

# AN12569

EdgeLock™ SE05x for secure access control in Industrial IoT

Rev. 1.2 — 7 December 2020

Application note

565712

## Document information

Information	Content
Keywords	EdgeLock SE05x, Industrial access control, MIFARE® DESFire® EV2
Abstract	This application note describes how EdgeLock SE05x, in combination with a microcontroller, supports secure access control in any industrial operation or environment. It gives insights into the integration of EdgeLock SE05x from a hardware and software perspective for this use case. It also provides detailed instructions to run a set of code examples that demonstrate how to leverage EdgeLock SE05x and LPC55S to support secure operation with a MIFARE DESFire EV2 card. In this case, the LPC55SS is used as an example and the same concept is applicable using another host MCU.



## Revision history

---

### Revision history

Revision number	Date	Description
1.0	2019-11-25	First document release
1.1	2019-11-27	Corrected USB port in Figure 11
1.2	2020-12-07	Updated to latest template and fixed broken URLs

## 1 EdgeLock SE05x for secure access control in industrial IoT use cases

In any industrial operation or environment there is equipment and restricted areas that need to be secured, for safety reasons as well as to protect key equipment from damage or loss. IoT opens new opportunities for designing advanced access control solutions for machines, production lines, and other types of valuable items in manufacturing and industrial settings.

Many industrial operations have supply cabinets, equipment lockers or data centers that must be protected. Electronic locks can restrict access to these areas to individuals who are authorized to enter or withdraw items from these secure locations. In addition, electronic access control solutions can ensure that only authorized personnel are allowed to operate potentially dangerous machinery or secure areas that are only accessed by trained, certified employees.

State-of-the-art electronic access control solutions use contactless cards or NFC-enabled smartphones as the standard means of authentication and validation of user credentials. The use of contactless cards to provide physical or logical access to facilities or equipment are extremely secure and much easier to manage and update than conventional mechanical keys. As such, they can provide a better and more simplified means of controlling and monitoring access to devices, systems, and equipment.

In this context, the EdgeLock SE05x can be used by a card reader to setup a secure transaction with MIFARE DESFire EV2 contactless cards. On the one hand, the EdgeLock SE05x stores the MIFARE secret key, authenticates the MIFARE DESFire EV2 and exports the MIFARE session key to the host MCU. In turn, the host MCU implements the MIFARE application logic and handles the MIFARE DESFire EV2 command set and secure messaging. On the other hand, MIFARE DESFire EV2 contains a security system with tamper-resistant properties and is capable of providing confidentiality, authenticity and integrity security services. The user credential is protected in this contactless IC based on the configured access rights and the knowledge of the secret keys that fulfill these access rights.

As such, EdgeLock SE05x helps you to provide a higher level of security in your access control system by:

- **Protecting the master keys:** The master keys used for card authentication are protected inside the EdgeLock SE05x and can not be read or manipulated.
- **Authenticating the card:** EdgeLock SE05x supports the authentication protocol and the session key generation algorithm of MIFARE DESFire EV2 card.
- **Performing securely related commands:** EdgeLock SE05x supports secure key change or key diversification of MIFARE DESFire EV2 cards.

## 2 EdgeLock SE05x application diagram

The EdgeLock SE05x works as a secure access module to increase the security of your IoT-enabled card reader for physical or logical access. The EdgeLock SE05x connection to the main host MCU is done using the chip's I<sup>2</sup>C interface. The host MCU is also connected to a contactless transceiver, used to communicate with the user contactless credential. Optionally, the host MCU can also interface with a backend system and with user interface components such as keypads, LEDs, LCD screens or others as described in [Figure 1](#):

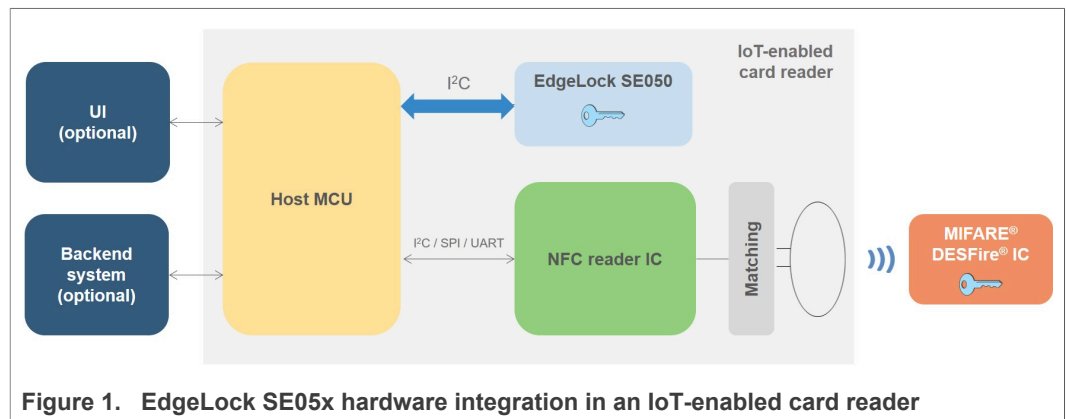


Figure 1. EdgeLock SE05x hardware integration in an IoT-enabled card reader

The OM-SE050ARD is an ideal development kit to evaluate the EdgeLock SE05x features, build a proof of concept or prototype our IoT-enabled card reader solution before going into production. It comes with headers and connectors that allow us to access the EdgeLock SE05x interfaces, including the I<sup>2</sup>C slave lines to connect a host MCU board. Check [Section 4](#) for more details about how the OM-SE050ARD is used in the EdgeLock SE05x secure access demo.

### 3 Software integration with EdgeLock SE05x Plug & Trust middleware

The host MCU implements the application logic and drives the operation of the IoT-enabled card reader. In this respect, the EdgeLock SE05x acts as a companion security IC to setup a secure transaction with a user credential stored in MIFARE DESFire EV2 contactless card. The host MCU uses the EdgeLock SE05x to authenticate the MIFARE DESFire EV2 credential and to derive the session key for each transaction. Then, this session key is exported to the host MCU and used to handle the standard MIFARE command set as part of the MCU application logic as depicted in [Figure 2](#).

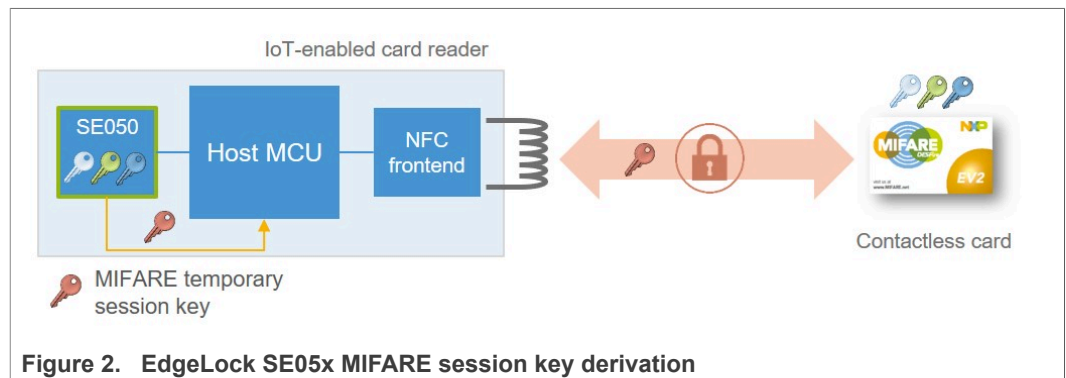


Figure 2. EdgeLock SE05x MIFARE session key derivation

The EdgeLock SE05x Plug & Trust Middleware is a single software stack designed to facilitate the integration of EdgeLock SE05x into your host MCU software. This middleware has built-in cryptographic and device identity features, abstracts the commands and communication interface exposed by the EdgeLock SE05x, and it is directly accessible from stacks like OpenSSL, mbedTLS or other cryptographic libraries. It also comes with support for various NXP MCU / MPU platforms and can be ported to multiple host platforms and host operating systems. [Figure 3](#) is a simplified representation of the layers and components which EdgeLock SE05x Plug & Trust Middleware is made of:

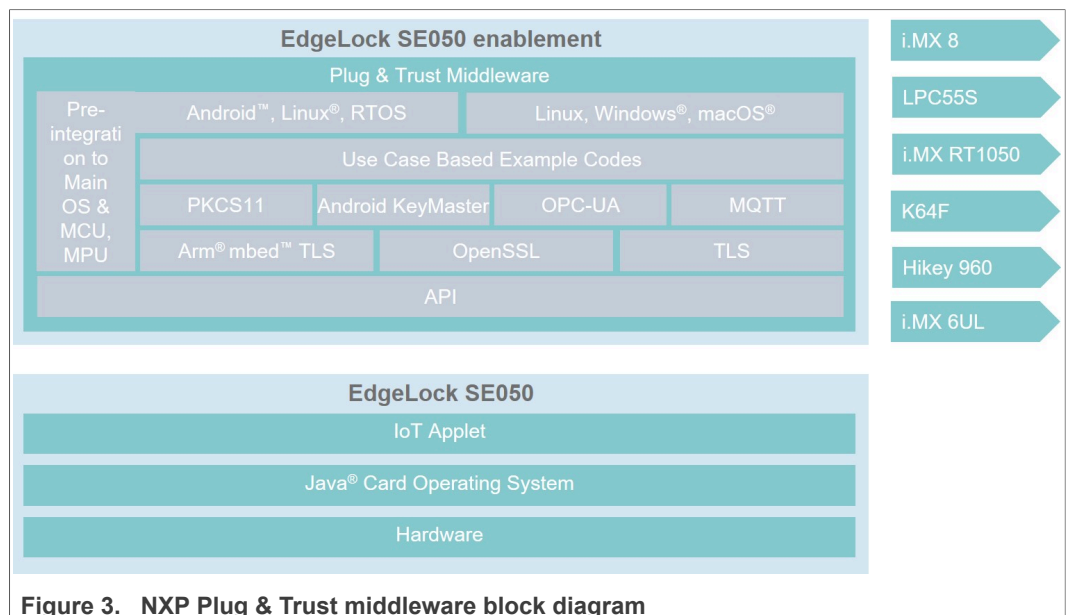


Figure 3. NXP Plug & Trust middleware block diagram

This section gives insights into the integration of EdgeLock SE05x from a software perspective. It includes an overview of the useful functions included in the EdgeLock SE05x Plug & Trust middleware for secure operation with MIFARE DESFire EV2 cards.

