Digital Logic

uFR Desfire Example - Simple Android v1.0

Table of contents

About

Software example written in Java programming language for Android, showcasing card read/write and card key change operations with Desfire® cards.

Git repository: https://www.d-logic.com/code/nfc-rfid-reader-sdk/ufr-ds-examples-android

uFR Series NFC Reader API: https://www.d-logic.com/code/nfc-rfid-reader-sdk/ufr-doc.git

Usage

Communication with the reader

Communication with uFR Series readers' can be established by utilizing options provided:

- **Reader type**: Based on the selection, baudrate is predetermined.
  - 0 - Auto, will test device communication at 1Mbps first, then 115200 and finally 250Kbps. OTG cable connection only.
  - 1: 1Mbps baudrate, OTG cable connection only.
  - 2: 115200bps, OTG cable connection only.
  - 3: for uFR Online series only, will display additional options **port name** and **port interface**

- **Port name** : depending on **port interface** this field requires:
  - **IP address**: when **port interface** is set to either UDP or TCP
  - **Serial number** or **mac address**: when **port interface** is set to BT Serial or BLE

- **Reader args**: Provided additional options for communication. Useful when the device has non standard baudrate or reset condition changed/disabled.

Authentication

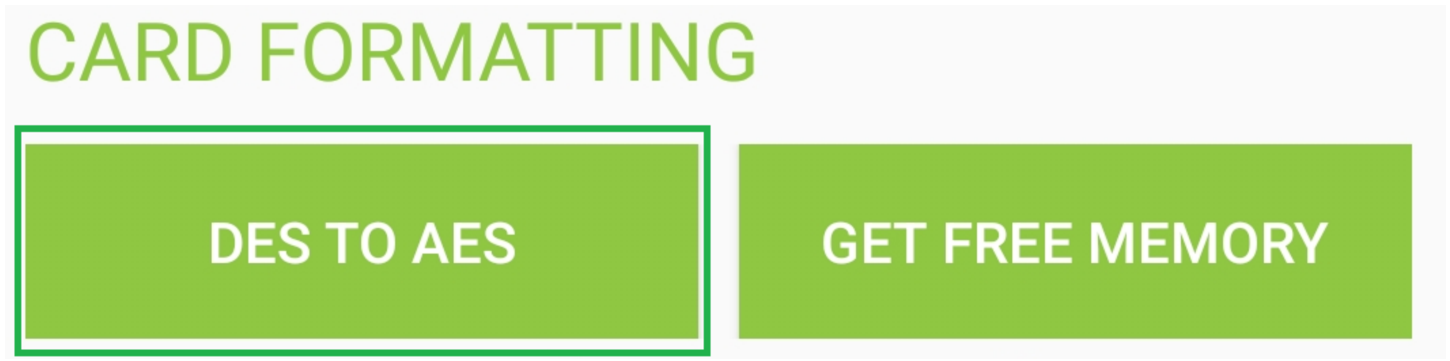
Important: This software will use **AES** keys and the reader key stored on index 0 for all card operations in the **MAIN** tab.
Any changes to this reader key will affect future card operations.

Card formatting

This software includes the following:

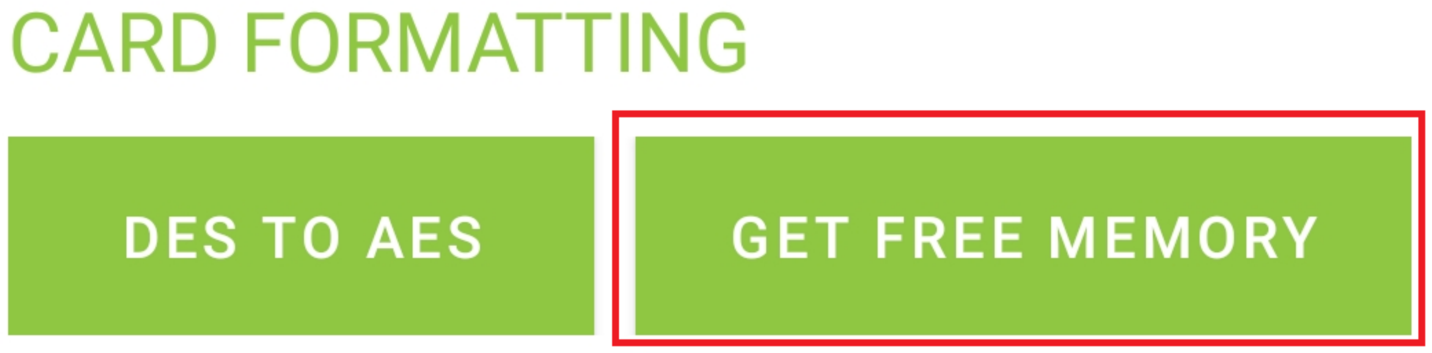
- Switching card master key from DES to AES
- Getting available free memory
- Formatting card via AES master key from the reader or providing one in the designated field

