AN14559

Migration guide from EdgeLock A5000 to EdgeLock A30

Rev. 1.2 — 2 September 2025

Application note

Document information

Information	Content
Keywords	EdgeLock A30 secure authenticator, NX Middleware
Abstract	This document describes the considerations for migrating an existing design based on EdgeLock A5000 to EdgeLock A30.

1 About EdgeLock A30 secure authenticator

EdgeLock A30 is a secure authentication IC for IoT platforms, electronic accessories and consumable devices such as home electronic devices, mobile accessories and medical supplies.

EdgeLock A30 supports on-chip ECC key generation to make sure that private keys are never exposed outside the IC. It performs cryptographic operations for security critical communication and control functions. EdgeLock A30 is Common Criteria EAL 6+ security certified with AVA_VAN.5 on product level and supports a generic Crypto API providing AES, ECDSA, ECDH, SHA, HMAC and HKDF cryptographic functionality.

- Asymmetric cryptography features support 256-bit ECC over the NIST P-256 and brainpool P256r1 curves.
- Symmetric cryptography features support both AES-128 and AES-256.
- PKI-based mutual authentication based on the Sigma-I protocol.
- Symmetric three-pass Mutual Authentication protocol.
- Secure messaging channel using either AES-128 or AES-256 session encryption/decryption and MAC.
- Number of supported persistent key entries:
  - Up to five persistent EC key entries for Sigma-I and crypto primitives.
  - Up to five persistent AES key entriesfor mutual authentication, Secure Dynamic Messaging and key update. These five AES key entries are not consuming any user memory.
  - Up to eight persistent AES key entriesfor crypto operations.

The Common Criteria security certification ensures that the IC security measures and protection mechanisms have been evaluated against sophisticated noninvasive and invasive attack scenarios.

- A30 supports an I^{2}C contact interface and has two additional GPIOs.
- A30 supports a low-power design, and consumes only 5 µA at Deep-Power-Down mode when an external VDD is supplied.

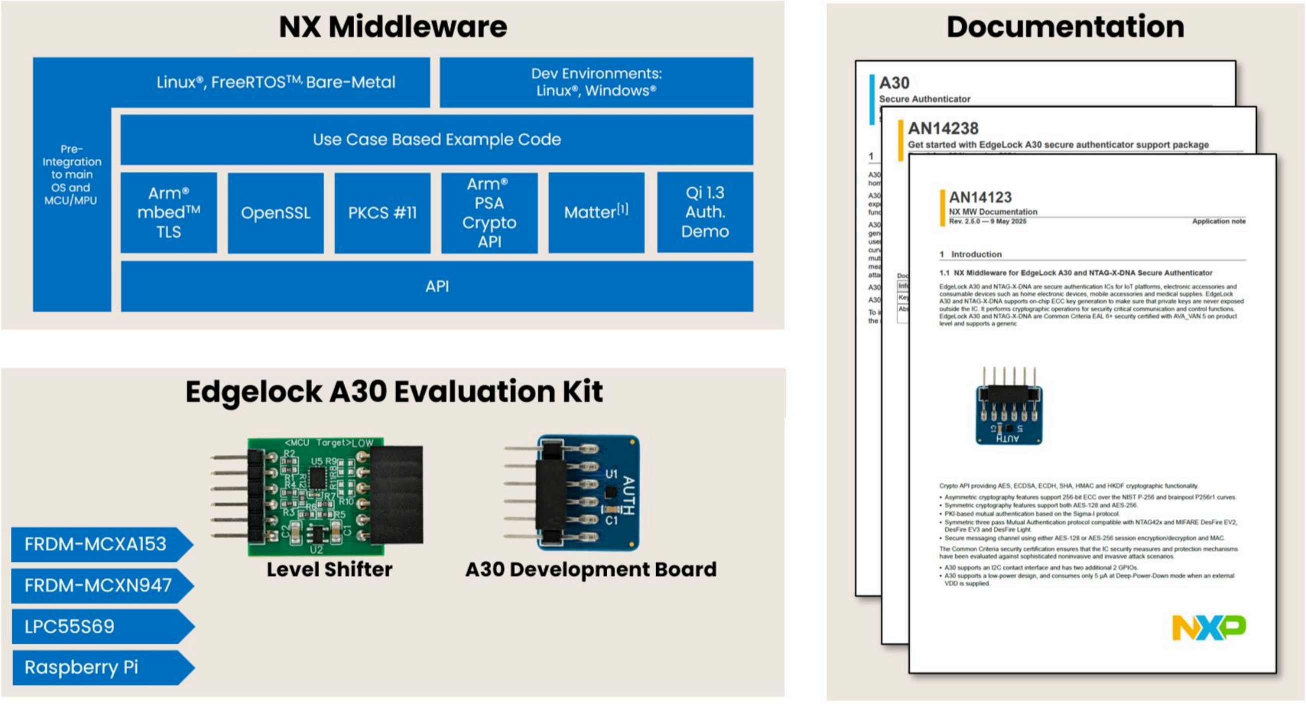

NX Middleware
Linux®, FreeRTOS™ Bare-Metal
Dev Environments: Linux®, Windows®
Pre-Integration to main OS and MCU/MPU
Use Case Based Example Code
Arm® mbed™ TLS
OpenSSL
PKCS #11
Arm® PSA Crypto API
Matter[1]
Qi 1.3 Auth. Demo
API
Edgelock A30 Evaluation Kit
FRDM-MCXA153
FRDM-MCXN947
LPC55S69
Raspberry Pi
Level Shifter
A30 Development Board
Documentation
A30
Secure Authenticator
AN14238
Get started with EdgeLock A30 secure authenticator support package
AN14123
NX MW Documentation
Rev. 2.5.0 — 9 May 2025
Application note
1 Introduction
1.1 NX Middleware for EdgeLock A30 and NTAG-X-DNA Secure Authenticator

