AN12664

EdgeLock SE05x for NFC late-stage configuration

Rev. 1.1 — 7 December 2020 Application note

583210

Document information

Information	Content
Keywords	EdgeLock SE05x, Late-stage configuration, IoT device lifecycle management
Abstract	This application note describe how to leverage EdgeLock SE05x to enable a secure and convenient late-stage parameter configuration of IoT devices in the factory, before shipment, or in the field.

Revision history

Revision history

Revision number	Date	Description
1.0	2020-03-19	First document release
1.1	2020-12-07	Updated to the latest template and fixed broken URLs

1 Identity management in IoT devices with EdgeLock SE05x

Identity management plays an important role in protecting IoT devices and the systems they interact with. A secure digital identity is the basis for device authentication and to make sure that a device attempting to connect into a secure network is the device it is claiming to be. With a unique strong device identity, smart devices can authenticate with other devices, services and users, protecting networks against cyberattacks, hacking and data breach.

Public Key Infrastructure (PKI) credentials and a hardware-based secure element can provide devices with secure and trustworthy identities. Typically, the credential injection is done before manufacturing so that devices leave the factory with trusted device identities. However, some manufacturers need more flexibility to configure devices either in the factory, before shipment, in the field, or during deployment, what we refer to *late-stage configuration*.

Late-stage parameter configuration lets manufacturers add specific settings, keys, or data to a generic IoT device in a flexible way at any point of the supply chain. Similarly, end customers can enter or edit specific parameters like P addresses, device IDs or many others before turning on their new devices.

The EdgeLock SE05x provides an immutable root of trust at the IC level. It is a tamper-resistant secure element that includes an ISO/IEC 14443-compliant contactless interface which can be used to wirelessly write data into the EdgeLock SE05x secure file system. As such, it gives manufactures a quick, secure way to support late-stage parameter configuration, by making it possible to use a standard NFC smartphone or NFC reader able to send ISO/IEC 14443-4 APDUs to load settings with a simple tap to the unpowered device. The smartphone or reader becomes the graphical user interface to the device, so it is easy to make selections, finalize settings, and transfer parameters.

2 IoT device lifecycle with EdgeLock SE05x for late-stage configuration

Some manufacturers of IoT devices need the flexbility to address different customer requirements while maintaining scalibility. These different customers might require IoT devices to come programmed with a particular setup, a specific set of network parameters, log data, or prepared for use in a given geographical region.

The EdgeLock SE05x enables a secure and convenient late-stage configuration of IoT devices in the factory, before shipment or in the field. We can use any NFC reader as an external user interface to load in the keys and enter or edit or system-specific settings, among others. Using the ISO/IEC 14443 interface of EdgeLock SE05x, this late-stage configuration can be done without opening the device enclosure, without the need of powering the device, and even with waterproof, dust-proof or fully sealed devices.

A generic lifecycle for IoT products with EdgeLock SE05x starts with NXP as the IC provider and it goes through different stages, from manufacturing the hardware, the device enrollment for operation, to the device decommissioning at the product end of life. Figure 1 shows a simplified representation of the lifecycle for an IoT product that leverages EdgeLock SE05x for late-stage configuration:

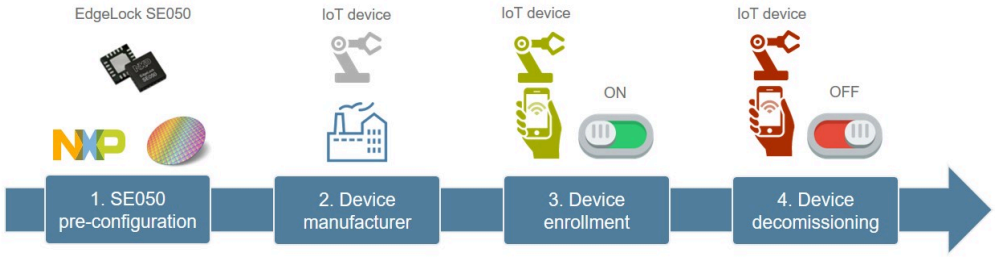

Figure 1. IoT device lifecycle with EdgeLock SE05x for late-stage configuration

Note: *The stakeholders responsible for or affected by these stages can vary from one system to another.*