DES to AES



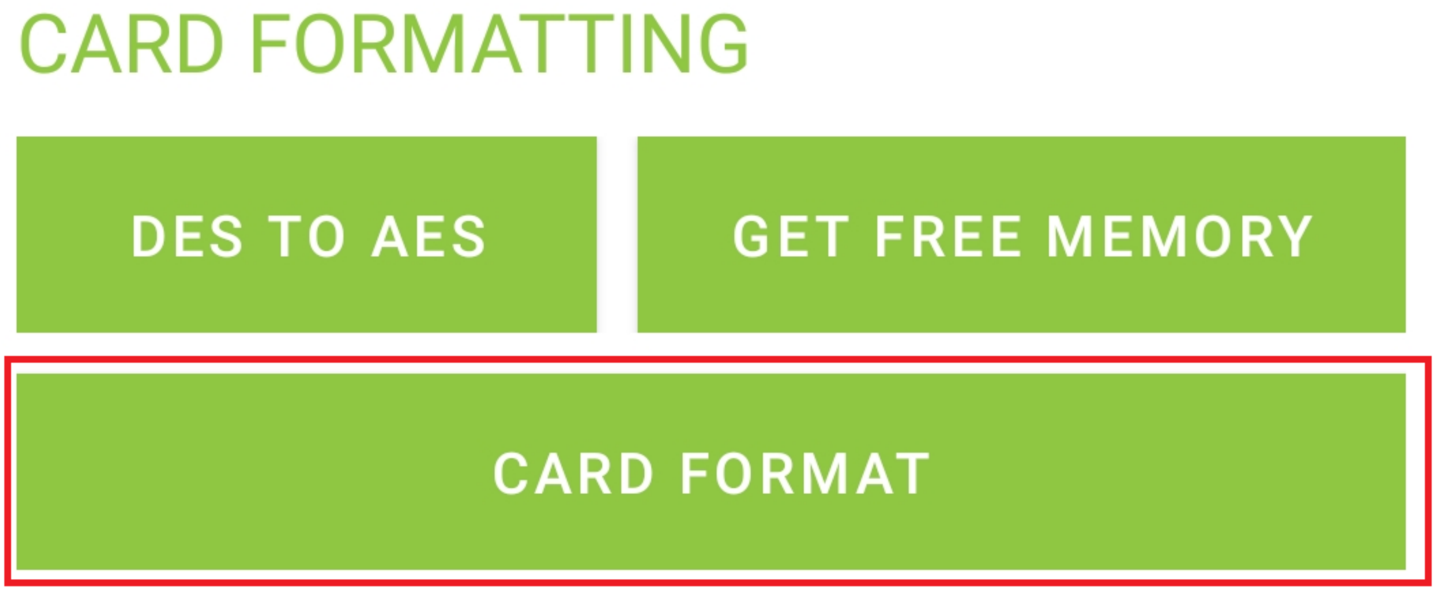
By using the provided **DES TO AES** button, the user will switch the card master key from the default DES key to the AES key.
This option relies on using default DES key (8 0x00 hex bytes) to authorize key change to AES (new key will be 16 0x00 hex bytes)

Get free memory



Button **GET FREE MEMORY** simply serves the purpose of getting information about free memory left on the tag.
No authentication is necessary.