Figure 1. EdgeLock A30 support package overview

Delivered as a ready-to-use solution, the EdgeLock A30 includes a complete product support package that simplifies design-in and reduces time to market. The EdgeLock A30 support package offers:

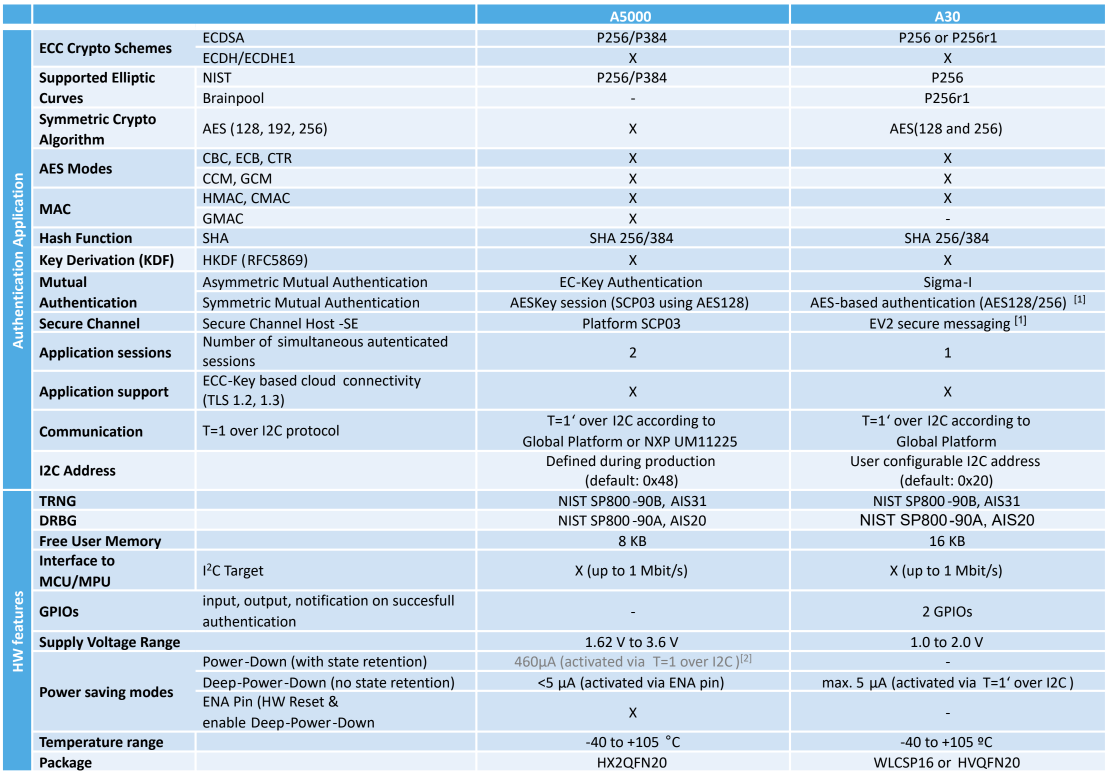
- EdgeLock A30 development kit
- NX Middleware
  - Software enablement for MCUs and MPUs.
  - Integration with the most common cryptographic libraries like OpenSSL, Mbed TLS and PKCS #11.
  - Multi-platform software enablement targeting freeRTOS and Linux as well as Windows as evaluation platform.
  - Sample code for major IoT and secure authentication use cases.
- Documentation

2 Migrating from EdgeLock EdgeLock A5000 to EdgeLock A30

This document describes the considerations for migrating an existing design based on EdgeLock A5000 to EdgeLock A30 solution. It is organized in the following sections:

- Section 2.1 "Hardware integration considerations"
- Section 2.2 "I2C interface and communicaton protocol considerations"
- Section 2.3 "Authentication application considerations"
- Section 2.4 "Middleware considerations"

Figure 2 shows the high-level comparison between EdgeLock A5000 and EdgeLock A30



			A5000	A30
Authentication Application	ECC Crypto Schemes	ECDSA	P256/P384	P256 or P256r1
		ECDH/ECDHE1	X	X
	Supported Elliptic Curves	NIST	P256/P384	P256
		Brainpool	-	P256r1
	Symmetric Crypto Algorithm	AES (128, 192, 256)	X	AES(128 and 256)
	AES Modes	CBC, ECB, CTR	X	X
		CCM, GCM	X	X
	MAC	HMAC, CMAC	X	X
		GMAC	X	-
	Hash Function	SHA	SHA 256/384	SHA 256/384
	Key Derivation (KDF)	HKDF (RFC5869)	X	X
	Mutual Authentication	Asymmetric Mutual Authentication	EC-Key Authentication	Sigma-I
		Symmetric Mutual Authentication	AESKey session (SCP03 using AES128)	AES-based authentication (AES128/256) [1]
	Secure Channel	Secure Channel Host -SE	Platform SCP03	EV2 secure messaging [1]
	Application sessions	Number of simultaneous autenticated sessions	2	1
	Application support	ECC-Key based cloud connectivity (TLS 1.2, 1.3)	X	X
	Communication	T=1 over I2C protocol	T=1' over I2C according to Global Platform or NXP UM11225	T=1' over I2C according to Global Platform
	I2C Address		Defined during production (default: 0x48)	User configurable I2C address (default: 0x20)
HW features	TRNG		NIST SP800-90B, AIS31	NIST SP800-90B, AIS31
	DRBG		NIST SP800-90A, AIS20	NIST SP800-90A, AIS20
	Free User Memory		8 KB	16 KB
	Interface to MCU/MPU	I^2C Target	X (up to 1 Mbit/s)	X (up to 1 Mbit/s)
	GPIOs	input, output, notification on succesfull authentication	-	2 GPIOs
	Supply Voltage Range		1.62 V to 3.6 V	1.0 to 2.0 V
	Power saving modes	Power-Down (with state retention)	460µA (activated via T=1 over I2C)[2]	-
		Deep-Power-Down (no state retention)	<5 µA (activated via ENA pin)	max. 5 µA (activated via T=1' over I2C)
		ENA Pin (HW Reset & enable Deep-Power-Down	X	-
	Temperature range		-40 to +105 °C	-40 to +105 ºC
	Package		HX2QFN20	WLCSP16 or HVQFN20

