Federal Office
for Information Security

Developer Documentation AGD - User Guidance

de.fac2 - FIDO U2F Authenticator Applet, v1.34

1.4 (14.06.2022)

Federal Office for Information Security
Post Box 20 03 63
D-53133 Bonn

Internet: https://www.bsi.bund.de

Table of Contents

Figures

Tables

1 Preparative Procedures

The de.fac2 U2F token is in operational state at its delivery. It's not necessary to initialize the token. However, it is possible to reset the token. That ensures the de.fac2 token generates new private keys which weren't used before. This reset process can be done via the de.fac2 Manager at any time (see chapter 3).

The de.fac2 U2F token only contains a Fido U2F applet. No other applets are allowed to be installed on the same card. Installing additional applets is not possible.

1.1 Operational environment security

The user has to keep the system used with the de.fac2 token secure. That includes using an up to date operation system with installed latest updates and patches. As the token is mainly used in web applications used with a web browser, the used browser has to be up to date. Also additional software that is used together with the de.fac2 token (e.g. Fido Clients as Google Authenticator) shall be installed via trusted sources (e.g. Google Play store, etc.).

The de.fac2 token shall only be used at trustworthy (web-) services. Before using the de.fac2, the user shall check (e.g. by verifying the TLS certificate) the service he is going to use.

Due the fact that the token can be used also contactless, it might be possible that the card could response to an attackers unwanted command if the token is near to an attackers reader. To avoid unwanted reset commands or so called "relay attacks" where an attacker can use the token by placing a reader near to the token, the user shall ensure the card is always under his control. To avoid unauthorized contactless usage of the token, the user may put the token into a shielded card sleeve when the token is not in use.

Note: As smart cards have no buttons, the FIDO user presence check with a smartcard is implemented by inserting the card into a reader or placing the card on an NFC field. After a single action, the presence check flag on the card is disabled.

Malware on the host PC / smartphone could send a reset command to the reader programmatically, thereby circumventing the presence check: It is not possible for the card to distinguish if the reader sent a reset command or if it was physically removed from the reader. This requires that the user keeps his system secure, i.e. that the host PC / smartphone is free of malware and under full control of the user.

1.2 Check authenticity

The authenticity of the de.fac2 U2F token can be checked by reading the attestation certificate out of the token. This can be done by using the U2F Registration Message (U2F_REGISTER) which is specified in chapter 4 in [FIDO_U2F_Msg]. The challenge parameter and the application parameter can be freely chosen (e.g. use 32 byte random bytes for each parameter). The token will return the attestation certificate in the Response APDU as specified in chapter 4.3 in [FIDO_U2F_Msg].

The attestation certificate is a X.509 conform certificate and contains the applets name and version in the subject field. The subject shall contain "CN=de.fac2 U2F token Version 1.34". The issuer shall contain "CN=BSI FIDO U2F Root CA". To ensure the attestation certificate is a trusted one, it must be checked if the attestation certificate is correctly signed with the BSI FIDO U2F root certificate with the subject "CN=BSI FIDO U2F Root CA ". which can be downloaded under the following URL:

https://www.bsi.bund.de/U2F-CA

If the signature of the attestation is valid and the subject is as defined above, the authenticity of the token is ensured.

To simplify the authenticity check process an applet on a smartphone can be used. The applet performs all steps described above and shows the checks result.

To ensure that the user has received a genuine de.fac2 U2F Token, the user may install the “de.fac2 Manager” from the Google Play store on his Android NFC-enabled smartphone.

It can be installed either via the Google Play App or via the Google Play website https://play.google.com/store/apps/details?id=de.tsenger.defac2manager

The authenticity of the de.fac2 token can be check by following the instruction inside the de.fac2 Manager app. In result the app shows the subject of the attestation certificate containing the name and version of the de.fac2 token and shows if the attestation certificate is signed and verifiable with the manufactures root certificate.

1.3 Additional software

For using the de.fac2 token on NFC-enabled Android devices it is recommended to install the latest version Google Authenticator[1] and the Chrome for Android browser on the users smartphone. These apps are not necessary for using the token and don’t have impact on the security of the token but enable the use of the de.fac2 token on many websites that support FIDO U2F logins.

The de.fac2 token can also be used on any computer with a contactless or contact-based reader and any tool which can send and receive APDUs to the token via this reader.

1 https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2

2 Operational User Guidance for end users

The de.fac2 token is a FIDO U2F authenticator that can be used on any system which supports FIDO U2F logins via NFC or compatible contactless interfaces. These are de facto Android smartphones which support NFC. The de.fac2 token may also be used via the contact-based interface with a compatible reader.

The de.fac2 token provides two main functions which are described in the following chapters.

2.1 Registration and Authentication

The main use cases are the registration and authentication process via the NFC interfaces on mobile devices like Android smartphones. The required steps for this use cases are described in the following.