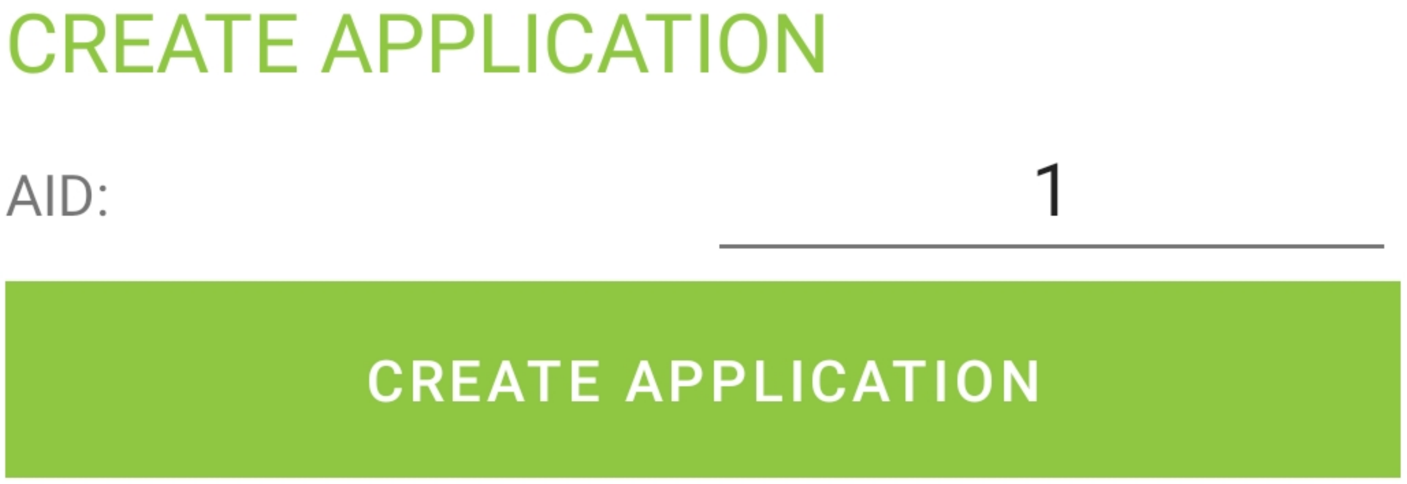
Card format



Card format serves to wipe the tag clean and reset back to defaults. All the created applications and files on the card will be erased.
This method relies on the card's **master** key for formatting.

Creating the applications and files

Create application



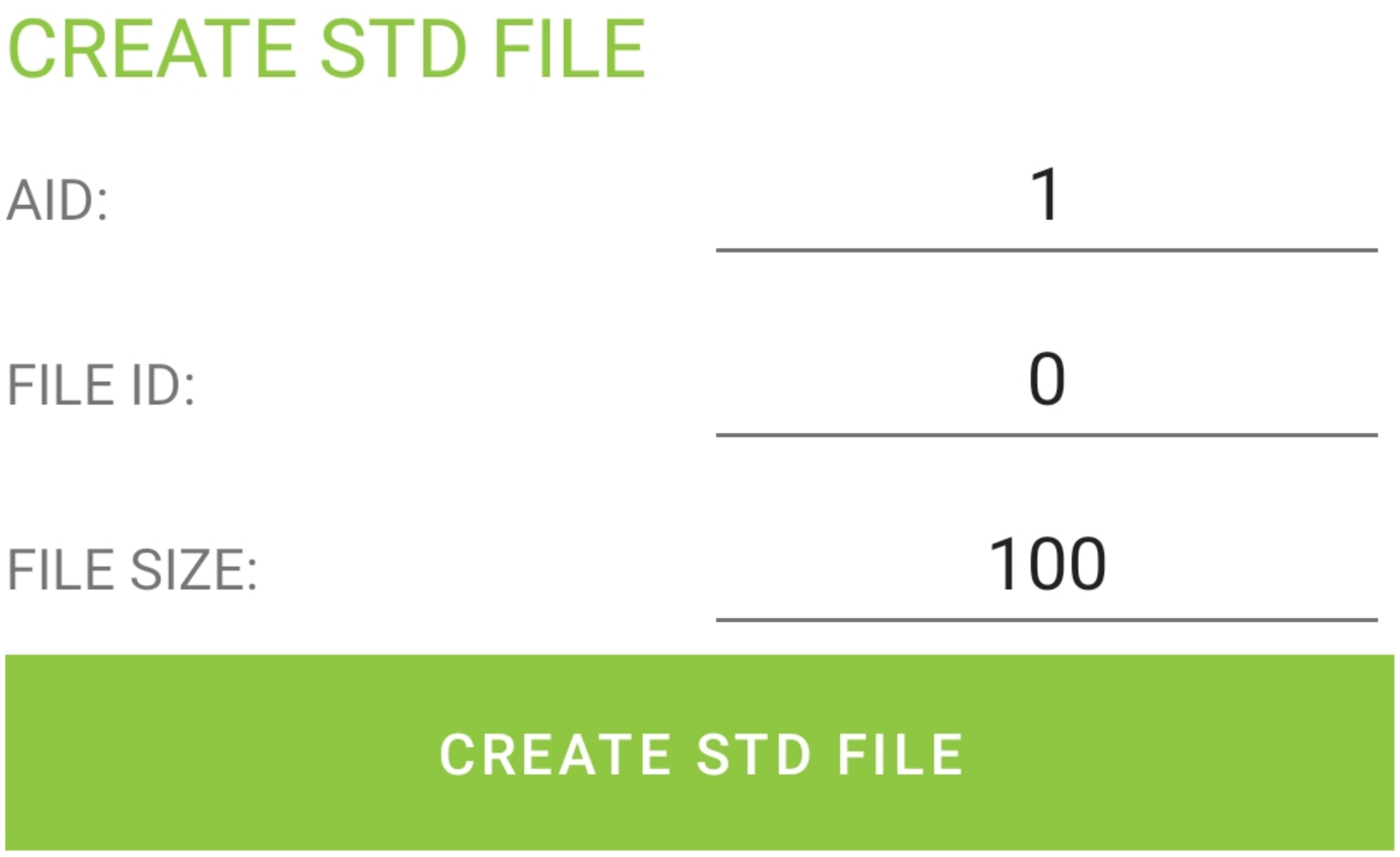
By providing a valid AID (Application Identifier) in the designated text field and using the **CREATE APPLICATION** button, a new Desfire® application will be created on the tag.
Valid AIDs for creation by default are in the range from 0x000001 to 0xFFFFFF
Application ID 0 (0x000000) is usually already defined and contains the card's master key.
Newly created application has predefined parameters:

- Application settings: **Set to 0x0F -** assumed bits 1111 defined below
- Maximum key number: **Set to 1.** Application will only use key index **0**.

Application settings bits

Bit 3	Bit 2	Bit 1	Bit 0
If set to 1 - Allows a future change of this configuration	If set to 1 - Master key is not required for creation/deletion of files	If set to 1 - allows directory list access without master key authentication	If set to 1 - allows application master key change

Create Std data file


CREATE STD FILE

AID: 1

FILE ID: 0

FILE SIZE: 100

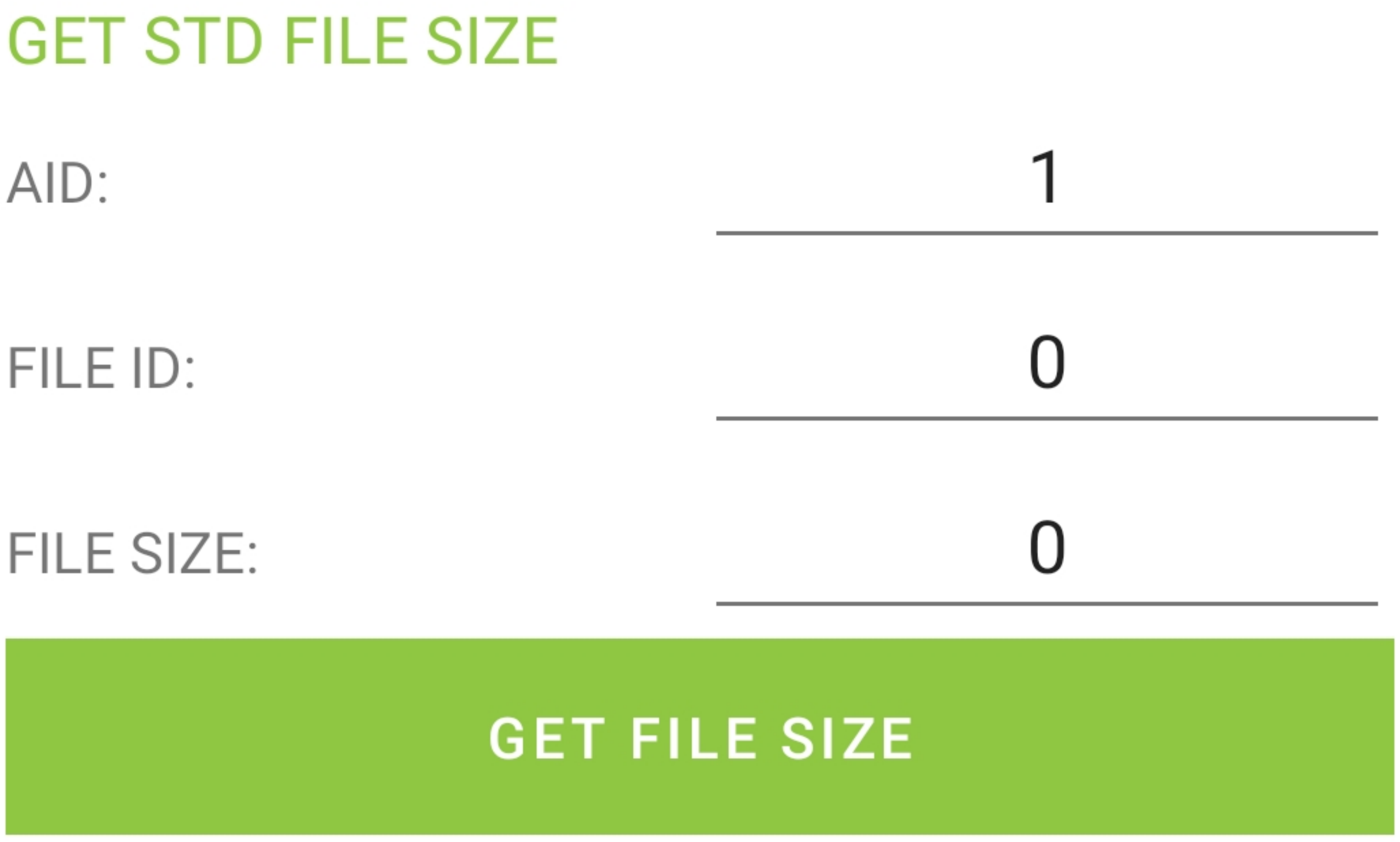
By using the **CREATE STD FILE**, a file that will be used for reading/writing operations will be created on the tag based on the parameters provided.