Figure 2. High-level comparison between EdgeLock A5000 and EdgeLock A30

2.1 Hardware integration considerations

EdgeLock A30 secure authenticator is designed for battery-operated applications and for MCU/MPUs with a supply voltage of 1.8 V. Therefore, from a hardware perspective, EdgeLock A30 is neither pin-to-pin nor package compatible with EdgeLock A5000. This means, a new PCB design is required in case of migrating from an existing EdgeLock A5000 hardware design to EdgeLock A30. The following tables compare the types regarding HW design.

Table 1. Package comparison

	A5000	A30
Package	HX2QFN20	WLCSP16 or HVQFN20

Table 2. Pin comparison

Pin function	A5000	A30
Power supply	VCC, GND	VCC, GND
I^2C	SDA, SCL	SDA, SCL
Power-on reset, Deep Power Down	ENA, VIN, VOUT	-[1] [2]
GPIO	-	GPIO1, GPIO2[3]

[1] EdgeLock A30 supports to trigger a power-on reset via T=1‘ over I^2C protocol.

[2] EdgeLock A30 enables Deep Power Down via T=1‘ over I^2C protocol.

[3] EdgeLock A30 has two configurable GPIOs. The GPIOs can be configured as input, output, and notification on successful authentication.

Table 3. Supply Voltage comparison

	A5000	A30
Power supply voltage	1.62 V to 3.6 V	1 V to 2 V [1]

[1] Level shifters are required in case the MCU/MPU boards are designed for a supply voltage of 3.3 V/5 V.

2.1.1 Application circuit diagram with depp power down support

This chapter compares application schematics with deep power down support between EdgeLock A5000 and EdgeLock A30.

2.1.1.1 A5000 application circuit diagram

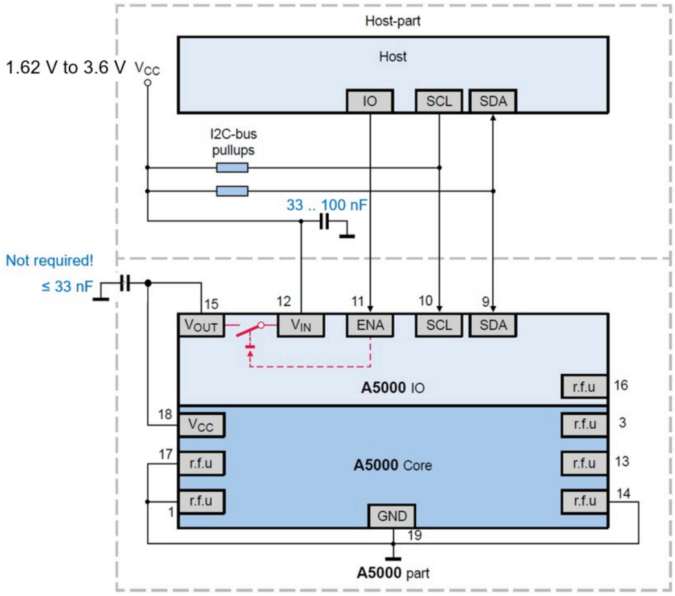


Figure 3. EdgeLock A5000 application circuit diagram with deep power down support

A5000 HW Reset sequence:

- Set ENA pin to logic zero level
- Wait 2 ms
- Set ENA pin to logic high level

A5000 Deep Power Down mode:

- ENA pin logic zero level .. Deep Power Down enabled
- ENA pin logic high level .. Deep Power Down disabled

2.1.1.2 A30 application circuit diagram

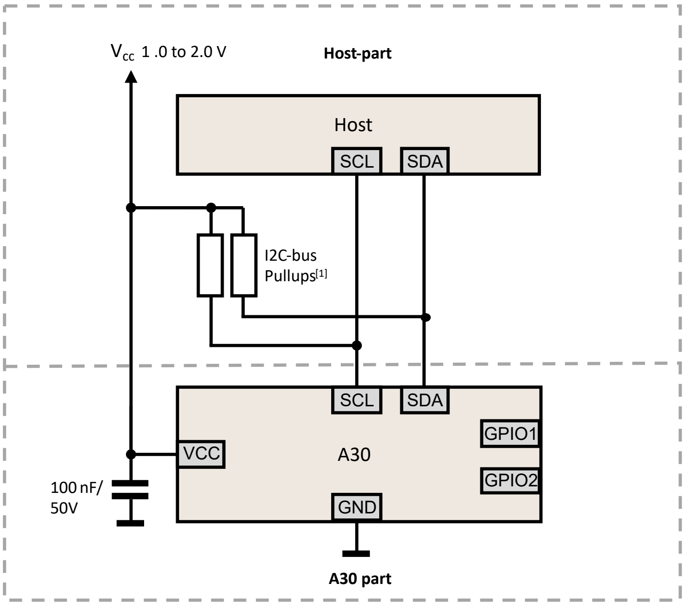


Figure 4. EdgeLock A30 application circuit diagram with deep power down support

Power-On-Reset:

- It is recommended that the host controller can perform a Power-On-Reset by controlling V_{CC}.
- It is possible to supply A30 via the MCU/MPU GPIO. The GPIO is able to deliver current up to 15 mA.

