# AN12450

EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

Rev. 4.2 — 4 August 2022 546942

**Application note** 

#### **Document information**

Information	Content
Keywords	EdgeLock SE05x, EdgeLock A5000, Plug & Trust middleware, i.MX RT1060, i.MX RT1170
Abstract	This document explains how to get started with the EdgeLock SE05x Plug & Trust middleware using the EdgeLock SE05x/A5000 development boards and i.MX RT1060 or i.MX RT1170 MCU boards. It provides detailed instructions to run projects imported either from the board SDKs or the CMake-based build system included in the EdgeLock SE05x Plug & Trust middleware.



## **Revision history**

Revision number	Date	Description
1.0	2019-07-18	First release
2.0	2019-11-25	Major update to incorporate details to import projects from i.MX RT1060 SDK and CMake- based build system.
2.1	2019-12-17	Corrected OM-SE050ARD J14 jumper setting.
2.2	2020-01-20	Fixed broken links Section 3.
3.0	2020-10-27	Updated for EdgeLock SE051 and i.MX RT1060
3.1	2020-12-07	Updated to latest template and fixed broken links
4.0	2021-12-23	Added instructions for i.MX RT1170
4.1	2022-03-28	<ul> <li>Add EdgeLock SE050E and EdgeLock A5000 product variants.</li> <li>Update <u>Table 1</u>, Figure 1, Figure 2, Figure 3, Figure 4, Figure 5, Figure 6, Figure 7, Figure 8</li> <li>Add note (step 3) in <u>Section 4.5</u> Build, run and debug project example</li> <li>Add <u>Section 4.6</u> Product specific build settings</li> <li>Add note in <u>Section 5.6.2</u> Run EdgeLock SE05x Plug &amp; Trust middleware examples</li> <li>Add <u>Section 6</u> Binding EdgeLock SE05x to a host using Platform SCP</li> </ul>
4.2	2022-08-04	Update to EdgeLock SE Plug & Trust Middleware version 04.02.xx. Update note (step 3) in <u>Section 4.5</u> Build, run and debug project example Update <u>Section 4.6</u> Product specific build settings Update note in <u>Section 5.6.2</u> Run EdgeLock SE05x Plug & Trust middleware examples Update <u>Section 5.7</u> Product specific CMake build settings Update <u>Section 6</u> Binding EdgeLock SE05x to a host using Platform SCP

## **1** How to use this document

The Plug & Trust middleware includes a set of project examples that demonstrate the use of EdgeLock SE05x in the latest IoT security use cases. These project examples can be either:

- Imported from the MCUXpresso SDKs available for i.MX RT1060 and i.MX RT1170 MCU boards. Using the board SDKs is recommended as it is the easiest and fastest way of importing and running the project examples.
- Imported from the CMake-based build system included in the Plug & Trust middleware package. The CMake-based option is provided for developers familiar with this build system or that are willing to run exactly the same project examples on PC/Windows/ Linux and embedded targets.

This document provides detailed instructions to run EdgeLock SE05x project examples imported either from the board SDKs or the CMake-based build system.

The main body of this document should be used in this sequence:

- 1. Order board samples. You can find the ordering details of the boards required in this document in <u>Section 2</u>.
- 2. Setup your boards. <u>Section 3</u> describes how to setup the OM-SE05xARD and your MCU board (either i.MX RT1060 board or i.MX RT1170 board).
- Run project examples. Go to <u>Section 4</u> for instructions on how to import projects from the board SDKs following the recommended way of working, or alternatively, go to <u>Section 5</u> for instructions on how to import projects from the CMake-based build system.

Additional material is provided in the appendices of this document.

## 2 Required hardware

The EdgeLock SE05x works as an auxiliary security device attached to a host controller through an I<sup>2</sup>C interface. To follow the instructions provided in this document, you need at least an EdgeLock SE05x development board and an MCU board (either i.MX RT1060 or i.MX RT1170) acting as a host controller.

#### EdgeLock SE05x development boards ordering details

The EdgeLock SE05x and EdgeLock A5000 product support packages are providing development boards for evaluating EdgeLock SE05x and EdgeLock A5000 features. Select the development board of the product you want to evaluate. <u>Table 1</u> details the ordering details of the EdgeLock SE05x and EdgeLock A5000 development boards.

Part number	12NC	Description	Picture
OM-SE050ARD-E	9354 332 66598	SE050E Arduino <sup>®</sup> compatible development kit	

Table 1. EdgeLock SE05x development boards.

Part number	12NC	Description	Picture
OM-SE050ARD-F	9354 357 63598	SE050 Arduino <sup>®</sup> compatible development kit	The second
OM-SE050ARD	9353 832 82598	SE050F Arduino <sup>®</sup> compatible development kit	
OM-SE051ARD	9353 991 87598	SE051 Arduino <sup>®</sup> compatible development kit	
OM-A5000ARD	9354 243 19598	A5000 Arduino <sup>®</sup> compatible development kit	

Table 1. EdgeLock SE05x development boards. ...continued

**Note:** The pictures in this guide will show EdgeLock SE05xE, but all boards in <u>Table 1</u> can be used as well with the same hardware configuration.

#### i.MX RT1060 MCU board ordering details

Table 2 provides the ordering details for the i.MX RT1060 development board.

Table 2. i.	MX RT1060	evaluation	kit	details
-------------	-----------	------------	-----	---------

Part number	12NC	Content	Picture
IMXRT1060-EVKB	935419011598	MIMXRT1060-EVK low cost evaluation kit for Cortex-M7	

#### i.MX RT1170 MCU board ordering details

Table 3 provides the ordering details for the i.MX RT1170 development board.

Table 3. i.MX RT1170 evaluation kit details

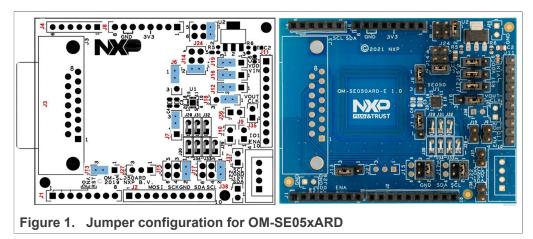
Part number	12NC	Content	Picture		
MIMXRT1170-EVK	935378982598	MIMXRT1170-EVK low cost evaluation kit for Cortex-M7			

## 3 Boards setup

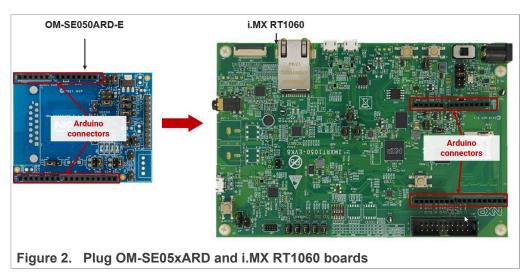
This section explains how to setup your i.MX RT1060/1170 and OM-SE05xARD boards to execute the Plug & Trust middleware:

 The OM-SE05xARD board has jumpers that allow you to use the EdgeLock SE05x I<sup>2</sup>C interface via the Arduino header. Configure the jumper settings as shown in <u>Figure 1</u> to enable this option.

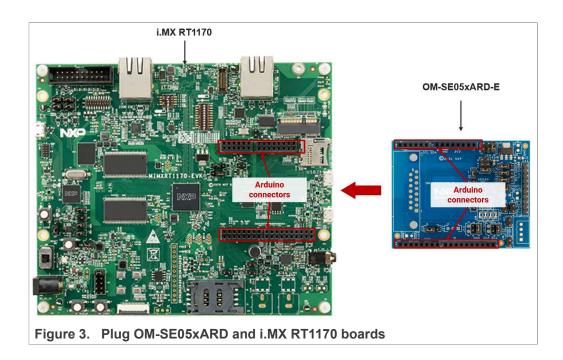
**Note**: For more information about the jumper settings, refer to <u>AN13539</u> OM-SE05xARD hardware overview.



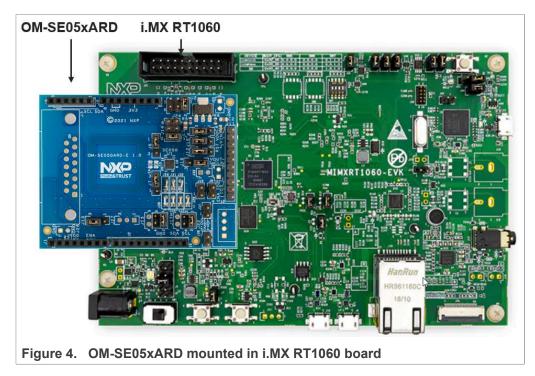
2. The OM-SE05xARD, and i.MX RT1060/1170 boards can be directly connected using the Arduino headers available in both boards. Figure 2 shows how to connect the OM-

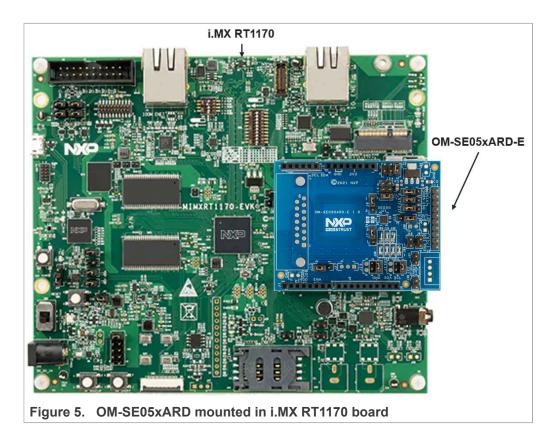


SE05xARD board on top of the i.MX RT1060. Figure 3 shows how to connect it on top of the i.MX RT1170 board.



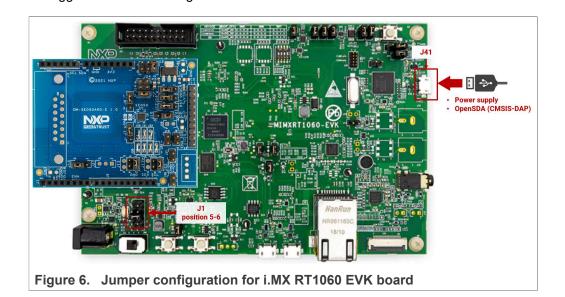
3. Double check that the two boards are connected as shown in <u>Figure 4</u> (for i.MX RT1060 board) or <u>Figure 5</u> (for i.MX RT1170 board):

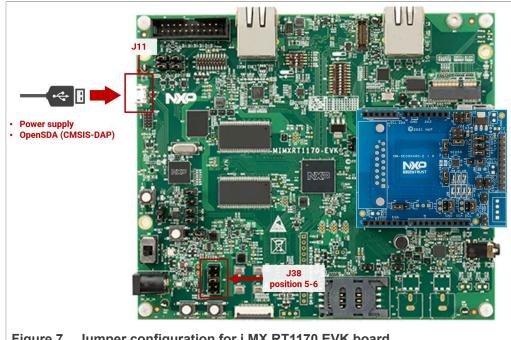




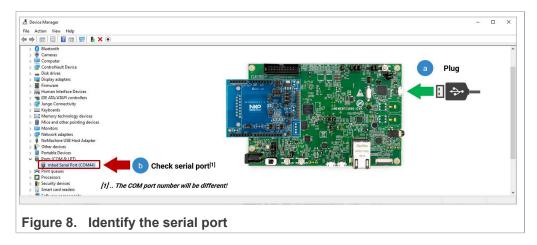
4. i.MX RT1060: configure the J1 jumper of the i.MX RT1060 EVK board in position 5-6 as shown in Figure 6. With this setting, the board power supply and the DAP-Link debugger will function using the J41 micro-USB connector. i.MX RT1170: configure the J38 jumper of the i.MX RT1170 EVK board in position 5-6 as shown in Figure 7. With this setting, the board power supply and the DAP-Link

debugger will function using the **J11** micro-USB connector.





- 5. Check that your laptop recognizes the i.MX RT1060/1170 board by following the steps shown in Figure 8:
  - a. Connect the board to your laptop using the micro-USB connector **J41** (for i.MX RT1060 board) or **J11** (for i.MX RT1170 board).
  - b. Check that the serial port is recognized in the category Ports (COM & LTP). In this document, it is recognized as USB Serial Device (COM12) but this naming might change depending on your computer. Therefore, it is important that you identify which device is recognized at the moment you plug the board to the PC.



## 4 Import project examples from board SDK

This section explains how to run EdgeLock SE05x projects by importing them from the i.MX RT1060 or i.MX RT1170 SDKs. This is the most recommended option since it implies that the MCU projects are self-contained standard MCUxpresso projects providing a better debug experience. If you are an expert user or you need to compile the example for different OSs, please use the CMake build system to import and compile examples as described in <u>Section 5</u>.

## 4.1 Prerequisites

The following software tools are required to run a project imported from the board SDK:

- 1. MCUXpresso IDE. Check <u>Section 7</u> for detailed installation instructions.
- 2. TeraTerm (or an equivalent serial application). You can download and run TeraTerm installer from this <u>link</u>.

## 4.2 Download the board SDK

The project examples for EdgeLock SE05x are included in the i.MX RT1060 and i.MX RT1170 SDKs. First, download the SDK for your board from the EdgeLock SE05x website. The SDKs available from EdgeLock SE05x website contain the most up-to-date and complete list of project examples to evaluate EdgeLock SE05x features.

**Note:** The i.MX RT1060 SDK or i.MX RT1170 SDK you can download from <u>MCUXpresso</u> <u>SDK Builder website</u> may not include all the EdgeLock SE05x project examples or the latest version of them.

## 4.3 Install the board SDK

After downloading the SDK for your board, you need to install it in MCUXpresso. To install the SDK, (1) drag and drop the SDK zip file in the Installed SDKs section in the bottom part of the MCUXpresso IDE and (2) click OK as shown in Figure 9:

My_workspace - MCUXpresso IDE				- 0 ×
e Edit Navigate Search Project ConfigTools Run FreeRTOS	Window Help ? + 🞯 🗐 🗊 😩 + 🖳 🍬   🕨 💷 📾 💐 🐎 🐟	m == == = = = = = = = = = = = = = = = =	1.2 I R. 101 - XI - 15	
		nn 🛲 oer en enriker i 🧿		iick Access
	MCUXpresso IDE SDK import	- 🗆 X		
] Pr X 🔀 Per IIII Re 🛊 Fa 🗶 Sy 🙂 🗖 🕒 🧐 🖽 🍫   ■ 🔹 🌣				L
	Are you sure you want to import the following SDK in the of folder?	common 'mcuxpresso'		
	C:\Users\PetarHlad\Downloads\SDK-EVK-INXRT1060-SE.zip	Â		
	Do not ask for confirmation on SDK Drag and Drop install	*		
Qui 🕬= Glo (x)= Vari 💁 Bre 🗄 Outl 🗖 🗖 🔞	talled		📼 Power Measurement T 📓 SWO	Trace Config 🛛 🗖
		OK Cancel		🎱 🍪 🔲 🖽 🖪
MCUXpresso IDE - Quickstart Panel 🏻 🍦 🕅	nstalled SDKs			
	tall an SDK, simply drag and drop an SDK (zip file/folder) into the 'Installed SDKs' view.	[Common 'mcuxpresso' folder]		
Create or import a project Na	e SDK Version Manifest Version Location		SDK Details	<b>U</b>
New project  New project  Import SDK example(s)  Import project(s) from file system			Selected SDK content.	
<ul> <li>Build your project</li> </ul>	SDK-EVK-IMXRT1060-SE	zip		
Relid				
9 my workspace			82	

Figure 9. Import the board SDK into MCUXpresso environment

If the SDK is successfully imported, you should see it listed in the Installed SDK window as shown in Figure 10:

🕅 Installed SDKs 🛛 🔲 Properties 🔮 Pr	roblems 📮 Console 🔎	Terminal 🔜 Image Inf	o 🙀 Debugger Console			
🕅 Installed SDKs					<b>9</b> 4	2  🗖
To install an SDK, simply drag and drop an SI	DK (zip file/folder) into the	'Installed SDKs' view. [Co	ommon 'mcuxpresso' folde	er]		
Installed SDKs Available Boards Availabl	e Devices					
Name	SDK Version	Manifest Ver	Location			
SDK_2.x_EVK-MIMXRT1060	2.9.0	3.8.0	Common>\SDK-EVK	-IMXRT1060-SE.zip		
SDK_2.x_MIMXRT1170-EVK	2.9.0	3.8.0	Common>\SDK-EVK	-IMXRT1170-SE.zip		
Figure 10. Imported S	DKs					

## 4.4 Import a project example in MCUXpresso

After importing the SDK for your board in the MCUXpresso workspace, follow these instructions to import an example project:

1. Click *Import SDK example(s)* in the MCUXpresso IDE quick start panel as shown in Figure 11:

🗙 workspace - MCUXpresso IDE					- 0
ile Edit Navigate Search Project ConfigTools Ru					
🖞 🕶 🔛 🔞 🐨 💊 📲 🖉 😒 🚳 🖷 💺 🗛	🕸 • 🖸 • 💁 • 🎒 🖉 🖬 🗐	📮 🔍 🕨 🗉 🖷 M 3.	🗇e.   🗮 🕱   🐌 🗓	🍬 3. To ik 🕹 🕺 🛛 🖓 🎢	
월 • 월 • 박 랴 랴 숀 • 라 • 1 쿱					Q 🛛 🔛
Proj 🕴 🔠 Regi 🏇 Faults 😤 Peri 😑 🗖					
E 😘 🏹 🖶 🍫 🔳 🕶 🖇					
There are no projects in your workspace. To add a project:					
Create a new MCUXpresso IDE C/C++ project.					
Import examples from SDK.					
Create a project					
in Import projects					
i) Quickstar 😒 (4)= Variables 🔍 Breakpoi 😐 🗖		blems 📮 Console 🍠 Termin	al 🔜 Image Info 🙀 Debu	gger Console 🛛 🔀 Offline Peripherals	808  🗆 = 🗆
	🕅 Installed SDKs	-			8 @ &  🗆 = 🗆
	(installed SDKs) To install an SDK, simply drag and drop an SDK	(zip file/folder) into the 'Installe			8 @ &  🗆 = 🗆
MCUXpresso IDE - Quickstart Pane	(installed SDKs) To install an SDK, simply drag and drop an SDK Installed SDKs Available Boards Available D	(zip file/folder) into the 'Installe Devices	d SDKs' view. [Common 'mcu	xpresso' folder]	8 @ 🍪 🔲 🖓 🗖
MCUXpresso IDE - Quickstart Pane No project selected Create or import a project	(1) Installed SDKs To install an SDK, simply drag and drop an SDK Installed SDKs Available Boards Available T Name	(zip file/folder) into the 'Installe Devices SDK Version	d SDKs' view. [Common 'mcu Manifest Version	xpresso' folder]	
MCUXpresso IDE - Quickstart Pane	Installed SDKs           To install an SDK, simply drag and drop an SDK (installed SDKS)         Available Boards)         Available D           Name         ## SDK,2.x., EVK-MIMXRT1060         ## SDK,2.x., MIMXRT1170-EVK	(zip file/folder) into the 'Installe Devices	d SDKs' view. [Common 'mcu	xpresso' folder]	p

2. The SDK import wizard will open. (1) You should see on the left side the list of all the SDKs imported in MCUXpresso IDE. Select the one corresponding to the board you are using (either i.MX RT1060 or i.MX RT1170). You should now see a figure of an i.MX RT1060 board (or i.MX RT1170 board in case you selected the i.MX RT1170

SDK) with an orange label. (2) Select the board and then (3) click on the Next button as shown in Figure 12:

SDK Import Wizard					- O X
SDK Import Wizard					
() Importing project(s) for device: MIMX	RT1062xxxxA using board: EVK-MIMXRT1060				NP 🖆
Board and/or Device se	election page				
→ SDK MCUs	Available boards				↓a <sub>z</sub> ↑a <sub>z</sub>   🖉
MCUs from installed SDKs. Please click	Please select an available board for your project.				
above or visit <u>mcuxpresso.nxp.com</u> to obtain additional SDKs.	Supported boards for device: MIMXRT1062xxxxA				
NXP MIMXRT1062xxxA  MIMXRT1062 MIMXRT1022xxxA  MIMXRT170 MIMXRT176 XXXX	Levening ties	I			
Selected Device: MIMXRT1062xxxxA	uring boards EVK_MIMYPT1060	SDKs for selected MCU			
Target Core: cm7		Name	SDK Version	Manifest Ve	Location
	00MHz, 512KB SRAM Microcontrollers (MCUs) based on ARM	SDK_2.x_EVK-MIMXRT1060		3.8.0	Common>\SDK-EVK-IMXRT1060
Contex-Iwi/ Core					
					_
?			< Back	Next >	3 Cancel
Figure 12 SDk	(import wizard				

Figure 12. SDK import wizard

Note: If there is not an SE05x orange label on top of the board image, MCUXpresso may be recognizing a board SDK with a higher version number, downloaded from MCUXpresso SDK Builder website. To access the most up-to-date and complete list of EdgeLock SE05x project examples, first you need to uninstall the SDK currently installed, and then repeat the process indicated in Figure 9.