### 3.1 EdgeLock SE05x Plug & Trust middleware MIFARE DESFire EV2 API

The EdgeLock SE05x Plug & Trust Middleware provides functions that support MIFARE DESFire EV2 card authentication, session key generation and key personalization. Refer to MIFARE DESFire EV2 datasheet ([ds226031](#)) for more details about MIFARE DESFire EV2 commands.

#### 3.1.1 Se05x\_API\_DFAuthenticateFirstPart1()

Performs the first part of the AuthenticateEV2First three-pass mutual authentication between MIFARE DESFire EV2 and the IoT-enabled card reader. Calling this function, the EdgeLock SE05x takes care of:

- Deciphering the random number (RndB) generated by MIFARE DESFire EV2 with the selected key.
- Rotating the first byte of RndB to the end (RndB').
- Concatenating RndB' with its own generated random number (RndA + RndB') and returning them encrypted to the host MCU with the selected key

#### 3.1.2 Se05x\_API\_DFAuthenticateFirstPart2()

Performs the second part of the AuthenticateEV2First three-pass mutual authentication between MIFARE DESFire EV2 and the IoT-enabled card reader. Calling this function, the EdgeLock SE05x takes care of:

- Deciphering the RndA' received from the MIFARE DESFire EV2.
- Comparing it with the RndA generated internally. If the comparison is successful, the authentication is successful.

#### 3.1.3 Se05x\_API\_DFAuthenticateNonFirstPart1()

Performs the first part of the AuthenticateEV2NonFirst three-pass mutual authentication between MIFARE DESFire EV2 and the IoT-enabled card reader. Calling this function, the EdgeLock SE05x takes care of::

- Deciphering the random number (RndB) generated by MIFARE DESFire EV2 with the selected key.
- Rotating the first byte of RndB to the end (RndB').
- Concatenating RndB' with its own generated random number (RndA + RndB') and returning them encrypted to the host MCU with the selected key

#### 3.1.4 Se05x\_API\_DFAuthenticateNonFirstPart2()

Performs the second part of the AuthenticateEV2NonFirst three-pass mutual authentication between MIFARE DESFire EV2 and the IoT-enabled card reader. Calling this function, the EdgeLock SE05x takes care of:

- Deciphering the RndA' received from the MIFARE DESFire EV2.
- Comparing it with the RndA generated internally. If the comparison is successful, the authentication is successful.

### 3.1.5 Se05x\_API\_DFDumpSessionKeys

Returns the transaction identifier (TI) and the session key of the current active authentication with the MIFARE DESFire EV2 to the host MCU.

### 3.1.6 Se05x\_API\_DFChangeKeyPart1 ()

Performs the first part of the ChangeKey command between MIFARE DESFire EV2 and the IoT-enabled card reader. Calling this function, the EdgeLock SE05x takes care of generating the cryptogram required to update a MIFARE DESFire EV2 key

### 3.1.7 Se05x\_API\_DFChangeKeyPart2

Performs the second part of the ChangeKey command between MIFARE DESFire EV2 and the IoT-enabled card reader. Calling this function, the EdgeLock SE05x takes care of verifying the MAC returned by the MIFARE DESFire EV2 after the ChangeKey command.

### 3.1.8 Se05x\_API\_DFDiversifyKey ()

The EdgeLock SE05x creates a diversified key using a diversification input up to 31 bytes. Refer to [AN10922](#) for more details about the AES128 key diversification algorithm used by EdgeLock SE05x

### 3.1.9 Se05x\_API\_DFKillAuthentication

The EdgeLock SE05x invalidates any active authentication with a MIFARE DESFire EV2 and clear the EdgeLock SE05x internal state.

## 3.2 EdgeLock SE05x Plug & Trust middleware MIFARE DESFire EV2 API documentation

You can refer to the code documentation provided as part of the EdgeLock SE05x middleware for full details about the MIFARE DESFire EV2 support. To open the HTML documentation:

1. Go to the directory where you unzipped the EdgeLock SE05x Plug & Trust middleware;
2. Go to the `simw-top\doc` folder;
3. Double click the `index.html` to open it in your default browser;

- The documentation landing page will appear as shown in [Figure 4](#). Use the left-hand side menu to navigate through the documentation or use the search textbox to find the functions described in [Section 3.1](#).

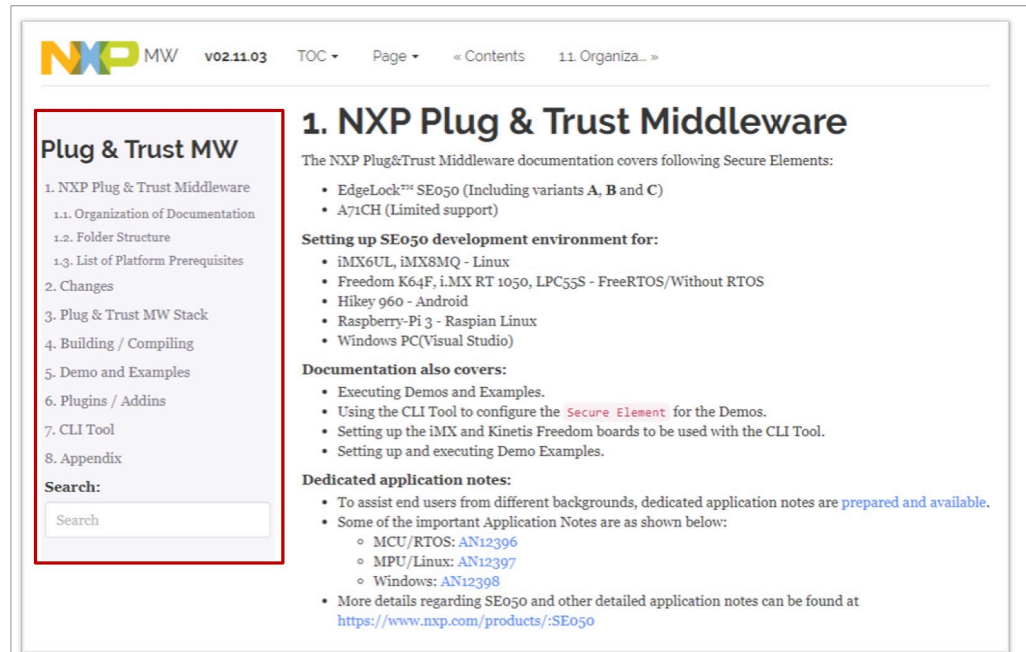


Figure 4. EdgeLock SE05x Plug & Trust MIFARE DESFire EV2 API documentation

If you are interested in the low level commands for secure operation with MIFARE DESFire EV2, you can refer to [AN12412 - SE050 APDU specification document](#).



## 4 Running the EdgeLock SE05x secure access module project examples

The EdgeLock SE05x Plug & Trust Middleware includes a set of project examples that demonstrate the use of EdgeLock SE05x to support secure operation with a MIFARE DESFire EV2 card. These project examples are:

- `ex_prepare_se05x`
- `ex_prepare_MFDFEV2`
- `ex_Ev2Auth_se05x`
- `ex_Ev2ChangeKey_se05x`
- `ex_Ev2DivChngKey_se05x`

The `ex_prepare_se05x` project example provisions the EdgeLock SE05x with sample keys and credentials. Executing this project example is a pre-requisite before running `ex_Ev2Auth_se05x`, `ex_Ev2ChangeKey_se05x` or `ex_Ev2DivChngKey_se05x` project examples.

The `ex_prepare_MFDFEV2` project example personalizes the MIFARE DESFire EV2 card with a sample application and keys. Executing this project example is a pre-requisite before running `ex_Ev2Auth_se05x`, `ex_Ev2ChangeKey_se05x` or `ex_Ev2DivChngKey_se05x` project examples.

The `ex_Ev2Auth_se05x` project example demonstrates how to authenticate and perform a sample secure transaction with a MIFARE DESFire EV2 leveraging the keys stored in the EdgeLock SE05x.

The `ex_Ev2ChangeKey_se05x` project example demonstrates how to change a MIFARE DESFire EV2 application key leveraging the keys stored in the EdgeLock SE05x

And, the `ex_Ev2DivChngKey_se05x` project example demonstrates how to change and diversify a MIFARE DESFire EV2 application key leveraging the keys stored in the EdgeLock SE05x

[Figure 5](#) depicts the setup to run the EdgeLock SE05x secure access project examples. It consists of an LPC55S board (LPC55S69), a CLRC663 development board (CLEV6630B), an EdgeLock SE05x board (OM-SE050ARD) and a MIFARE DESFire EV2 acting as the user credential.

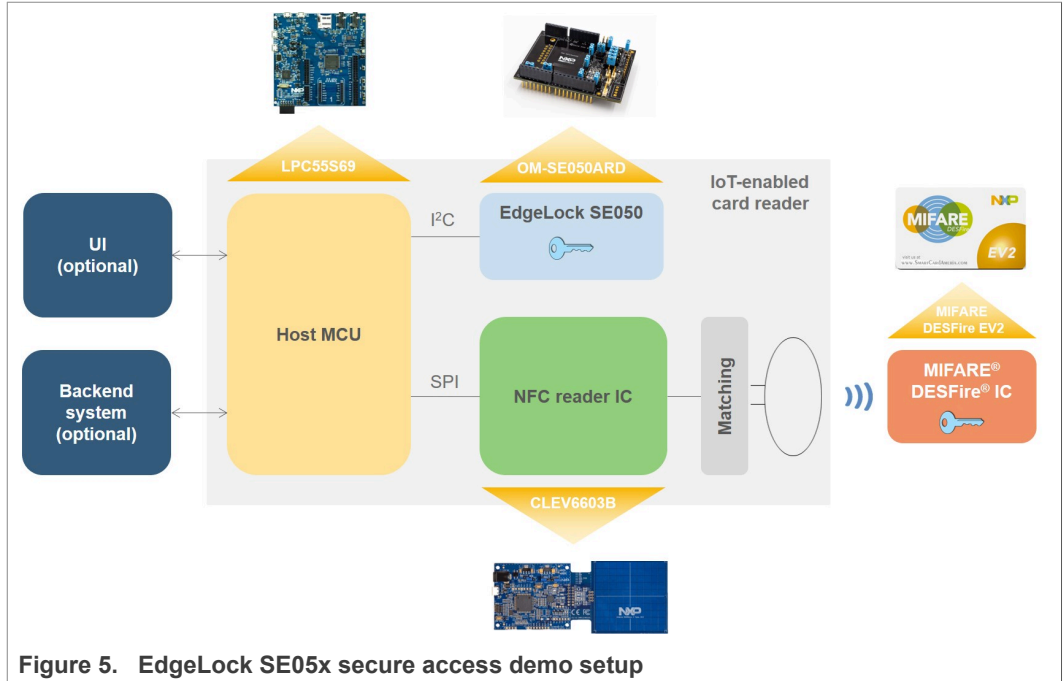


Figure 5. EdgeLock SE05x secure access demo setup

The steps required to run the project examples that use EdgeLock SE05x as a secure access module to operate with MIFARE DESFire EV2 cards are:


1. [Get the required hardware.](#)
2. [Download and install the NDA-protected LPC55S69 SDK version.](#)
3. [Prepare the boards setup and wiring.](#)
4. [Run the different project examples.](#)

#### 4.1 Hardware required

The hardware required to run the project examples related with the EdgeLock SE05x secure access demo is:

1. OM-SE050ARD development kit:

Table 1. OM-SE050ARD development kit details

Part number	12NC	Content	Picture
<a href="#">OM-SE050ARD</a>	935383282598	EdgeLock SE050 development board	


2. LPC55S69 evaluation kit

Table 2. LPC55S69 evaluation kit details

Part number	12NC	Content	Picture
<a href="#">LPC55S69-EVK</a>	935377412598	LPCXpresso55S69 Development Board	

3. CLEV6630B development board

Table 3. CLEV6630B

Part number	12NC	Content	Picture
<a href="#">CLEV6630B</a>	935339149699	CLRC663 Frontend Development Board	

4. MIFARE DESFire EV2 card. You can order samples free of charge by getting in touch with your local sales representative

4.2 Download and install the NDA-protected LPC55S69 SDK version

The project examples for the EdgeLock SE05x secure access module use case are included as part of the LPC55S69 under NDA version.

1. Download the NDA-protected LPC55S69 SDK version from [NXP Docstore](#), under the `IOT solutions /SE050` content class category.
2. Drag and drop the NDA-protected LPC55S69 SDK zip file in the *Installed SDKs* section in the bottom part of the MCUXpresso IDE.

3. Check that the NDA-protected LPC55S69 SDK is installed successfully.
  - a. Click *Import SDK examples* from the MCUXpresso IDE quick start panel.
  - b. Make sure that a picture of an LPC55S69 with a red ribbon with the label "SE050 (NDA)" on it is available as shown in [Figure 6](#)

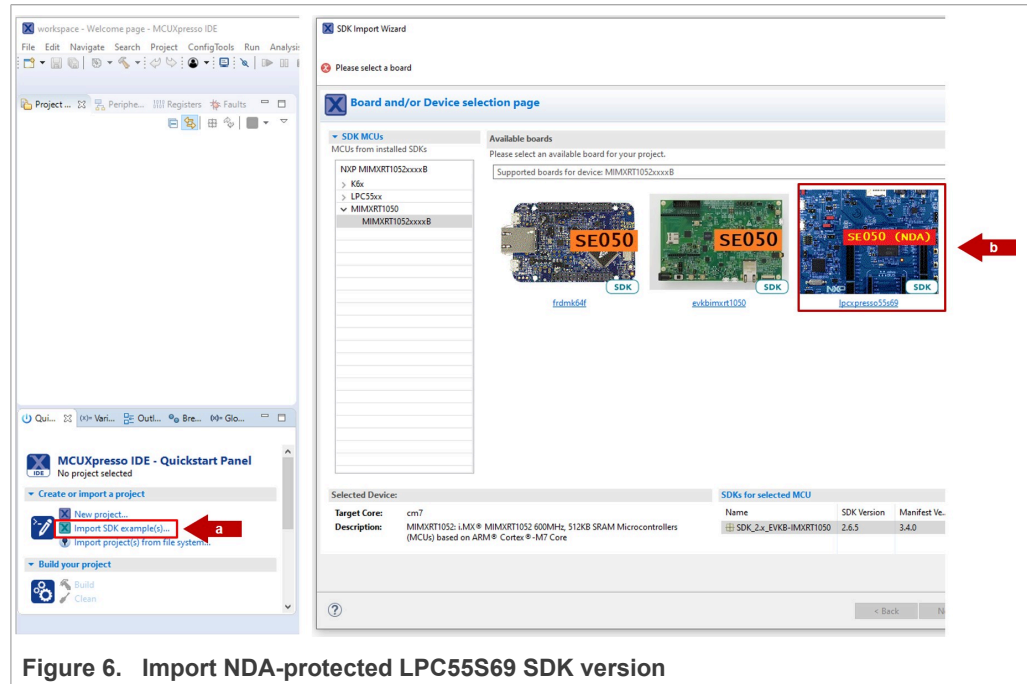


Figure 6. Import NDA-protected LPC55S69 SDK version

**Note:** For more detailed instructions on how to install the LPC55S69 SDK into our MCUXpresso workspace, refer to [AN12542 - Quick start guide with LPC55S69](#).

### 4.3 Board setup and wiring

This section describes the board setup and wiring of the OM-SE050ARD), the LPC55S69 and the CLEV6630B boards.

#### 4.3.1 OM-SE050ARD jumper configuration

We use the Arduino headers to connect the host MCU board to the OM-SE050ARD. The jumper settings to enable the I<sup>2</sup>C slave interface over the Arduino header are shown in [Figure 7](#):

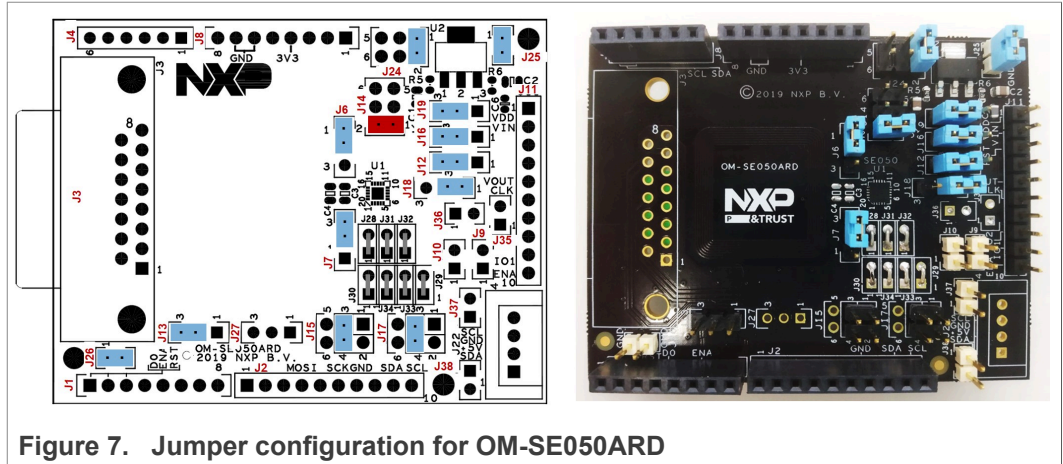


Figure 7. Jumper configuration for OM-SE050ARD

Remember to change J14 to position 1-2, which connects SE\_VDD directly to 3V3 and bypasses enable signal. This is required because enable pin on LPC55S coincides with Silex-2401 SPI pins so we cannot use SE\_EN signal to drive SE\_VDD.

#### 4.3.2 OM-SE050ARD connection with host MCU board (LPC55S69)

We use the Arduino connectors to mount the OM-SE050ARD board on top of the LPC55S69 as shown in Figure 8. Note that OM-SE050ARD should be aligned with A5 pin in LPC55S69 P19 header and D0 pin in LPC55S69 P18 header. The two last pins in P16 and the two first pins in P18 should be left open.

