

# Userguide PersoSim

V1.4

This userguide is intended to help users during installation and basic usage of PersoSim and related components.

## Contents

<b>1</b>	<b>Components &amp; Installation</b>	<b>1</b>
1.1	Simulator . . . . .	2
1.2	Driver . . . . .	2
1.3	Editor . . . . .	3
<b>2</b>	<b>Basic usage of PersoSim simulator</b>	<b>4</b>
2.1	User interface . . . . .	4
2.2	Load a personalisation . . . . .	5
2.3	Change the reader type . . . . .	5
2.4	Logging and Console . . . . .	7
2.5	Smartphone as card reader . . . . .	7
<b>3</b>	<b>Creation of user defined profiles with PersoSim Editor</b>	<b>10</b>

## 1 Components & Installation

Prerequisite for installation and operation of PersoSim is a Java Runtime Environment version 1.8 or above.

## 1.1 Simulator

The PersoSim main application contains the card simulation as well as the virtual card reader which is connected to the operation system via driver (see below). The application does not require a classical installation. It can be executed immediately after extracting the zip file or tar ball.

Depending on the operating system execute either `PersoSim.exe` in Windows or `PersoSim` on Linux or MacOS. The GUI shall start similar to figure 1. In case your use case does not require the virtual driver (i.e. when connecting via RemoteIFD) the simulator can already be used now.

## 1.2 Driver

In order to use the virtual card reader (and thus the simulated id card) through the regular PCSC stack operating system dependent driver implementations are available. These drivers integrate PersoSim in the respective PCSC stack and thus allow access from every compatibel application.

The installation procedure for these drivers is operating system dependent. Below the installation is described for Windows, Linux and MacOs in that order.

**Windows** Minimum requirement for PersoSim driver on Windows is Windows 7 or newer.

The driver installer package automatically detects which version of Windows (Version 7, 8, 8.1 oder 10) and which architecture(32 oder 64 Bit) needs to be used. In order to install the driver just execute the the installer package and follow the instructions. In contrast to earlier versions the manual installation of test certificates is no longer required. On Windows 10 it might be needed to deactivate driver signature validation during installation. After successful installation the virtual card reader shows up as "PersoSim Virtual Reader" in device manager similar to physical card readers and can be used.

**Linux** The following steps to install the driver on Linux where created and tested on Ubuntu 12.04 LTS. But it should be basically work on most Linux derivates that support PCSC through libpcsclite.

Based on the Ubuntu 12.04 LTS default installation the following additional packages need to be installed:

- pcsd
- pcsc-tools
- libpcsclite-dev

These can be installed by executing the following command:

```
sudo apt-get install pcsd pcsc-tools libpcsclite-dev
```

The driver is distributed in source form within the tar ball. After extraction it needs to be compiled by executing `make` in the extracted directory. Afterwards the compiled driver can be installed via `sudo make install`. In order to load the newly installed driver PCSC daemon needs to be restarted: (`sudo service pcsd restart`). On following starts of `pcsd` the driver will be loaded automatically.

**MacOS** In order to provide the required tool chain on MacOS you need to install Xcode and "command line developer tools" on your system. The exact process depends on the used versions of MacOS an XCode. For current versions the execution of the console command `xcode-select --install` should lead to a dialog that also installs the command line tools.

The driver is distributed in source form within the tar ball. After extraction it needs to be compiled by executing `make` in the extracted directory. Afterwards the compiled driver can be installed via `sudo make install`. After a system reboot the virtual driver should be automatically detected by MacOS and PersoSim can be used.

### 1.3 Editor

PersoSim Editor is a tool to modify existing profiles and even allows to create completely new profiles for the PersoSim simulator. This allows to simulate id cards with arbitrary content.

Simliar to the simulator PersoSim Editor does not need a classic installation. Installation and execution work similiarly just by extracting the archive and starting the contained executable. UI should start up similiar to figure 6 in chapter 3.