3. Under the se hostlib examples drop down list, you have the list of supported project examples. Select the examples that you want to import in your MCUXpresso workspace and click on the Finish button as shown in Figure 13. In this case, we

selected the  $se05x_{Minimal}$  project as an example. The same process can be followed for importing all other examples.

Import projects			
oject name prefix: evkmimxrt1060		2 Project name suffix:	<i>B</i>
Use default location			
cation: C:\Users\AVANT STUDIO\Documents\M	CUXpressoIDE_11.0.1_2563\wor	kspace\evkmimxrt1060	Browse.
roject Type		Project Options	
C Project C++ Project C Static Library	C++ Static Library	SDK Debug Console () Semihost () UAR Copy sources Import other files	F 🔿 Example default
amples			🔤 🖉 🐼 🍇 🖽 🛙
vpe to filter			
Vame  V	This example This is a bare This demo de This demo de This demo de This demo de This example This example This example This example This example	an be used to get SE05X platform information. reads Accelerometer data via the 12C master interfar minimum example for se050. This gets the amount o monstrates connection to AVVS IoT Console using pr monstrates connection to Google Cloud Platform us monstrates connection to IBM Watson IoT platform us monstrates connection to IBM Watson IoT platform us does a elliptic curve cryptography signing and verify does a HMAC Key derivation operation based on the does a Message Digest hashing operation. does a As signing and verify operation. does a symmetric cryptography AES encryption and 50 demo application allows the board to be used as	of free memory. e-provisioned rovisioned devi. ing pre-provisi y operation. e info and salt.

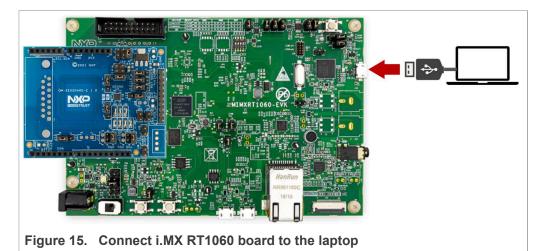
4. The projects you selected should now be visible in your MCUXpresso workspace as shown in Figure 14:

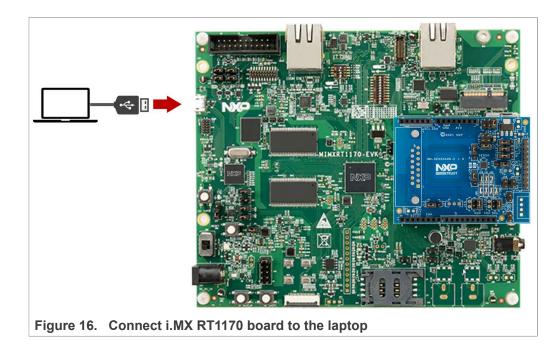
Proj:       32       IIII Regi	🗙 workspace - MCUXpresso IDE					- 0	
Proj	ile Edit Navigate Search Project ConfigTools Ru	un RTOS Analysis Window Help					
Proj. 32       1111 Regi	🖻 • 🔛 🔞   💩 • 🗞 • 📓 🖉 🗠   🏟 • 🛼	💠 🎋 • 🔘 • 💁 • 🎒 🖉 🕫	1 🖸 🔌 🗈 🖬 🕅	3. (9. je) 🗟 🕱 🚺	00 🐘 32, 78, 92, 82, 82, 63, 64, 69, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7,		
© exhiminal CoBurger       © exhiminal CoBurger         © Project String       © Installed SDKs         © Console       © Terminal         © Gomponent       © Gomponent         © Gomponent       © Installed SDKs         © Startup       © Installed SDKs         © Startup       © Installed SDKs         © Installed SDKs	월 • 월 • 약 약 Ҿ • ♀ · I 러					Q	8
Image: Antimital Dispects       Image: Antimital Dispects         Image: Antimital Dispects       Image: Antip         Image: Antip       <							
<ul> <li>✓ So Artificial SDKs</li> <li>✓ Project Strings</li> <li>✓ Orkinal Cobuges</li> <li>✓ Project Strings</li> <li>✓ Orkinal SDKs</li> <li>✓</li></ul>							-
<ul> <li>Project Stining</li> <li>Project Stining</li> <li>Project Stining</li> <li>Protect Stining</li> <li>Pro</li></ul>							
3 Stores       3 Stores         4 Stores       3 Stores         5 Stores       5 Stores         5 Stores       5 Stores         6 Stores       5 Stores         6 Stores       5 Stores         7 Stores       5 Stores         8 Stores       5 Stores         9 Stores       5 Stores         9 Stores       5 Stores         9 St							
<ul> <li>So component</li> <li>So c</li></ul>							
Software softwa							
<ul> <li>a device device</li></ul>							
Softwars Softwars							
Se Apostilio							
Se surce	> 🐸 mbedtls						
Softwardset      Softwardset     Softwar							
Softwires							
	> 😕 startup						
Import project       Import project         Import project       SN Kerry (from file system	> 😕 startup > 冯 utilities						
MCUXpresso IDE - Quickstart Pane	> 28 startup > 28 utilities > 28 xip						
Circle orkinant/Box/sets_Minimal (Debug)       Installed SDKS_Available Boards] Available Boards]	> ᢙ startup > ᢙ utilities > ᢙ xip > ᢙ doc	🍘 Installed SDKs 🕴 🥅 Properties 🌊 Prob	olems 📮 Console 🖉 Termin	al 🔜 Image Info 🖳 Debuş	gger Console 😹 Offline Peripherals	🗴 🥥 🕄 🗆 c	- 8
Create or import a project     Mane     SOK \_vvaliable Beards  \vvaliable Devices     SoK \\vvaliable Devic	> ᢙ startup > ᢙ utilities > ᢙ xip > ᢙ doc		olems 📮 Console 🖑 Termin	al 🚡 Image Info 🙀 Debuş	gger Console 😤 Offline Peripherals	🗴 🥥 🍪 🕅 c	- 0
Image: Solution of the system     Solution of the system       Image: Solution of the system     Solution of the system	> @ startup > @ utilities > @ sip > @ ouickstar	) Installed SDKs				🗴 🔿 🍪 🖂 =	- 0
Import SDK example(s)         Import project(s) from file system              Import project(s) from file system	> @ startup > @ vtilities > @ vtilities > @ objectstar	(installed SDKs) To install an SDK, simply drag and drop an SDK	(zip file/folder) into the 'Installe			🗴 🔿 🍪 🗆 -	
Import project(s) from file system	> @ startup > @ vtilities > @ vtilities > @ objectstar	Installed SDKs     To install an SDK, simply drag and drop an SDK     Installed SDKs Available Boards Available D	(zip file/folder) into the 'Installe Devices	d SDKs' view. [Common 'mcu	xpresso' folder]	🗴 🔿 🍪 🗆 -	
	> @ startup > @ velikies > @ velikies > @ odec ) Quickstar ☆ O> Variables �o Breakpoi	O Installed SDKs To install an SDK, simply drag and drop an SDK Installed SDKs Available Boards Available D Name     O	(zip file/folder) into the 'Installe Devices SDK Version	d SDKs' view. [Common 'mcu Manifest Version	xpresso' folder] Location Common's \SDK-EVK-IMXRT1060-SE.zig		-
	> @ stanup > @ stanup > @ viiifies > @ doc ) Quicktar ☆ (>> Variables % Breakpoi	O Installed SDKs To install an SDK, simply drag and drop an SDK Installed SDKs Available Boards Available D Name     O	(zip file/folder) into the 'Installe Devices SDK Version 2.9.0	d SDKs' view. [Common 'mcu Manifest Version 3.8.0	xpresso' folder] Location Common's \SDK-EVK-IMXRT1060-SE.zig		-
S evtmimut1060_se05x_Minimal	) @ startup ) @ vlifies ) @ v	⑦         Installed SDKs           To install an SDK, simply drag and drop an SDK           Installed SDK3         Available Boards Available D           Name         Image: SDK_2x_EVK-MIMXRT1060           Image: BDK_2x_MIMXRT1170-EVK	(zip file/folder) into the 'Installe Devices SDK Version 2.9.0	d SDKs' view. [Common 'mcu Manifest Version 3.8.0	xpresso' folder] Location Common's \SDK-EVK-IMXRT1060-SE.zig		2
	> of startup > of startup > of utilities > of xip > of version of the startup > of version of the startup Project: versionment1000_uceds_Minimal (Debug) □ □ • Create or import a project • Create or import a project • Of the project • Import project(), • Import project(), • Import project(), • Import project(), from file system	⑦         Installed SDKs           To install an SDK, simply drag and drop an SDK           Installed SDK3         Available Boards Available D           Name         Image: SDK_2x_EVK-MIMXRT1060           Image: BDK_2x_MIMXRT1170-EVK	(zip file/folder) into the 'Installe Devices SDK Version 2.9.0	d SDKs' view. [Common 'mcu Manifest Version 3.8.0	xpresso' folder] Location Common's \SDK-EVK-IMXRT1060-SE.zig		2 0
		⑦         Installed SDKs           To install an SDK, simply drag and drop an SDK           Installed SDK3         Available Boards Available D           Name         Image: SDK_2x_EVK-MIMXRT1060           Image: BDK_2x_MIMXRT1170-EVK	(zip file/folder) into the 'Installe Devices SDK Version 2.9.0	d SDKs' view. [Common 'mcu Manifest Version 3.8.0	apreso' folder]  Location  Ccantion  Ccantion Common \SDK-EVK-IMXRT1060-5E.zig  Common \SDK-EVK-IMXRT1170-5E.zig	2	

## 4.5 Build, run and debug project example

After importing project examples in the MCUXpresso workspace, follow these instructions to build, run and debug a project:

1. Attach a USB cable from the computer to the debug USB connector of your board. Figure 15 shows the connection with i.MX RT1060 board. Figure 16 shows the connection with i.MX RT1170 board.





AN12450 **Application note** 

- 2. Launch and configure the TeraTerm application as shown in Figure 17:
  - a. Click *Serial* option and select from the drop down list the COM port number assigned to your board.
  - b. Go to Setup > Serial Port and configure the terminal to 115200 baud rate, 8 data bits, no parity and 1 stop bit and click OK.

		^	Port:	COM44 ~	<u>N</u> ew setting
era Term: New cor	nnection	×	Sp <u>e</u> ed: Data:	8 bit	
O TCP/IP	Host: myhost.example.com		Parity:	none v	Cancel
	History Service: O Telnet TCP port#: 22		Stop bits:	1 bit v	<u>H</u> elp
	SSH SSH version: SSH2     Other	~	Elow control:	none v	
	IP version: AUTO	$\sim$		nit delay	
• Serial	Port: COM44: mbed Serial Port (COM44)		0	msec/ <u>c</u> har 0	msec/line
	OK Cancel Help			nbed -2009	
			<		, ×

- 3. Note: The default build configuration of the Plug & Trust middleware ≥ V04.02.0x generates code for the OM-SE050ARD-E development board. You need to adapt settings in the feature header file fsl\_sss\_ftr.h in case you are using a different EdgeLock secure element development board or a different secure element product IC. The settings are described in <u>Section 4.6</u>.
- 4. Go to the MCUXpresso Quickstart Panel and click on the *Build* button as shown in <u>Figure 18</u>. Wait a few seconds and check that the build process has finished successfully in the MCUXpresso console window.

X workspace - MCUXpresso IDE	- 0	>
le Edit Navigate Search Project ConfigTools Run	RTOS Analysis Window Help	
	\$ • 0 • 9 • 10 • 1 • 10 • 1 • 10 • 1 • 10 • 1 • 1	
1、101-101-101-10-10-10-10-10-10-10-10-10-1		-01
	Q is	- 10
🖢 Proj 🔀 🕮 Regi 🎋 Faults 🛼 Peri 🦈 🗖	E	
P 😫 T 🖶 🍫 🕅 🕶 3		
evkmimxrt1060_se05x_Minimal <debug></debug>		
> Project Settings		
> 🖑 Binaries		
> 🔊 Includes		
> 🥴 CMSIS		
> 🖉 board > 🖉 component		
> 🙆 device		
> 29 drivers		
> 🔗 mbedtis		
> 🕮 se_hostlib		
> 🕮 source		
> 🐸 startup		
> 😕 utilities		
> 🕮 xip		
> 🚁 Debug > 🗁 doc		
	🕜 Installari SDK s 🗇 Donastias 📧 Donasla 🕫 🖉 Consola 🕫 🖉 Tarminal 🔿 Imana Info 🔛 Dahunnar Consola 号 Officina Darinharais 🗢	-
) Quickstar 🕅 🔅 🕬 Variables 💁 Breakpoi 🖓 🗖 🦞	Change and Change To counce the state of the	
	¥ + + + + + + + + + + + + + + + + + + +	37
- · · · · · · · · · · · · · · · · · · ·	DT Build Console [evkmimxt1060_se05x_Minimal]	_
	akeno-print-directory post-build	^
	erforming post-build steps rm-none-eabi-size "evkmimxrt1060_se05x_Minimal.axf"; # arm-none-eabi-objcopy -v -0 binary "evkmimxrt1060_se05x_Minimal.axf" "evkmimxrt10	161
	text data bss dec hex filename	
Clean i	194308 464 42992 237764 3a0c4 evkmimxrt1060_se05x_Minimal.axf	
- Debug your project 🛛 💽 - 🔛 -		
	5:34:27 Build Finished. 0 errors, 0 warnings. (took 55s.37ms)	
1 Permanan huid are Deeve		
The formation had and based		>
evkmimxrt1060_se05x_Minimal	O NXP MIMXRT062xxxxA* (exkmi_nim	mall
		and the

 Go to the MCUXpresso Quickstart Panel and click on the *Debug* button as shown in <u>Figure 19</u>. If there is more than one probe attached, you have to select the CMSIS-DAP debug probe from the list. Wait a few seconds until the project executes.

🔀 workspace - MCUXpresso IDE								- 0	×
File Edit Navigate Search Project ConfigTools Run	TTTD: And a	in Minuters Male							
						- D X	📕 🕹 🖞 🕲 🕲 🖄		
「別・別・広点な・ウール」	· • · • • •	Probes discovered				U X	1 24 79 26 16 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	n in	a local
		Connect to target: MIMXR	T1062xxxxA					Q, 189	A COLORADO
🚡 Proj 😂 🕮 Regil 🌞 Faults 🛼 Peri 📟 🗖		1 probe found. Select the prol	e to use:					- 0	1 #
🖂 😵 🍸 🛛 🗮 🍫  📓 🕶 🖇		1							
evkmimxrt1060_se05x_Minimal <debug></debug>		Available attached p	ohes						(k)=
Project Settings		reason and a second a pr							8
> 侯 Binaries > 劒 Includes		Name	Serial number	Туре	Manufactur	IDE Debug Mo			8
> P CMSIS		CMSIS-DAP	02440000092e	Link	ARM	Non-Stop			(2)=
board									200
> 😛 component									
> 😬 device									
> 🙆 drivers									
) 🥵 mbedtls ) 🥵 se_hostlib									
) A source									
> 🖾 startup		Supported Probes (tick/untic	the second of the second second				1		
> 🥵 utilities									
> 🥵 xip		MCUXpresso IDE LinkSen P&E Micro probes	er (inc. CMSIS-DAP	) probes					
> 🗁 Debug > 🗁 doc		SEGGER J-Link probes							
	_	SEGOLICI SELINE probes							_
😲 Quickstar 💢 🕪 Variables 🤷 Breakpoi 🖓 🔲	nstailed SD	Probe search options					Console 🧏 Offline Peripherals	- 0	
		Search again					a 🖬 🖬 🖉 🔗	🖌   📷 🖾 🖛 📑 :	-
	RedlinkServer	Scalenogan							_
	CoresConfigu PacketSize =	Remember my selection (fo	r this Launch config	uration)				,	^
Con & Rolld	Reference Co								
	HasSWV = FAL HasETM = FAL	?	2		OK	Cancel			
	HasJTAG = TR						1		
	HasSWD = TRU Probe Type =								
		nce Count = 0							
🙊 🐺 Internation I and Carling 🛛 💌									~
< >	1							3	
👺 evkmimxrt1060_se05x_Minimal					1		U NXP MIMXRT1062	xxxA* (evkminimal)	)
Figure 19. Debug	nroie	cts in MCI	IXnros	ser	wor	ksnac	9		
riguie is. Debug	pioje		whies	30	, ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	Nopac	6		

6. When the example executes, it will automatically stop in a breakpoint. Click on *Resume* to allow the software to continue its execution as shown in <u>Figure 20</u>:

Debug your project           * Debug           * Terminate, Build and Debug	<b>∑ - ∷ - № -</b> • <		(evkbiimal)	>
Clean				
Build your project				
New project Import SDK example(s) Import project(s) from file syst	[P	■ 第二次 (金田) (金田) (金田) (金田) (金田) (金田) (金田) (金田)		
Come (3 interes (6 online +0 bit	C. W. GO			
Qui 22 (0): Vari 😤 Outl 💁 Bri	e. 10+90 • • • 10	Installed SDKs 🔟 Properties 🐏 Problems 🖸 Console 😥 🦑 Terminal 🔐 Image Info 👖 Memory 144- Heap and Stack Usage 🙀 Debugger Console		- 1
s b fd Ini2r.h		<pre>10 (EA_SSS_BOUL_SKP_SELECL_APPLE: == 1) 31 (PCONTEXT)-&gt;se05x open ctx.skip select apple: = 1:</pre>		
> h fsl_iomuxc.h		29 #if defined(EX_SSS_BOOT_SKIP_SELECT_APPLET) && \ 30 (EX_SSS_BOOT_SKIP_SELECT_APPLET == 1)		
> h fsl_gpio.h		27 }		
> c fsl_gpio.c		126 goto cleanup;		
> c fsl_dcp.c > h fsl_dcp.h		<pre>124 if (kStatus_SSS_Success != status) { 125 LOG E("ex sss boot connectstring Failed"); 126 E("ex sss boot connectstring Failed"); 127 LOG E("ex sss boot connectstring Failed"); 128 LOG E("ex sss boot connectstring Failed"); 129 LOG E("ex ssss boot connectstring Failed"); 129 LOG E("ex sss boot connec</pre>		1
> h fsl_common.h		<pre>status = ex_sss_boot_connectstring(argc, argv, &amp;portName);</pre>		4
) c fsl_common.c		21 #endif		
> [c] fsl_clock.c > [h] fsl_clock.h		<pre>120 memset((EX_SSS_BOOT_PCONTEXT), 0, sizeof(*(EX_SSS_BOOT_PCONTEXT)));</pre>		
drivers		118 119 #ifdef EX SSS BOOT PCONTEXT		
e 😂 device		17 LOG_I(PLUGANDTRUST_PROD_NAME_VER_FULL);		
e component		15 Mendif 16		
board		14 ex_sss_main_ksdk_bm();		1
Signature     Signature	0	Welcome 🗟 fsl_clock.c 🕞 ex_sss_main_inc.h 🕄		2
Project Settings     We Binaries		<ul> <li></li></ul>		
evkmimurt1060_se05x_Minimal <debu< td=""><td>ug&gt;</td><td>✓ Pevkbimxrt106Qse_hostlib_se05x_Minimal.axf [MIMXRT1052cocx8 (cortex·m7)]</td><td></td><td></td></debu<>	ug>	✓ Pevkbimxrt106Qse_hostlib_se05x_Minimal.axf [MIMXRT1052cocx8 (cortex·m7)]		
P 9	Contraction of the second s	Keykbimxrt1060.se, hostlib_se05x_Minimal LinkServer Debug [C/C++ (NXP Semiconductors) MCU Application]		
Project 🙁 😕 Periphe IIII Registi	ers the Faults	Debug 🛛	1 ito =	
		Quick Acces	s 🖪 🖌	Š
• 🖩 🔞   🖲 • 🗞 • 📓 🖉 🛇	🖉 🕶 🗳 💌 💷	• ゆや・ほ・出す 回帰し • ゆ • ゆ • の • 幸 # 2 2 4 4 4 1 0 1 0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	• • •	
		Run Analysis FreeRTOS Window Help		

7. Once the program execution begins, logs are printed on the terminal application indicating the execution status. For the se05x Minimal project example, the logs

should indicate the available memory in the secure element (in this case, 32767) as can be seen in Figure 21.