Valid **FILE ID**s for creation are in the range of 0-31. Every AID can have up to 32 files and the number of **AID**s can be expanded based on the card's size.

Other parameters used for creation:

- **FILE SIZE** - size reserved on the card for the newly created file.

Get file size


GET STD FILE SIZE

AID: 1

FILE ID: 0

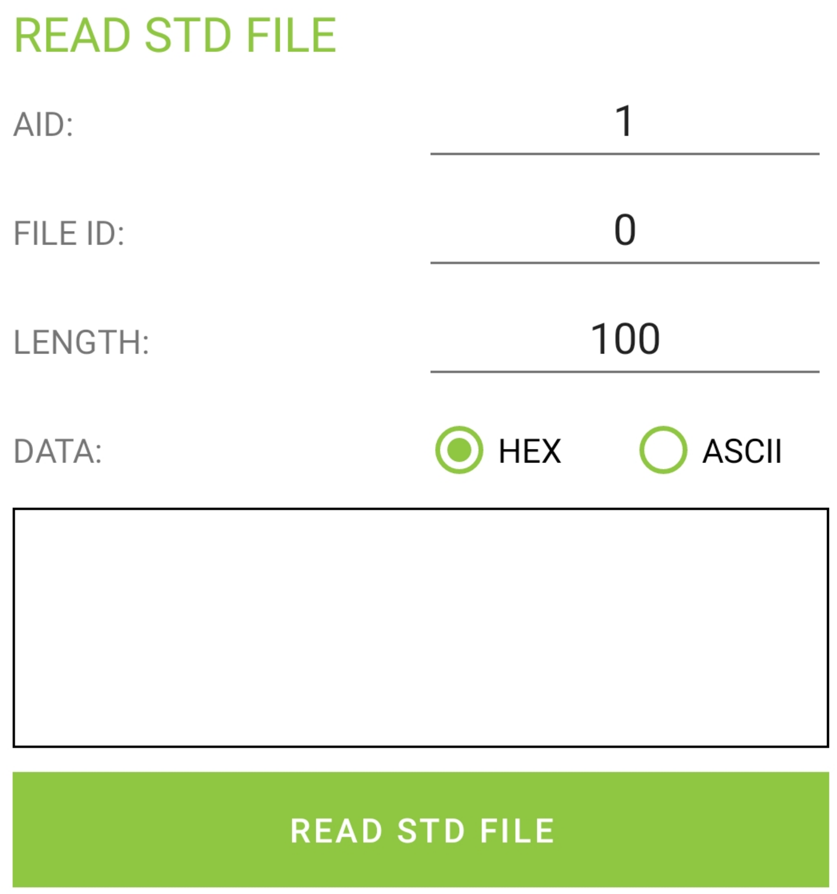
FILE SIZE: 0

By using the **GET FILE SIZE** button, the user can retrieve information about the size of the existing std data file on the card.
Valid **AID** and **FILE ID** are mandatory for this operation.
Parameters returned on success are:

- **File size:** total amount of data that this file can store.

