or a USIM template?**

**A:** SIM templates should be used when the user wants to create a 2G profile.

USIM templates should be used when the user wants to create a 3G profile.

### 11.1.5 Why Link Files?

**Q: Why should I link a 2G file to a 3G file?**

**A:** A link should be made when a 2G file and a 3G file are to use shared data. This type of mapping of data reduces the amount of memory used for this file on the SIM. For example, if a 2G file and 3G file are linked it means any update of the data in the 2G file will be represented in the same data update in the 3G file. A 2G file can also link to another 2G file.

### 11.1.6 GSM Algorithms?

**Q: Which are the differences between GSM algorithms?**

**A:** The most commonly used authentication algorithms are COMP V1, COMP V2, COMP V3 and MILENAGE. These algorithms are used by the card and the network to successfully connect to the network.

MILENAGE is used for 3G networks while the COMPVx algorithms are used for 2G networks.

### 11.1.7 Install Parameters?

**Q: What are install parameters?**

**A:** Install parameters are to Java applets what command-line arguments are to applications. They enable the user to customize the applets operation. By defining parameters, you can increase your applets flexibility, making your applet work in multiple situations without recoding and recompiling it.

### 11.1.8 Disabling Applet Settings?

**Q: What difference does disabling Applet settings make to personalisation?**

**A:** The user can disable applet settings. This option should be selected when user is loading a library i.e. no installation required when loading a library as there is no install method in the application code.

## 11.2 Personalisation

### 11.2.1 Applet Loading?

**Q: Is an applet loaded onto the card when it is added into the Load Applet module or when the card personalisation is done?**

**A:** An applet is loaded onto the card during personalisation. When applets are loaded into the Load Applet Module they are simply added to a list of applets to load onto the card during personalisation. The order in which the applets are loaded into the module is the order in which they will be personalised onto the card.

### 11.2.2 Personalised Image?

**Q: What is a personalised image?**

**A:** A personalised image is an image of the memory of the card at a specific time.

**Q: Why would I want to create a personalised image?**

**A:** A personalised image is used to personalise the card by writing blocks of data to memory area rather than personalising using the usual standard commands. In general this script is mainly used during the production process to allow the image of the static perso be programmed to large quantities of cards at a much faster rate.

### 11.2.3 Multiple Personalisations?

**Q: Can I personalise more than one card reader simultaneously at one time?**

**A:** No, this is not currently possible. One card must be personalised fully before the next card is personalised.

### 11.2.4 Options Always Selected?

**Q: Why can I not personalise without selecting pre-perso, file system editor, network configuration and simba security configuration?**

**A:** The user cannot personalise the card without these options selected as the card requires these to function.

### 11.2.5 Run Script or Generate Script & Personalise?

**Q: When should I use the Run Script button, and when should I use the Generate script & Personalise Card button?**

**A:** The Generate Script & Personalise Card button should be used when the user wants to create LDR file which contains all the APDU commands that are sent during card personalisation as well as loading the current profile configuration onto the card. This .ldr file can be re used to personalise other cards with the same data.

The Run Script button should be used when the user already has the required LDR file, and wants to personalise the card with this file. This option speeds up the personalisation process for the user when they require to personalise multiple cards with the same data.

### 11.2.6 Referenced Access Condition - How do I set the Record ID?

**Q: How Do I configure the Referenced Access Condition?**

**A:** The Referenced Access condition sets the security of a file by using a record from the ARR file. This ARR file can have many records. Each record will contain a different access condition rule.

To set the Access Condition of a file to Referenced first click on the referenced radio button. This should change the Access Condition Panel. In the File Id text-box enter the file ID of the ARR file. To select the ARR record to use enter the record index into the Record 1 text-box. The text-box is called Record 1 because a single file can have 2 access condition rules assigned to it. To set the second access condition rule enter the second record index into the text-box labelled Record 2.

### 11.2.7 Enabling the use Fixed Dialling Numbers (FDN)?

**Q: How can I enable FDN with the Profile Editor?**

**A:** All profile editor templates have the FDN set to *'allocated but not activated'* in the SST table.

Here's how to enable the FDN for 2G and 3G.

**2G Profile:**

If you're starting with no file system, use the add template feature to add in the default template for a 2G profile.

In these template the SST table has the FDN set to *'allocated but not activated'* this means that the FDN will be disabled in the phone and you won't be able to enable the FDN via the phone because the FDN menu item will be hidden. To enable FDN you'll have to change the content of the SST file to set the ADN to *'allocated but not activated'* and set the FDN to *'allocated and activated'*. Now when you insert the SIM into the phone the menu option to enable the FDN should now be visible.

**3G Profile:**

If you're starting with no file system, use the add template feature to add in the default template for a 3G profile.

With a 3G profile the default template will allow you to activate the FDN directly from the phone. There's no need to change the content of the SST table.

#### 11.2.8 Referenced Access Condition – Security Environment

**Q: What is the Referenced Access Condition Security Environment value?**

**A:** The profile editor will allow you to set the security environment variable to 2 values, Table 27 explains what each value represents.

Table 27: Security Environment Variable Properties

| Security Value | Properties |
|---|---|
| 0 | The global Application PIN is disabled AND the Universal PIN is disabled OR<br>The global Application PIN is disabled AND the Universal PIN is enabled and used. |
| 1 | The global Application PIN is enabled OR The global Application PIN<br>is disabled AND the Universal PIN is enabled but not used. |

#### 11.2.9 File Already Exists During Personalisation

**Q: Why does it say that the file already exists during personalisation?**

**A:** This will happen if the Short File Identifier of a file contains the same value as another file in the same level of the file system. This value must be unique, just like the file ID.

#### 11.2.10 What access condition corresponds to PIN2

**Q: Which access condition should I use to signify PIN2? Should I use CHV2 or 2PIN1?**

**A:** PIN1 means that this is the first pin on the first application. Therefore 2PIN1 is the second pin on the first application. Therefore 2PIN1 should be used to signify PIN2.

On the other hand, CHV2 is the first pin on the second application. As Profile Editor does not allow the user to create more than one ADF file, CHV2 has been removed from the access conditions in Profile Editor.