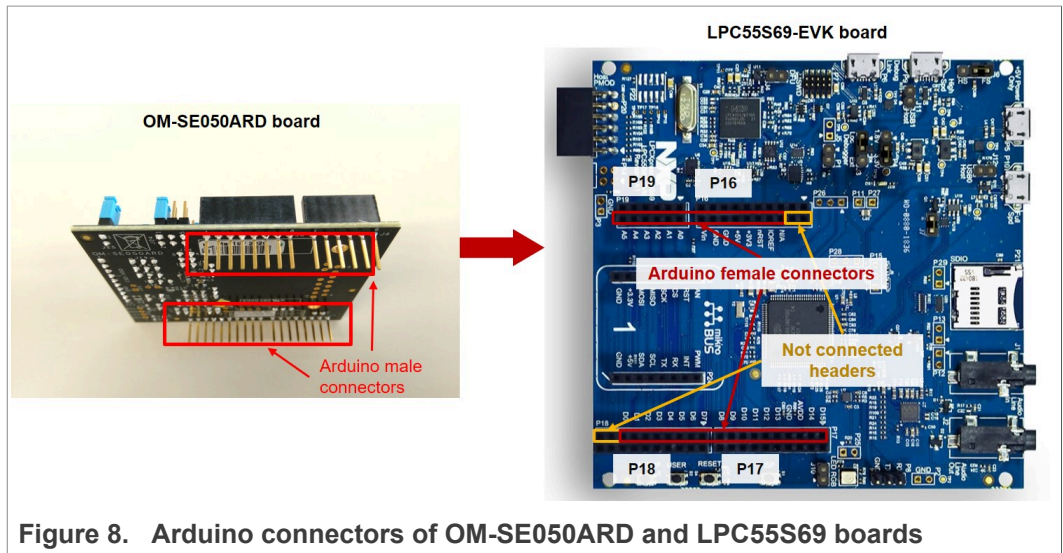


Figure 8. Arduino connectors of OM-SE050ARD and LPC55S69 boards

Double check that the two boards are connected as shown in Figure 9

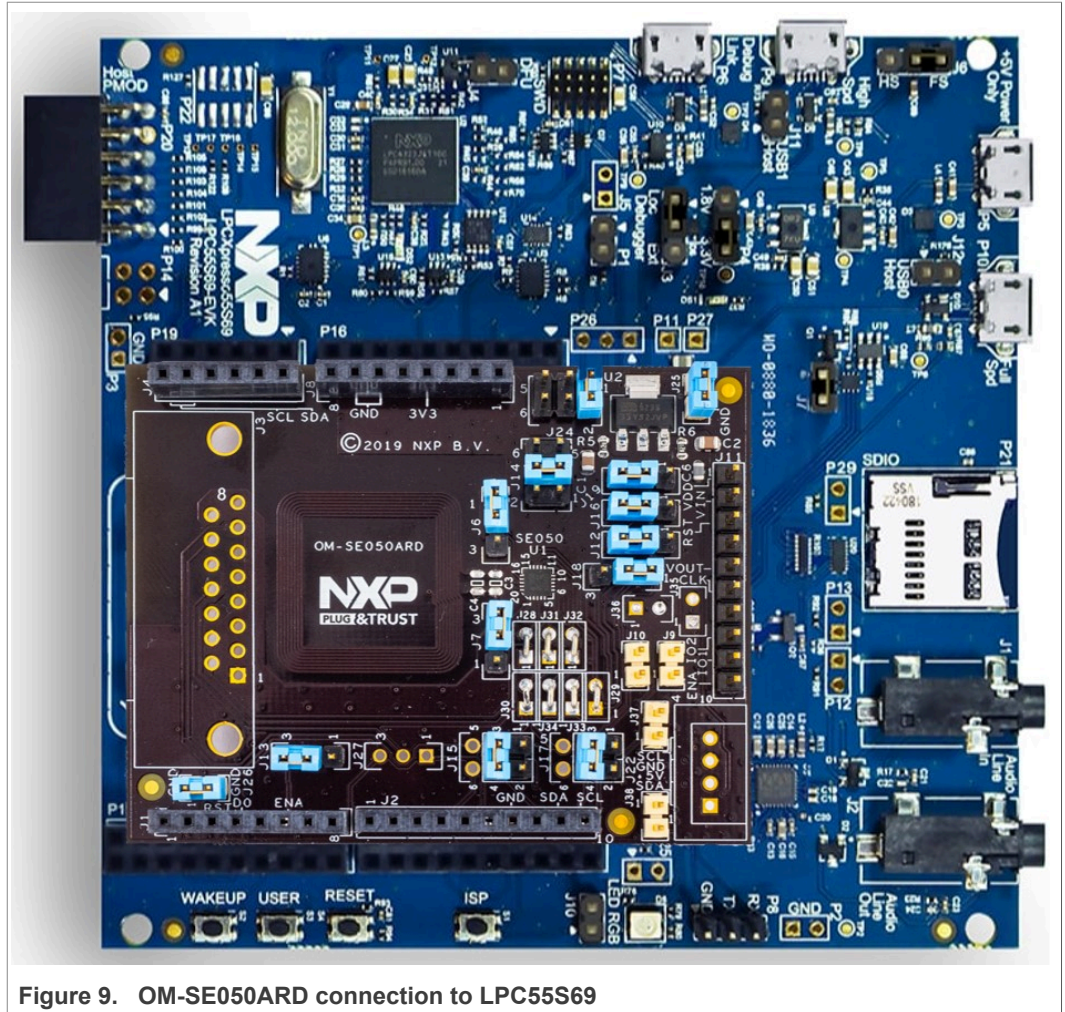


Figure 9. OM-SE050ARD connection to LPC55S69

### 4.3.3 Host MCU connection to NFC frontend (CLEV6630B)

We need to connect the LPC55S69 board via SPI to the CLEV6630B board, acting as the NFC frontend of the system. For that, we wire the LPC55S69 and CLEV6630B boards as shown in [Figure 10](#):

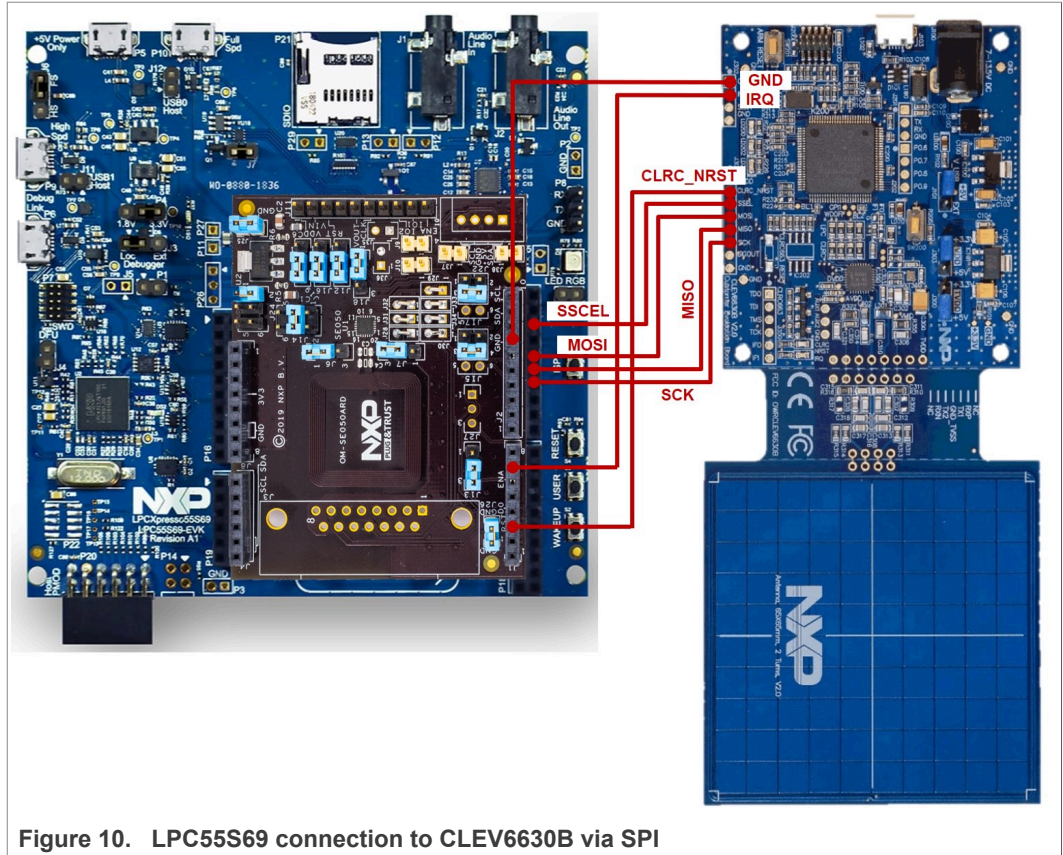


Figure 10. LPC55S69 connection to CLEV6630B via SPI

Table 4 shows the detailed connection of the LPC55S69 to CLEV6630B

Table 4. LPC55S69 connection to CLEV6630B via SPI

LPC55S69 (# jumper - # pin)	CLEV6630B (# jumper - # pin)
P18-11 (RESET)	J301-2 (CLRC_NRST)
P17-6 (SPI CSEL)	J301-3 (SSEL)
P17-10 (MOSI)	J301-4 (MOSI)
P17-12 (MISO)	J301-5 (MISO)
P17-14 (SPI SCK)	J301-6 (SCK)
P17-7 (GND)	J302-1 (GND)
P18-3 (IRQ)	J302-2 (IRQ)

#### 4.3.4 Laptop connection and TeraTerm configuration

Attach a USB cable to the LPC55S69 board debug link port and to the CLEV6630B board as shown in Figure 11.

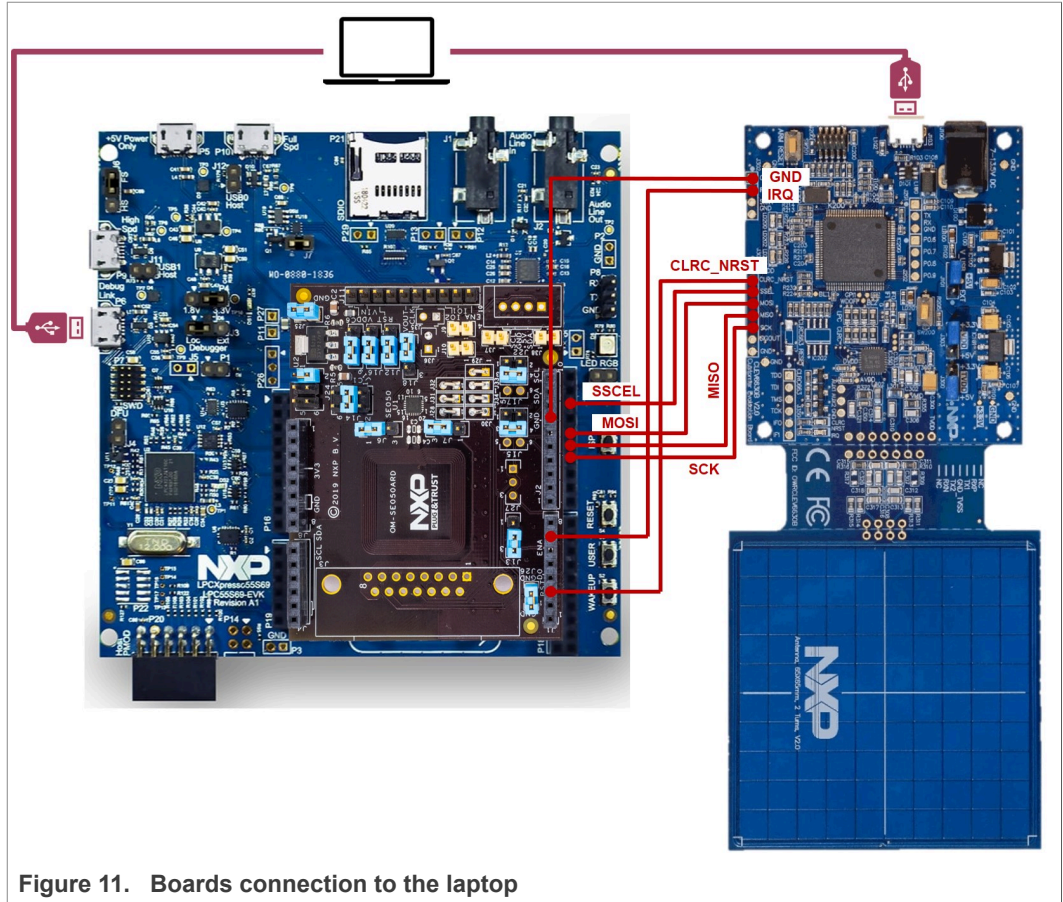


Figure 11. Boards connection to the laptop

After connecting the boards to the laptop, open a TeraTerm session and configure the serial communication, as shown in [Figure 12](#):