Read/Write to file

Reading the file


READ STD FILE

AID: 1

FILE ID: 0

LENGTH: 100

DATA: [✓ HEX] ASCII

By specifying the necessary parameters and using the **READ STD FILE** button, the reader will try to read and display the data in the selected format.

Parameters for card read operation are:

- **AID**

Application identifier that contains the file

- **FILE ID**

ID of the file that will be read

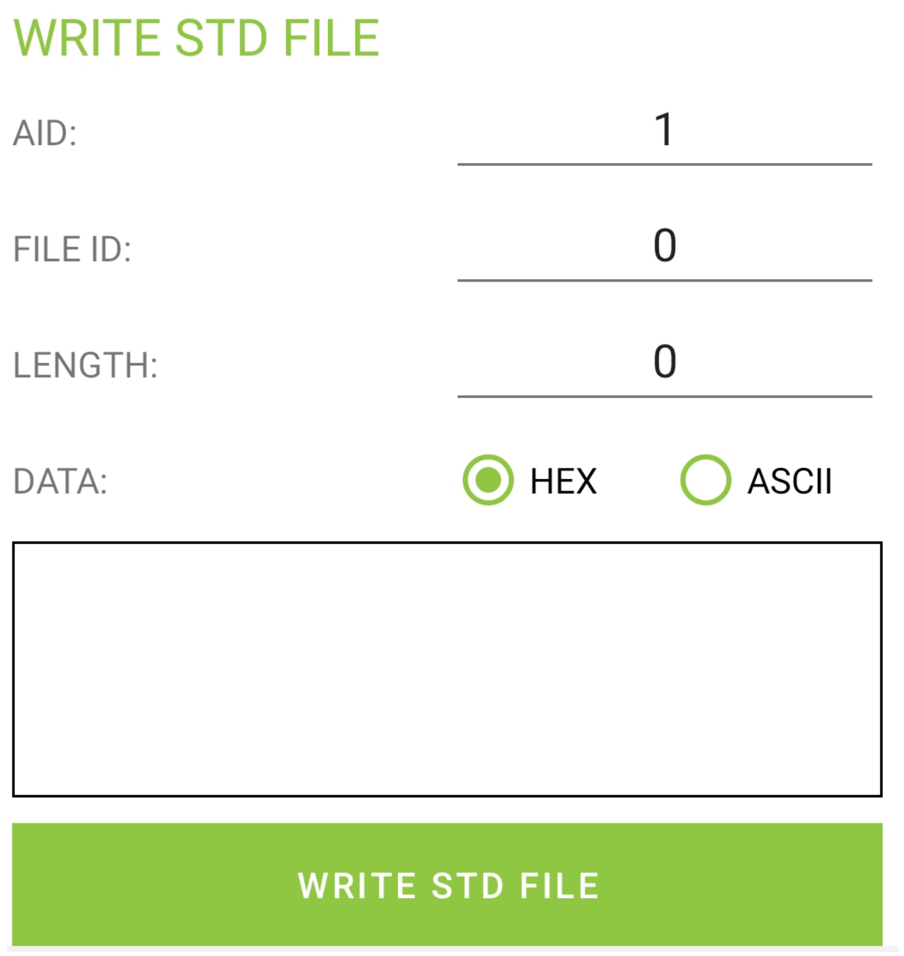
- **LENGTH**

Amount of data to be read expressed as number of bytes.