The IoT device lifecycle with EdgeLock SE05x for late-stage configuration consists of the following phases:

1. **EdgeLock SE050 pre-configuration phase:** NXP delivers an amount of EdgeLock SE05x ICs based on a customer purchase order. The ICs are pre-provisioned with die-individial credentials, injected by NXP or its distribution partners.
2. **Manufacturing phase:** Constructing the device hardware and deploying the initial software onto the hardware takes place. The EdgeLock SE05x is integrated in the device, including the connection with the host MCU and with the NFC antenna coil.
3. **Enrollment phase**: The device is installed into its final location and prepared for operation and secure communication within the network. The device is programmed with specific settings and cryptographic material using an NFC phone or tablet as an external graphical user interface.
4. **Decommissioning phase**: The device is securely removed from the system. For this purpose, an NFC phone or tablet can also be used to reset the device to factory default values or to erase the device configuration.

3 Late-stage configuration example in smart industry

The EdgeLock SE05x enables a flexible IoT device identity lifecycle management. The connection of an external contactless antenna to EdgeLock SE05x allows you to use an NFC phone or tablet to provision digital identities before shipment, in the field or during deployment. This facilitates industries with convenient device renewals, ownership transfers, service revocation or end of life decommissioning in the field.

For illustrative purposes, let's assume a communication network in a smart factory facility used to connect sensors, actuators, PLCs, robots or any other equipment to some control servers. These networked machines require occasional maintenance, replacement or reconfiguration. As such, they need to be securely onboarded and disconnected from the network to avoid privacy breaches, data theft, unauthorized network access or threats to critical infrastructure.

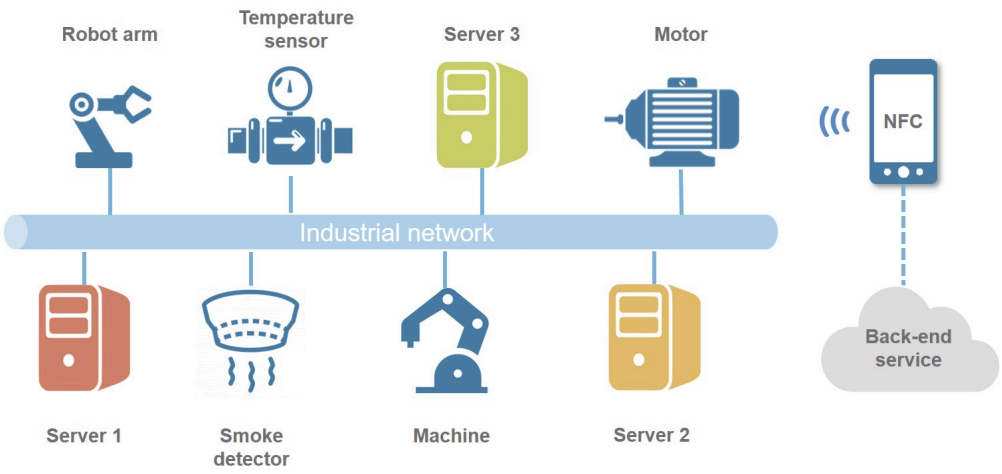

Figure 2. Industrial network setting example

In order to minimize the time spent on device onboarding or reconfiguration, this smart factory implements a phone application that technicians use to: verify device authenticity, and securely connect / disconnect it from one of the factory network servers leveraging EdgeLock SE05x. This phone application connects with the company PKI infrastructure, which is used to authenticate and provision device-individual credentials into EdgeLock SE05x.

Based on this industrial setting example, the following sections elaborate on relevant implementation details and give insights into the integration of EdgeLock SE05x from a hardware and software perspective.

3.1 EdgeLock SE05x pre-configuration for ease of use

NXP is the starting point of any design based on EdgeLock SE05x. The product lifecycle starts with the delivery of an amount of EdgeLock SE05x ICs pre-configured by NXP secure facilities or its distribution partners for ease-of-use during operational phase.