A30 triggers a Reset via T=1‘ over I^2C protocol:

- Proprietary NXP S-Blocks SE chip reset request/response

A30 enables Deep Power Down via T=1‘ over I^2C protocol:

- Proprietary NXP S-Block Deep Power Down request/response

2.2 I²C interface and communicaton protocol considerations

Table 4 compares the I²C target interface differences between EdgeLock A5000 and EdgeLock A30 in detail.

Table 4. I²C target interface comparison

I²C target	A5000	A30
Configurable I²C address	Defined during production	User configurable [1]
Default I²C address	0x48	0x20
Communication speed	up to 1 Mbit/s	up to 1 Mbit/s

[1] Configured via the SetConfiguration command.

Table 5 lists the differences with respect to the supported data link layer protocol.

Table 5. Communication protocol comparison

	A5000	A30

Data link layer protocol	T=1‘ over I²C according to NXP UM11225 [1] and
	T=1‘ over I²C according to Global Platform [2]
		T=1‘ over I²C according to Global Platform

NXP proprietary S-Blocks	Chip reset	Chip reset and Deep Power Down

[1] UM11225 NXP SE05x T=1 Over I²C Specification
[2] GlobalPlatform Technology - APDU Transport over SPI / I²C - Version 1.0. Version 1.0, January 2020

2.3 Authentication application considerations

2.3.1 EdgeLock A30 Authentication overview

A30 supports two protocols to establish a secure messaging channel:

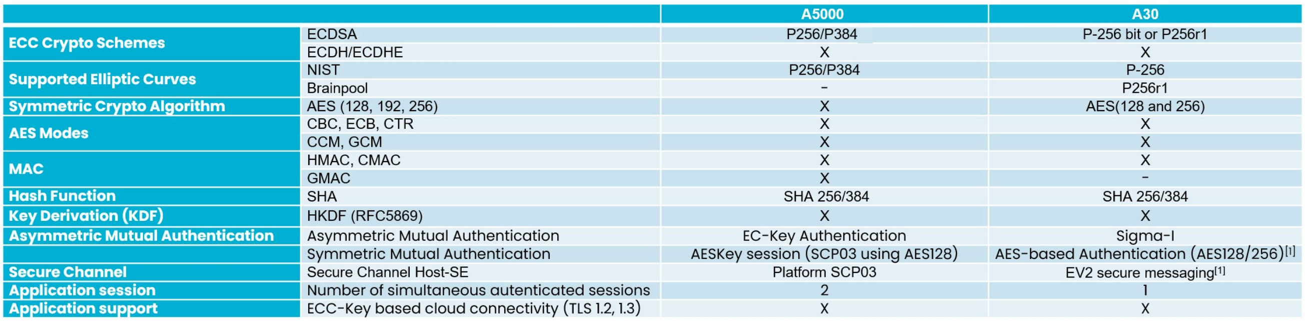
- **PKI-based Asymmetric Mutual Authentication**
  - It is based on **Sigma-I 256-bit ECC** (NIST P-256 or brainpoolP256r1).
  - Generates AES-128 or 256 session keys for Sigma-I mutual authentication message exchange.
  - Generates AES-128 or 256 session keys used for EV2 secure messaging channel.
- **AES-based Symmetric Mutual Authentication**
  - The same protocol as introduced in MIFARE DESFire EV2 products.
  - Based on AES-128 or AES-256.
  - Generates AES-128 or 256 session keys for EV2 secure messaging channel.
- Both mutual authentication methods initiate a MIFARE DESFire and NTAG42x compatible **EV2 secure messaging channel** (authenticated session).
  - AES-128 or AES-256 session encryption/decryption and MAC keys.
  - Access rights to subsequent commands and files granted after successful mutual authentication depending on configuration.
  - A30 supports one open secure messaging channel (authenticated session) at one time.

2.3.2 Crypto algorithms and protocols

From application point of view the following differences shall be considered before migrating from EdgeLock A5000 to EdgeLock A30:

- EdgeLock A30 supports almost the same crypto algorithms and schemes like A5000.
- EdgeLock A30 supports different authentication and secure messaging protocols.
  - EdgeLock A30 does not support more than one application sessions like A5000.
  - EdgeLock A30 grant access rights to subsequent commands and files after successful symmetric or asymmetric mutual authentication.
- EdgeLock A30 supports different object policies.
- EdgeLock A30 does not support secure attestation and Platform Configuration Register (PCR) like A5000.

The comparision from crypto algorithms and authentication protocol point of view between EdgeLock A5000 and EdgeLock A30 can be found in Figure 5.