The same operations can be repeated to run any of the other Plug & Trust middleware project examples.

M	COM44 - Ter	a Term VT		-		]	$\times$
<u>F</u> ile	Edit Setur	Control					
App	:INFC	:PlugAndTrust_v04.01.00_20211217					^
ss	:INFC	:atr (Len=35)					
		01 A0 00 00 03 96 04 03 E8 00 FE 02	ØB	03	E8	00	
		01 00 00 00 00 64 13 88 0A 00 65 53	45	30	35	31	
		00 00 00					
pp	:INFC	mem=32767					
pp	:INFC	:se05x_Minimal Example Success !!!					
pp	:INFC	:ex sss Finished					
		TeraTerm logs - se05x Minimal project example					

## 4.6 Product specific build settings

The NXP Plug & Trust middleware supports the SE05x Secure Element, the A5000 Secure Authenticator, and the legacy A71CH products.

The Plug & Trust Middleware uses the feature file fsl\_sss\_ftr.h to select a dedicated EdgeLock product IC and the corresponding IoT applet or Authenticator application. The fsl\_sss\_ftr.h header file is located in the project source folder.

The SE050 product identification can be obtained as described in <u>AN12436</u> chapter 1 *Product Information*. <u>AN12973</u> describes the same procedure for the SE051 product family.

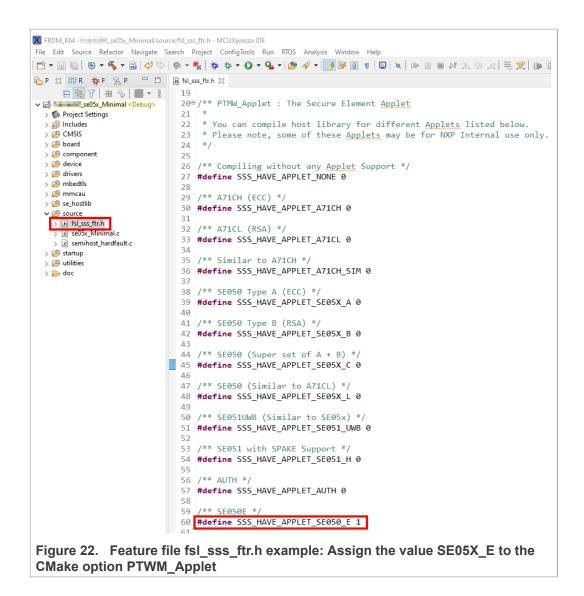
The fsl\_sss\_ftr.h header file includes several compilation options to select a dedicated product variant like: PTWM\_Applet, PTMW\_FIPS, PTMW\_SE05X\_Ver, PTMW\_SE05X\_Auth and PTMW\_SCP.

Select the desired value of the compilation option by setting exclusively the corresponding C-preprocessor define to 1 (enable). All other values for the same option (represented by C-preprocessor defines) must be set to 0.

Example: Assign the value SE050\_E to the compilation option PTWM\_Applet.

# AN12450

#### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170



The following tables show the required PTMW options to build the MCUXpresso SDK for a dedicated product variant. The SSSFTR\_SE05X\_RSA option is used to optimize the memory footprint for product variants that do not support RSA.

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
SE050E Dev. Board OM-SE050ARD-E	A921	SSS_ HAVE_	SSS_ HAVE_	SSS_ HAVE_	any option	SSS_ HAVE_	disabled
SE050E2	A921	APPLET_ SE05X_E	FIPS_ NONE	SE05X_ VER_ 07_02		SCP_NONE or SSS_ HAVE_ SCP_ SCP03_ SSS	

#### Table 4. Feature file fsl\_sss\_ftr.h settings for SE050E product variants

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
SE050F Dev.Board OM-SE050ARD-F	A92A	SSS_ HAVE_	SSS_ HAVE_	SSS_ HAVE_	SSS_HAVE_SE05X_AUTH_ PLATFSCP03	SSS_ HAVE_	enabled
SE050F2	A92A	APPLET_ SE05X_C	FIPS_ SE050	SE05X_ VER_ 03_XX	or SSS_HAVE_SE05X_AUTH_ USERID_PLATFSCP03 or SSS_HAVE_SE05X_AUTH_ AESKEY_PLATFSCP03 or SSS_HAVE_SE05X_AUTH_ ECKEY_PLATFSCP03	SCP_ SCP03_ SSS	

#### Table 5. Feature file fsl\_sss\_ftr.h settings for SE050F product variants

#### Table 6. Feature file fsl\_sss\_ftr.h settings for SE050 Previous Generation product variants

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
SE050A1	A204	sss_	sss_	sss_	any	SSS_	disabled
SE050A2	A205	HAVE_	HAVE_	HAVE_	option	HAVE_	
		APPLET_	FIPS_	SE05X	-	SCP_NONE or	
		SE05X_A	NONE	VER_ 03_XX		SSS	
						HAVE_	
						SCP_	
						SCP03_ SSS	
SE050B1	A202	SSS	SSS_	SSS	any	SSS	enabled
SE050B2	A203	HAVE	HAVE	HAVE	option	HAVE	
		APPLET	FIPS	SE05X		SCP_NONE	
		SE05X_B	NONE	VER_	-	or	
				03_XX		SSS_ HAVE	
						SCP	
						SCP03_	
						SSS	
SE050C1	A200	SSS	SSS_	SSS	any	SSS	enabled
SE050C2	A201	HAVE_	HAVE_	HAVE_	option	HAVE_	
SE050 Dev Board	A1F4	APPLET_	FIPS_	SE05X	-	SCP_NONE	
OM-SE050ARD		SE05X_C	NONE	VER_		or	
				03_XX		SSS_ HAVE_	
						SCP_	
						SCP03_	
						SSS	

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
SE050F2	A77E <sup>[1]</sup>	SSS_ HAVE_ APPLET_ SE05X_C	SSS_ HAVE_ FIPS_ SE050	SSS_ HAVE_ SE05X_ VER_ 03_XX	SSS_HAVE_SE05X_AUTH_ PLATFSCP03 Or SSS_HAVE_SE05X_AUTH_ USERID_PLATFSCP03 Or SSS_HAVE_SE05X_AUTH_ AESKEY_PLATFSCP03 Or SSS_HAVE_SE05X_AUTH_ ECKEY_PLATFSCP03	SSS_ HAVE_ SCP_ SCP03_ SSS	enabled

## Table 6. Feature file fsl\_sss\_ftr.h settings for SE050 Previous Generation product variants...continued

[1] All SE050F2 with variant A77E have date code in year 2021. All the SE050F2 with date code in the year 2022 have the variant identifier A92A.

#### Table 7. Feature file fsl\_sss\_ftr.h settings for SE051 product variants

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
SE051A2	A920	SSS_ HAVE_ APPLET_ SE05X_A	SSS_ HAVE_ FIPS_ NONE	SSS_ HAVE_ SE05X_ VER_ 07_02		SSS_ HAVE_ SCP_NONE or SSS_ HAVE_ SCP_ SCP03_ SSS	disabled
SE051C2	A8FA	SSS_ HAVE_ APPLET_ SE05X_C	SSS_ HAVE_ FIPS_ NONE	SSS_ HAVE_ SE05X_ VER_ 07_02	any option	SSS_ HAVE_ SCP_NONE or SSS_ HAVE_ SCP_ SCP03_ SSS	enabled
SE051W2	A739	SSS_ HAVE_ APPLET_ SE05X_C	SSS_ HAVE_ FIPS_ NONE	SSS_ HAVE_ SE05X_ VER_ 07_02	any option	SSS_ HAVE_ SCP_NONE Or SSS_ HAVE_ SCP_ SCP03_ SSS	enabled

# AN12450

### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
SE051A2	A565	SSS_ HAVE_ APPLET_ SE05X_A	SSS_ HAVE_ FIPS_ NONE	SSS_ HAVE_ SE05X_ VER_ 06_00	any option	SSS_ HAVE_ SCP_NONE Or SSS_ HAVE_ SCP_ SCP03_ SSS	disabled
SE051C2	A564	SSS_ HAVE_ APPLET_ SE05X_C	SSS_ HAVE_ FIPS_ NONE	SSS_ HAVE_ SE05X_ VER_ 06_00 _VER_ 06_00	any option	SSS_ HAVE_ SCP_NONE or SSS_ HAVE_ SCP_ SCP03_ SSS	enabled

## Table 7. Feature file fsl\_sss\_ftr.h settings for SE051 product variants...continued

#### Table 8. Feature file fsl sss ftr.h settings for A5000 product variants

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
OM-A5000ARD	A736	SSS_	SSS_	SSS_	any	SSS_	disabled
A5000	A736	HAVE_ APPLET_ AUTH	HAVE_ FIPS_ NONE	HAVE_ SE05X_ VER_ 07_02	option	HAVE_ SCP_NONE or SSS_ HAVE_ SCP_ SCP03_ SSS	

## 4.6.1 Example: SE050E build settings

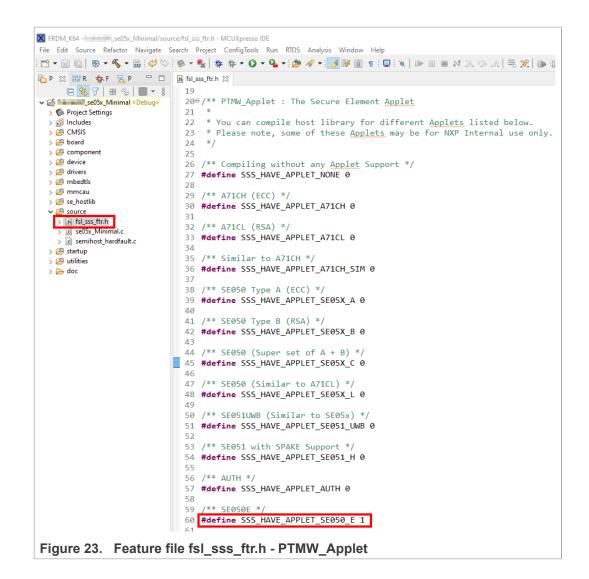
The following images show the configuration for the SE050E development board OM-SE05ARD-E according to Table 4.

1. Select the Applet variant SE050E.

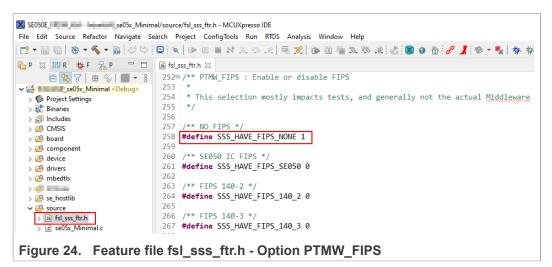
AN12450 **Application note** 

# AN12450

#### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170



#### 2. Select FIPS none.



AN12450 Application note

3. Select Applet version 7.02.

SE050E_SE05x_Minimal/s	source/fsl_sss_ftr.h - MCUXpresso IDE
<u>File Edit Source Refactor Navigate Searce</u>	h <u>P</u> roject ConfigTools <u>R</u> un RTOS Analysis <u>W</u> indow <u>H</u> elp
: 📩 🕶 🔚 🐚   🥹 🕶 🗞 🕶 🔜 : 🛷 😒 : 🚍	×   D = H = M = 0 (R   = 🛪   = 🛪   D = 🕅 = 2. C (R   2   2   2   0   A   .
🖻 P 🖾 🛲 R 🔆 F 🚼 P 🗖 🖪	j fsl_sss_ftr.h ⊠
E 🔄 🏹 🖶 🗞 🔳 🕶 🖇	93
✓ 🚰 🔜 se05x_Minimal <debug></debug>	94⊖/** PTMW_SE05X_Ver : SE05X Applet version.
> 伦 Project Settings	95 *
> 🐉 Binaries	96 * Selection of Applet version 03_XX enables SE050 features.
> 🗊 Includes	97 * Selection of Applet version 06_00 enables SE051 features.
> 📇 CMSIS	98 *
> 🔑 board	99 */
/ / component	100 101 /** SE050 */
/ / device	102 #define SSS HAVE SE05X VER 03 XX 0
	103
/ / mbeutis	103 104 /** SE051 */
	105 <b>#define</b> SSS HAVE SE05X VER 06 00 0
/ _ se_noseno	106
V Source	107 /** SE051 */
	108 #define SSS_HAVE_SE05X_VER_07_02 1
Figure 25. Feature file fsl	sss ftr.h - Option PTMW SE05x Ver

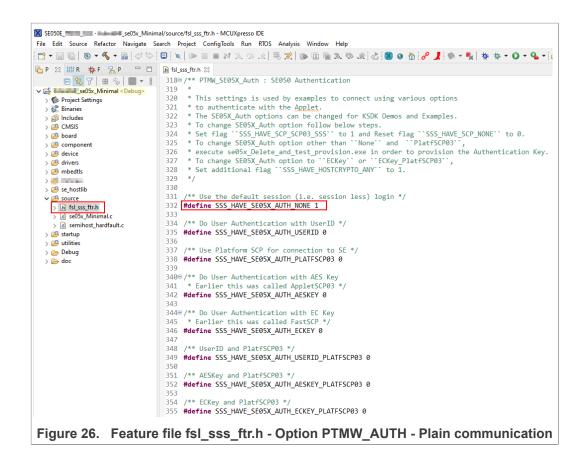
4. In this example we use plain communication. Plain communication for the example execution is enabled by selecting the following options:

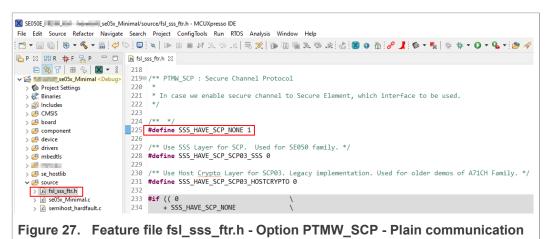
- Set the #define SSS\_HAVE\_SE05X\_AUTH\_NONE option to 1 and disable all other options be setting the flags to 0.
- Set the #define SSS\_HAVE\_SCP\_NONE option to 1 and disable all other options be setting the flags to 0.

How to enable Platform SCP is described in <u>Section 6.3</u>.

## AN12450

#### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

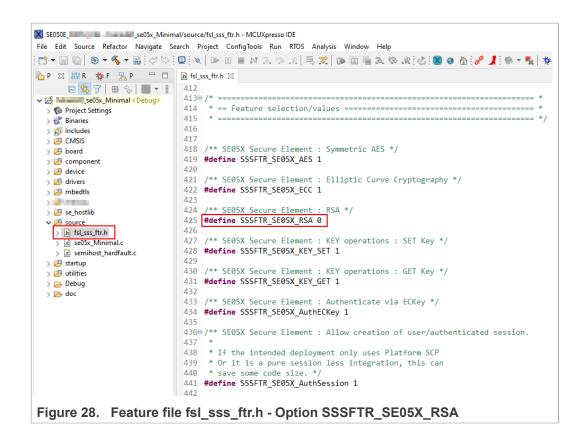




5. To reduce the Plug & Trust middleware memory footprint we disable RSA for the SE050E product variant.

# AN12450

EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170



## 5 Import project examples from CMake-based build system

This section explains how to run EdgeLock SE05x project examples using the CMakebased build system. Although this solution offers the possibility to quickly build the same example code for multiple platforms, the debug experience may be affected by MCUxpresso not being able to make use of the *defines* chosen in CMake.

## 5.1 Prerequisites

The following software tools are required to run projects generated from the CMakebased build system:

- 1. MCUXpresso IDE. Check <u>Section 7</u> for detailed installation instructions.
- 2. CMake. Check Section 8 for detailed installation instructions.
- Python ≥ 3.7.x and ≤ 3.9.x 32-bit version. Check <u>Section 9</u> for detailed installation instructions.
  - Note: higher Python versions may work as well.
- TeraTerm (or an equivalent serial application). You can download and run the TeraTerm installer from this <u>link</u>.

## 5.2 Download the Plug & Trust middleware

Follow these steps to download the Plug & Trust middleware in your local machine:

1. Download Plug & Trust middleware from the NXP website.

- 🍤 🔽 🖛 🛛 Drive Tools Local Disk (C:) Home Share View Manage ← → · ↑ 💺 > This PC > Local Disk (C:) \* ^ Pictures Name Date modified Туре Size Music Intel 2/25/2019 4:12 AM File folder Projects 3/7/2019 1:28 AM File folder nxp Videos PEMicro 3/7/2019 1:34 AM File folder PerfLogs 4/11/2018 4:38 PM File folder a OneDrive 3/11/2019 4:05 AM File folder Program Files S This PC 3/11/2019 3:28 AM Program Files (x86) File folder 🔓 3D Objects Projects 3/11/2019 6:17 AM File folder Python27 3/11/2019 4:53 AM File folder Desktop se05x middleware 3/11/2019 6:27 AM File folder Documents Users 2/25/2019 5:06 AM File folder Downloads 3/11/2019 3:38 AM Windows File folder Music Recovery 2/25/2019 12:46 PM Text Document 0 KB Pictures Videos Local Disk (C:) USB DISK (E:) USB DISK (E:) Figure 29. Create se05x\_middleware folder
- 2. Create a folder called **se05x\_middleware** in C: directory as shown in Figure 29:

 Unzip the Plug & Trust middleware inside the se05x\_middleware folder. After unzipping, you will see a folder called simw-top. The contents of the simw-top directory should look as shown in Figure 30:

* ↑ → This PC → OS	(C:) → se05x_middleware → simw-top		✓ ♂		
	Name	Date modified	Туре	Size	
ick access	akm	28/07/2020 10:22	File folder		
bileKnowledge	binaries	28/07/2020 10:22	File folder		
eDrive - MobileKnowledge	demos	28/07/2020 10:22	File folder		
eprive - wobileknowledge	🔄 doc	28/07/2020 10:22	File folder		
s PC	📙 ext	28/07/2020 10:22	File folder		
D Objects	hostlib	28/07/2020 10:22	File folder		
esktop	📙 mal	28/07/2020 10:22	File folder		
ocuments	nxp_iot_agent	28/07/2020 10:22	File folder		
ownloads	projects	28/07/2020 10:22	File folder		
	pycli	28/07/2020 10:22	File folder		
lusic	scripts	28/07/2020 10:22	File folder		
ictures	sss .	28/07/2020 10:22	File folder		
ideos	tools	28/07/2020 10:22	File folder		
IS (C:)	Android.mk	02/04/2020 22:52	MK File	7 KB	
ture de	CleanSpec.mk	02/04/2020 22:52	MK File	2 KB	
twork	CMakeLists.txt	02/04/2020 22:52	Text Document	4 KB	
	🔒 EULA.pdf	02/04/2020 22:52	Adobe Acrobat D	134 KB	

**Note:** It is recommended to keep  $se05x\_middleware$  with the <u>shortest</u> path possible and <u>without spaces</u> in it. This avoids some issues that could appear when building the middleware if the path contains spaces.

## 5.3 Build the Plug & Trust middleware project examples

The Plug & Trust middleware uses CMake for building the project examples. To build the Plug & Trust middleware, open a Command Prompt and use the following steps as shown in Figure 31:

- Go to the folder where you unzipped the Plug & Trust middleware: (1) Send >> cd C:\se05x\_middleware\simw-top\scripts
- 2. Define the environment:
   (2) Send >> env setup.bat
- 3. Generate the Plug & Trust middleware project examples:
  - (3) Send >> create\_cmake\_projects.py
    Note: This command may take a few seconds to complete.