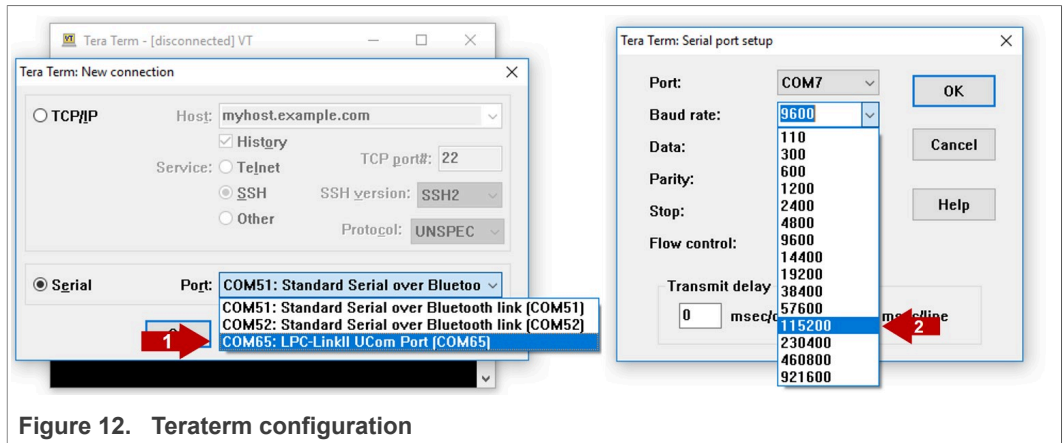


Figure 12. TeraTerm configuration

Optionally, you may configure the TeraTerm terminal window to optimize the visualization of the logs. For that, change the New-line receive setting to **AUTO** as shown in [Figure 13](#):



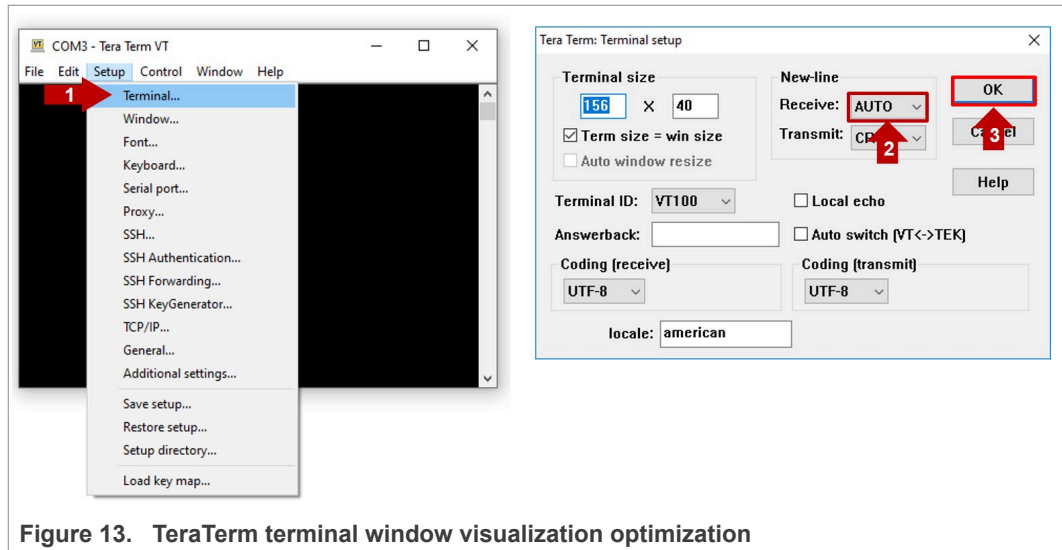


Figure 13. TeraTerm terminal window visualization optimization

#### 4.4 Project examples execution

After completing the board setup and wiring, import and run the project examples as described in this section.

##### 4.4.1 Import the project examples in MCUXpresso workspace

The LPC55S69 SDK under NDA version include project examples that use EdgeLock SE05x for MIFARE DESFire EV2 card authentication, session key generation and key personalization operations as detailed in [Section 4](#).

Import the project examples in your MCUXpresso workspace as shown in [Figure 14](#):

1. Click *Import SDK examples* from the MCUXpresso IDE quick start panel.
2. Select the LPC55S69 board from the list of available boards. The board picture should include a red ribbon with the label "SE050 (NDA)" on it as shown in [Section 4.2](#).
3. Select the following project examples from the list and click the *Finish* button:
  - a. se\_SE05x\_ex\_Ev2Auth\_SE05x
  - b. se\_SE05x\_ex\_Ev2ChangeKey\_SE05x
  - c. se\_SE05x\_ex\_Ev2DivChngKey\_SE05x
  - d. se\_SE05x\_ex\_Prepare\_MFDFEV2
  - e. se\_SE05x\_ex\_Prepare\_SE05x
4. The project should now be visible in your MCUXpresso workspace.

**Note:** For detailed instructions on how to import project examples from LPC55S69 SDK, check [AN12542 - Quick start guide with LPC55S](#)

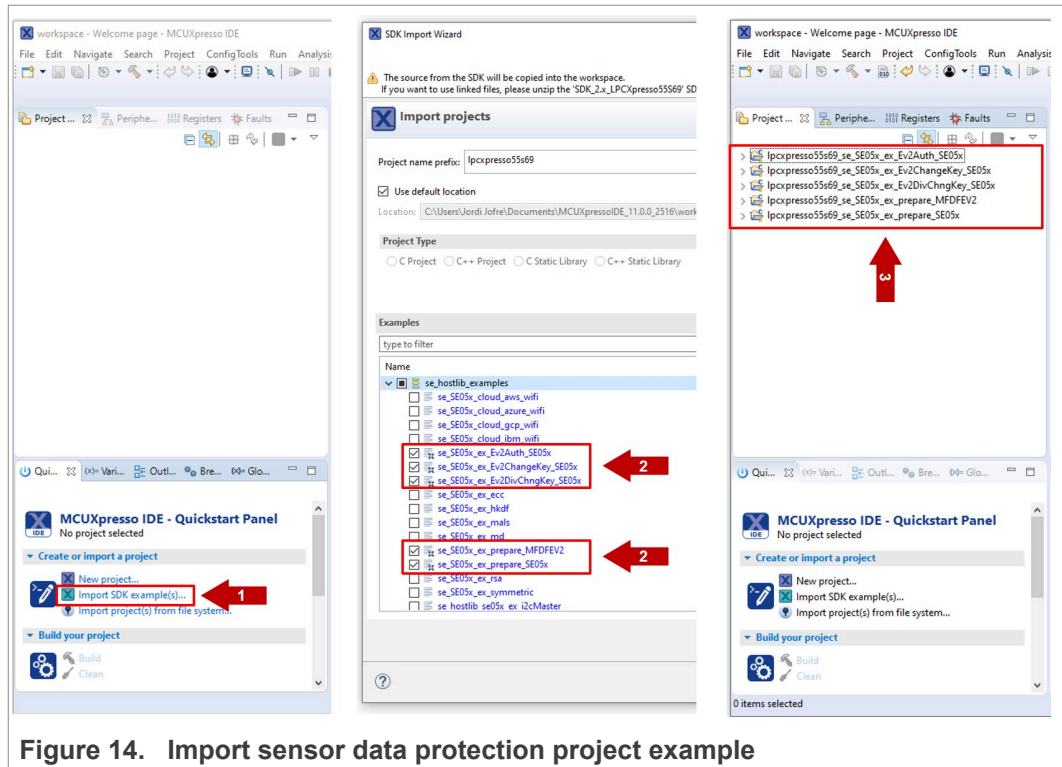


Figure 14. Import sensor data protection project example

#### 4.4.2 Provision EdgeLock SE05x (ex\_prepareSe050)

The first step is to inject in the keys that will be used for authenticating the MIFARE DESFire EV2 card in the EdgeLock SE05x. This can be done by executing the `ex_prepareSe050` project example, which injects two AES keys into the EdgeLock SE05x as depicted in [Figure 15](#)

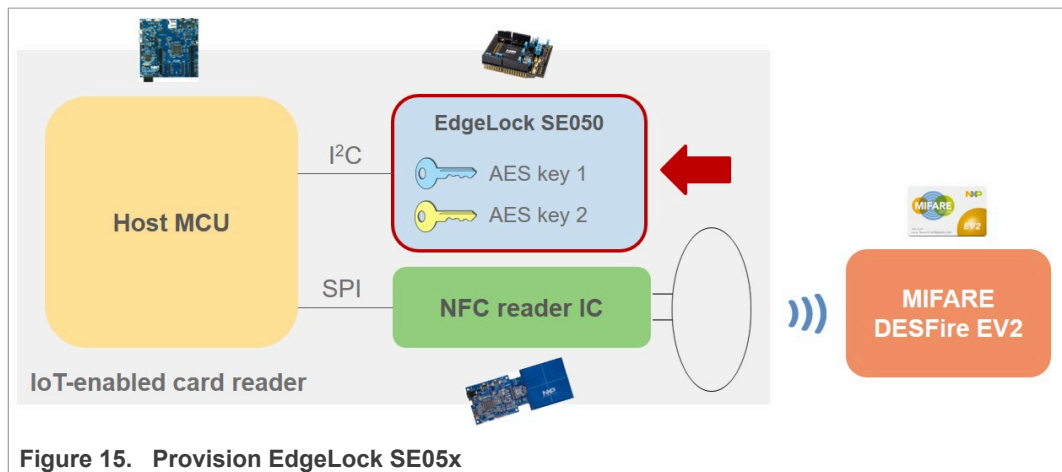


Figure 15. Provision EdgeLock SE05x

To run the `ex_prepareSe050` project example, follow these steps:

1. Select the `ex_prepareSe050` project example from your workspace.

- Go to the MCUXpresso Quickstart Panel and click *Debug* button, wait a few seconds until the project executes and click on *Resume* as shown in [Figure 16](#)