		A5000	A30
ECC Crypto Schemes	ECDSA	P256/P384	P-256 bit or P256r1
	ECDH/ECDHE	X	X
Supported Elliptic Curves	NIST	P256/P384	P-256
	Brainpool	-	P256r1
Symmetric Crypto Algorithm	AES (128, 192, 256)	X	AES(128 and 256)
AES Modes	CBC, ECB, CTR	X	X
	CCM, GCM	X	X
MAC	HMAC, CMAC	X	X
	GMAC	X	-
Hash Function	SHA	SHA 256/384	SHA 256/384
Key Derivation (KDF)	HKDF (RFC5869)	X	X
Asymmetric Mutual Authentication	Asymmetric Mutual Authentication	EC-Key Authentication	Sigma-I
	Symmetric Mutual Authentication	AESKey session (SCP03 using AES128)	AES-based Authentication (AES128/256)[1]
Secure Channel	Secure Channel Host-SE	Platform SCP03	EV2 secure messaging[1]
Application session	Number of simultaneous autenticated sessions	2	1
Application support	ECC-Key based cloud connectivity (TLS 1.2, 1.3)	X	X

Figure 5. Crypto algorithms and authenticaton protocol comparison between EdgeLock A5000 and EdgeLock A30

2.3.3 Secure objects

The following differences between EdgeLock A5000 and EdgeLock A30 are considered:

Table 6. Key and certification object comparison between EdgeLock A5000 and EdgeLock A30

	A5000	A30
Total available free user memory	8 KB	16 KB
Key locking (symmetric and asymmetric)	Policy-based locking.	**Deleting persistent keys is not supported.** The key update policy defines if it is possible to update the key value.
Key storage AES	Number of keys is not fixed. Consumes user memory.	Up to five persistent AES key entries are available for mutual authentication, Secure Messaging Data (SMD), and key update operations. These keys are stored in dedicated memory and **do not consume user memory**. Up to eight persistent AES key entries can be allocated for general cryptographic operations. These keys are stored within the user memory.
Key Storage EC	Number of keys is not fixed. Consumes user memory.	Up to five persistent EC key[1] entries. Consumes user memory.

Table 6. Key and certification object comparison between EdgeLock A5000 and EdgeLock A30 *...continued*

	A5000	A30
EC public key part handling on key pair generation.	Stored after key generation inside a key object.	The public key value is returned but not stored inside an A30 key object (to optimize memory consumption).
EC public key storage	Stored in memory as public key and as part of a X509 certificate.	Stored as part of a X509 certificate only (FileType.StandardData). **Deleting files is not supported.**
Sigma-I certificate repository	Not supported, because A5000 does not support the Sigma-I asymmetric authentication protocol.	Dedicated certificate repository for Sigma-I certificates.
Import External Object	Supported	Not supported

[1] A30 supports up to five persistent EC keys for different crypto operations incl. SIGMA-I Mutual Authentication.

Table 7. Data files

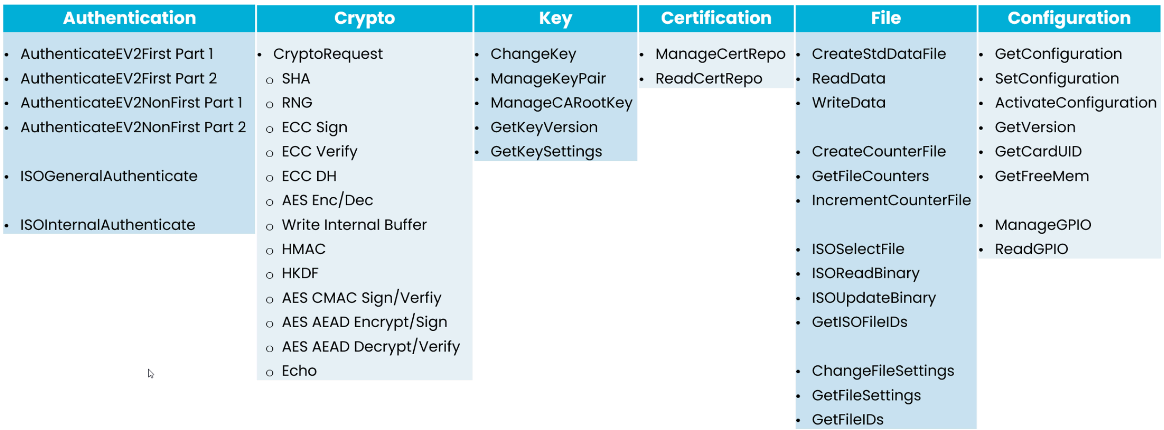
	A5000	A30
Total available free user memory	8 KB	16 KB
Store raw data	Binary File	FileType.StandardData
Monotonic Counter	1–8 byte length	4 byte length FileType.Counter
File locking	Policy-based locking	**Deleting files is not supported.**
File Access Rights	Policy with rights on object creation	Access Rights: FileAR.Read,File AR.Write, FileAR.ReadWrite FileAR. Change

2.3.4 Commands

EdgeLock A30 supports different authenticator commands (APDUs) as EdgeLock A5000.

Note: *The NX MW sss (Secure Sub System) APIs, OpenSSL, PKSC11# and Mbed TLS libraries are abstracting the APDU layer.*

Figure 6 gives a brief overview of the EdgeLock A30 APDU commands.