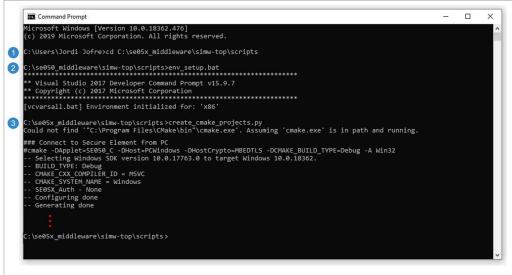


Figure 31. Generate Plug & Trust middleware project examples

Depending on your PC installation you may need to update the application file locations within the <code>env setup.bat</code> file.

4. Your project directory should now contain two folders: a (1) simw-top folder and a (2) simw-top build folder as shown in Figure 32:

	Name	Date modified	Туре	Size
Quick access	1 simw-top	28/07/2020 10:22	File folder	
MobileKnowledge	2 simw-top_build	28/07/2020 10:22	File folder	
OneDrive - MobileKnowledg	e			

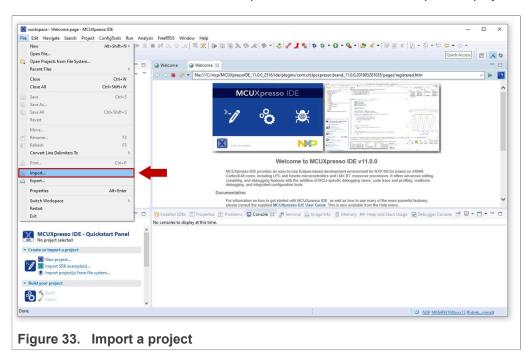
## 5.4 Import PlugAndTrustMW project example in MCUXpresso workspace

After generating the projects in your local machine using the create\_cmake\_projects.py script, you need to import the PlugAndTrustMW project
in your MCUXpresso workspace. Follow these steps to import the project:

AN12430	
Application	note

AN112450

Go to File → Import using the top bar menu as shown in Figure 33.
 Note: In this case, do not use the MCUXpresso Quickstart Panel to import the project.



2. In the import wizard menu, (1) select *Existing Projects into Workspace* from the *General* folder, then (2) click on *Next* as shown in Figure 34:

	_	🔀 Import		×	Quick Acces	
Project 😫 🌄 Periphe 💷 Registers 🎋 Faults 📟 🗖	Welcom	Select				-
	(+ +) <b>=</b>	Create new projects from an archive file or directory.	r 🚬 r		10.201905281035/pages/registered.htm	~ •
		Select an import wizard:				
		type filter text			Balancia Balancia Balancia	
Que 13 10-144_ 32 Out 40 Brg_ 10-160_ 9 1	nstaller	Sensel     1       Barting Roject into Workspace     Fill Schwart into Workspace     1       Fill Schwart     Fill Schwart     1	I	~	the second	• <b>P</b> • □
	No console			_		
MCUXpresso IDE - Quickstart Panel	l		2 Cancel			
Create or import a project						
New project Import SDK example(s) Import project(s) from file system						
Build your project						
S Suild Clean						
			1		U NXP MK64FN1M0xxx12 0	forders to a force of

3. First, you need to import Plug & Trust middleware project in MCUXpresso. For that, in the *Select root directory* option, (1, 2) browse to the location of your Plug & Trust

middleware directory (in this case *C*:\se05x\_middleware\simw-top\_build) and (3) click on *Select folder* as shown in Figure 35:

Pro	N	≡ el le ≡ × □			• 0 • 9 •		✓ Quick Access     ✓ Quick	8 × 8
	Import Projects Select a directory to search for existing Eclipse projects.		p/1/*	Select Folder	ware > cimw-t		Search simw-top, build	X
	Select root directory:	Browse	״ו	Organize - New folder			ji∷ •	0
	O Select archive file:	Browse		3D Objects	^	Name	Date modified	Type
	Projects:	Select All Deselect All	1	Desktop Documents Downloads Music		se_x88 simw-top-eclipse_arm simw-top-eclipse_arm_el2go simw-top-eclipse_ns	05-Aug-20 16:00 05-Aug-20 16:16 05-Aug-20 16:00 05-Aug-20 16:00	File fr File fr File fr
		Refresh	DE es,			simw-top-eclipse_s	05-Aug-20 16:00	File fo
Quic	Options  Sector for meted projects  Cacepy projects into workspace  Cacepy projects into workspace  Helde projects that already exist in the workspace  Working sets		d in n on the tUX t dc	USB DISK (E) USB DISK (E) Conate screenshots conate screenshots 1	~ .	<		
1	Add project to working sets Working sets	New Select		Folder: simw-top_build		3	Select Folder Canc	el
Buil			46					
Deb	Content of the section of the s	Cancel						
	ected V					10	workspace	

 After selecting C:\se05x\_middleware\simw-top\_build folder, a project called *PlugAndTrustMW-Debug@simw-top-eclipse\_arm* should be visible in the *Projects*  area. (1) Select the project and (2) click on the *Finish* button to import this project into your workspace as shown in <u>Figure 36</u>:

X workspace - Welcome page - MCUXpresso IDE					- 🗆 X
File Edit Navigate Search Project ConfigTools Run Ani	1		2012 2022	1	
🗂 • 🔝 🐚   🕲 • 🗞 • 🤣 🖓 🗶 • 💷 🔪   🕨 🛙	0 # N 2. 7 .6	X Import	o x	Ø - ○ · ○ · ○ · ○ · ○ · ○ · ○ · ○ · ○ · ○	Quick Access
🔓 Pro 😥 🎋 De IIII Re 🗱 Fa 🐕 Per 😑 🗖	🥥 Welcome 😫	Import Projects	A.	- C (V) (X)	ъвым б∢г №2 🗖 🗖
🖻 😵 📾 🗣 🛛 🗮 🔹 🤝	00 <b>= 8</b> .	Select a directory to search for existing Eclipse projects.		281035/pages/registere 🗸 🕨 🎦	10 10 10 10 10 10 10 10 10 10 10 10 10 1
		Select root directory: C:\se05x_middleware\simw-top_build	Browse		
		○ Select archive file:	Browse	*** *****	
		Projects:			
		PlugAndTrustMW-Debug@simw-top-eclipse_arm (C:\se05x_middl	Select All		
		PlugAndTrustMW-Debug@simw-top-eclipse_arm_el2go (C:\se05x			
		PlugAndTrustMW-Debug@simw-top-eclipse_ns (C:\se05x_middle PlugAndTrustMW-Debug@simw-top-eclipse_s (C:\se05x_middlew	Deselect All		
			Refresh		
		· · · · · · · · · · · · · · · · · · ·		d on ARM®	
				advanced editing, pfiling, multicore	
	Docu	Options  Search for nested projects			
	- Doite	Copy projects into workspace		verful features,	
U Quickstart Panel		Close newly imported projects upon completion		and the second se	
MCUXpresso IDE - Quickstart Panel		Hide projects that already exist in the workspace		<b>*</b>	,
No project selected	Console 23	Working sets		DKs S Instruction Trace 👖 Memory	
Create or import a project	No consoles to displa	Add project to working sets	New		
New project     Import SDK exemple(s)     Import project(s) from file system		Working sets:	Select		
* Build your project					
Clean		? < Back 2 Finish	Cancel		
→ Debug your project 💽 → 🔛 →					
🗱 🍄 Debug 🏧 Terminate, Build and Debug					
items selected			1	U works	ace

5. The PlugAndTrustMW project should now be imported in your workspace as shown in

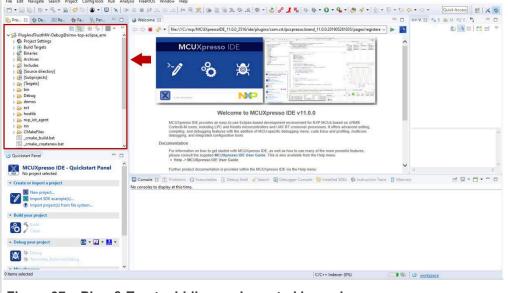
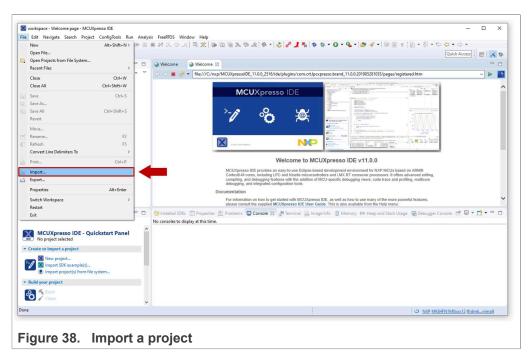


Figure 37. Plug & Trust middleware imported in workspace

### 5.5 Import board project example in MCUXpresso workspace

After importing the *PlugAndTrustMW* project example in MCUXpresso, you need to import the *cmake\_projects\_evkbimxrt1060* project (for i.MX RT1060) or the *cmake\_projects\_evkbimxrt1170* project (for i.MX RT1170). Follow these steps to import the projects:

Go to File → Import using the top bar menu as shown in Figure 38.
 Note: In this case, do not use the MCUXpresso Quickstart Panel to import project.



2. In the import wizard menu, (1) select *Existing Projects into Workspace* from the *General* folder, then (2) click on *Next* as shown in Figure 39:

3 • 🛛 🖏   🥸 • 🔦 •   ∅ ♡   🎱 •   🖳 🔌   ▶ 🗉				L • (2)
Project 💥 🗏 Periphe ነ Registers 🏇 Faults 😐 🗖	Welcon	M Import	— 🗆 ×	
■ 😵 🖬 🖘 🗐 🖛 🍫 🛛 🗮 🔻 🤝		Select Create new projects from an archive file or directory.	Ľ	0.201905281035/pages/registered.htm
		Select an import wizard:		
		type filter text		Marcola Marcola Alexandro Ale
) QuiL 23 10- VerL 32: OutL 90 510- 104 500- 111 1	(i) Installer	Correct     Correct     Correction     Correct	• •	We want of the more powerful features, table to the help mark.
	No console			
MCUXpresso IDE - Quickstart Panel No project selected		? < Back Next >	2 Cancel	
Create or import a project	1			
Mew project Import SDK example(s) Import SDK example(s) Import project(s) from file system				
Build your project				
So Suild Clean				
ne			1	<ul> <li>NXP MK64FN1M0xxx12 (frdmknimal)</li> </ul>

 In the Select root directory option, (1, 2) browse to the location of your i.MX RT1060/1170 projects directory (in our case C:\se05x\_middleware\simw-top\projects), then (3) select the cmake\_projects\_evkbimxrt1060 if you are using the i.MX RT1060

AN12450	
Application note	

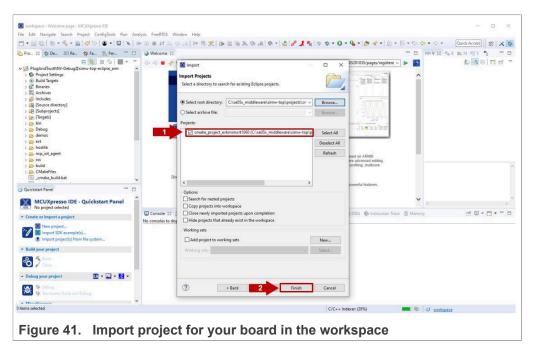
# AN12450

EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

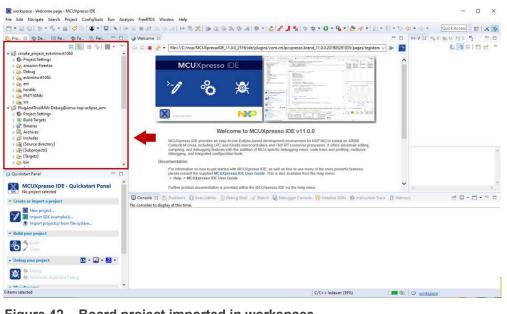
Import  Impo		9 🗐 T 📴 🗙   🍽 🛛 🖷 🖂 🖘 🔅 🗮 🗮 🕲			-1-4	Q	2 1
			■ w w w (W) (W) (W) (W) (W) (W) (W) (W)		10	4	- 18
elect a directory to search for existing belipse projects.		Select Folder					×
Select root directory:	Browse	2 This PC > OS (C:) > se05x_middlewar	e > simw-top > projects >		✓ Ŏ	n projects	
Select archive file:	Browsen	Organize - New folder				8:: •	0
	Browse	MobileKnowledge	^ Name	Date modified	Туре	Size	
ojects:		Finance & Administration Site - Petar Hlad	cmake_project_evkmimxrt1060	22/11/2021 18:33	File folder		
	Select All	MK Collaborative - Documents	cmake_project_evkmimxrt1170	25/11/2021 16:22	File folder		
	Deselect All	<ul> <li>OneDrive - MobileKnowledge</li> </ul>	cmake_project_frdmk64f cmake_project_lpc55s	22/11/2021 17:08 22/11/2021 17:08	File folder File folder		
	Refresh	Microsoft Teams Chat Files	cmake_project_lpc55s_ns	22/11/2021 17:08	File folder		
			cmake_project_lpc55s_s	22/11/2021 17:08	File folder		
		This PC					
		Desktop					
		Documents					
Options Search for nested projects		Downloads					
Copy projects into workspace		h Music					
Close newly imported projects upon completion		E Pictures					
Hide projects that already exist in the workspace		Videos					
Working sets		Prop nut106 S (C:)					
Add project to working sets		bostir BACKUP PHC (D:)					
Working sets:	Select	RT1060-EVK (E:)					
		Google Drive (G:)					
	2	Folder: cmake_project_evkmimurt1060					
					4 Select Folde	er Cancel	_
Kack Next > Finish				_	Selectronde	Caricel	
Sack Next > Finish	Cancel	-					
Clean							
ebug your project 💽 🔹 🔛 🔹							
a 75: Dehua 👻	¢						

board or the *cmake\_projects\_evkbimxrt1170* project if you are using the i.MX RT1170 board. Finally, (4) click *Select folder* as shown in <u>Figure 40</u>:

4. After selecting C:\se05x\_middleware\simw-top\projects folder, the cmake\_projects\_evkbimxrt1060 (or cmake\_projects\_evkbimxrt1170) project should be visible in the Projects area. Click on the Finish button to import this project into your workspace as shown in Figure 41:



 Both The *PlugAndTrustMW* and *cmake\_projects\_evkbimxrt1060* (or *cmake\_projects\_evkbimxrt1170*) projects should now be imported in your workspace as shown in Figure 42:



#### Figure 42. Board project imported in workspace

The two projects need to be imported in the same MCUXpresso workspace. The cmake\_project\_evkmimxrt1060 (or cmake\_project\_evkmimxrt1170) project is used to compile the binary file and debug the solution while the PlugAndTrustMW-Debug@simw-top-eclipse\_arm project contains the source files. Note: In order to be able to set breakpoints within the source code upfront, you need to navigate through the PlugAndTrustMW-Debug@simw-top-eclipse\_arm project files to set the breakpoints. For instance, by navigating to PlugAndTrustMW-Debug@simw-top-eclipse\_arm/[Source\_directory]/demos/se05x/ se05x\_Minimal directory, we can add the desired breakpoints in the project execution of the se05x\_Minimal.c project example.

6. Continue to <u>Section 5.6</u> for instructions on how to execute the project examples.

## 5.6 Execute Plug & Trust middleware examples

This section explains how to:

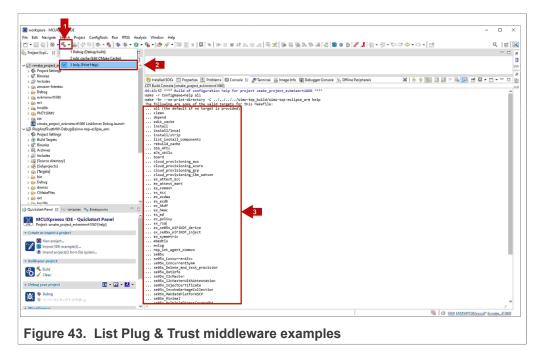
- List the Plug & Trust middleware examples.
- Edit Plug & Trust middleware CMake options.
- Execute a Plug & Trust middleware example.

#### 5.6.1 List the Plug & Trust middleware examples

The Plug & Trust middleware comes with several examples used to verify atomic SE05x security IC features. To get the list of examples, follow these steps:

- 1. Click on the arrow next to the hammer icon in the top bar menu of MCUXpresso.
- 2. Select 3 help (Print help) option. Wait a few seconds until the operation is completed.

3. The MCUXpresso console will display the list of Plug & Trust middleware examples which can be compiled with the currently chosen CMake settings (see Figure 43).



#### 5.6.2 Edit Plug & Trust middleware example CMake options.

The Plug & Trust middleware is delivered with the CMake files that include the set of directives and instructions describing the project's source files and targets. In addition, it includes the CMake configuration files used to enable or disable several features, portability and setting flags to generate the build files for your platform and native build environment.

**Note:** The default build configuration of the Plug & Trust middleware  $\geq \lor 04.02.0x$  generates code for the OM-SE050ARD-E development board. You need to adapt the CMake settings in case you are using a different EdgeLock secure element development board or a different secure element product IC. The settings are described in <u>Section 5.7</u>.

To edit the CMake options, follow the steps shown in Figure 44:

- 1. Click on the arrow next to the *hammer* icon in the top bar menu of MCUXpresso.
- 2. Select 2 edit\_cache (Edit CMake Cache).
- 3. The CMake GUI window will open in your laptop. Using this GUI, change your host platform to evkbimxrt1060 (for i.MX RT1060 board) or to evkbimxrt1170 (for i.MX RT1170 board).
- 4. Click on the **Configure** button.

- \_ - 101 File Edit Navigate St. ch Project ConfigTools Run Analysis FreeRTOS V - Quick Access 🖘 🕫 i 🛕 CMake 3.22.0 - C:/se05x\_middleware/simw-top\_build/simw-top-eclipse\_arm  $\times$ ... XX 🌣 De... 🗸 2 File Tools Options Help ere is the source code: C:/se05x\_middleware/sim Browse Source... Browse Build... s: C:/se05x\_mid -top\_build top-eclipse\_arm 2 🐈 Add Entry 🛛 💥 Re Grouped Advanced Entry Environment. Value X CN x32 MCUXpresso IDE - Quickstart Panel Project: cmake\_project\_evkmimcrt1060 (edit\_ceche import a proje . . . . . . . . . 2-1 4 Configure Generate t lpcxpresso55s n 8 2 - Debug your project 💽 • 🔛 • 🔜 • 😹 🏶 Debug cmake project evkmimxrt1060 Build Project: (45%)
- 5. Click on the **Generate** button to apply the configuration, then close the CMake GUI window.

## Figure 44. Configure CMake options of Plug & Trust middleware examples.

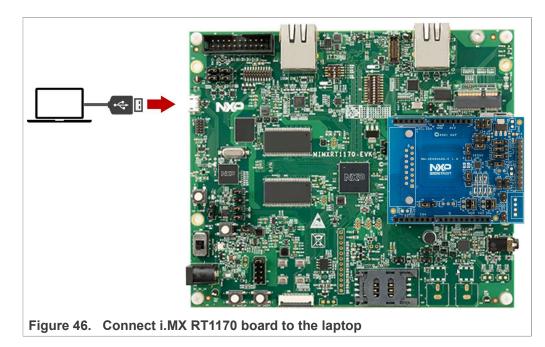
#### 5.6.3 Build and run a Plug & Trust middleware project example

This section explains how to run the Plug & Trust middleware example called  $se05x\_Minimal$ . The  $se05x\_Minimal$  project outputs the memory left in the secure element.

**Note:** The execution of the *se05x\_Minimal* project is shown as an example. The steps detailed in this section can be replicated to run any other example included as part of the Plug & Trust middleware.