- **DATA**

Successfully read data will be displayed in this field, based on the selected **HEX** or **ASCII** format

Writing to the file


WRITE STD FILE

AID: 1

FILE ID: 0

LENGTH: 0

DATA: [✓ HEX] ASCII

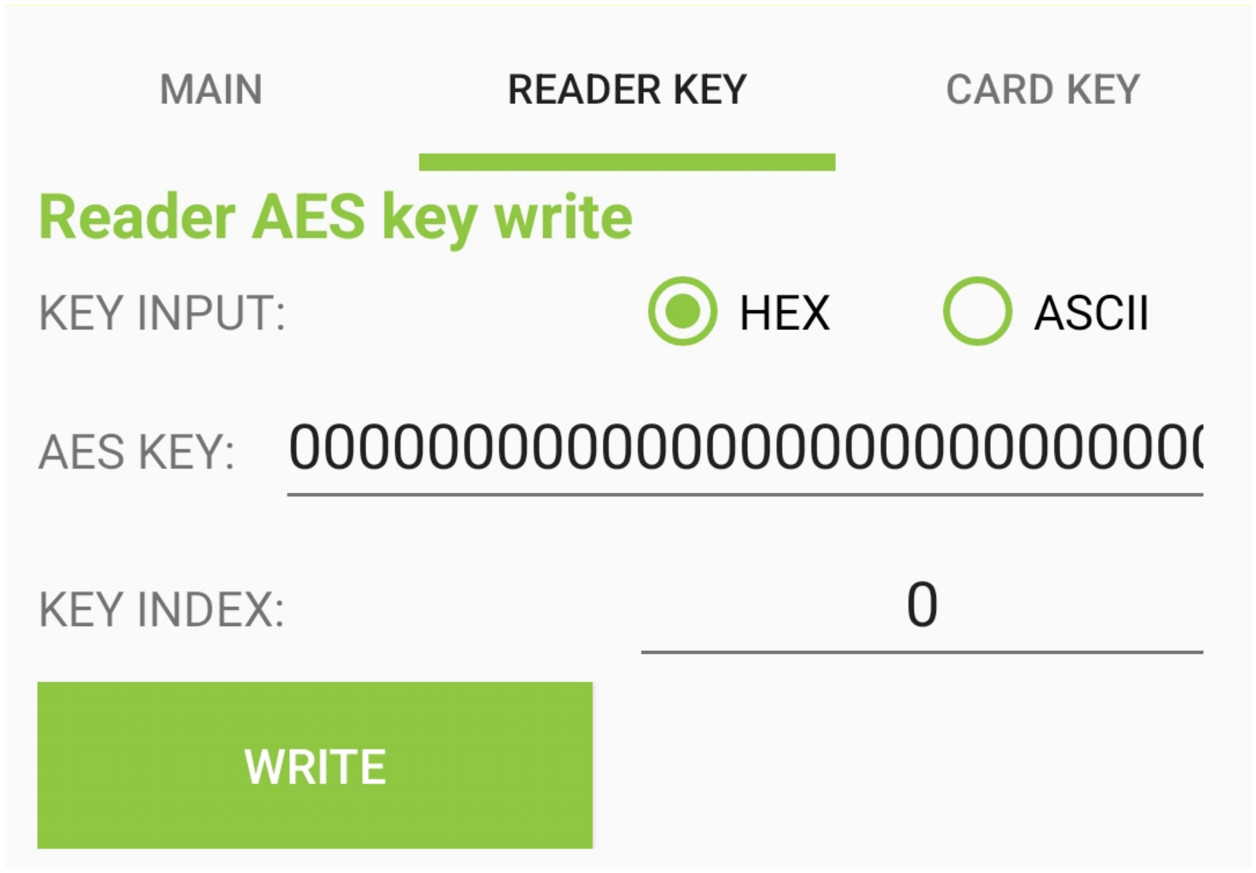
Same parameter format applies as with the previous **READ** option.

Provide the **AID** and **FILE ID** of the file, and finally input the data. The **LENGTH** field will be populated automatically as the user enters the data and will display the length of the current input. Length will be calculated based on the format selected (HEX/ASCII).

Finally, use the **WRITE FILE** button to store the data on the tag, in the selected format.

Reader key write

Switch to the tab **READER KEY** to gain access to these options.