1. On the Android device start Google Chrome and open the site you want to register or login with your de.fac2 token and login with user name and password.
2. When you see the message “tap and hold” hold your token against the NFC antenna of your phone.
3. After the registration / authentication process, remove the token from the phones’ NFC antenna.

It is also possible to use the de.fac2 token on systems with a contact-based card reader. The required steps may slightly differ on different systems. Please follow the instruction of the system you are using.

It is always necessary to remove the de.fac2 token from the NFC device / card reader after every operation (registration, authentication and reset) before the next operation can be performed. Removing and replacing the token will ensure that the user has proven its presence and only one operation can be performed after the user presence check.

2.2 Reset

To set the token to its delivery state it can be reset, which means that the internal keys will be securely erased and new keys will be generated. This can be done with the de.fac2 Manager App. It can be installed either via the Google Play App or via the Google Play website https://play.google.com/store/apps/details?id=de.tsenger.defac2manager.

The reset process can be started by following the instruction inside the de.fac2 Manager app.

After a reset all previous registrations are invalid and can’t be used any longer without a new registration.

3 Technical User Guidance

This chapter describes how to use the APDU interface of the de.fac2 U2F token. It is mainly for technical users which are going to communicate directly with the TOE without using the standard FIDO U2F end user process. This chapter might also be useful for developers of FIDO Authenticator Clients and for the composite product integrator which has to set the attestation certificate.

All APDUs the de.fac2 U2F token can process, are described in [FIDO_U2F_Msg]. The commands and their parameters as well as the possible responses and their meanings are described therein. The token doesn't differentiate between different user roles at the APDU interface. All commands will be processed in dependence on the internal state of the token and may be only usable in certain states.

Three different internal states are defined:

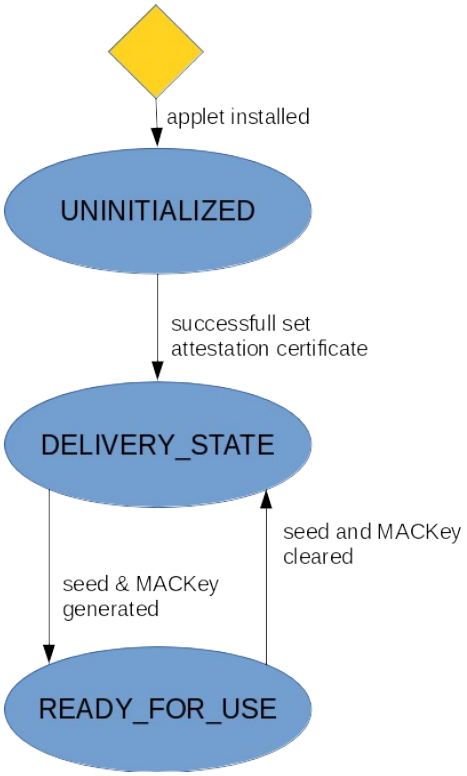

Figure 1: Applet state diagram

- UNINITIALIZED: After installation of the de.fac2 applet onto the platform, the token will be in this state. In this state the token will only accept the SET_ATTESTATION_CERT command. All other commands will be rejected. After successful setting the attestation certificate, the token will switch to the state DELIVERY_STATE. The UNINITIALIZED state can't be reached again if the token was in any other state.
- DELIVERY_STATE: In this state the only allowed operation is the generation of the internal keys seed and MACKey. After successful key generation, the token will switch to the state READY_FOR_USE.
- READY_FOR_USE: In this state all operational commands except the SET_ATTESTATION_CERT can be processed. The command RESET will clear the seed and MACKey and switch the token to the state DELIVERY_STATE.

The de.fac2 token accepts all commands specified in the FIDO U2F Specification[FIDOSpec]. The respective APDU formats are described in [FIDO_U2F_Msg]. All therein specified APDUs and their parameters are implemented in the token. The responses are also conform to this specification. The token also has implemented the GET_RESPONSE command. This command is used for chaining (see [ISO 7816-4]) if extended length APDUs can't be used and more data have to transferred in response to a command then fit in one APDU.

Additional to the FIDO specified commands the token processes two proprietary commands which are described in the following.

- The SET_ATTESTATION_CERT command is only usable as long as the card is in an uninitialized state. As soon as the attestation certificate is loaded into the token, this command can't be used anymore. This command will never be available for the end user, because the attestation certificate will be set before the delivery of the token.
- The RESET command can be used to reset the seed and MACKey. The old keys will be deleted and new keys will be generated. After a reset all registrations which were performed previously will be invalid and can't be used any longer without a new registration.

The format and parameters of these two additional commands are described in the following chapters:

3.1 SET_ATTESTATION_CERT

This APDU is used to set the TOEs attestation certificate. The command is only available as long as the certificate wasn't transferred completely to the TOE. Once the TOE is in the state READY_FOR_USE, the instruction can't be used anymore. The command has the following structure and parameters:

CLA	INS	P1	P2	L_c	Data	L_e
0x01	0x09	MSB Offset	LSB Offset	Data length	Attestation certificate	0x00

If the attestation certificate will be transferred in multiple parts to the TOE, the parameters P1 and P2 define the offset of the attestation certificate data chunk that will be transferred with the actual command APDU.