To execute the se05x Minimal example, follow these steps:

- Figure 45. Connect i.MX RT1060 board to the laptop
- 1. Connect the i.MX RT1060 board to your laptop as shown in <u>Figure 45</u>, or your i.MX RT1170 as shown in <u>Figure 46</u>:



2. Open TeraTerm. Click *Serial* option and select from the drop-down list the COM port number assigned to your i.MX RT1060 (or i.MX RT1170). Then go to Setup → Serial

Port and configure the terminal to 115200 baud rate, 8 data bits, no parity and 1 stop bit and click OK as shown in Figure 47:

era Term: New co	nnection		×	Sp <u>e</u> ed:	115200 - L	
O TCP/Į₽		myhost.example.com	~	<u>D</u> ata: P <u>a</u> rity:	8 bit ~ none ~	Cancel
	Service:	History         TCP port#: 22           Telnet         TCP port#: 524           SSH         SSH version: SSH2           Other         Other		<u>S</u> top bits: <u>F</u> low control:	1 bit v none v	<u>H</u> elp
◉ S <u>e</u> rial		IP version: AUTO		Device Friendly I Device Instance	nit delay msec/ <u>c</u> har 0 Name: mbed Serial Port ID: USB\VID_0D28&PID	
	ОК	Cancel <u>H</u> elp		Device Manufact Provider Name: r Driver Date: 8-24 Driver Version: 2	mbed  -2009	

- 3. Select the se05x\_Minimal as the project to be executed. For that, follow the steps shown in Figure 48:
  - a. In the Project Explorer window, go to the **Debug** folder and open the **Makefile** file (under *cmake\_project\_evkbimxrt1060* or *cmake\_project\_evkbimxrt1170*).
  - b. The **BUILD\_TARGET** contains the name of the project to be executed. Write se05x Minimal in the **BUILD\_TARGET** variable.
  - c. Click on the arrow on the *hammer* icon in the top bar menu of MCUXpresso.
  - d. Select **1** Debug (Debug build). Wait a few seconds until the build operation completes.

irkspace - cmaket_evkmimxrt1060/Debug/Makefile - I	and the second second					
dit Navigate Search Project ConfigTools Run Ani						
	▶ <u>■                                   </u>	6-1初 - 約 - 40 -	() • · · ·	Quick Access	B	×
😢 🎋 De 💙 1 Debug (Debug build)	Vie d 🕞 Makefile 🕄	- 0	0+V 23 %	B M M GOE	»2	-
2 edit_cache (Edit CMake Cache)		^		At the P	1 19	nŵ.
cmake_project_e 3 help (Print Help)	<pre># cmake_project_evkmimxrt1060/Debug/Makefile</pre>			100 1000 -	1.64	-
Project Settings	4#Update these variables if needed					
Binaries	5 TOP_NAME=sime-top					
🖇 Includes	5 BUILD DIR?=, /, /, //S(TOP_NAME) build/S(TOP_NAME}-eclipse_arm 7 BUILD TARGET=se05x Minimal					
amazon-freertos	7 BUILD_TARGET=se05x_Minimal	_				
- Debug	9# a7x utils					
transfer cmake project extremiment1060.axf - [arm/le]	10# a71ch_Hain					
🚡 Makefile < a	11# a7x_a7ichMainTst					
my selox_MinimaLbin	12 # a7x_vcom 13 # ex common					
in evkmimxrt1060	14# ex_common 14# ex_asymmetric					
b ext	15					
it hostlib	<pre>16ifeq (help, \$(ConfigName))</pre>					
PN7150Nfc	17 BUILD_TARGET+help					
in 555	18 all: help_target 19 else					
PlugAndTrustMW-Debug@simw-top-eclipse_arm	20 ifeq (edit cache, \$(ConfigName))					
Project Settings	21 BUILD TARGET=edit_cache					
Build Targets	22 all: help_target					
Binaries	23 else 24 all: build all					
the Archives	25 endif					
kstart Panel	26 endif					
	27 28# a7x vcom a71ch Nain test session test LoopBack					
MCUXpresso IDE - Quickstart Panel	zo # arx_vcum arzen_moin test_session test_Loopback	Ŷ	120			
Project: cmake_project_evkmimxrt1060 [Debug]		3	1.171			
	😅 Console 💢 🔮 Problems 🔘 Executables 🗍 Debug Shell 🛷 Search 🚇 Debugger Console 🍈 Installed SDKs 🔞 Ins	struction Trace [ Memo	iry.			.0
ate or import a project			0.000			-
Newproject.	CDT Build Console [cmake_project_evkmimurt1060]		A 0 601 79	1 (0021 1026   1		
import SDK example(s)	Scanning dependencies of target se05x Minimal					-
2 Inspant plaject(1) from file system	[ 95%] Building C object demos/se05x/se05x Minimal/CMakeFiles/se05x Minimal.dir/ / / /ext/mcu-s	dk/devices/MTMXRT10	62/xin/fs1 fle	exspi nor boot		
	[ 95%] Building C object demos/se05y/se05y Minimal/CNakeFiles/se05y Minimal.dir/ / /ksdk/common/	boards/evkeimyct1864	R/vin/eukeiewr	t1868 sdram i	ni der	d.c
ild your project	[ 97%] Building C object demos/se05x/se05x_Minimal/CMakeFiles/se05x_Minimal.dir//ksdk/common/	'boards/evkmimxrt1064	8/xip/evkmimxr	t1060_flexspi	_nor_c	con
Suid	[97%] Building ASM object demos/se05x/se05x Minimal/CMakeFiles/se05x Minimal.dir/_/_/ext/mcu	-sdk/devices/MIMXRT	1062/gcc/start	up_MIMXRT1062	.5.0	
Suid	[100%] Building C object demos/se05x_Minimal/CMakeFiles/se05x_Minimal.dir/se05x_Minimal.c.o [100%] Linking C executable///bin/se05x_Minimal.axf					
	se05x Hinimal.axf -> se05x Minimal.bin					
bug your project 💽 • 🔛 •	[100%] Built target se05x_Minimal					
	The second se					
t Debug	17:47:33 Build Finished. 0 errors, 0 warnings. (took 45s.847ms)					
🕸 Terminate, Build and Debug						
	c c					
efile - cmake_project_evkmimkrt1060/Debug			-	(RT1062xxxxA* (cr		-
			NAP MIMA	ALL DREAMAR (C)		

example

4. Go to the MCUXpresso Quickstart Panel and (1) click on the **Debug** button as shown in <u>Figure 49</u>. If there is more than one probe attached, you have to (2) select the CMSIS-DAP debug probe from the list. Wait a few seconds until the project executes:

Pro 😥 🌼 De IIII Re 🏘 Fa 🎘 Per 😁 🗂		Probes discovered					×	• 🙆 🛷 • 🕅 •	- [i] = 🔅	\$ • \$ •	Quick Ac	cess 🗄 🖽	二大な
	@ Welcome	Connect to target: MIM	XRT1062xxxxA						- 0	00+V 23	BBM	r ∈ *2	- 0
🚍 😪 🖽 🗞 🛛 🕶 🌫	1 2# cmake_pro	1 probe found. Select the							^		松	4 B   C	°≊ ⊽
<ul> <li>Project Settings</li> <li>Binaries</li> </ul>		Available attached	probes										
> 👔 Includes > 🚰 amazon-freertos	6 BUILD_DIR?- 7 BUILD_TARGE	Name	Serial number/ID	Type	Manu	IDE Debug Mod	e		- 1				
V Ca Debug	8 9 # a7x ut	CMSIS-DAP	0227000047784	LinkS	ARM	Non-Stop							
> 🕸 cmake_project_evkmimxrt1060.axf - [arm/le]	10 # a71ch												
Makefile	11# a7x_a7												
iii se05x_Minimal.bin	12# a7x_vc 13# ex com						- 11						
> Cate evkmimxrt1060 > Cate ext	14 # ex_asy						- 11						
> Cost hostlib	15 16 ifeg (help.						- 11						
> PN7150Nfc	17 BUILD TARGE						- 11						
> 🕞 SSS	18 all: help_t 19 else	Supported Probes (tick/ur	· · · · · · · · · · · · · · · · · · ·				-						
PlugAndTrustMW-Debug@simw-top-eclipse_arm	20 ifeq (edit	MCUXpresso IDE Link											
>  Project Settings >  Build Targets	21 BUILD_TARGE	P&E Micro probes	server (inc. Civisis-DAP) p	robes									
> & Binaries	22 all: help_t 23 else	SEGGER J-Link probes					- 11						
> III. Archives	24 all: build	ET SEGGER 3-Link probes											
	25 endif 26 endif	Probe search options											
Quickstart Panel	27	Search again											
MCUXpresso IDE - Quickstart Panel	28 # a7x_vcom	Search again							~				
Project: cmake_project_evkmimurt1060 [Debug]	4	Remember my selection	(for this Launch configur	tion)					2	5			3
Create or import a project	🕒 Console 😫 [	Carlot and a second second					- 0	Ks 🚯 Instruction Tra	ce 🚺 Memi	ory			
and the second se		?	2		OK	Cancel				B. 51	8 1 2		- 19 -
New project	RedlinkServer	-	_			1.2	_						
Mimport SDK example(s)	IsOpen = FALSE WireInitialized						-						-
<ul> <li>Import project(s) from file system</li> </ul>	WireInitialized												
Build your project	CoresConfigured												
Build	PacketSize = 64 Reference Count	. 8											
Clean	HasSWV = FALSE												
- Debug your project 💽 + 🔛 + 🔛 -	HasETM = FALSE HasJTAG = TRUE												
The Debug	HasSND = TRUE Probe Type = CMS Probe Reference												
The Territoria Brand and Defrug													Y
• MicroBusses	<												>
Makefile - cmake_project_evkmimxrt1060/Debug										U NXP N	IIMXRT1062xxxx	A* (cmake.	11050)

5. When it executes, it will automatically stop in a breakpoint. Click on *Resume* to allow the software to continue its execution as shown in <u>Figure 50</u>.

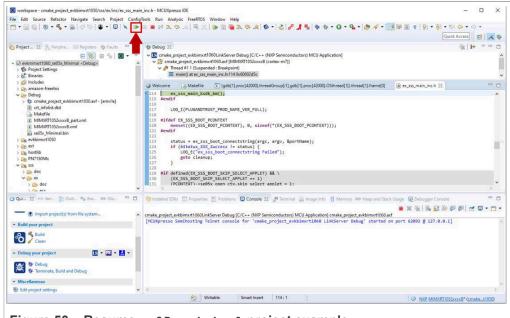


Figure 50. Resume se05x\_minimal project example

6. The project example should now be running in your board. If it is running successfully, the TeraTerm logs should indicate the available memory in the secure element (in this case 32767), as can be seen in Figure 51.

The same operations can be repeated to run any of the other Plug & Trust middleware project examples.

ØB	03	E8	00	
45	30	35	31	
				0B 03 E8 00 45 30 35 31

### 5.7 Product specific CMake build settings

The NXP Plug & Trust middleware supports the SE05x Secure Elements, the A5000 Secure Authenticator, and the legacy A71CH products.

The EdgeLock Plug & Trust middleware is delivered with CMake files that include the set of directives and instructions describing the project's source files and the build targets. The CMake files are used to select a dedicated EdgeLock product IC and the corresponding IoT applet or Authenticator application.

The SE050 product identification can be obtained as described in AN12436 chapter 1 Product Information. AN12973 describes the same procedure for the SE051 product family.

The following tables show the required PTMW CMake options to build a dedicated product variant. The SSSFTR SE05X RSA CMake option is used to optimize the memory footprint for product variants that do not support RSA.

#### Table 9. CMake Settings for SE050E product variants

Variant	OEF	PTMW_	PTMW_	PTMW_	PTMW_SE05X_Auth	PTMW_	SSSFTR_
	ID	Applet	FIPS	SE05X_ Ver		SCP	SE05X_ RSA
SE050E Dev. Board	A921	SE05X_E	None	07_02	any	None	disabled
OM-SE050ARD-E					option	or	
SE050E2	A921	1				SCP03_ SSS	

#### Table 10. CMake Settings for SE050F product variants

Variant	OEF	PTMW_	PTMW_	PTMW_	PTMW_SE05X_Auth	PTMW_	SSSFTR_
	ID	Applet	FIPS	SE05X_ Ver		SCP	SE05X_ RSA
SE050F Dev.Board	A92A	SE05X_C	SE050	03_XX	PlatfSCP03	SCP03_	enabled
OM-SE050ARD-F					or	SSS	
SE050F2	A92A				UserID_PlatfSCP03		
					or		
					AESKey_PlatfSCP03		
					or		
					ECKey_PlatfSCP03		

### Table 11. CMake Settings for SE050 Previous Generation product variants

Variant	OEF	PTMW_	PTMW_	PTMW_	PTMW_SE05X_Auth	PTMW_	SSSFTR_
	ID	Applet	FIPS	SE05X_ Ver		SCP	SE05X_ RSA
SE050A1	A204	SE05X_A	None	03_XX	any	None	disabled
SE050A2	A205	-			option	or SCP03_ SSS	
SE050B1	A202	SE05X_B	None	03_XX	any	None	enabled
SE050B2	A203	-			option	or scp03_ sss	
SE050C1	A200	SE05X_C	None	03_XX	any	None	enabled
SE050C2	A201				option	or	
SE050 Dev Board OM-SE050ARD	A1F4					SCP03_ SSS	
SE050F2	A77E <sup>[1]</sup>	SE05X_C	SE050	03_XX	PlatfSCP03 or UserID_PlatfSCP03 or AESKey_PlatfSCP03 or ECKey_PlatfSCP03	SCP03_ SSS	enabled

[1] All SE050F2 with variant A77E have date code in year 2021. All the SE050F2 with date code in the year 2022 have the variant identifier A92A.

#### Table 12. CMake Settings for SE051 product variants

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
SE051A2	A920	SE05X_A	None	07_02	any	None	disabled
					option	or	
						SCP03_ SSS	

AN12450 Application note

Variant	OEF ID	PTMW_ Applet	PTMW_ FIPS	PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
SE051C2	A8FA	SE05X_C	None	07_02	any option	None <b>or</b> SCP03_ SSS	enabled
SE051W2	A739	SE05X_C	None	07_02	any option	None or SCP03_ SSS or SCP03_ SSS	enabled
SE051A2	A565	SE05X_A	None	06_00	any option	None or SCP03_ SSS	disabled
SE051C2	A564	SE05X_C	None	06_00	any option	None or SCP03_ SSS	enabled

### Table 12. CMake Settings for SE051 product variants...continued

### Table 13. CMake Settings for A5000 product variants

Variant	OEF ID	PTMW_ Applet		PTMW_ SE05X_ Ver	PTMW_SE05X_Auth	PTMW_ SCP	SSSFTR_ SE05X_ RSA
OM-A5000ARD	A736	AUTH	None	07_02	any	None	disabled
A5000	A736	-			option	or	
						SCP03_ SSS	

### 5.7.1 Example: SE050E CMake build settings

The following images show the configuration for the SE050E development board OM-SE05ARD-E according to Table 9.

- Select SE05X E for the CMake option PTWM\_Applet.
- Select None for the CMake option PTWM FIPS.
- Select 07 02 for the CMake option PTWM SE05X Ver.
- Disable the CMake option SSSFTR SE05X RSA.

In this example we use plain communication. Plain communication for the example execution is enabled by selecting the following options:

- Select None for the CMake option PTMW SE05X Auth.
- Select None for the CMake option PTMW SCP.

How to enable Platform SCP is described in Section 6.

AN12450	All information provided in this document is subject to legal disclaimers.	© NXP B.V. 2022. All rights reserved.
Application note	Rev. 4.2 — 4 August 2022	
	546942	42 / 69