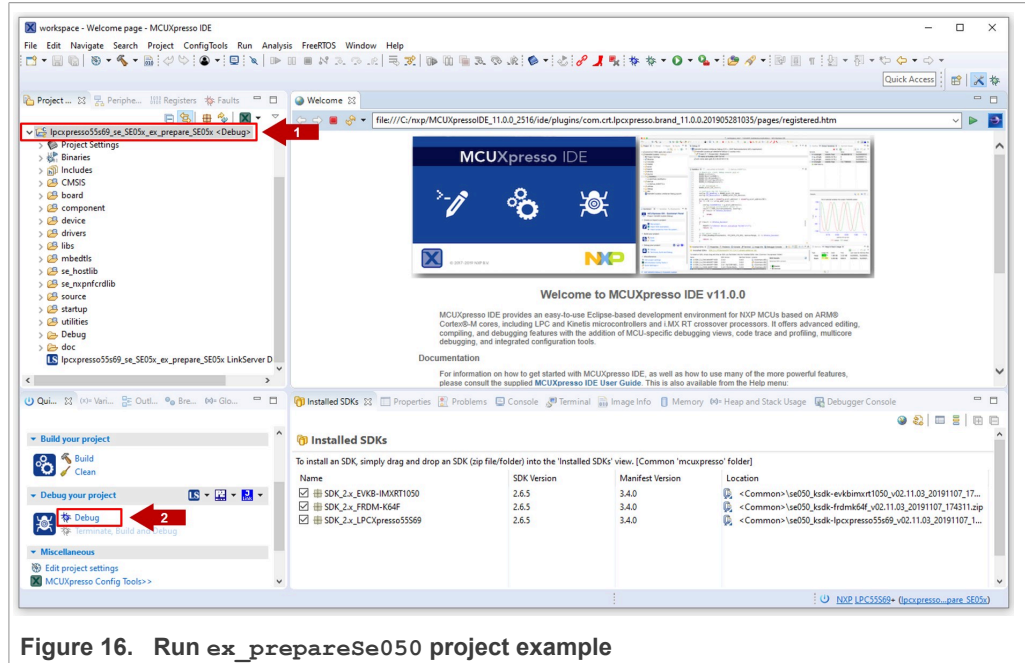


Figure 16. Run ex\_prepareSe050 project example

- Go to TeraTerm, you should see in logs similar to the one shown in [Figure 17](#):

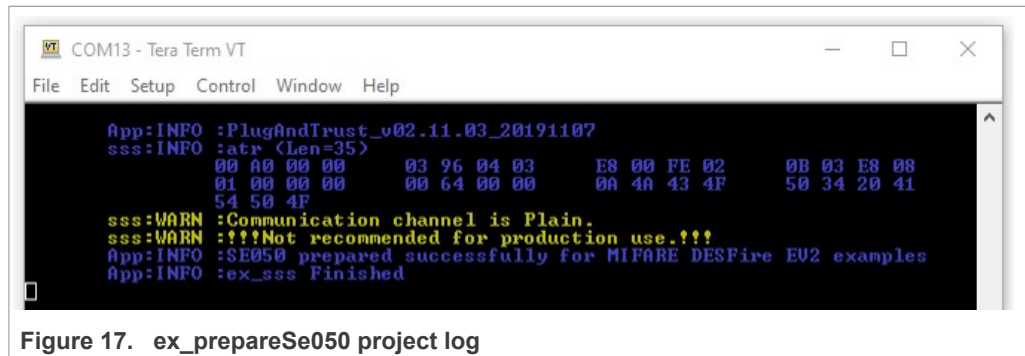
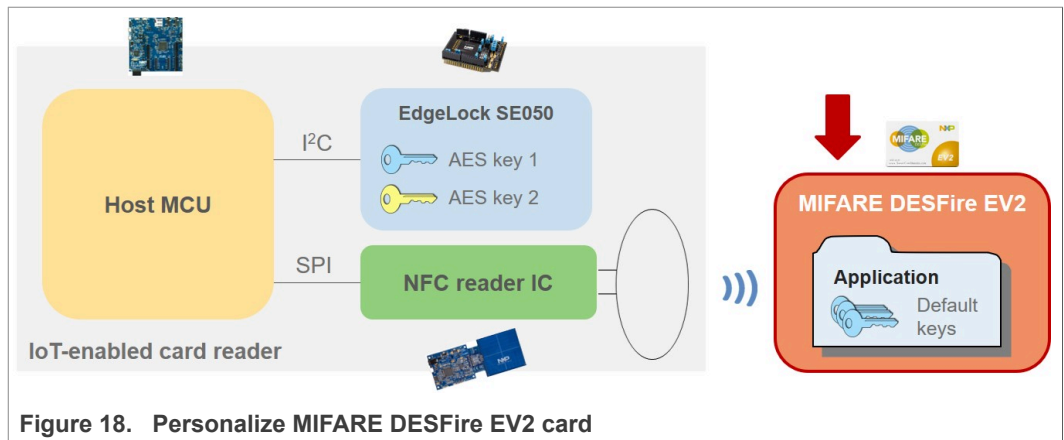


Figure 17. ex\_prepareSe050 project log

#### 4.4.3 Personalize MIFARE DESFire EV2 card (ex\_prepareMFDFEV2)

The second step is to personalize a blank MIFARE DESFire EV2 card. This can be done by executing the ex\_prepareMFDFEV2 project example, which formats the card and creates a sample application with default key values inside the MIFARE DESFire EV2 card as depicted in [Figure 18](#):

**Note:** The card personalization performed by the ex\_prepareMFDFEV2 project example is provided for illustrative purposes. This card personalization needs to be adapted based on the card memory map required for each specific application.



To run the `ex_prepareMFDFEV2` project example, place your blank MIFARE DESFire EV2 card over the CLEV6630B antenna and follow these steps:

1. Select the `ex_prepareMFDFEV2` project example from your workspace.
2. Go to the MCUXpresso Quickstart Panel and click *Debug* button, wait a few seconds until the project executes and click on *Resume*

- If the card personalization succeeds, your Teraterm window should look similar to [Figure 19](#):

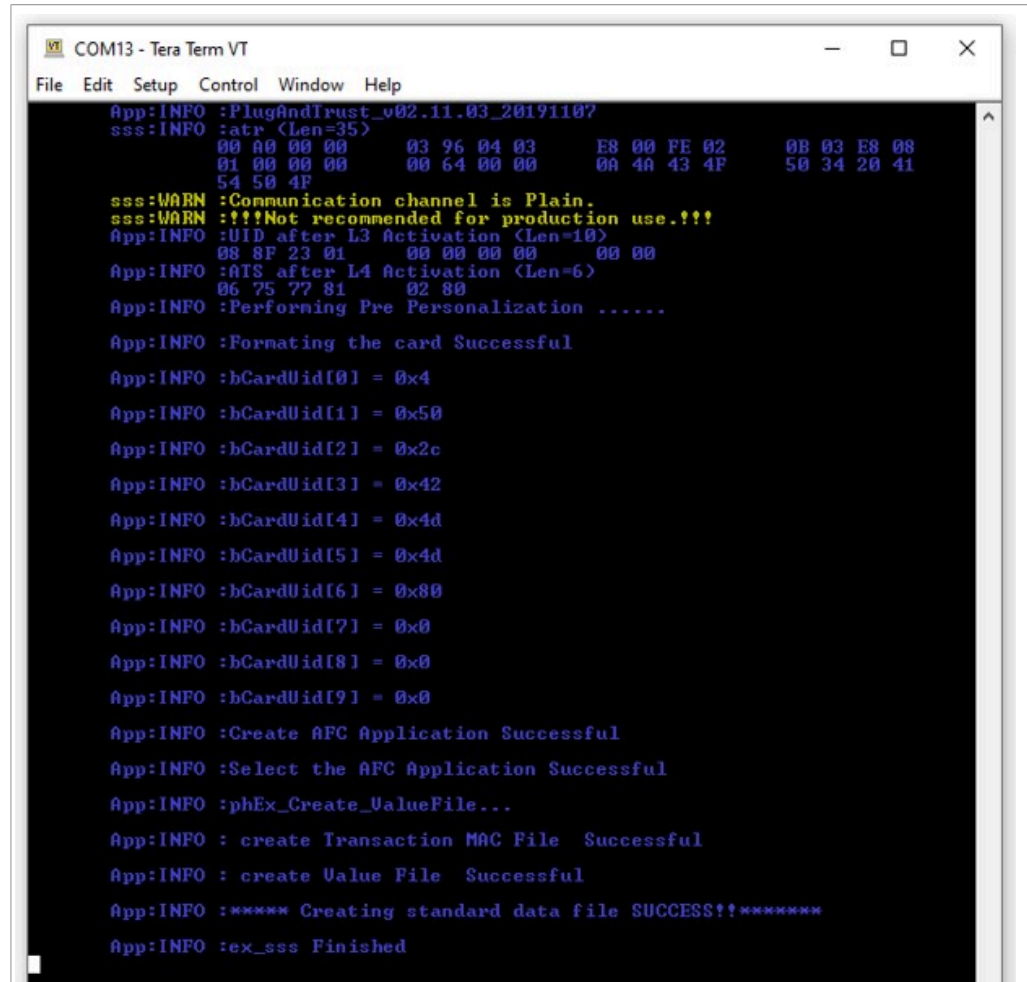
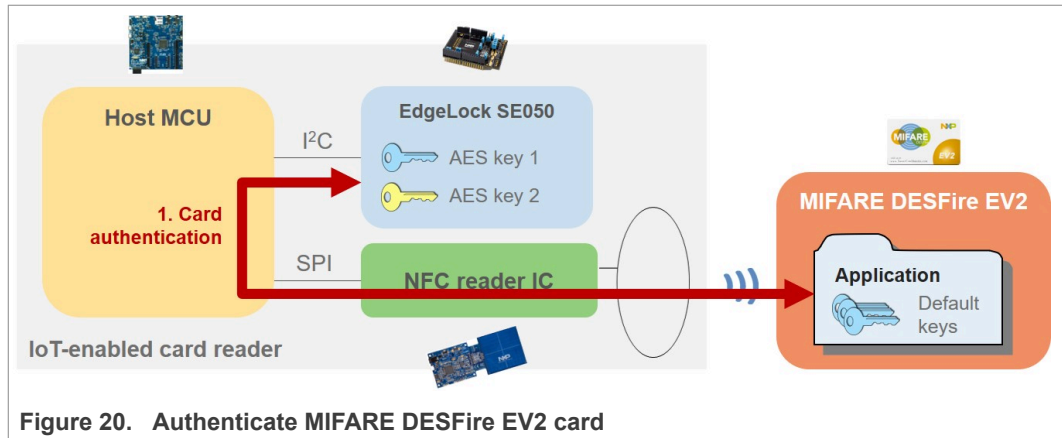