Authentication

- AuthenticateEV2First Part 1
- AuthenticateEV2First Part 2
- AuthenticateEV2NonFirst Part 1
- AuthenticateEV2NonFirst Part 2
- ISOGeneralAuthenticate
- ISOInternalAuthenticate

Crypto

- CryptoRequest
  - SHA
  - RNG
  - ECC Sign
  - ECC Verify
  - ECC DH
  - AES Enc/Dec
  - Write Internal Buffer
  - HMAC
  - HKDF
  - AES CMAC Sign/Verfiy
  - AES AEAD Encrypt/Sign
  - AES AEAD Decrypt/Verify
  - Echo

Key

- ChangeKey
- ManageKeyPair
- ManageCARootKey
- GetKeyVersion
- GetKeySettings

Certification

- ManageCertRepo
- ReadCertRepo

File

- CreateStdDataFile
- ReadData
- WriteData
- CreateCounterFile
- GetFileCounters
- IncrementCounterFile
- ISOSelectFile
- ISOReadBinary
- ISOUpdateBinary
- GetISOFileIDs
- ChangeFileSettings
- GetFileSettings
- GetFileIDs

Configuration

- GetConfiguration
- SetConfiguration
- ActivateConfiguration
- GetVersion
- GetCardUID
- GetFreeMem
- ManageGPIO
- ReadGPIO

Figure 6. EdgeLock A30 APDU command overview

2.4 Middleware considerations

This chapter provides a brief comparison between EdgeLock A5000 Plug & Trust Middleware and EdgeLock A30 Plug & Trust Middleware and the estimated middleware migration effort.

2.4.1 EdgeLock A5000 Plug & Trust Middleware

Figure 7 shows a simplified representation of the EdgeLock A5000 Plug & Trust Middleware components:

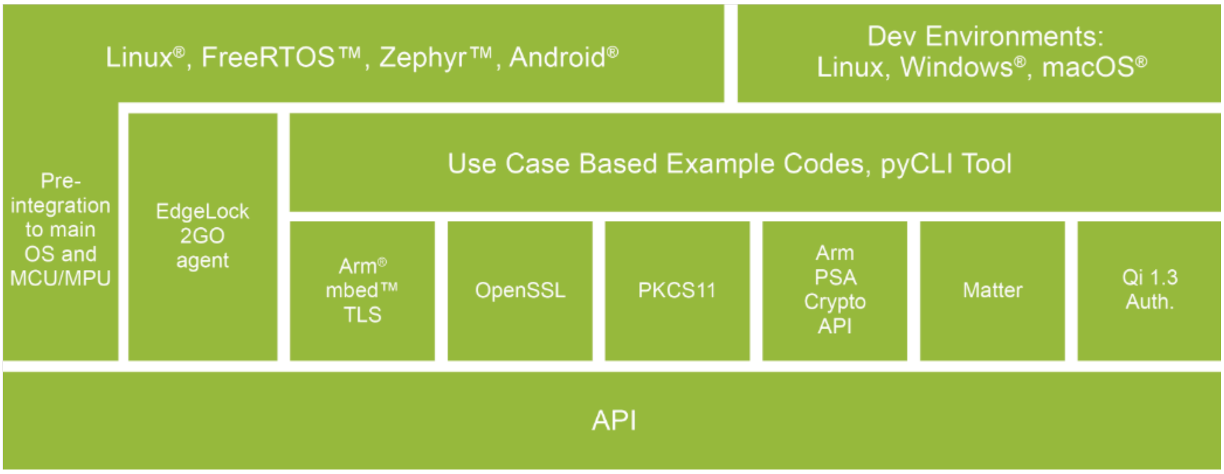

Linux®, FreeRTOS™, Zephyr™, Android®
Dev Environments: Linux, Windows®, macOS®
Pre-integration to main OS and MCU/MPU
EdgeLock 2GO agent
Use Case Based Example Codes, pyCLI Tool
Arm® mbed™ TLS
OpenSSL
PKCS11
Arm PSA Crypto API
Matter
Qi 1.3 Auth.
API

Figure 7. Plug & Trust Middleware block diagram

- EdgeLock Plug & Trust Middleware is distributed via different packages:
  - Full Multiplatform Plug & Trust middleware package (www.nxp.com/A5000).
  - Plug & Trust Mini Package (GitHub) is a subset of the Plug & Trust middleware for Linux use.
  - Plug & Trust Nano Package (GitHub) is a minimalistic version of the Plug & Trust middleware optimized for constrained devices. It also provides an integration with Zephyr OS and an example of Qi 1.3 authentication.
- Supported MCU/MPU platforms out of the box:
  - MCUs: MIMXRT1170-EVK, MIMXRT1060-EVK, FRDM-64F and the LPC55S69-EVK
  - MPUs: Raspberry Pi and MCIMX8M-EVK
- Command line provisioning tool: *ssscli*

2.4.2 EdgeLock A30 Plug & Trust Middleware

Figure 8 gives a brief overview of the EdgeLock A30 NX Middleware components:

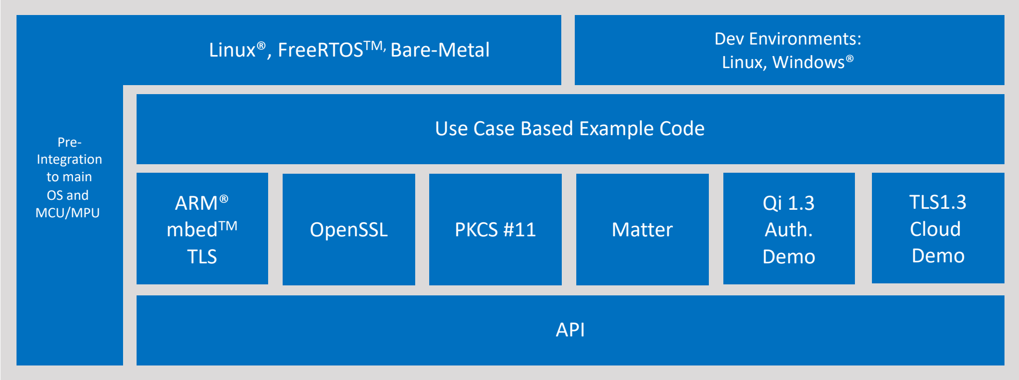

Linux®, FreeRTOS™, Bare-Metal
Dev Environments: Linux, Windows®
Pre-Integration to main OS and MCU/MPU
Use Case Based Example Code
ARM® mbed™ TLS
OpenSSL
PKCS #11
Matter
Qi 1.3 Auth. Demo
TLS1.3 Cloud Demo
API