### **NXP Semiconductors**

# AN12450

e source code: C:/se05x_mw_v04.02.00_20220527_125816/simw-top <ul> <li><ustom></ustom></li> <li><ustom></ustom></li></ul>	
uild the binaries: C:/se05x_mw_v04.02.00_20220527_125816/simw-top.  Grouped Intries CE V V V V V V V V V V V V V V V V V V	
✓ Grouped       ✓ Advan         Value       Value         puped Entries       ✓         CE       ✓         MW_A71CH_AUTH       None         IMW_Applet       SE050_E         IMW_Host       ✓         IMW_Host       ✓         IMW_Host       ✓         IMW_Host       ✓         IMW_FIPS       None         IMW_Host       ✓         IMW_ROS       Default         IMW_SEDSX_Auth       None         IMW_SE05X_Auth       None         IMW_SE05X_Ver       07_02         IMW_SMCOM       T1ol2C         IMW_mbedTLS_ALT       None         ssorCount       ✓         R       ✓         SFTR_SE05X_AuthECKey       ✓         SFTR_SE05X_AuthECKey       ✓         SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ       ✓         SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ       ✓         SFTR SE05X_KEY_GET       ✓         SFTR_SE05X_KEY_GET       ✓         SFTR_SW_AES       ✓         SFTR_SW_KEY_GET       ✓         SFTR_SW_KEY_GET       ✓         SFTR_SW_KEY_GET       ✓         SFTR_SW_KEY_GET	
Value         vupped Entries         CE         V         IMW_A71CH_AUTH       None         IMW_Applet       SE050_E         IMW_Host       Mupplet         IMW_Host       MBEDTLS         IMW_Log       Default         IMW_StoCrypto       MBEDTLS         IMW_StoCrypto       Default         IMW_StoS       Default         IMW_SEQ       None         IMW_SEDSX_Ver       07_02         IMW_SE05X_Ver       07_02         IMW_SE05X_Ver       07_02         IMW_SE05X_Auth       None         IMW_SE05X_ALT       None         SSFTR_SE05X_ALT       Vone         SSFTR_SE05X_ALT       Vone         SSFTR_SE05X_AUTHSEsion       V         SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ       V         SSFTR_SE05X_KEY_GET       V         SSFTR_SE05X_KEY_SET       V         SSFTR_SE05X_KEY_SET       V         SSFTR_SUP_SET       V         SSFTR_SUP_SET       V         SSFTR_SW_KEY_SET       V         SSFTR_SW_KEY_SET       V         SSFTR_SW_KEY_SET       V         SSFTR_SW_KEY_SET       V	d 🖶 Add Entry 🗱 Remove Entry Environment.
Augued Entries CE V IMW_A71CH_AUTH IMW_Applet IMW_FIPS IMW_Host IMW_Host IMW_Host IMW_Host IMW_Gap IMW_SIC IMW_SOS IMW_SSD IMW_SSD IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD IMW_SCP IMW_SSD I	
KE V IMW_A71CH_AUTH None IMW_Applet SE050_E IMW_FIPS None IMW_Host IMW_Host IMW_Host IMW_Gap Default IMW_SE057 IMW_SE057 IMW_SE057 IMW_SE057 IMW_SE057 IMW_SE057 IMW_SE057 IMW_SMCOM T1012C IMW_mbedTLS_ALT None SSFTR_SE057 IMW_SMCOM T1012C IMW_mbedTLS_ALT None SSFTR_SE057 IMW_SMCOM T1012C IMW_mbedTLS_ALT None SSFTR_SE057 IMW_SMCOM IN02C IMW_SMCOM IN02C	
V IMW_A71CH_AUTH None IMW_Applet SE050_E IMW_Host IMW_Host IMW_Host IMW_Host IMW_ROSCP IMW_SE05X_AUTH IMW_SE05X_AUTH IMW_SE05X_AUTH IMW_SE05X_Ver 07_02 IMW_SE05X_Ver 07_02 IMW_SE05X_Ver 07_02 IMW_SE05X_Ver 07_02 IMW_SE05X_AUTH IMW_MEDTLS_ALT None SSFTR_SE05X_ALT SSFTR_SE05X_AUTHECKey SSFTR_SE05X_AUTHECKey SSFTR_SE05X_AUTHECKey SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_RAA SSFTR_SE05X_RAA SSFTR_SUSSX_RAA SSFTR_SUSSX_RAA SSFTR_SW_SEC SSFTR_SW_AES SSFTR_SW_AES SSFTR_SW_AES SSFTR_SW_AES SSFTR_SW_AES SSFTR_SW_SET SSFTR_SW_SET SSFTR_SW_RSA SSFTR_SW_TESTCOUNTERPART	
IMW_Applet     SE050_E       IMW_Host     None       IMW_Host     MBEDTLS       IMW_Log     Default       IMW_RTOS     Default       IMW_SBL     None       IMW_SCP     None       IMW_SEO5X_Auth     None       IMW_SE05X_Ver     07_02       IMW_SMCOM     T1o12C       IMW_SMCOM     T1o12C       IMW_SMCOM     T1o12C       IMW_SSTR_SE05X_AuthECKey     Immediate       SFTR_SE05X_AuthSession     Immediate       SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ     Immediate       SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ     Immediate       SFTR_SE05X_REY_GET     Immediate       SFTR_SE05X_REY_SET     Immediate       SFTR_SE05X_REY_GET     Immediate       SFTR_SE05X_REY_SET     Immediate       SFTR_SW_AES     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_TESTCOUNTERPART     Immediate	
IMW_FIPS     None       IMW_Host     MBEDTLS       IMW_Host     Default       IMW_Log     Default       IMW_RTOS     Default       IMW_SBL     None       IMW_SCP     None       IMW_SCSX_Auth     None       IMW_SE05X_Auth     None       IMW_SMCOM     T1ol2C       IMW_SMCOM     T1ol2C       IMW_SMCOM     T1ol2C       IMW_SSTR_SE05X_ALT     None       SSFTR_SE05X_ALT     None       SSFTR_SE05X_ALT     None       SSFTR_SE05X_CALS     Image: SSFTR_SE05X_ACS       SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ     Image: SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ       SSFTR_SE05X_KEY_GET     Image: SSFTR_SE05X_KEY_GET       SSFTR_SE05X_KEY_GET     Image: SSFTR_SE05X_KEY_SET       SSFTR_SW_AES     Image: SSFTR_SW_KEY_GET       SSFTR_SW_KEY_GET     Image: SSFTR_SW_KEY_GET       SSFTR_SW_KEY_GET     Image: SSFTR_SW_TESTCOUNTERPART	
IMW_Host       MBEDTLS         IMW_Log       Default         IMW_RTOS       Default         IMW_STOS       Default         IMW_STOS       Default         IMW_SE05X_AUTH       None         IMW_SE05X_Auth       None         IMW_SE05X_Ver       07_02         IMW_BEDTLS_ALT       None         ssorCount       R         SFTR_SE05X_AES       SSFTR_SE05X_AuthECKey         SSFTR_SE05X_AuthSession       SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ         SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ       SSFTR_SE05X_ECY_GET         SSFTR_SE05X_KEY_SET       SSFTR_SE05X_KEY_SET         SSFTR_SE05X_RSA       SSFTR_SUSST_ST         SSFTR_SUSST_RSA       SSFTR_SW_AES         SSFTR_SW_AES       SSFTR_SW_SET         SSFTR_SW_AES       SSFTR_SW_SET         SSFTR_SW_AES       SSFTR_SW_SET         SSFTR_SW_KEY_SET       SSFTR_SW_KEY_SET         SSFTR_SW_KEY_SET       SSFTR_SW_KEY_SET         SSFTR_SW_RSA       SSFTR_SW_TESTCOUNTERPART	
IMW_HostCrypto     MBEDTLS       IMW_Log     Default       IMW_RTOS     Default       IMW_STOS     Default       IMW_SBL     None       IMW_SEO5X_Auth     None       IMW_SEO5X_Ver     07_02       IMW_SMCOM     Tiol2C       IMW_mbedTLS_ALT     None       ssorCount     R       SFTR_SE05X_AES     Immediate       SFTR_SE05X_AuthECKey     Immediate       SFTR_SE05X_AuthSession     Immediate       SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ     Immediate       SFTR_SE05X_KEY_GET     Immediate       SFTR_SE05X_RSA     Immediate       SFTR_SE05X_RSA     Immediate       SFTR_SE05X_RSA     Immediate       SFTR_SE05X_RSA     Immediate       SFTR_SUBST_SENT     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_KEY_SET     Immediate       SFTR_SW_TESTCOUNTERPART     Immediate	
IMW_Log     Default       IMW_RTOS     Default       IMW_RTOS     Default       IMW_SBL     None       IMW_SEDSX_Ver     None       IMW_SEO5X_Ver     07_02       IMW_SMCOM     Tiol2C       IMW_mbedTLS_ALT     None       ssorCount     R       R     SFTR_SE05X_AuthECKey       SFTR_SE05X_AuthECKey     Image: SFTR_SE05X_AuthSession       SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ     Image: SFTR_SE05X_REV_GET       SFTR_SE05X_REV_GET     Image: SFTR_SE05X_REV_SET       SFTR_SE05X_RSA     Image: SFTR_SW_REY_SET       SFTR_SW_KEY_GET     Image: SFTR_SW_KEY_SET       SFTR_SW_KEY_SET     Image: SFTR_SW_KEY_SET       SFTR_SW_TESTCOUNTERPART     Image: SFTR_SW_TESTCOUNTERPART	
IMW_RTÖS Default IMW_SBL None IMW_SCP None IMW_SCP None IMW_SE05X_Auth None IMW_SE05X_Ver 07_02 IMW_SMCOM Tiol2C IMW_mbedTLS_ALT None ssorCount R ISSFTR_SE05X_ALT SSFTR_SE05X_ACT SSFTR_SE05X_AUTHECKey ISSFTR_SE05X_AUTHECKey ISSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ ISSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ ISSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ ISSFTR_SE05X_KEY_GET ISSFTR_SE05X_RSA ISSFTR_SW_AES ISSFTR_SW_AES ISSFTR_SW_AES ISSFTR_SW_AES ISSFTR_SW_ECC ISSFTR_SW_KEY_GET ISSFTR_SW_KEY_GET ISSFTR_SW_KEY_GET ISSFTR_SW_RSA ISSFTR_SW_TESTCOUNTERPART ISSFTR_SW_TESTCOUNTERPART	
IMW_SBL None IMW_SCP None IMW_SCP IMW_SCPSX_Auth None IMW_SE05X_Ver 07_02 IMW_SMCOM T1ol2C IMW_mbedTLS_ALT None ssorCount R SSFTR_SE05X_ALT SSFTR_SE05X_ALTHECKey SSFTR_SE05X_AuthSession SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_KEY_GET SSFTR_SE05X_REY_GET SSFTR_SW_AES SSFTR_SW_AES SSFTR_SW_AES SSFTR_SW_REY_GET SSFTR_SW_KEY_GET SSFTR_SW_KEY_GET SSFTR_SW_KEY_GET SSFTR_SW_KEY_GET SSFTR_SW_KEY_GET SSFTR_SW_RES SSFTR_STR_SS SSFTR_SS	
INW_SE05X_Auth     None       INW_SE05X_Ver     07_02       INW_SMCOM     Tiol2C       INW_mbedTLS_ALT     None       sorCount     ************************************	
INW_SE05X_Ver 07_02 INW_SMCOM Tiol2C INW_mbedTLS_ALT None ssorCount R ISFTR_SE05X_AES SFTR_SE05X_AUTHECKey SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SFTR_SE05X_REY_GET SFTR_SE05X_REY_GET SFTR_SU_AES SFTR_SW_AES SFTR_SW_AES SFTR_SW_ECC SFTR_SW_KEY_GET SFTR_SW_KEY_GET SFTR_SW_KEY_GET SFTR_SW_KEY_SET SFTR_SW_RSA SFTR_SW_TESTCOUNTERPART SFTR_SW_TESTCOUNTERPART	
INW_SMCOM Tiol2C INW_mbedTLS_ALT None ssorCount R SSFTR_SE05X_AES  SSFTR_SE05X_AuthECKey SSFTR_SE05X_AuthSession SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_ECC SSFTR_SE05X_KEY_GET SSFTR_SE05X_KEY_GET SSFTR_SW_AES SSFTR_SW_AES SSFTR_SW_ECC SSFTR_SW_KEY_GET SSFTR_SW_KEY_GET SSFTR_SW_KEY_GET SSFTR_SW_KEY_GET SSFTR_SW_RES SSFTR_SW_RES SSFTR_SW_RES SSFTR_SW_TESTCOUNTERPART	
IMW_mbedTLS_ALT None ssorCount R SSFTR_SE05X_AES SSFTR_SE05X_AuthECKey SSFTR_SE05X_AuthSession SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SW_CREATE SSFTR_SW_AES SSFTR_SW_AES SSFTR_SW_ECC SSFTR_SW_ECC SSFTR_SW_REATE SSFTR_SW_REATE SSFTR_SW_TESTCOUNTERPART SSFTR_SW_TESTCOUNTERPART SSFTR_SW_TESTCOUNTERPART	
ssorCount R SSFTR_SE05X_AES SSFTR_SE05X_AuthECKey SSFTR_SE05X_AuthSession SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SSFTR_SW_KEY_SET SSFTR_SW_KEY_SET SSFTR_SW_TESTCOUNTERPART SSFTR_SW_TESTCOUNTERPART	
R SFTR_SE05X_AES SFTR_SE05X_AuthECKey SFTR_SE05X_AuthECKey SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SFTR_SE05X_CREATE_DELETE_CRYPTOOBJ SFTR_SE05X_KEY_GET SFTR_SE05X_REY_GET SFTR_SE05X_RSA SFTR_SW_AES SFTR_SW_AES SFTR_SW_AES SFTR_SW_KEY_SET SFTR_SW_KEY_SET SFTR_SW_KEY_SET SFTR_SW_RSA SFTR_SW_TESTCOUNTERPART	
SFTR_SE05X_AES	
ISFTR_SE05X_AuthECKey	
ISFTR_SW_AES	
ISFTR_SW_TESTCOUNTERPART	
Press Configure to undate and display new values in red. then pres	
Press Configure to undate and display new values in red, then pres	
Press Configure to undate and display new values in red, then pres	
Press Configure to update and display new values in red, then pres	
ress configure to aparte and asplay new values in real uter pres	Generate to generate selected build files.
e Generate Open Project Current Generator: Edipse CDT4 -	ix Makefiles

### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

### 6 Binding EdgeLock SE05x to a host using Platform SCP

Binding is a process to establish a pairing between the IoT device host MPU/MCU and EdgeLock SE05x, so that only the paired MPU/MCU is able to use the services offered by the corresponding EdgeLock SE05x and vice versa.

AN12450
Application note

A mutually authenticated, encrypted channel will ensure that both parties are indeed communicating with the intended recipients and that local communication is protected against local attacks, including man-in-the-middle attacks aimed at intercepting the communication between the MPU/MCU and the EdgeLock SE05x and physical tampering attacks aimed at replacing the host MPU/MCU or EdgeLock SE05x.

EdgeLock SE05x natively supports Global Platform Secure Channel Protocol 03 (SCP03) for this purpose. PlatformSCP uses SCP03 and can be enabled to be mandatory.

This chapter describes the required steps to enable Platform SCP in the middleware for EdgeLock SE05x.

The following topics are discussed:

- <u>Section 6.1</u> Introduction to the Global Platform Secure Channel Protocol 03 (SCP03)
- <u>Section 6.2</u> How to configure the Platform SCP keys in the i.MX RT1060 MCUXpresso SDK
- Section 6.3 How to enable Platform SCP in the i.MX RT1060 MCUXpresso SDK
- Section 6.4 How to configure the Platform SCP keys in CMake-based build system
- Section 6.5 How to enable Platform SCP in the CMake-based build system

# 6.1 Introduction to the Global Platform Secure Channel Protocol 03 (SCP03)

The Secure Channel Protocol SCP03 authenticates and protects locally the bidirectional communication between host and EdgeLock SE05x against eavesdropping on the physical I2C interface.

EdgeLock SE05x can be bound to the host by injecting in both the host and EdgeLock SE05x the same unique SCP03 AES key-set and by enabling the Platform SCP feature in the Plug & Trust middleware. The <u>AN12662</u> *Binding a host device to EdgeLock SE05x* describes in detail the concept of secure binding.

SCP03 is defined in <u>Global Platform Secure Channel Protocol '03' - Amendment D v1.2</u> specification.

SCP03 can provide the following three security goals:

#### Mutual authentication (MA)

 Mutual authentication is achieved through the process of initiating a Secure Channel and provides assurance to both the host and the EdgeLock SE05x entity that they are communicating with an authenticated entity.

### Message Integrity

- The Command- and Response-MAC are generated by applying the CMAC according NIST SP 800-38B.
- Confidentiality
  - The message data field is encrypted across the entire data field of the command message to be transmitted to the EdgeLock SE05x, and across the response transmitted from the EdgeLock SE05x.

The SCP03 secure channel is set up via the EdgeLock SE05x Java Card OS Manager using the standard ISO7816-4 secure channel APDUs.

The establishment of an SCP03 channel requires three static 128-bit AES keys shared between the two communicating parties: Key-ENC, Key-MAC and Key-DEK. These keys

© NXP	B.V.	2022.	All	rights	reserved

AN12450

are stored in the Java Card Secondary Security Domain (SSD) and not in the secure authenticator applet.

Key-ENC and Key-MAC keys are used during the SCP03 channel establishment to generate the session keys. Session Keys are generated to ensure that a different set of keys are used for each Secure Channel Session to prevent replay attacks.

Key-ENC is used to derive the session key S-ENC. The S-ENC key is used for encryption/decryption of the exchanged data. The session keys S-MAC and R-MAC are derived from Key-MAC and used to generate/verify the integrity of the exchanged data (C-APDU and R-APDU).

Key-DEK key is used to encrypt new SCP03 keys in case they get updated.

Table 14. Static SCP03 keys

Key	Description	Usage	Кеу Туре
Key-ENC	Static Secure Channel Encryption Key	Generate session key for Decryption/ Encryption (AES)	AES 128
Кеу-МАС	Static Secure Channel Message Authentication Code Key	Generate session key for Secure Channel authentication and Secure Channel MAC Verification/Generation (AES)	AES 128
Key-DEK	Data Encryption Key	Sensitive Data Decryption (AES)	AES 128

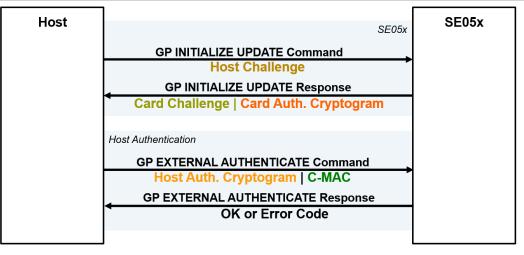
The session key generation is performed by the Plug & Trust middleware host crypto.

#### Table 15. SCP03 session keys

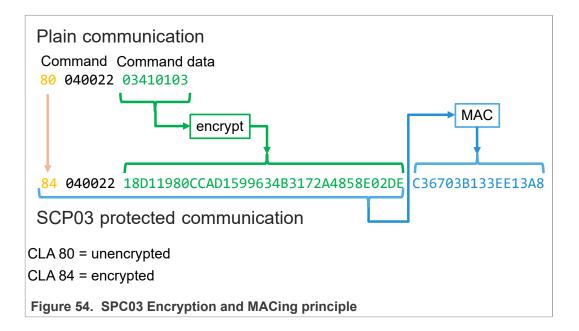
Кеу	Description	Usage	Кеу Туре
S-ENC	Session Secure Channel Encryption Key	Used for data confidentiality	AES 128
S-MAC	Secure Channel Message Authentication Code Key for Command	Used for data and protocol integrity	AES 128
S-RMAC	Secure Channel Message Authentication Code Key for Response	User for data and protocol integrity	AES 128

*Note:* For further details please refer to <u>Global Platform Secure Channel Protocol '03' -</u> <u>Amendment D v1.2</u>.

EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170







# 6.2 How to configure the Platform SCP keys in the i.MX RT1060 MCUXpresso SDK

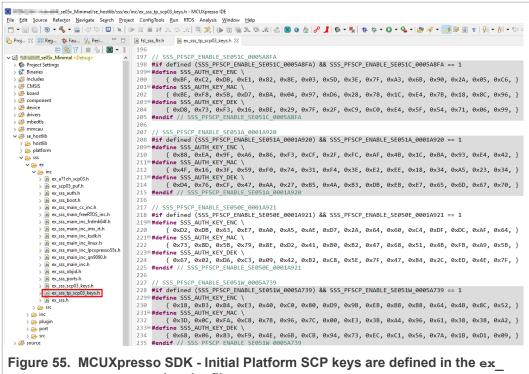
The product specific initial Platform SCP key values are described for the EdgeLock SE05x product variants in <u>AN12436</u> and for the EdgeLock SE051 variants in <u>AN12973</u>.

The Plug & Trust middleware header file <code>ex\_sss\_tp\_scp03\_keys.h</code> contains the initial values of all EdgeLock SE05x, EdgeLock SE051, A5000 and A71CH product variants.

The <code>ex\_sss\_tp\_scp03\_keys.h</code> header file can be found in the following location:

.\se hostlib\sss\ex\inc\

### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170



sss\_tp\_scp03\_keys.h header file.

The fsl\_sss\_ftr.h header file inlcudes compilation options to select one of the predefined initial Platform SCP keys.

Select the desired value of the compilation option by setting exclusively the corresponding C-preprocessor define  $SSS\_PFSCP\_ENABLE\_xx$  to 1 (enable). All other values for the same option (represented by C-preprocessor defines  $SSS\_PFSCP\_ENABLE\_xx$ ) must be set to 0.

EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

se05x_Minim	nal/source/fsl_sss_ftr.h - MCUXpresso IDE
<u>File Edit Source Refactor Navig</u>	jate Se <u>a</u> rch <u>P</u> roject ConfigTools <u>R</u> un RTOS Analysis <u>W</u> indow <u>H</u> elp
📑 🗝 🔚 🐚   📎 🖛 🗞 🖛 🔜   <	성 등 🖻 🛯 🖉 🐨 💷 🔤 것 🗇 다 🗏 🗢 🖄 👘 💷 💷 🖉 👘 🖏
P 🛛 1889 R 🔭 🗆 🗖	h fsl_sss_ftr.h ☆
🕒 🔄 🏹 🖶 🎭 🖾 🕶 🕴	590⊕ /* Enable one of these
✓ 🚰 Internation_se05x_Minimal < Debute	
> 🌮 Project Settings	592 */
> 🐉 Binaries	593 #define SSS PFSCP ENABLE SE050A1 0
> 🗊 Includes	594 #define SSS PFSCP ENABLE SE050A2 0
> 😂 CMSIS	595 #define SSS PFSCP ENABLE SE05082 0
> 🔑 board	596 #define SSS PFSCP ENABLE SE050B1 0
> 🔑 component	597 #define SSS PFSCP ENABLE SE050C1 0
> 😕 device	598 #define SSS PFSCP ENABLE SE050C2 0
> 📇 drivers	599 #define SSS PFSCP ENABLE SE050 DEVKIT 0
> 🔁 mbedtls	600 #define SSS PFSCP ENABLE SE051A2 0
> 😂 mmcau	601 #define SSS PFSCP ENABLE SE051C2 0
> 🔑 se_hostlib 🗸 🚰 source	602 #define SSS PFSCP ENABLE SE050F2 0
> h fsl sss ftr.h	603 #define SSS PFSCP ENABLE SE051C 0005A8FA 0
> .c se05x_Minimal.c	604 #define SSS PFSCP ENABLE SE051A 0001A920 0
> c semihost_hardfault.c	605 #define SSS PFSCP ENABLE SE050E 0001A921 1
> 📴 startup	606 <b>#define</b> SSS PFSCP ENABLE SE051W 0005A739 0
> 🔑 utilities	607 <b>#define</b> SSS PFSCP ENABLE A5000 0004A736 0
> 👝 Debug	608 <b>#define</b> SSS PFSCP ENABLE SE050F2 0001A92A 0
> 🗁 doc	609 #define SSS PFSCP ENABLE OTHER Ø
Figure 56. Select the a file.	actual Platform SCP keys in the fsl_sss_ftr.h header

The following tables contains the the Platform SCP key header file define to be set to 1 (enable) for the different secure element and secure authenticator product variants.

Table 16.	Platform SCP	key define	prefix for	SE050E	product variants
-----------	--------------	------------	------------	--------	------------------

Variant	OEF ID	Platform SCP key define to be set to '1'
SE050E Dev. Board OM-SE050ARD-E	A921	SSS_PFSCP_ENABLE_SE050E_0001A921
SE050E2	A921	SSS_PFSCP_ENABLE_SE050E_0001A921

#### Table 17. Platform SCP key define prefix for SE050F product variants

Variant	OEF ID	Platform SCP key define to be set to '1'
SE050F Dev.Board	A92A	SSS_PFSCP_ENABLE_SE050F2_0001A92A
OM-SE050ARD-F		
SE050F2	A92A	SSS_PFSCP_ENABLE_SE050F2_0001A92A

# Table 18. Platform SCP key define prefix for SE050 Previous Generation product variants

Variant	OEF ID	Platform SCP key define to be set to '1'
SE050A1	A204	SSS_PFSCP_ENABLE_SE050A1
SE050A2	A205	SSS_PFSCP_ENABLE_SE050A2
SE050B1	A202	SSS_PFSCP_ENABLE_SE050B1
SE050B2	A203	SSS_PFSCP_ENABLE_SE050B2
SE050C1	A200	SSS_PFSCP_ENABLE_SE050C1
SE050C2	A201	SSS_PFSCP_ENABLE_SE050C2

 Table 18. Platform SCP key define prefix for SE050 Previous Generation product

 variants...continued

Variant	OEF ID	Platform SCP key define to be set to '1'	
SE050 Dev Board	A1F4	SSS_PFSCP_ENABLE_SE050_DEVKIT	
OM-SE050ARD			
SE050F2	A77E <sup>[1]</sup>	SSS_PFSCP_ENABLE_SE050F2	

[1] All SE050F2 with variant A77E have date code in year 2021. All the SE050F2 with date code in the year 2022 have the variant identifier A92A.