The data part contains the whole or only a part of the attestation certificate. If the size of the attestation certificate exceeds the size of data that can be transferred in one APDU, the certificate can be split into multiple parts which can be transferred in separate APDUs.

Incoming attestation certificate data (parts) will be proceeded and stored in reserved internal storage. The storage size has to be set up as parameter at the installation process of the applet. This APDU generates the TOEs private keys seed and MACKey after the storage is completely filled. After the keys are generated the TOE will switch to the state READY_FOR_USE. Certificate bytes can be send in chunks of arbitrary size. As long as the storage is not completely filled, the TOE will stay in the state UNINITIALIZED. This function will not check if the stored private key matches the received attestation certificate.

The Response APDU contains no data field. The SW will be 0x9000 for successful processing the Command APDU or 0x6A80 if the size of the of all previous data in the SET_ATTESTATION_CERT command APDUs plus the actual data field size exceeds the size that was defined at the installation process of the applet.

If the token is already in state READY_FOR_USE the token will reject the command with SW 0x6982.

3.2 RESET

This APDU is used to request a reset of the internal keys and the FIDO signing counter.

CLA	INS	P1	P2	L_c	Data	L_e
0x00	0x8E	0x5E	0x70	-	-	0x00

Parameter P1 has to be 0x5E and P2 has to be 0x70. Otherwise the command will not be executed.

This APDU expects no data and will ignore incoming data.

This function will check if user presence check is needed, enforce it if needed and check if the internal state is READY_FOR_USE. The keys seed and MACKey will be securely cleared. After successful clearing the keys the FIDO signing counter will be reset to zero and new keys for seed and MACKey will be generated by using a DRG.4.

The response APDU will contain no data field. The SW bytes will be

- SW 0x9000 if the reset was successful or
- SW 0x6200 if the reset failed. The keys are kept untouched or
- SW 0x6985 if the user presence check failed. The keys are kept untouched.

3.3 Generic status words

Beside the status words (SW) that are command specific and described above, the applet also may return the following generic SW:

- SW 0x6982 (Security conditions not fulfilled): Will be returned if proprietary CLA byte (0x01) was used.
- SW 0x6D00 (Instruction not supported): Will be returned if CLA byte was 0x00, but the INS byte doesn't contain any of the known instructions.
- SW 0x6E00 (Class not supported): Will be returned if CLA byte was neither set to 0x00 nor 0x01.

4 Product Integrator Guidance

At the composite product integration process the de.fac2 applet will be installed onto the JavaCard platform and the FIDO attestation certificate will be set. The installation process steps and the parameters needed to bring the token into the READY_FOR_USE state are described in this chapter.

The applets cap file must be installed via the platforms Global Platform Manager. The applets constructor expects the following parameters which shall be provided at the installation of the cap file.

- 1 byte user presence check flag: If this flag is set to 0x01, the user presence check of the token will be disabled. This is helpful for automated testing (e.g. FIDO NFC interoperability test). If this flag is set to 0x00 the user presence check is active. **This flag has to be set to 0x00 for the production of the final TOE.**
- 2 bytes length of attestation certificate (big endian encoded) in bytes: length of the attestation certificate which will be loaded into the token via the SET_ATTESTATION_CERT command.
- 32 bytes private key: The private key of the attestation certificate which will be loaded into the token via the SET_ATTESTATION_CERT command.

The install parameter is a concatenation of the all parameter bytes described above. Its length is 35 bytes. All other lengths will abort the cap file installation.

After the installation the attestation certificate has to be uploaded into the token with the SET_ATTESTATION_CERT command. The structure and parameters for this command are described in chapter 3.1. The product integrator has to ensure that the uploaded attestation certificate matches with the parameters which were set at the installation. This means the length of the attestation certificate matches the given length and the private key which was set at the installation is the corresponding key to the public key in the attestation certificate. The TOE doesn't verify if the keys are corresponding by itself.

When the length which was set at the installation is reached by using one or multiple SET_ATTESTATION_CERT commands, the TOE will store the attestation certificate and switch its internal state to READY_FOR_USE. Changing the certificate or its keys is not possible once this state is reached.

Glossary

U2F Universal Second Factor

For further FIDO related terms see the „FIDO Technical Glossary“ of [FIDOSpec].

Reference Documentation

FIDO_U2F_Msg	FIDO Alliance: FIDO U2F Raw Message Formats, https://fidoalliance.org/specs/fido-u2f-v1.2-ps-20170411/fido-u2f-raw-message-formats-v1.2-ps-20170411.pdf
FIDOSpec	FIDO Alliance: Fido Alliance Universal 2nd Factor 1.1 Specifications, https://fidoalliance.org/specifications/download/
ISO 7816-4	ISO/IEC: Identification cards - Integerated circuit cards, Part 4: Organization, security and commands for interchange