Figure 8. NX Middleware block diagram

- NX Middleware is distributed via GitHub.
- No dedicated Mini and Nano Package required as this is already part to the NX Middleware release.
- Supported MCU/MPU platforms out of the box:
  - MCUs: FRDM-MCXA153, FRDM-MCXN947 and the LPC55S69-EVK
  - MPU: Raspberry Pi
- Command line provisioning tool: *nxclitool*

2.4.3 Migrating from EdgeLock A5000 Plug & Trust Middleware and EdgeLock A30 NX Middleware

Figure 9 shows the high level EdgeLock A30 NX Middleware architecture and gives an overview of the estimated application migration effort. For example, the application migration effort is minimal when using one of the higher layers such as the plug-in modules. The opposite is true when the application uses the low-level secure authenticator APDU layer. The following two subchapters are describing the migration effort on MCU platforms (bare metal, freeRTOS, ...) and on embedded Linux platforms.

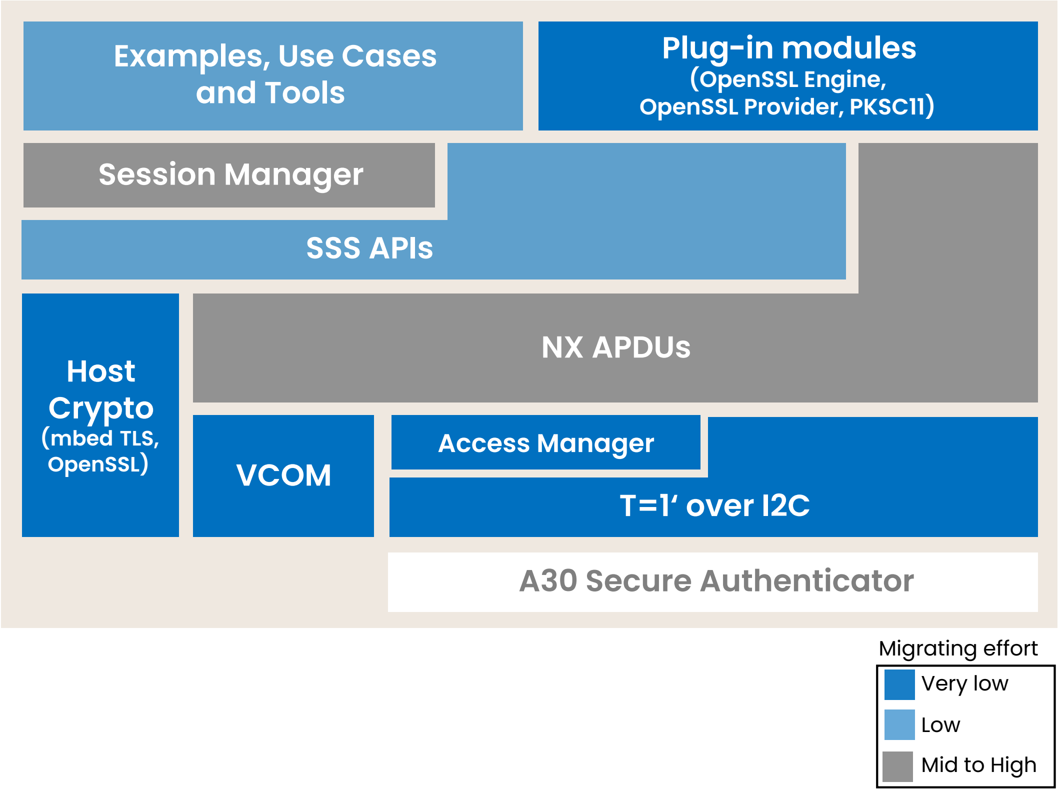


Figure 9. NX Middleware architecture - estimated migration effort

- Session Manager
  - APIs to open a session. The following sessions are supported:
    - Plain session
    - PKI based asymmetric Mutual Authentication (Sigma-I-Verifier or Sigma-I-Prover)
    - AES-based Symmetric Mutual Authentication
  - Note: Both mutual authentication methods initiate a MIFARE DESFire compatible EV2 secure messaging channel (authenticated session).
- SSS APIs
  - Provides abstraction APIs for EdgeLock A30, OpenSSL and mbedTLS host crypto.
  - SSS APIs are supporting common crypto operations.
- NX APDUs
  - Implements the EdgeLock A30 authenticator commands (APDUs)
- Access Manager
  - Manage access from multiple Linux processes to EdgeLock A30. Client processes connect over the JRCPv1 protocol to the Access Manager.

T=1‘ over I^{2}C communication protocol according to Global Platform.

2.4.3.1 Middleware on MCU platforms (bare metal, freeRTOS, …)

NX Middleware provides two different layers of crypto APIs to access the secure authenticator EdgeLock A30 as shown in Figure 10. In addition, the different key and file management must be taken into account.

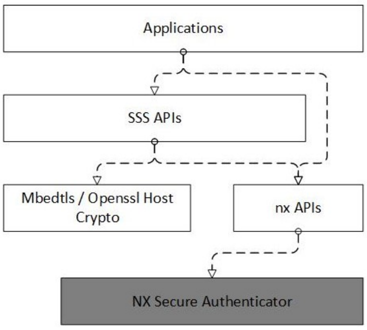


Figure 10. NX Middleware crypto API layers

The *sss_APIs* are independent from the secure authenticator hardware and reduces the porting effort compared to the APDU API layer. More details can be found in Table 8.