Variant	OEF ID	Platform SCP key define to be set to '1'
SE051A2	A920	SSS_PFSCP_ENABLE_SE051A_0001A920
SE051C2	A8FA	SSS_PFSCP_ENABLE_SE051C_0005A8FA
SE051W2	A739	SSS_PFSCP_ENABLE_SE051W_0005A739
SE051A2	A565	SSS_PFSCP_ENABLE_SE051A2
SE051C2	A564	SSS_PFSCP_ENABLE_SE051C2

#### Table 19. Platform SCP key define prefix for SE051 product variants

#### Table 20. Platform SCP key define prefix for A5000 product variants

Variant	OEF ID	Platform SCP key define to be set to '1'
A5000 Dev. Board OM-A5000ARD	A736	SSS_PFSCP_ENABLE_A5000_0004A736
A5000	A736	SSS_PFSCP_ENABLE_A5000_0004A736

In the next step it is necessary to enable Platfrom SCP in the Plug & Trust middleware. Section 6.3 describes how to enable Platform SCP in the <u>Binding EdgeLock SE05x to a</u> host MCU/MPU using Platform SCP.

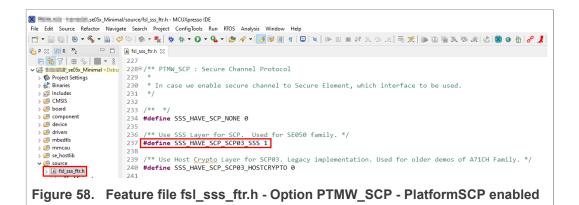
### 6.3 How to enable Platform SCP in the i.MX RT1060 MCUXpresso SDK

To enable Platform SCP is required to rebuild the SDK with the following options:

- Set exclusively the C-preprocessor define SSS\_HAVE\_SE05X\_AUTH\_PLATFSCP03 to 1 to configure PTMW SE05X Auth.
- Set exclusively the C-preprocessor define SSS\_HAVE\_SCP\_SCP03\_SSS to 1 to configure PTMW SCP.

### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

	୬ % : ⊗ + 💺   🐐 ☆ + O + 🂁 + ! 🥙 🖋 + 🚺 😡 🗐 π : 🖳 🍬   🕨 🗉 🖬 X ス େ ル! ≒ 🕱   ℔ 🛍 🖕 ス ጭ ル! ሪ ኔ ! (
P ⊠ 1119 R <sup>3</sup> 2	h fsl_sss_ftr.h ⊠
🕒 😫 🏹 🖶 🍫 🔳 🔻 🕴	325
😂 🔜 📲 _se05x_Minimal < Debu	
> 🍥 Project Settings	3270/** PTMW_SE05X_Auth : SE050 Authentication
> 🖑 Binaries	328 *
> 🔊 Includes > 🤔 CMSIS	329 * This settings is used by examples to connect using various options
> 😂 board	330 * to authenticate with the Applet.
> 🔑 component	331 * The SE05X_Auth options can be changed for KSDK Demos and Examples.
> 🔑 device	332 * To change SE05X_Auth option follow below steps.
> 🕒 drivers	333 * Set flag ``SSS_HAVE_SCP_SCP03_SSS`` to 1 and Reset flag ``SSS_HAVE_SCP_NONE`` to 0. 334 * To change SE05X Auth option other than ``None`` and ``PlatfSCP03``,
> 🔑 mbedtls	334 * To change SE05X_Auth option other than None and Plattscr03 , 335 * execute se05x_Delete_and_test_provision.exe in order to provision the Authentication Key
> 冯 mmcau	336 * To change SE05X_Auth option to ``ECKey`` or ``ECKey_PlatfSCP03``,
> 🔑 se_hostlib	337 * Set additional flag `SSS_HAVE_HOSTCRYPTO_ANY`` to 1.
✓ <sup>™</sup> source	338 */
> h fsl_sss_ftr.h	339
> c se05x_Minimal.c	340 /** Use the default session (i.e. session less) login */
> 🖻 semihost_hardfault.c	341 #define SSS HAVE SE05X AUTH NONE 0
> 🔑 utilities	342
> 🅞 Debug	343 /** Do User Authentication with UserID */
> > doc	344 #define SSS HAVE SE05X AUTH USERID 0
	345
	346 /** Use Platform SCP for connection to SE */
	347 #define SSS_HAVE_SE05X_AUTH_PLATFSCP03 1
	348
	349⊜/** Do User Authentication with AES Key
	350 * Earlier this was called AppletSCP03 */
	351 #define SSS_HAVE_SE05X_AUTH_AESKEY 0
	352
	353⊖/** Do User Authentication with EC Key
	354 * Earlier this was called FastSCP */
	355 #define SSS_HAVE_SE05X_AUTH_ECKEY 0
	356
	357 /** UserID and PlatfSCP03 */
	358 #define SSS_HAVE_SE05X_AUTH_USERID_PLATFSCP03 0
	360 /** AESKey and PlatfSCP03 */
	361 #define SSS_HAVE_SE05X_AUTH_AESKEY_PLATFSCP03 0 362
	362 363 /** ECKey and PlatfSCP03 */
	364 #define SSS_HAVE_SE05X_AUTH_ECKEY_PLATFSCP03 0



# 6.4 How to configure the Platform SCP keys in CMake-based build system

The product specific initial Platform SCP key values are described for the EdgeLock SE05x product variants in <u>AN12436</u> and for the EdgeLock SE051 variants in <u>AN12973</u>.

The Plug & Trust middleware header file <code>ex\_sss\_tp\_scp03\_keys.h</code> contains the initial values of all EdgeLock SE05x, EdgeLock SE051, A5000 and A71CH product variants.

The ex\_sss\_tp\_scp03\_keys.h header file location in the following location: .\simw-top\sss\ex\inc\

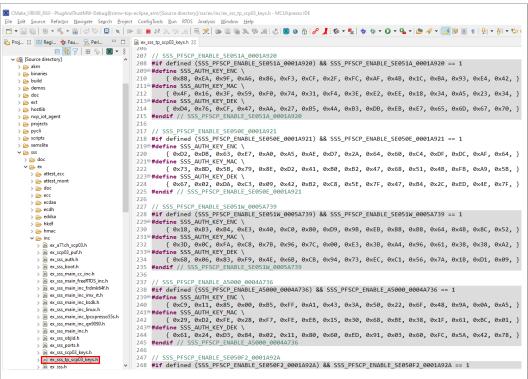


Figure 59. MCUXpresso - Initial Platform SCP keys are defined in ex\_sss\_tp\_ scp03\_keys.h header file

The fsl\_sss\_ftr.h.in file includes options to select one of the predefined initial Platform SCP keys in the ex\_sss\_tp\_scp03\_keys.h header file. This file is located in: .\simw-top\sss\inc.

Select the desired value of the compilation option by setting exclusively the corresponding C-preprocessor define  $SSS\_PFSCP\_ENABLE\_xx$  to 1 (enable). All other values for the same option (represented by C-preprocessor defines  $SSS\_PFSCP\_ENABLE\_xx$ ) must be set to 0.

### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

▼ 🗐 🕼   🗞 ▼ 🗞 ▼ 🔜   🖓 🐤   📮   🍬	D= 11 = M 2. O 12   7, 7   10 10 = 2 7 10 12 2 18 0 18 17 18 17 18 18 18 18 18 18 18 18 18 18 18 18 18
roj 🔀 👭 Regi 株 Fau 🧏 Peri 🖓	🗆 📄 fsl_sss_ftr.h.in 😒
E 🔄 🏹 🖶 🗞 🕅 🗸	<pre>§ 574/* Import Export Key is enabled */</pre>
> 🔁 scripts	575 #cmakedefine01 SSS_HAVE_IMPORT
> 🗁 semslite	576
✓ → SSS	577 /* With NXP NFC Reader Library */
> 🗁 doc	578 #cmakedefine01 SSS_HAVE_NXPNFCRDLIB
> 🗁 ex	579
🗸 🗁 inc	580 #define SSS HAVE A71XX \
> h fsl_sscp_a71ch.h	581 (SSS HAVE APPLET A71CH   SSS HAVE APPLET A71CH SIM)
> 庙 fsl_sscp_a71cl.h	582
> h fsl_sscp_commands.h	583 #define SSS_HAVE_SSCP (SSS_HAVE_A71XX)
> h fsl_sscp_mu.h	584
> h fsl_sscp.h	585/* For backwards compatibility */
> h fsl_sss_api_ver.h	586 #define SSS_HAVE_TESTCOUNTERPART (SSSFTR_SW_TESTCOUNTERPART)
> h fsl_sss_api.h	587
> h fsl_sss_base_apis.hpp > h fsl_sss_config.h	588/* ======= Miscellaneous values : END ===================================
in fsl_sss_config.n in fsl_sss_ftr_default.h	589
> h fsl_sss_keyid_map.h	590/* Enable one of these
in fsl_sss_kcytd_map.in	591 * If none is selected, default config would be used
in fsl_sss_lpc55s_types.h	592 */
h fsl_sss_mbedtls_apis.h	593#define SSS PFSCP ENABLE SE050A1 0
h fsl_sss_mbedtls_apis.hpp	594 #define SSS_PFSCP_ENABLE_SE050A2 0
in fsl_sss_mbedtls_types.h	595#define SSS_PFSCP_ENABLE_SE050B1 0
> h fsl_sss_openssl_apis.h	596 #define SSS PFSCP ENABLE SE050B2 0
h fsl_sss_openssl_apis.hpp	597 #define SSS_PFSCP_ENABLE_SE050C1 0
> in fsl_sss_openssl_types.h	598 #define SSS PFSCP ENABLE SE050C2 0
> h fsl_sss_policy.h	599 #define SSS_PFSCP_ENABLE_SE050_DEVKIT_0
h fsl_sss_se05x_apis.h	600 #define SSS PFSCP ENABLE SE051A2 0
> In fsl_sss_se05x_apis.hpp	601#define SSS PFSCP ENABLE SE051C2 0
> h fsl_sss_se05x_policy.h	602#define SSS PFSCP ENABLE SE050F2 0
h fsl_sss_se05x_scp03.h	603 #define SSS PFSCP ENABLE SE051C 0005A8FA 0
In fsl_sss_se05x_types.h	604#define SSS_PFSCP_ENABLE_SE051A_0001A920 0
In fsl_sss_sscp_apis.hpp	605 #define SSS PFSCP ENABLE SE050E 0001A921 1
in fsl_sss_sscp_config.h	606 #define SSS PFSCP ENABLE SE051W 0005A739 0
> h fsl_sss_sscp.h > h fsl_sss_user_apis.h	607#define SSS PFSCP ENABLE A5000 0004A736 0
> h fsl_sss_user_apis.h > h fsl_sss_user_types.h	608#define SSS_PFSCP_ENABLE_SE050F2_0001A92A 0
in tsi_sss_user_types.n in fsi_sss_util_asn1_der.h	609#define SSS PFSCP ENABLE OTHER 0
in tsi_sss_util_asn1_der.n in fsl_sss_util_rsa_sign_utils.h	610

Figure 60. Select the actual Platform SCP keys in the fsl\_sss\_ftr.h.in CMake input file

The Plug & Trust Middleware uses a feature file to select/detect used/enabled features within the middleware stack. The file  $fsl\_sss\_ftr.h$  is automatically generated into the used build directory. CMake is overwritting the  $fsl\_sss\_ftr.h$  file every time CMake is invoked. CMake is using the SCP key settings of the  $fsl\_sss\_ftr.h.in$  file as input to generate the the  $fsl\_sss\_ftr.h$  file. You do not have to manually edit the  $fsl\_sss\_ftr.h$  feature file. Selections from CMake edit cache automatically updates into the generated feature file.

**Note:** The Platform SCP key selection in the <code>fsl\_sss\_ftr.h.in</code> CMake input file is persistent.

The location of the generated fsl\_sss\_ftr.h feature header file is: .\simw-top build\simw-top-eclipse arm.

The following tables contains the the Platform SCP key header file define to be set to 1 (enable) for the different secure element and secure authenticator product variants.

[	Variant	OEF ID	Platform SCP key define to be set to '1'
	SE050E Dev. Board	A921	SSS_PFSCP_ENABLE_SE050E_0001A921
	OM-SE050ARD-E		

 Table 21. Platform SCP key define prefix for SE050E product variants...continued

Variant	OEF ID	Platform SCP key define to be set to '1'
SE050E2	A921	SSS_PFSCP_ENABLE_SE050E_0001A921

#### Table 22. Platform SCP key define prefix for SE050F product variants

Variant	OEF ID	Platform SCP key define to be set to '1'
SE050F Dev.Board OM-SE050ARD-F	A92A	SSS_PFSCP_ENABLE_SE050F2_0001A92A
SE050F2	A92A	SSS_PFSCP_ENABLE_SE050F2_0001A92A

### Table 23. Platform SCP key define prefix for SE050 Previous Generation product variants

Variant	OEF ID	Platform SCP key define to be set to '1'
SE050A1	A204	SSS_PFSCP_ENABLE_SE050A1
SE050A2	A205	SSS_PFSCP_ENABLE_SE050A2
SE050B1	A202	SSS_PFSCP_ENABLE_SE050B1
SE050B2	A203	SSS_PFSCP_ENABLE_SE050B2
SE050C1	A200	SSS_PFSCP_ENABLE_SE050C1
SE050C2	A201	SSS_PFSCP_ENABLE_SE050C2
SE050 Dev Board	A1F4	SSS_PFSCP_ENABLE_SE050_DEVKIT
OM-SE050ARD		
SE050F2	A77E <sup>[1]</sup>	SSS_PFSCP_ENABLE_SE050F2

[1] All SE050F2 with variant A77E have date code in year 2021. All the SE050F2 with date code in the year 2022 have the variant identifier A92A.

#### Table 24. Platform SCP key define prefix for SE051 product variants

Variant	OEF ID	Platform SCP key define to be set to '1'
SE051A2	A920	SSS_PFSCP_ENABLE_SE051A_0001A920
SE051C2	A8FA	SSS_PFSCP_ENABLE_SE051C_0005A8FA
SE051W2	A739	SSS_PFSCP_ENABLE_SE051W_0005A739
SE051A2	A565	SSS_PFSCP_ENABLE_SE051A2
SE051C2	A564	SSS_PFSCP_ENABLE_SE051C2

Table 25.	Platform SCP	key define prefix	for A5000 product variants
-----------	--------------	-------------------	----------------------------

Variant	OEF ID	Platform SCP key define to be set to '1'
A5000 Dev. Board OM-A5000ARD	A736	SSS_PFSCP_ENABLE_A5000_0004A736
A5000	A736	SSS_PFSCP_ENABLE_A5000_0004A736

In the next step it is necessary to enable Platfrom SCP in the Plug & Trust middleware. <u>Section 6.5</u> describes how to enable Platform SCP in the CMake-based build system.

### 6.5 How to enable Platform SCP in the CMake-based build system

To enable Platform SCP is required to rebuild the SDK with the following CMake options:

• Select SCP03\_SSS for the CMake option PTMW\_SCP.

• Select PlatfSCP03 for the CMake option PTMW\_SE05X\_Auth.

The following images show the configuration for the SE050E development board OM-SE05ARD-E.

ere is the source code: C:/se05x mw v04.02.0	Browse Source
	, browse gource
<custom></custom>	·
ere to build the binaries: C:/se05x_mw_v04.02.0	/ Browse Build.
rch:	Grouped Advanced 🕂 Add Entry 🗱 Remove Entry Environment
ame	Value
Ungrouped Entries CMAKE	
PTMW	
PTMW_A71CH_AUTH	None Croso F
PTMW_Applet PTMW_FIPS	SE050_E None
PTMW_Host	Name of Street Stre
PTMW_HostCrypto	MBEDTLS
PTMW_Log PTMW_RTOS	Default Default
PTMW SBL	None
PTMW_SCP	SCP03_SSS
PTMW_SE05X_Auth PTMW_SE05X_Ver	PlatfSCP03 07_02
PTMW_SMCOM	T1ol2C
PTMW_mbedTLS_ALT	None
ProcessorCount SSSFTR	
SSSFTR_SE05X_AES	
SSSFTR_SE05X_AuthECKey	
SSSFTR_SE05X_AuthSession	
SSSFTR_SE05X_CREATE_DELETE_CRYPTOO SSSFTR_SE05X_ECC	SJ ⊻
SSSFTR_SE05X_KEY_GET	
SSSFTR_SE05X_KEY_SET	
SSSFTR_SE05X_RSA SSSFTR_SW_AES	
SSSFTR_SW_ECC	
SSSFTR_SW_KEY_GET	
SSSFTR_SW_KEY_SET	
SSSFTR_SW_RSA SSSFTR_SW_TESTCOUNTERPART	
Press Configure to update and displ	y new values in red, then press Generate to generate selected build files.
Configure Generate Open Project Cur	ent Generator: Edipse CDT4 - Unix Makefiles

### 7 Appendix A: Install MCUXpresso IDE

MCUXpresso is a free-of-charge, code size unlimited, easy-to-use IDE for Kinetis and LPC MCUs, and i.MX RT crossover processors. To install it, do the following:

1. Go to <u>MCUXpresso</u> and click the download button as indicated in <u>Figure 62</u>:

		DOWNLOADS	DEVEL	OPMENT TOOLS	TRAINING & SUPPORT
o To view & Features orted Devices im Requirements	Eclipse-based di MCUs based on general purpose MCUs. The MCU compiling, and d MCU-specific de multicore debug	♥ So IDE brings developers an evelopment environment for Arm® Cortex®-M cores, incline crossover and wireless - en JXpresso IDE offers advance ebugging features with the a subugging views, code trace as ing. and integrated configure IDE debug connections at ID	NXP® uding its labled ed editing, addition of und profiling, ration tools.	<ul> <li>Advanced editing, coloring, MCU-spe and profiling</li> <li>Use built-in SDK s built packages ma</li> </ul>	unlimited code size, easy-to-use compiling and editing with synta coffic debugging views, code trac election tool, or drag and drop pr de with SDK Builder / 20.04.2 LTS, Github project vort

2. You will be asked to sign-in with your account at the NXP website. If you do not have an account, click on *Register Now* as shown in <u>Figure 63</u>:

NP	PRODUCTS	APPLICATIONS	DESIGN	SUPPORT	COMPANY
Home / Sign In	or Register				
			Sign	In	
			Email A	ddress or NXI	P Company ID
			Passwo	rd	
			SIGN I	N	
				Forgot your	password?
			Do	on't have an A	ccount? Register Now
<b>E</b> :	Deviator				
Figure 63.	Register yo	ur NXP accour	It		

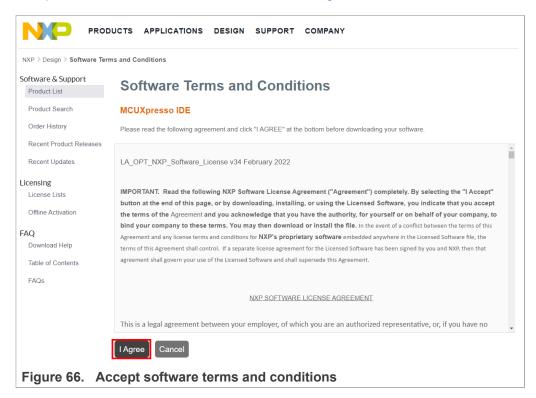
3. If you already have an account, you can directly type your (1) email address, (2) password and (3) click sign-in button as shown in Figure 64:

NP	PRODUCTS	APPLICATIONS	DESIGN	SUPPORT	COMPANY
Home / Sign In o	or Register				
				1 Passw 2 SIGN	Address or NXP Company ID
Figure 64.	Sign-in i	n NXP websi	te		

4. Click on MCUXpresso IDE as shown in Figure 65:

NXP > Design > Product Info	rmation : MCUXpresso IDE	
Software & Support Product List	Product Information	
Product Search	MCUXpresso IDE	
Order History	Select a version. To access older versions, click on the " Previous " tab	
Recent Product Releases Recent Updates	Current Previous	
Licensing	Version         Description           11.5.0         MCUXpresso IDE	

5. Accept software terms and conditions as shown in Figure 66:



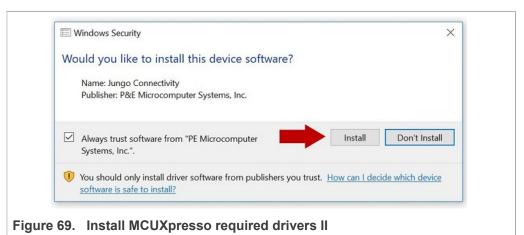
6. Select your MCUXpresso product version and click on the corresponding *File Name* to start the download as shown in Figure 67:

NXP > Design > MCUXpress	IDE MCUXpresso IDE : Files		
oftware & Support	Product Download		
Product List	Troduct Download		
Product Search	MCUXpresso IDE		
Order History	Files License Keys Notes		O Download He
Recent Product Releases	,		
Recent Updates	Show All Files		3 Fil
	+ File Description	File Size  File Name	
Icensing License Lists	+ MCUXpressoIDE_11.5.0 - Linux	928.6 MB mcuxpressoide-11.5.0_7232.x86_64.deb.t	bin
License Lists	+ MCUXpressoIDE_11.5.0 - MAC	889.6 MB & MCUXpressolDE_11.5.0_7232.pkg	
Offline Activation	+ MCUXpressoIDE 11.5.0 - Windows	840.9 MB & MCUXpressoIDE 11.5.0 7232.exe	

7. Double click on the installer file and follow the setup wizard until MCUXpresso installation is completed. Please, make sure you allow the installation of the additional

drivers required by MCUXpresso during the installation process as shown in <u>Figure 68, Figure 69, Figure 70</u> and <u>Figure 71</u>:





# Windows Security Would you like to install this device software? Name: Freescale, P&E Micro (http://www.pemicro... Publisher: P&E Microcomputer Systems, Inc. Always trust software from "PE Microcomputer Systems, Inc.". You should only install driver software from publishers you trust. How can I decide which device software is safe to install?