Figure 19. ex\_prepareCard output project log

#### 4.4.4 Authenticate and operate MIFARE DESFire EV2 card (ex\_Ev2Auth\_se05x)

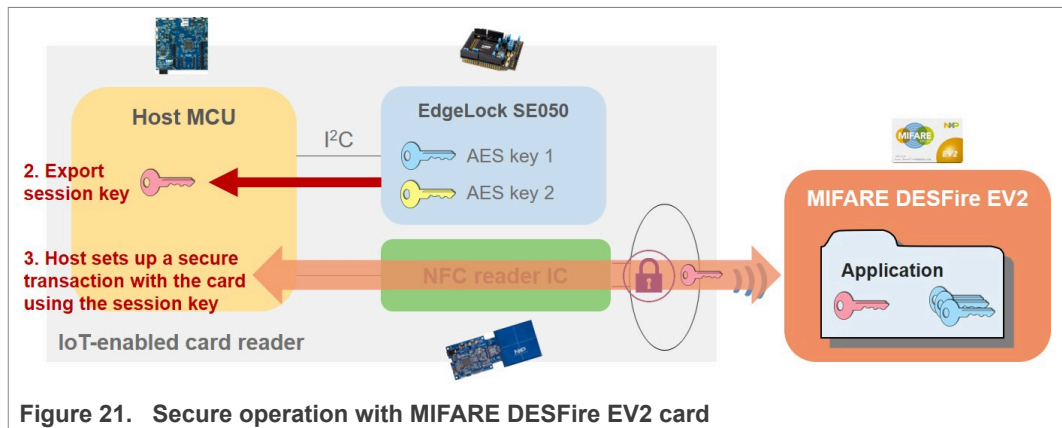
Now that both the EdgeLock SE05x IC and the MIFARE DESFire EV2 card have been provisioned with the required keys, it is possible to authenticate the card with the EdgeLock SE05x and to start a secure transaction. This can be done by executing the ex\_Ev2Auth\_se05x project example.

First, the ex\_Ev2Auth\_se05x leverages EdgeLock SE05x to authenticate the card, as depicted in [Figure 20](#)



After that, it generates and exports the MIFARE DESFire EV2 session key to the host MCU, which uses this session key to set up a secure transaction with the card, as depicted in [Figure 21](#).

**Note:** The secure transaction between the EdgeLock SE05x and the MIFARE DESFire EV2 card performed by the `ex_Ev2Auth_se05x` project example is provided for illustrative purposes. The commands required in a transaction needs to be adapted based on each specific application.



To run the `ex_Ev2Auth_se05x` project example, place your blank MIFARE DESFire EV2 card over the CLEV6630B antenna and follow these steps:

1. Select `ex_Ev2Auth_se05x` project example from your workspace
2. Go to the MCUXpresso Quickstart Panel and click *Debug* button, wait a few seconds until the project executes and click on *Resume*

- In Teraterm you should see the logs of the `ex_Ev2Auth_se05x` project example, which should look similar to [Figure 22](#):

```

COM13 - Tera Term VT
File Edit Setup Control Window Help
sss:WARN :Communication channel is Plain.
sss:WARN :!!!Not recommended for production use!!!
App:INPO :UID after L3 Activation (Len=10)
          08 ED 2A 5C 00 00 00 00 00 00
App:INPO :ATS after L4 Activation (Len=6)
          06 75 77 81 02 80
App:INPO :Select the AFC Application Successful
App:INPO :attempting to authenticate with cardkey = 0 and Se0Obj ID = 2103308288
App:INPO :
          CARD =====> SE050 16-byte Ek(RndB) = (Len=16)
          85 76 49 BC 00 3F 2D 58 48 24 04 BD FF 1E 62 39
App:INPO :
          CARD <===== SE050 E(Kx, RandA || RandB') = (Len=32)
          ED 57 9D 0C 07 75 BD C5 92 CC DD 32 C7 30 B8 46
          95 8A 02 63 4E E5 81 23 C2 95 D9 FB 27 7F 4A DE
App:INPO :
          CARD =====> SE050 32-byte E(Kx, T1||RandA'!!PCDcap2!!PCDcap2) = (Len=32)
          45 7D 61 AD 5A 5B 86 57 69 6E F5 EA FC C1 E6 3E
          05 9A 28 B6 D1 97 6E 1B D9 1A 93 BE C1 BC 80 40
App:INPO :
          CARD <===== SE050 E(Kx, RandA || RandB') = (Len=12)
          00 00 00 00 00 00 00 00 00 00 00 00
App:INPO :Dumped Session Key is (Len=16)
          DA E1 B5 49 EC F9 5D 86 C2 57 81 08 BA 0E 18 09
App:INPO :Dumped Session Mac is (Len=16)
          BA 08 73 51 6F EC 68 CD 70 C6 4F 6E AA 60 A8 33
App:INPO :Dumped T1 is (Len=4)
          DC 97 CA 7B
App:INPO :pDataParams->uCndCtr=0
App:INPO : E02 First Authenticate Successful
App:INPO :
          CARD =====> SE050 16-byte Ek(RndB) = (Len=16)
          7D D7 38 C5 F4 5E 54 82 9A 78 56 5E 95 DF 98 D5
App:INPO :
          CARD <===== SE050 E(Kx, RandA || RandB') = (Len=32)
          6C A1 3B 71 6B 8A FE 03 85 4F C1 93 23 55 FA 8D
          8E 98 79 83 B1 E3 3F 67 1A 4B 73 92 4B 78 A7 B4
App:INPO :
          CARD =====> SE050 32-byte E(Kx, T1||RandA'!!PCDcap2!!PCDcap2) = (Len=16)
          11 DE 46 3C 08 5D 6E D9 7E 99 06 37 C1 DA 5E D8
App:INPO :Dumped Session Key is (Len=16)
          D1 37 88 EB D5 83 A3 F1 63 39 AC 69 C8 A7 2B 78
App:INPO :Dumped Session Mac is (Len=16)
          5B 16 1D 5C 64 02 34 78 DD DD 94 DE 02 E4 4A AC
App:INPO :Dumped T1 is (Len=4)
          DC 97 CA 7B
App:INPO :pDataParams->uCndCtr=0
App:INPO : E02 Following Authenticate Successful
App:INPO :Authenticated with cardkey = 0 and Se0Obj ID = 2103308288
App:INPO :CARD UID is as below (Len=7)
          04 50 2C 42 4D 4D 80
App:INPO :phEx_Use_ValueFile...
App:INPO :Performing Accreditation in AFC App...
App:INPO :Performing Accreditation in AFC App Successful
App:INPO :<Plain Communication>Trying to Get the Current Value. Plain Communication
App:INPO :Getting current value Successful
App:INPO :<Enc Communication using session Key>Trying to Add money to the account
App:INPO :Add money to the account successful
App:INPO : The amount in your account After credit is 0 0 0 23
App:INPO : Accreditation DONE!
App:INPO : Auth session is reset in software
App:INPO : Auth session is killed in SE
App:INPO :ex sss Finished
    
```

Figure 22. `ex_Ev2Auth_se05x` log output project log

#### 4.4.5 Change MIFARE DESFire EV2 application key (`ex_Ev2ChangeKey_se05x`)

The EdgeLock SE05x also allows you to securely change MIFARE DESFire EV2 application keys. This can be done by executing the `ex_Ev2ChangeKey_se05x` project example or the `ex_Ev2DivChngKey_se05x` project example in case we want to use key diversification.

In this context, we can leverage EdgeLock SE05x to authenticate with the current key value and then, to securely compute the MIFARE DESFire EV2 change key command as depicted in [Figure 23](#):

**Note:** The MIFARE DESFire EV2 application key change performed by the *ex\_Ev2DivChngKey\_se05x* project example is provided for illustrative purposes. This process might need to be adapted depending on the MIFARE DESFire EV2 personalization, memory map, or specific application requirements.

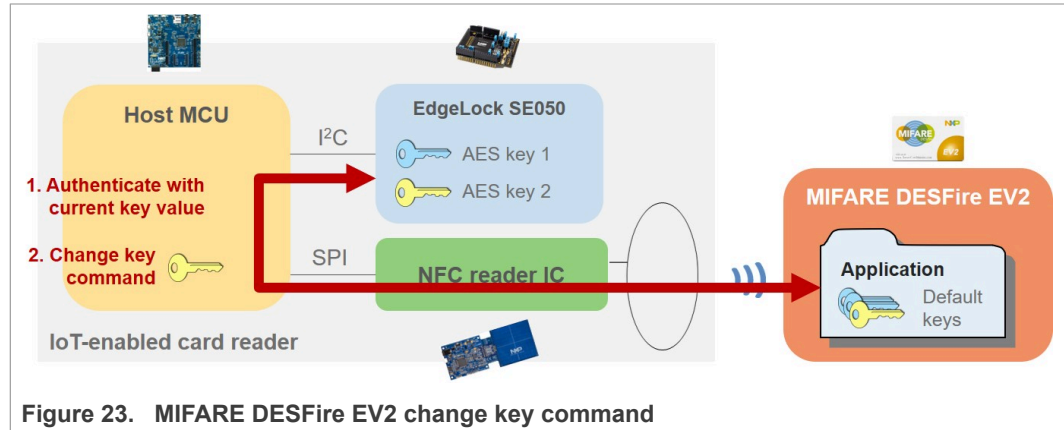


Figure 23. MIFARE DESFire EV2 change key command

To run the *ex\_Ev2ChangeKey\_se05x* project example, place your blank MIFARE DESFire EV2 card over the CLEV6630B antenna and follow these steps:

1. Select *ex\_Ev2ChangeKey\_se05x* project example from your workspace.
2. Go to the MCUXpresso Quickstart Panel and click *Debug* button, wait a few seconds until the project executes and click on *Resume*