Reader AES key write

KEY INPUT: [✓ HEX] ASCII

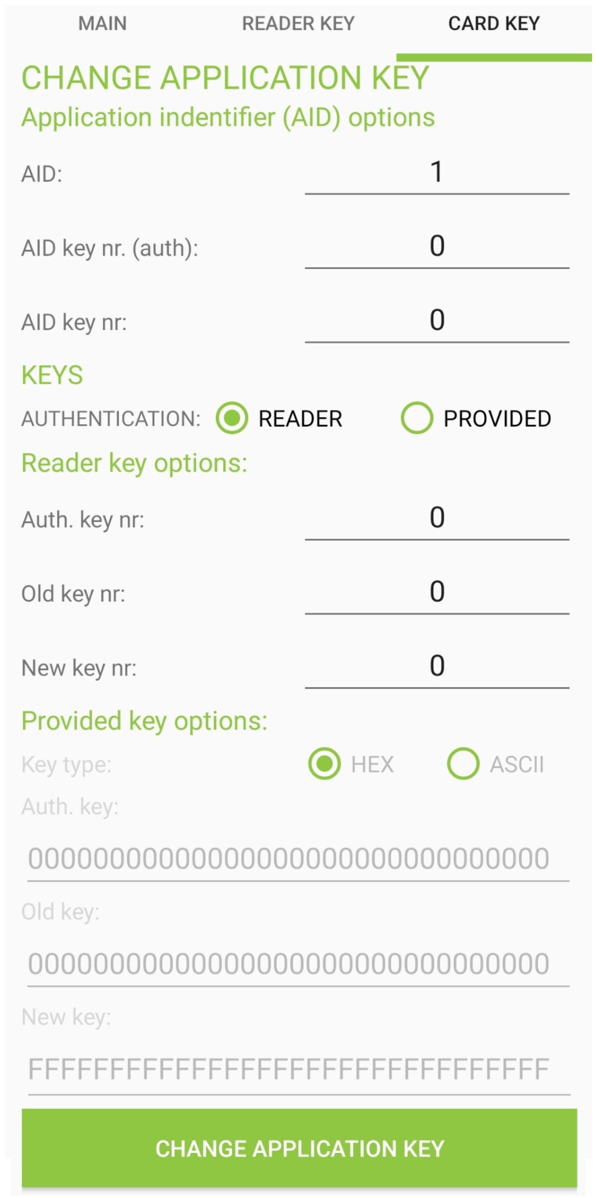
AES KEY: 0000000000000000000000000000000(

KEY INDEX: 0

By using the **WRITE KEY** button, the AES key will be stored in the reader based on the **KEY INDEX** provided. The available key index range for AES keys is 0-15.

Card key write

Switch to the tab **CARD KEY** to gain access to these options.



CHANGE APPLICATION KEY
Application indentifier (AID) options

AID:	1
AID key nr. (auth):	0
AID key nr:	0

KEYS
AUTHENTICATION: [✓ READER] PROVIDED
Reader key options:

Auth. key nr:	0
Old key nr:	0
New key nr:	0

Provided key options:
Key type: [✓ HEX] ASCII
Auth. key: 00000000000000000000000000000000
Old key: 00000000000000000000000000000000
New key: FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

These options give access to changing keys stored on the Desfire® card based on the parameters provided and card key settings.
When changing the key for **AID 0** - you will be notified that it will be the **card master key** that will be changed, in this case, parameters **AID key nr. (auth.)** and **AID key nr.** will be ignored since they are related to the other card applications that contain more than one key and have their AID different from 0.

Parameters

Application identifier options (AID):
AID - Application Identifier whose key will be changed
AID key nr (auth.) - Index of the application key that will be used to authenticate key changing.
AID key nr. - Index of the application key that will be changed

Key type: Specifies the input source of the keys. Whether using the keys stored in the uFR reader or providing them manually in the designated fields below (Under **Provided key options**)

Reader key options:
Auth. key nr - Reader key index that will be used for the authentication
Old key nr - Key that will be changed, provided from the reader.
New key nr - New key to be stored in the card, provided from the reader.

Provided key options:
Keys format - Specifies format of the provided keys. **HEX** input requires 16 hex bytes, **ASCII** requires 16 characters long keys.

Auth key - Provided key that will be used to authenticate card key change operation
Old key - Provided old key that currently exists on the card
New key - Provided new key that will be stored on the card.

Methods used for changing keys in this example are:

- **uFR_int_DesfireChangeMasterKey**(_PK) - when changes are made to the card Master key, and
- **uFR_int_DesfireChangeAesKey_aes**(_PK) - when changing card application keys

For more details on how these methods work, refer to **uFR Series NFC Reader API**:
https://www.d-logic.com/code/nfc-rfid-reader-sdk/ufr-doc.git

Revision history

Date	Version	Comment
2023-03-21	1.0	Base document