## 2 Basic usage of PersoSim simulator

### 2.1 User interface

The foreground of PersoSim resembles the user interface of the integrated virtual card reader (see figure 1). All other functions are accessible through tabs and menus which will be described in the following.

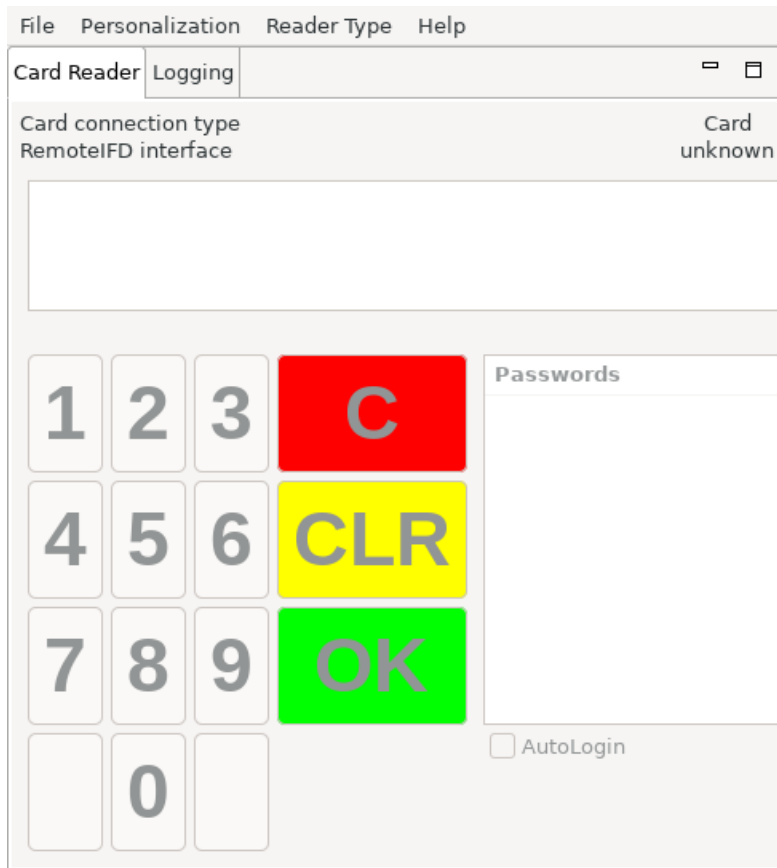


Figure 1: PersoSim main window

## 2.2 Load a personalisation

Personalisations can be loaded either through the command line or through menu items within the graphical user interface. If no other personalisation is selected the default profile 1 is used. When a profile is loaded it is immediately active. This is similar to putting a card on a physical reader.

Currently 16 predefined standard profiles are available. These are separated in 11 german ID cards as well as 5 Unionsbürgerkarten (german id cards for european union citizens). The current overview of available profiles including details to the data used within them can be found on the PersoSim website in the Downloads section.

**Load a personalisation via menu** The most convenient option to load a personalisation is the application menu. The menu item "Personalization" offers the following sub items:

- **Select Perso from file** opens a file dialog which allows to select an arbitrary personalisation from an existing XML file.
- **Select predefined Perso template** allows to select on of the standard profiles described above.
- **Remove card** removes the simulated card from the reader.

**Load a personalisation via command line** The PersoSim console (see section 2.4) allows loading of personalisations. The command `loadperso<file name>` loads a personalisation from a given file. Instead of the filename a single number can be entered which will be mapped to the standard profiles.

## 2.3 Change the reader type

The virtual card reader used within PersoSim can be configured to simulate either a basic or standard reader. Selection of the reader type can be performed easily using the menu item "Reader Type".