Table 8. Middleware migration on MCU platforms

API	EdgeLock A5000 Plug & Trust Middleware	EdgeLock A30 NX Middleware	
Secure Authenticator specific commands (APDUs)	Se05x_ APIs	nx_ APIs	Depending on the use case the porting effort is mid to high.
Security Sub System APIs	sss_ APIs	sss_ APIs	Porting effort is minimal. The SSS APIs are functional APIs to abstract the access to various types of cryptographic sub systems.

2.4.3.2 Migrating on embedded Linux platforms

On embedded Linux platforms, the NX Middleware provides an OpenSSL engine, an OpenSSL provider and a PKSC#11plug-in module in addition to the sss_API and APDU APIs. The migration effort for the sss_API and APDU layer is similar to that for MCUs. Plug-in modules mainly require the replacement of the corresponding Linux libraries and minimal adjustments at the application level. In addition, the different key and file management must be taken into account.

Table 9. Middleware migration on embedded Linux platforms

Plug-in modules	EdgeLock A5000 Plug & Trust Middleware	EdgeLock A30 NX Middleware	Porting effort
Secure Authenticator specific commands (APDUs)	Se05x_ APIs	nx_ APIs	Depending on the use case the porting effort is mid to high.

Table 9. Middleware migration on embedded Linux platforms*...continued*

Plug-in modules	EdgeLock A5000 Plug & Trust Middleware	EdgeLock A30 NX Middleware	Porting effort
Security Sub System APIs	sss_ APIs	sss_ APIs	Porting effort is minimal. The SSS APIs are functional APIs to abstract the access to various types of Cryptographic Sub Systems.
OpenSSL engine	Plug & Trust OpenSSL engine	NX OpenSSL engine	Minimum effort as abstracted by the OpenSSL engine.
OpenSSL provider	Plug & Trust OpenSSL provider	NX OpenSSL provider	Minimum effort as abstracted by the OpenSSL provider.
PKCS#11	Plug & Trust PKCS #11 plug-in	NX PKCS #11 plug-in	Minimum effort as abstracted by the PKCS #11 plugin.
Access Manager[1]	Plug & Trust Access Manager	NX Access Manager	Minimum effort as abstracted by the Access Manager.

[1] Manage access from multiple Linux processes to A30. Client processes connect over the JRCPv1 protocol to the Access Manager.

3 Revision history

Table 10. Revision history

Document ID	Release date	Description
AN14559 v.1.2	2 September 2025	Editorial changes (typos, etc.). • Section 1 "About EdgeLock A30 secure authenticator": updated. • Section 2.3 "Authentication application considerations": Table 6: updated. • Section 2.4.2 "EdgeLock A30 Plug & Trust Middleware": updated.
AN14559 v.1.1	23 January 2025	Correct Table 7 Data files
AN14559 v.1.0	21 January 2025	Initial version

Legal information

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.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

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

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

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

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

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

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.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

Translations — A non-English (translated) version of a document, including the legal information in that 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 vulnerabilities or may support established security standards or specifications with known limitations. 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.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

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

NXP — wordmark and logo are trademarks of NXP B.V.

Tables

Tab. 1. Package comparison 5
Tab. 2. Pin comparison 5
Tab. 3. Supply Voltage comparison 5
Tab. 4. I2C target interface comparison 8
Tab. 5. Communication protocol comparison 8
Tab. 6. Key and certification object comparison between EdgeLock A5000 and EdgeLock A30 9
Tab. 7. Data files 10
Tab. 8. Middleware migration on MCU platforms 14
Tab. 9. Middleware migration on embedded Linux platforms 14
Tab. 10. Revision history 16

Figures

Fig. 1. EdgeLock A30 support package overview 2
Fig. 2. High-level comparison between EdgeLock A5000 and EdgeLock A30 4
Fig. 3. EdgeLock A5000 application circuit diagram with deep power down support 6
Fig. 4. EdgeLock A30 application circuit diagram with deep power down support 7
Fig. 5. Crypto algorithms and authenticaton protocol comparison between EdgeLock A5000 and EdgeLock A30 9
Fig. 6. EdgeLock A30 APDU command overview 11
Fig. 7. Plug & Trust Middleware block diagram 11
Fig. 8. NX Middleware block diagram 12
Fig. 9. NX Middleware architecture - estimated migration effort 13
Fig. 10. NX Middleware crypto API layers 14

Contents

1 About EdgeLock A30 secure authenticator 2
2 Migrating from EdgeLock EdgeLock A5000 to EdgeLock A30 4
2.1 Hardware integration considerations 4
2.1.1 Application circuit diagram with depp power down support 5
2.1.1.1 A5000 application circuit diagram 6
2.1.1.2 A30 application circuit diagram 7
2.2 I2C interface and communicaton protocol considerations 8
2.3 Authentication application considerations 8
2.3.1 EdgeLock A30 Authentication overview 8
2.3.2 Crypto algorithms and protocols 9
2.3.3 Secure objects 9
2.3.4 Commands 11
2.4 Middleware considerations 11
2.4.1 EdgeLock A5000 Plug & Trust Middleware 11
2.4.2 EdgeLock A30 Plug & Trust Middleware 12
2.4.3 Migrating from EdgeLock A5000 Plug & Trust Middleware and EdgeLock A30 NX Middleware 12
2.4.3.1 Middleware on MCU platforms (bare metal, freeRTOS, …) 14
2.4.3.2 Migrating on embedded Linux platforms 14
3 Revision history 16
Legal information 17

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