- 3. In Teraterm you should see the example logs of the ex\_Ev2ChangeKey\_se05x project example, which should look similar to Figure 24:

```
COM13 - Tera Term VT
File Edit Setup Control Window Help
App:INFO :PlugAndTrust_v02.11.03_20191107
sss:INFO :atr (Len=35)
00 00 00 00 03 96 04 03 E8 00 FE 02 0B 03 E8 08
01 00 00 00 00 64 00 00 0A 4A 43 4F 50 34 20 41
54 50 4F
sss:WARN :Communication channel is Plain.
sss:WARN :!!!Not recommended for production use.!!!
App:INFO :UID after L3 Activation (Len=10)
08 E6 38 04 00 00 00 00 00 00
App:INFO :ATS after L4 Activation (Len=6)
06 75 77 81 02 80
App:INFO :Select the AFC Application Successful

App:INFO :attempting to authenticate with cardkey = 0 and Se00bj ID = 2103308288
App:INFO :
CARD ===== SE050 16-byte Ek(RndB) = (Len=16)
D7 0E 64 9E 0B 21 56 86 14 0E E0 F9 D5 25 67 A5
App:INFO :
CARD <===== SE050 ECKx, RandA !! RandB' = (Len=32)
43 05 98 F1 5D 84 9D D2 2B 3C BA B3 12 54 48 ED
4E A4 86 FC A7 C8 18 62 8F 93 95 DB 03 9F 56 8F
App:INFO :
CARD ===== SE050 32-byte ECKx, TI!!RandA'!!PCap2!!PCDeap2 = (Len=32)
DB 17 C2 9D 2C D6 30 78 6A 6D 97 34 A8 16 FF 12
AA B9 5B 29 35 49 AD 15 A3 F0 07 16 62 55 93 51
App:INFO :
CARD <===== SE050 ECKx, RandA !! RandB' = (Len=12)
00 00 00 00 00 00 00 00 00 00 00 00
App:INFO :Dumped Session Key is (Len=16)
C9 41 DD 33 AF 97 7E EC 2C 5B A2 CE C6 E3 F5 0C
App:INFO :Dumped Session Mac is (Len=16)
5F F8 E4 2F F9 4A 34 3E C2 B3 9A A7 67 A3 36 67
App:INFO :Dumped TI is (Len=4)
BE 90 09 4B
App:INFO :pDataParams->wCmdCtr=0
App:INFO : E02 First Authenticate Successful

App:INFO :
CARD ===== SE050 16-byte Ek(RndB) = (Len=16)
78 91 60 06 AA 63 A8 2F E6 C1 BA 4F 69 4A E1 25
App:INFO :
CARD <===== SE050 ECKx, RandA !! RandB' = (Len=32)
B0 21 36 74 8F F4 84 D7 F7 5A AF D2 67 08 69 CB
03 7D C1 DA BA E6 33 5F 60 62 4F BA 41 25 71 47
App:INFO :
CARD ===== SE050 32-byte ECKx, TI!!RandA'!!PCap2!!PCDeap2 = (Len=16)
18 8F 04 99 D3 2B 37 3E CF F9 97 43 40 6A 95 B9
App:INFO :Dumped Session Key is (Len=16)
12 F8 87 7E D8 5C 77 C2 95 D3 76 BB B2 2D CC 6B
App:INFO :Dumped Session Mac is (Len=16)
20 42 DB A7 19 4F B8 32 8E 65 C3 3F 76 AD 9E 49
App:INFO :Dumped TI is (Len=4)
BE 90 09 4B
App:INFO :pDataParams->wCmdCtr=0
App:INFO : E02 Following Authenticate Successful

App:INFO :Authenticated with cardkey = 0 and Se00bj ID = 2103308288
App:INFO :attempting to change cardkey = 2 from Old Se050bjID= 2103308288 to new Se050bj
App:INFO : Change Key for card key 2 is Successful to Se050bjID= 2103308289
App:INFO :Checking that the previous auth session is still valid by trying an encrypted co
App:INFO :CARD UID is as below (Len=7)
04 50 2C 42 4D 4D 80
App:INFO :Previous auth session is still valid
App:INFO :Auth with the changed cardkey 2
App:INFO :Select the AFC Application Successful

***
***
***
```

Figure 24. ex\_Ev2ChangeKey\_se05x output project log

## 5 Conclusions

The EdgeLock SE05x works as a secure access module for an IoT-enabled card reader, attached through a standard I<sup>2</sup>C interface to the device host controller. It is used to increase the security of an access control system by protecting the MIFARE secret keys, authenticating MIFARE DESFire EV2 credentials and performing secure key exchange or key diversification.

The application logic running on the host controller is responsible for handling the contactless communication and implementing the MIFARE DESFire EV2 command set. The host controller relies on the EdgeLock SE05x to authenticate cards and to derive the corresponding session key for each transaction. Using this session key, the host controller is able to establish a secure channel with MIFARE DESFire EV2 to perform a specific transaction.

The EdgeLock SE05x support package is designed in order to reduce complexity, speed up design and add flexibility in each part of the product development process. It offers libraries for different MCUs, integration with the most common OSs and source code examples for the latest IoT security use cases.

The EdgeLock SE05x Plug & Trust Middleware facilitates the integration of EdgeLock SE05x into your host MCU software. In the context of secure access control in industrial IoT applications, it offers a high level API for secure operation with MIFARE DESFire EV2 and comes with project examples for MIFARE DESFire EV2 card authentication, session key generation and key personalization operations.

In addition, the EdgeLock SE05x support package includes project examples to evaluate EdgeLock SE05x in a secure access control framework as explained in [Section 4](#). You can adapt these project examples and re-use the source code examples as needed to speed up the development of your IoT-enabled card reader design.

## 6 Legal information

### 6.1 Definitions

**Draft** — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

### 6.2 Disclaimers

**Limited warranty and liability** — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors. In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory. Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

**Right to make changes** — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

**Suitability for use** — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

**Applications** — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification. Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products. NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based

on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

**Export control** — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

**Evaluation products** — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer. In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out of the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages. Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US\$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

**Translations** — A non-English (translated) version of a document is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

**Security** — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

### 6.3 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

## Tables

---

Tab. 1.	OM-SE050ARD development kit details .....	10	Tab. 4.	LPC55S69 connection to CLEV6630B via SPI .....	15
Tab. 2.	LPC55S69 evaluation kit details .....	11			
Tab. 3.	CLEV6630B .....	11			

Figures

Fig. 1.	EdgeLock SE05x hardware integration in an IoT-enabled card reader ..... 4	Fig. 12.	TeraTerm configuration ..... 16
Fig. 2.	EdgeLock SE05x MIFARE session key derivation ..... 5	Fig. 13.	TeraTerm terminal window visualization optimization ..... 17
Fig. 3.	NXP Plug & Trust middleware block diagram ..... 5	Fig. 14.	Import sensor data protection project example ..... 18
Fig. 4.	EdgeLock SE05x Plug & Trust MIFARE DESFire EV2 API documentation ..... 8	Fig. 15.	Provision EdgeLock SE05x ..... 18
Fig. 5.	EdgeLock SE05x secure access demo setup ..... 10	Fig. 16.	Run ex_prepareSe050 project example ..... 19
Fig. 6.	Import NDA-protected LPC55S69 SDK version ..... 12	Fig. 17.	ex_prepareSe050 project log ..... 19
Fig. 7.	Jumper configuration for OM-SE050ARD ..... 13	Fig. 18.	Personalize MIFARE DESFire EV2 card ..... 20
Fig. 8.	Arduino connectors of OM-SE050ARD and LPC55S69 boards ..... 13	Fig. 19.	ex_prepareCard output project log ..... 21
Fig. 9.	OM-SE050ARD connection to LPC55S69 ..... 14	Fig. 20.	Authenticate MIFARE DESFire EV2 card ..... 22
Fig. 10.	LPC55S69 connection to CLEV6630B via SPI ..... 15	Fig. 21.	Secure operation with MIFARE DESFire EV2 card ..... 22
Fig. 11.	Boards connection to the laptop ..... 16	Fig. 22.	ex_Ev2Auth_se05x log output project log ..... 23
		Fig. 23.	MIFARE DESFire EV2 change key command ..... 24
		Fig. 24.	ex_Ev2ChangeKey_se05x output project log ..... 25

## Contents

<b>1</b>	<b>EdgeLock SE05x for secure access control in industrial IoT use cases</b> .....	<b>3</b>
<b>2</b>	<b>EdgeLock SE05x application diagram</b> .....	<b>4</b>
<b>3</b>	<b>Software integration with EdgeLock SE05x Plug &amp; Trust middleware</b> .....	<b>5</b>
3.1	EdgeLock SE05x Plug & Trust middleware	
	MIFARE DESFire EV2 API .....	6
3.1.1	Se05x_API_DFAuthenticateFirstPart1() .....	6
3.1.2	Se05x_API_DFAuthenticateFirstPart2() .....	6
3.1.3	Se05x_API_DFAuthenticateNonFirstPart1() .....	6
3.1.4	Se05x_API_DFAuthenticateNonFirstPart2() .....	6
3.1.5	Se05x_API_DFDumpSessionKeys .....	7
3.1.6	Se05x_API_DFChangeKeyPart1() .....	7
3.1.7	Se05x_API_DFChangeKeyPart2 .....	7
3.1.8	Se05x_API_DFDiversifyKey() .....	7
3.1.9	Se05x_API_DFKillAuthentication .....	7
3.2	EdgeLock SE05x Plug & Trust middleware	
	MIFARE DESFire EV2 API documentation .....	7
<b>4</b>	<b>Running the EdgeLock SE05x secure access module project examples</b> .....	<b>9</b>
4.1	Hardware required .....	10
4.2	Download and install the NDA-protected	
	LPC55S69 SDK version .....	11
4.3	Board setup and wiring .....	12
4.3.1	OM-SE050ARD jumper configuration .....	12
4.3.2	OM-SE050ARD connection with host MCU	
	board (LPC55S69) .....	13
4.3.3	Host MCU connection to NFC frontend	
	(CLEV6630B) .....	14
4.3.4	Laptop connection and TeraTerm	
	configuration .....	15
4.4	Project examples execution .....	17
4.4.1	Import the project examples in	
	MCUXpresso workspace .....	17
4.4.2	Provision EdgeLock SE05x (ex_	
	prepareSe050) .....	18
4.4.3	Personalize MIFARE DESFire EV2 card	
	(ex_prepareMFDfEV2) .....	19
4.4.4	Authenticate and operate MIFARE DESFire	
	EV2 card (ex_Ev2Auth_se05x) .....	21
4.4.5	Change MIFARE DESFire EV2 application	
	key (ex_Ev2ChangeKey_se05x) .....	23
<b>5</b>	<b>Conclusions</b> .....	<b>26</b>
<b>6</b>	<b>Legal information</b> .....	<b>27</b>

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

© NXP B.V. 2020.

All rights reserved.

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: [salesaddresses@nxp.com](mailto:salesaddresses@nxp.com)

Date of release: 7 December 2020

Document identifier: 565710

Document number: 565712