**Basic reader** usually neither have a display nor pin pad. Accordingly the basic reader in PersoSim only shows the two information fields in the upper area. The show whether a card is present and via which connection type the virtual reader is connected to the applications (see figure 2)

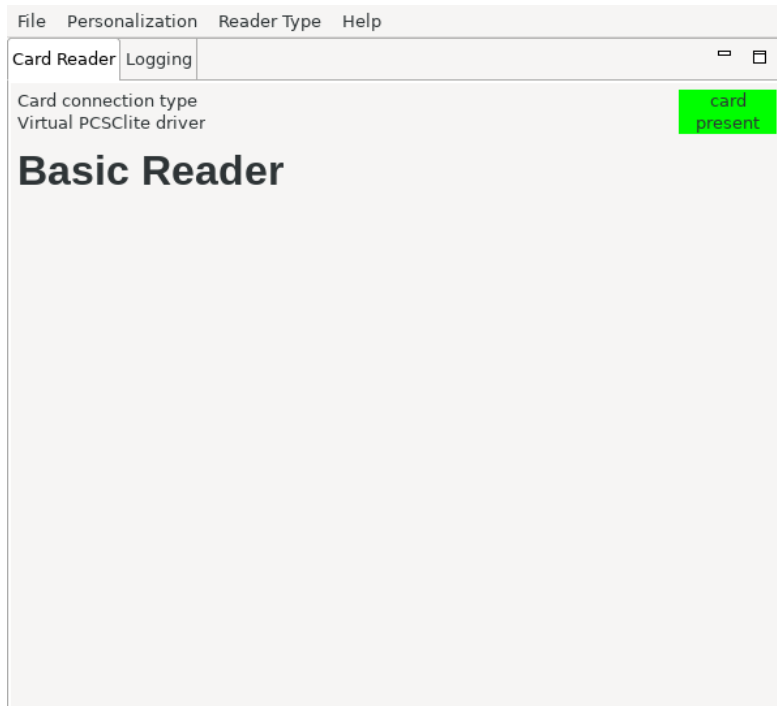


Figure 2: PersoSim as basic reader

**Standard reader** need to have a display and a PIN pad. These allow the presentation of data according to BSI TR-03119 as well as input of passwords. Accordingly the PersoSim standard reader shows a display field and a key pad for PIN entry. Below the display the transmitted CHAT will be displayed (when hovering above it with the mouse cursor it will be broken down in a human readable form).

Additionally the PersoSim standard reader allows to store frequently used passwords on the right-hand side. These can be quickly used by double-click on an entry.

When performing multiple tests this manual input requires significant amounts of test time. Therefore PersoSim has an AutoLogin feature implemented. Figure 3 shows this behaviour as an example: passwords shown are '123456' (PersoSim default PIN) as well as '500540' (CAN in all default profiles)

When PersoSim is started the standard reader is active.

Input buttons on the standard reader are only activated when communication requires this. When the program is started those buttons are without any function. As soon as user input is needed a message is shown in the display field accordingly and the buttons get activated.

