PV204: Project Design

Rudolf Wittner, Peter Sooky, Deepak Kumar Vishwakarma

April 14, 2016

Abstract

The aim of this document is to cover the initial design of our course project work. We chose open source application from *https://github.com/gaborbata/jpass.git* and we will implement additional security features using the Javacard smart card.

1 Open Source Application

JPass is a simple, small, portable password manager application with strong encryption. It allows to store user names, passwords, URLs and generic notes in an encrypted file protected by one master password.

2 Javacard Smart Card Applet Design

We will create a JavaCard applet for JPass password manager. Here are the scenarios, where the applet will be used:

- **Generate password** - The password will be generated on the smart card. For simplicity, user will have no options for choosing parameters of the password - numbers, uppercase and lowercase letters will be used. This operation is available without authentication.
- **Save file** - During this operation, all passwords will be stored (updated) on the card. While saving the passwords for the first time, the user will have to choose his PIN (4-6 digits), which is used as a master key for accessing the saved structure. User will also have to choose unique string identifier of his data, which will be later used for accessing the passwords (it will be used instead of file name). If this operation is successful, the card goes to authenticated state, so then the user don't need to pass PIN and identify again. Authenticated state is always tied to the identifier.
- **Open file** - Loads stored data from the card. User have to provide PIN and the identifier of the data. If this operation is successful, card goes to authenticated state.
- **Change PIN** - changes the pin for the current identifier which is already authenticated - so it is accessible only when the applet is in authenticated state.

3 Javacard Smart Card Interface

In this section we discuss the interface, which will be used by the JPass application to communicate with the JavaCard applet. The interface will translate method calls to APDU commands and send it to the applet:

- `string GeneratePwd()`
  Generates new random password.
- `bool SetPin(string newPin)`
  Sets pin for the applet. Returns true for success, otherwise returns false.

- `bool ChangePin(string oldPin, string newPin)`
  Changes the pin for authenticated user. Returns the outcome of the operation.
- `bool VerifyPin(string pin)`
  Changes the state of the applet from normal to authenticated - is used to log on.
- `ADT GetData()`
  Return data stored on the card for current user. This call is available only in authenticated state.
- `bool StoreData(ADT userData)`
  Stores (updates) the data for current user. This call is available only in authenticated state.

The applet will have 2 states:

- `BASIC` - state for not authenticated operations - GeneratePwd, SetPin[1] and VerifyPin
- `AUTHENTICATED` - state for authenticated operations - ChangePin, GetData, StoreData

The communication channel between the JavaCard and the application running at the host machine will use secure sessions. Further, we will use MAC to ensure the integrity of the data.

4 Attacker Model and Threat Mitigation

Our development will cater to the active attacker in the communication channel. Attacker will have following powers

- Listen to the channel
- Modify the data being communicated
- Store and replay the data (replay attack)

We will make the communication between the application and the java card secured using the encryption. To ensure the data integrity we will use MAC.

[1]For SetPin() functionality it is assumed that this operation is exercised in a safe environment