### 8 Appendix B: Install CMake

CMake is an open-source, cross-platform family of tools that helps you build C/C++ projects on multiple platforms using a compiler-independent method. It has minimal dependencies, requiring only a C++ compiler on its own build system. SE05x middleware leverages on CMake to generate native makefiles and workspaces that can be used in the compiler environment of your choice.

To install CMake:

1. Go to CMake downloads page: https://cmake.org/download/

2. Scroll down and select your binary distribution. For this guide, the binary distribution is Windows as shown in Figure 72:

CMake	About v Services v Resources v Download
atest Release (3.22.3)	
ne release was packaged with CPack which is included as part of the release. The .sh files are illow the directions. The OS-machine.tar.gz files are gziped tar files of the install tree. The OS- e distributions can be untared in any directory. They are prefixed by the version of CMake. Fo ux-x86_64. This prefix can be removed as long as the share, bin, man and doc directories are npack them with zip or tar and follow the instructions in README.rst at the top of the source	machine.tar.7_files are compressed tar files of the install tree. The ta or example, the linux-x86_64 tar file is all under the directory cmake- e moved relative to each other. To build the source distributions,
burce distributions:	
Platform	Files
Unix/Linux Source (has \n line feeds)	cmake-3.22.3.tar.gz
Windows Source (has \r\n line feeds)	cmake-3.22.3.zip
nary distributions:	
Platform	Files
Windows x64 Installer: Installer tool has changed. Uninstall CMake 3.4 or lower first!	cmake-3.22.3-windows-x86_64.msi
Windows x64 ZIP	cmake-3.22.3-windows-x86_64.zip
Windows i386 Installer: Installer tool has changed. Uninstall CMake 3.4 or lower first!	cmake-3.22.3-windows-i386.msi
Windows i386 ZIP	cmake-3.22.3-windows-i386.zip
macOS 10.13 or later	cmake-3.22.3-macos-universal.dmg
	cmake-3.22.3-macos-universal.tar.gz
macOS 10.10 or later	cmake-3.22.3-macos10.10-universal.dmg
	cmake-3.22.3-macos10.10-universal.tar.gz
Linux x86_64	cmake-3.22.3-linux-x86_64.sh
	cmake-3.22.3-linux-x86_64.tar.gz
Linux aarch64	cmake-3.22.3-linux-aarch64.sh

AN12450

3. Double click on the downloaded installer file. Windows Defender SmartScreen might pop-up the wizard shown in Figure 73:

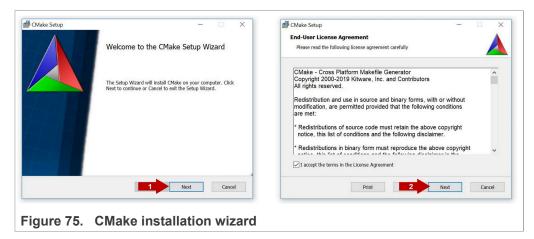
in

4. If this is your case: Click (1) on *More info* and then (2) click on *Run anyway* as shown in Figure 74:

Vindows Defender SmartScreen prevented an unrecognized app from tarting. Running this app might put your PC at risk. <u>tore info</u>	Windows Defender SmartScreen prevented an unrecognized app from starting. Running this app might put your PC at risk. App: cmake-3:14.0-rc4-win64-x64.msi Publisher: Unknown publisher
Don't run	2 Run anyway Don't run

AN12450 Application note

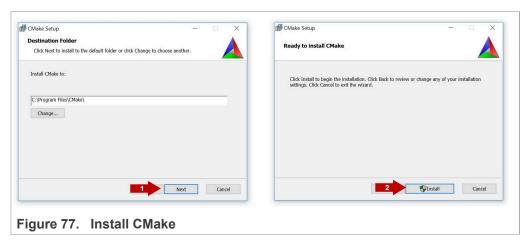
 The CMake installation wizard will open. Click (1) Next and (2) accept the End-User License Agreement as shown in Figure 75:



 As part of the CMake setup, (1) Add Cmake to the system PATH for all users and (2) click Next as shown in Figure 76:

Install Options Choose options for installing CMake By default CMake does not add its directory to the system PATH. O Do not add CMake to the system PATH O Add CMake to the system PATH for all users
By default CMake does not add its directory to the system PATH.
○ Do not add CMake to the system PATH
Add CMake to the system DATH for all users
Add Chake to the system PATTION of users
$\bigcirc$ Add CMake to the system PATH for the current user
Create CMake Desktop Icon
2 Next Cancel

 Select a destination folder, (1) click *Next* and then (2) click *Install* as shown in <u>Figure 77</u>:



8. Wait a few seconds until the installation is completed and click *Finish* as shown in Figure 78:

CMake Setup -
Completed the CMake Setup Wizard
Click the Finish button to exit the Setup Wizard.
Finish Cano

### 9 Appendix C: Install Python

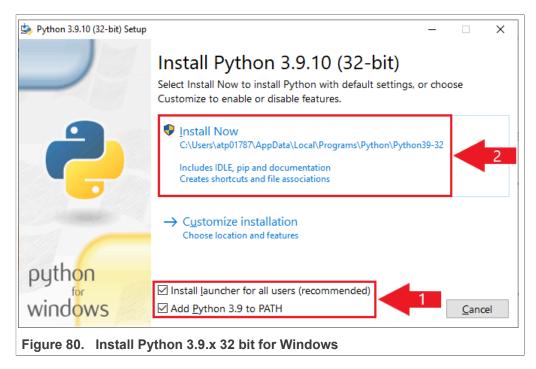
This section explains how to install Python  $\ge$  3.7.x and  $\le$  3.9.x 32-bit version, but the same procedure can be applied for more recent versions. Follow these steps to install Python in your local machine:

1. Go to <u>https://www.python.org/downloads</u> and download **Python** ≥ **3.7.x** and ≤ **3.9** 32bit version. Make sure you download the Python 32 bit version.

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		1440acb71471e2394befdb30b1a958d1	25800844	SIG
XZ compressed source tarball	Source release		e754c4b2276750fd5b4785a1b443683a	19154136	SIG
macOS 64-bit Intel-only installer	macOS	for macOS 10.9 and later, deprecated	2714cb9e6241cf7e2f9022714a55d27a	30395760	SIG
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	c2393ab11a423d817501b8566ab5da9f	38217233	SIG
Windows embeddable package (32-bit)	Windows		c1d2af96d9f3564f57f35cfc3c1006eb	7671509	SIG
Windows embeddable package (64-bit)	Windows		b8e8bfba8e56edcd654d15e3bdc2e29a	8509821	SIG
Windows help file	Windows		784020441c1a25289483d3d8771a8215	9284044	SIG
Windows installer (32-bit)	Windows		457d648dc8a71b6bc32da30a7805c55b	27767040	SIG
Windows installer (64-bit)	Windows	Recommended	747ac35ae667f4ec1ee3b001e9b7dbc6	28909456	SIG

Figure 79. Download Python 3.9.x 32 bit version

2. Double click on the downloaded installer file. Select the "*Install launcher for all users*" and "*Add Python 3.7 to Path*" options and click *Install Now* as indicated in Figure 80:



3. Wait a few seconds until the installation is completed as indicated in Figure 81

Python 3.9.10 (32-bit) Setup	>	< > Python 3.9.10 (32-bit) Setup	×
	Setup Progress		Setup was successful
	Installing:		New to Python? Start with the <u>online tutorial</u> and <u>documentation</u> . At your terminal, type "py" to launch Python, or search for Python in your Start menu.
	Python 3.9.10 Standard Library (32-bit)		See <u>what's new</u> in this release, or find more info about <u>using</u> Python on Windows
			Disable path length limit Changes your machine configuration to allow programs, including Python, to bypass the 260 character *MAX_PAIH* limitation.
python		python	
windows	Cancel	] wind <mark>ows</mark>	<u>C</u> lose
Figure 81.	Python 3.9.x 32 bit installa	tion comple	oted

#### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

### 10 Legal information

### **10.1 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.

### 10.2 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 products products products 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 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.

### 10.3 Trademarks

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

EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

### **Tables**

Tab. 1.	EdgeLock SE05x development boards
Tab. 2.	i.MX RT1060 evaluation kit details4
Tab. 3.	i.MX RT1170 evaluation kit details5
Tab. 4.	Feature file fsl_sss_ftr.h settings for
	SE050E product variants 19
Tab. 5.	Feature file fsl_sss_ftr.h settings for
	SE050F product variants20
Tab. 6.	Feature file fsl_sss_ftr.h settings for SE050
	Previous Generation product variants
Tab. 7.	Feature file fsl_sss_ftr.h settings for SE051
	product variants21
Tab. 8.	Feature file fsl_sss_ftr.h settings for A5000
	product variants22
Tab. 9.	CMake Settings for SE050E product
	variants40
Tab. 10.	CMake Settings for SE050F product
	variants41
Tab. 11.	CMake Settings for SE050 Previous
	Generation product variants41
Tab. 12.	CMake Settings for SE051 product variants41
Tab. 13.	CMake Settings for A5000 product variants 42
Tab. 14.	Static SCP03 keys 45

Tab. 15.	SCP03 session keys	45
Tab. 16.	Platform SCP key define prefix for SE050E	
	product variants	48
Tab. 17.	Platform SCP key define prefix for SE050F	
	product variants	48
Tab. 18.	Platform SCP key define prefix for SE050	
	Previous Generation product variants	48
Tab. 19.	Platform SCP key define prefix for SE051	
	product variants	49
Tab. 20.	Platform SCP key define prefix for A5000	
	product variants	49
Tab. 21.	Platform SCP key define prefix for SE050E	
	product variants	52
Tab. 22.	Platform SCP key define prefix for SE050F	
	product variants	53
Tab. 23.	Platform SCP key define prefix for SE050	
	Previous Generation product variants	53
Tab. 24.	Platform SCP key define prefix for SE051	
	product variants	53
Tab. 25.	Platform SCP key define prefix for A5000	
	product variants	53

### **Figures**

Fig. 1. Fig. 2.	Jumper configuration for OM-SE05xARD5 Plug OM-SE05xARD and i.MX RT1060
· ·g	boards
Fig. 3.	Plug OM-SE05xARD and i.MX RT1170 boards
Fig. 4.	OM-SE05xARD mounted in i.MX RT1060
Fig. 5.	board7 OM-SE05xARD mounted in i.MX RT1170
Fig. 6.	board
Fig. 7.	board
	board
Fig. 8.	Identify the serial port
Fig. 9.	Import the board SDK into MCUXpresso environment10
Fig. 10.	Imported SDKs
Fig. 11.	Import projects from SDK11
Fig. 12.	SDK import wizard
Fig. 13.	Select projects to import
Fig. 14.	Imported projects in MCUXpresso
	workspace14
Fig. 15.	Connect i.MX RT1060 board to the laptop 15
Fig. 16.	Connect i.MX RT1170 board to the laptop15
Fig. 17.	TeraTerm setup16
Fig. 18.	Build projects in MCUXpresso workspace16
Fig. 19.	Debug projects in MCUXpresso workspace 17
Fig. 20.	Run projects in MCUXpresso workspace 17
Fig. 21.	TeraTerm logs - se05x_Minimal project
	example18
AN12450	All information provided in this do

E05xARD5 RT1060	Fig. 22.	Feature file fsl_sss_ftr.h example: Assign the value SE05X E to the CMake option	
6		PTWM_Applet	19
RT1170	Fig. 23.	Feature file fsl_sss_ftr.h - PTMW_Applet	
6	Fig. 24.	Feature file fsl_sss_ftr.h - Option PTMW_	
X RT1060		FIPS	23
7	Fig. 25.	Feature file fsl_sss_ftr.h - Option PTMW_	
X RT1170		SE05x_Ver	24
7	Fig. 26.	Feature file fsl_sss_ftr.h - Option PTMW_	
RT1060 EVK		AUTH - Plain communication	25
8	Fig. 27.	Feature file fsl_sss_ftr.h - Option PTMW_	
RT1170 EVK		SCP - Plain communication	25
8	Fig. 28.	Feature file fsl_sss_ftr.h - Option SSSFTR_	
9		SE05X_RSA	26
JXpresso	Fig. 29.	Create se05x_middleware folder	27
10	Fig. 30.	Unzip se05x middleware	27
10	Fig. 31.	Generate Plug & Trust middleware project	
11		examples	28
12	Fig. 32.	SE05x middleware project structure	28
13	Fig. 33.	Import a project	29
SO	Fig. 34.	Import a project (II)	29
14	Fig. 35.	Select Plug & Trust middleware build folder .	30
the laptop 15	Fig. 36.	Import Plug & Trust middleware	30
the laptop15	Fig. 37.	Plug & Trust middleware imported in	
		workspace	
workspace16	Fig. 38.	Import a project	32
workspace 17	Fig. 39.	Import a project (II)	32
orkspace 17	Fig. 40.	Select project folder for your board	33
project	Fig. 41.	Import project for your board in the	
		workspace	33
All information provided in this doc	ument is subject to lea	gal disclaimers. © NXP B.V. 2022. All rights r	eserved

### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

Fig. 42.	Board project imported in workspace
Fig. 43.	List Plug & Trust middleware examples
Fig. 44.	Configure CMake options of Plug & Trust
	middleware examples
Fig. 45.	Connect i.MX RT1060 board to the laptop 37
Fig. 46.	Connect i.MX RT1170 board to the laptop37
Fig. 47.	Configure TeraTerm
Fig. 48.	Debug build Plug & Trust middleware
	se05x_minimal project example
Fig. 49.	Debug se05x_minimal project example
Fig. 50.	Resume se05x_minimal project example39
Fig. 51.	TeraTerm logs - se05x_minimal project
	example40
Fig. 52.	SE050E CMake Settings - Plain
	communication43
Fig. 53.	SPC03 mutual authentication – principle46
Fig. 54.	SPC03 Encryption and MACing principle46
Fig. 55.	MCUXpresso SDK - Initial Platform SCP
	keys are defined in the ex_sss_tp_scp03_
	keys.h header file47
Fig. 56.	Select the actual Platform SCP keys in the
	fsl_sss_ftr.h header file 48
Fig. 57.	Feature file fsl_sss_ftr.h - Option PTMW_
	SE05X_Auth - PlatformSCP enabled
Fig. 58.	Feature file fsl_sss_ftr.h - Option PTMW_
	SCP - PlatformSCP enabled50

Fig. 59.	MCUXpresso - Initial Platform SCP keys	
	are defined in ex_sss_tp_scp03_keys.h	
	header file	51
Fig. 60.	Select the actual Platform SCP keys in the	
	fsl_sss_ftr.h.in CMake input file	52
Fig. 61.	SE050E CMake Settings - PlatformSCP	
	enabled	
Fig. 62.	Go to MCUXpresso website	55
Fig. 63.	Register your NXP account	55
Fig. 64.	Sign-in in NXP website	56
Fig. 65.	Select MCUXpresso	56
Fig. 66.	Accept software terms and conditions	57
Fig. 67.	Download MCUXpresso	57
Fig. 68.	Install MCUXpresso required drivers I	58
Fig. 69.	Install MCUXpresso required drivers II	58
Fig. 70.	Install MCUXpresso required drivers III	58
Fig. 71.	Install MCUXpresso required drivers IV	59
Fig. 72.	Download CMake	60
Fig. 73.	Execute CMake installer	61
Fig. 74.	Run the CMake installer (II)	61
Fig. 75.	CMake installation wizard	62
Fig. 76.	Add CMake path	62
Fig. 77.	Install CMake	63
Fig. 78.	Complete CMake installation	63
Fig. 79.	Download Python 3.9.x 32 bit version	64
Fig. 80.	Install Python 3.9.x 32 bit for Windows	64
Fig. 81.	Python 3.9.x 32 bit installation completed	65

### EdgeLock SE05x Quick start guide with i.MX RT1060 and i.MX RT1170

### Contents

1	How to use this document	
2	Required hardware	3
3	Boards setup	5
4	Import project examples from board SDK	9
4.1	Prerequisites	
4.2	Download the board SDK	
4.3	Install the board SDK	
4.4	Import a project example in MCUXpresso	
4.5	Build, run and debug project example	14
4.6	Product specific build settings	18
4.6.1	Example: SE050E build settings	22
5	Import project examples from CMake-	
	based build system	26
5.1	Prerequisites	
5.2	Download the Plug & Trust middleware	26
5.3	Build the Plug & Trust middleware project	
	examples	28
5.4	Import PlugAndTrustMW project example in	
	MCUXpresso workspace	28
5.5	Import board project example in	
	MCUXpresso workspace	31
5.6	Execute Plug & Trust middleware examples	34
5.6.1	List the Plug & Trust middleware examples	34
5.6.2	Edit Plug & Trust middleware example	
	CMake options	35
5.6.3	Build and run a Plug & Trust middleware	
	project example	
5.7	Product specific CMake build settings	
5.7.1	Example: SE050E CMake build settings	42
6	Binding EdgeLock SE05x to a host using	
	Platform SCP	43
6.1	Introduction to the Global Platform Secure	
	Channel Protocol 03 (SCP03)	44
6.2	How to configure the Platform SCP keys in	
	the i.MX RT1060 MCUXpresso SDK	46
6.3	How to enable Platform SCP in the i.MX	
	RT1060 MCUXpresso SDK	49
6.4	How to configure the Platform SCP keys in	
	CMake-based build system	50
6.5	How to enable Platform SCP in the CMake-	
-	based build system	
7	Appendix A: Install MCUXpresso IDE	
8	Appendix B: Install CMake	
9	Appendix C: Install Python	
10	Legal information	66

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. 2022.

All rights reserved.

For more information, please visit: http://www.nxp.com For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 4 August 2022 Document identifier: AN12450 Document number: 546942