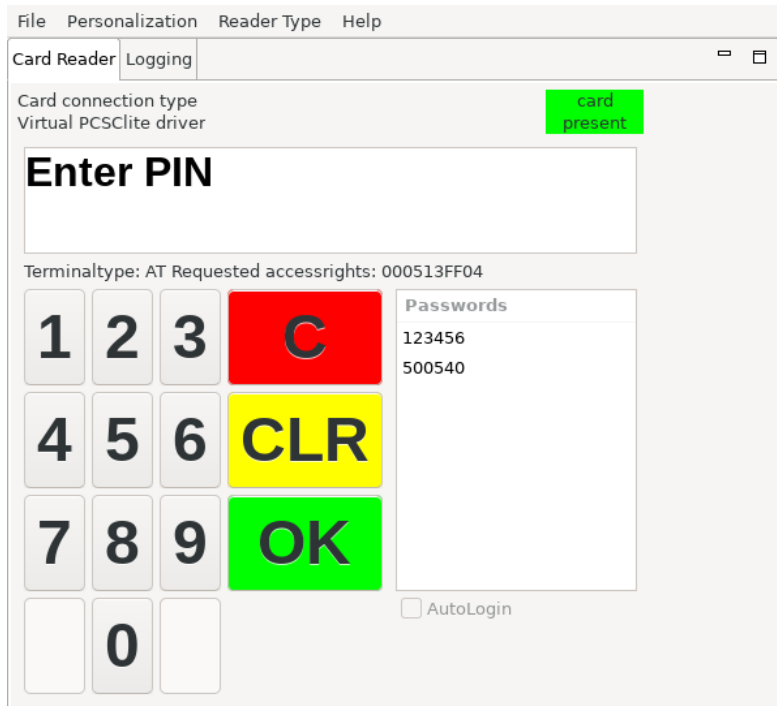


Figure 3: PersoSim as standard reader

## 2.4 Logging and Console

The tab "Logging" contains the console window which shows all log output of all commands executed on the simulator as well as (intermediate) results and execution steps (compare figure 4). This view provides a context menu which allows access to logging configuration. Additionally this menu can be used to export the current log output to a file.

Below the log message display is an input box where you can enter commands to interact with the simulator directly. To get an overview of supported commands enter `help`

## 2.5 Smartphone as card reader

BSI TR-03112-6 defines the protocol "Remote IFD", also known as "Smartphone as card reader". This allows to use a NFC-capable smart phone as card reader to access the id card. PersoSim supports this protocol in the desktop application as well as in the android app and thus allows to access the simulated card from eID-Clients via network instead of the operating systems PCSC stack.

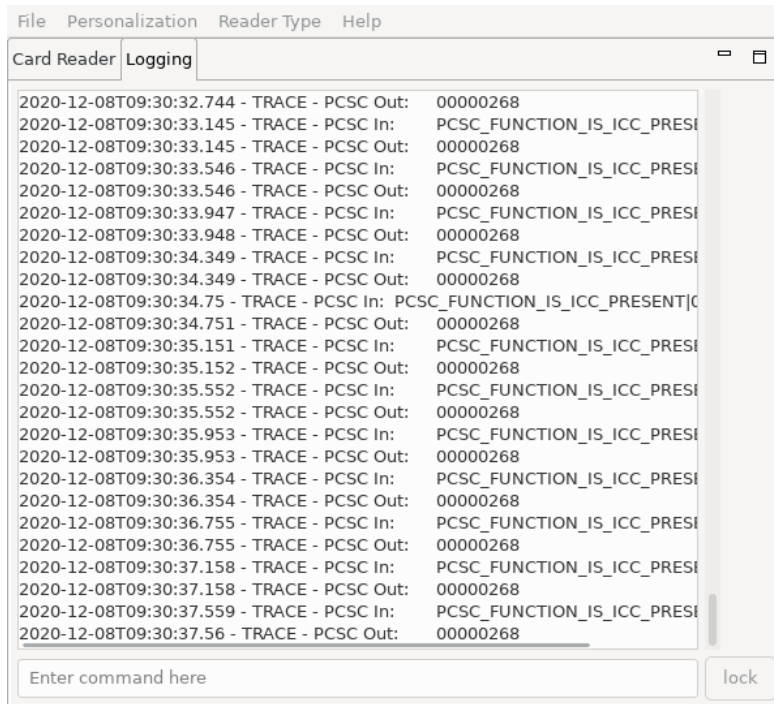


Figure 4: PersoSim Console

In order to use this feature it needs to be selected in the menu as connection type. Select "Reader Type" → "UseRemote IFD" to do so. The currently active connection type is always shown in the upper part of the virtual card reader interface. This notification is present regardless of the selected reader type (on standard reader this is above the pin pad on basic reader in a similar position)

Before an eID-Client can communicate with PersoSim via Remote IFD both devices/applications need to be paired. On PersoSim the required dialog can be opened via menu item "ReaderType" → "Configure RemoteIFD" (figure 5).

The upper area lists paired devices/applications. In case pairing needs to be revoked this can be done here within the context menu of each element.

The middle area allows to initiate a new pairing process to pair with a new device. When pairing mode is started the randomly generated pairing code is shown here. This pairing code needs to be entered in the eID-Client in order to establish the very first connection to PersoSim (see the documentation of your eID-Client for details). When pairing was successful following connections establishment is handled based on exchanged certificates and no further input of the pairing code is needed.

The lower part of the configuration dialog allows to change the name of this PersoSim





Figure 5: Configuration Remote IFD

instance. This will be shown in the eID-Client during pairing and to identify the certificate when a connection is reestablished.

### 3 Creation of user defined profiles with PersoSim Editor

PersoSim Editor allows to create user defined personalisations which can be used in PersoSim. Profiles are stored within an XML file which contains all relevant data needed for the simulation. For user convenience PersoSim Editor allows graphical editing of the most relevant fields in these files. Figure 6 shows the user interface of PersoSim Editor, which can be downloaded from PersoSim website.

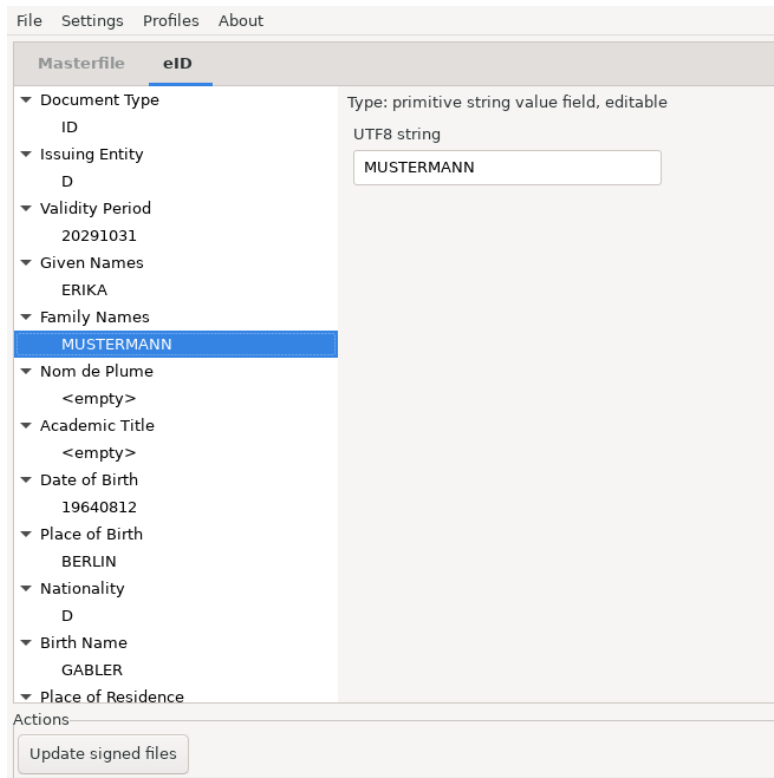


Figure 6: PersoSim Editor

PersoSim Editor displays existing values of a profile and allows modification or removal of those values as well as adding new fields. Invalid values will be highlighted so that the user can adjust them easily (e.g. invalid country codes will be filtered). The base for these modifications can either be an empty new profile, an existing personalisation loaded from a file or one of the default profiles.

In the signature configuration dialog the user can configure document signer certificate and key, to allow the final signature generation for the newly created profile. This will recreate the relevant files EF.CardSecurity and EF.ChipSecurity to complete the profile. In case the profile does not yet contain an EF.CardAccess this can be generated as well. These files will be filled with default cryptographic parameters and protocols as defined

for the german id card. EF.CardAccess, EF.CardSecurity and EF.ChipSecurity can be regenerated automatically, but can not be edited manually.