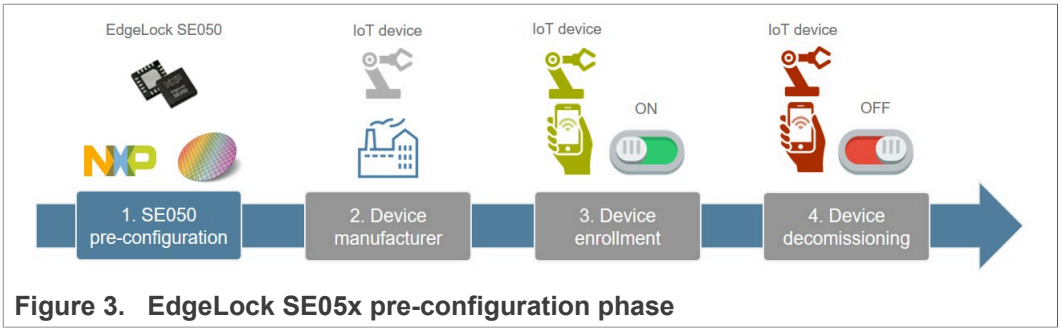

Figure 3. EdgeLock SE05x pre-configuration phase

NXP offers the EdgeLock SE05x variants off-the-shelf pre-provisioned with individual keys and credentials. These pre-injected credentials offer out-of-the-box device integrity protection and attestation, and can be read out from the chip and installed on different cloud services as needed. In AN12436, you can find the definition of available EdgeLock SE05x configurations.

Note: *Provisioning of further user specific credentials is possible in NXP production facilities or at programming centers. Get in touch with your NXP representative for more information.*

Based on the industrial setting described in Section 3, we assume that the factory policy mandates to attest the new device before any credentials are provisioned on it. Device attestation is the affirmation that the identity of a device is correct, true or genuine. NXP delivers every EdgeLock SE05x trust provisioned with an attestation key and its associated signed certificate. Table 1 shows the details of the attestation credentials trust provisioned in each EdgeLock SE05x

Table 1. Attestation credentials in EdgeLock SE05x variant C

Use case	Key type	Certificate
Attestation, proof of origin	Root of trust signing key, ECC256, die individual.	Attestation cert, ECC signed
	Root of trust signing key, RSA2048, die individual.	Attestation cert, RSA signed

Note: *Refer to AN123436 - SE050 configurations for more details about the EdgeLock SE05x pre-configuration for ease of use*

Using these pre-installed attestation credentials, the factory technicians can verify the device authenticity before provisioning any key to the device or before it is integrated into their existing infrastructure. Optionally, the manufacturer certificate can also be used for attestation purposes if it is provisioned in EdgeLock SE05x during production.

3.2 Device manufacturing

The integration of EdgeLock SE05x into the IoT device hardware has minimal impact on design complexity or footprint. The EdgeLock SE05x is designed to serve as an add-on or auxiliary device attached to the host MCU/MPU.

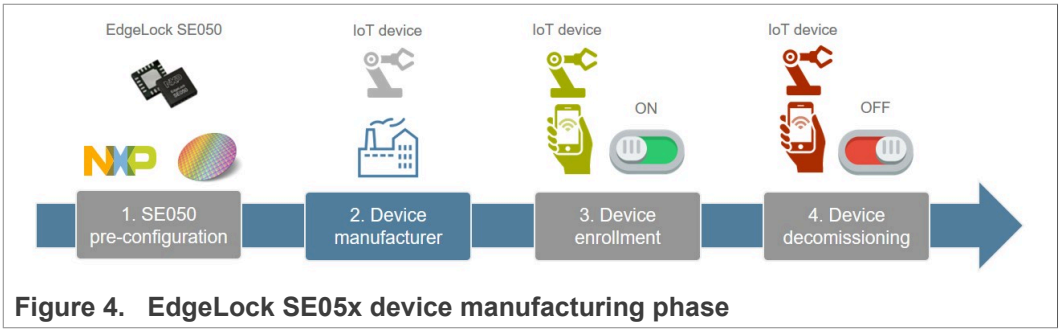

Figure 4. EdgeLock SE05x device manufacturing phase

To communicate with the host MCU/MPU, the EdgeLock SE05x uses an I²C slave interface. Besides this mandatory connection to the host MCU/MPU, the EdgeLock SE05x supports ISO/IEC 14443 contactless connectivity and an I²C master interface for connections to digital sensors.

Figure 5 describes the application circuit that uses the I²C slave for host connection, the I²C master for sensor data protection and the ISO/IEC 14443 for NFC communication.

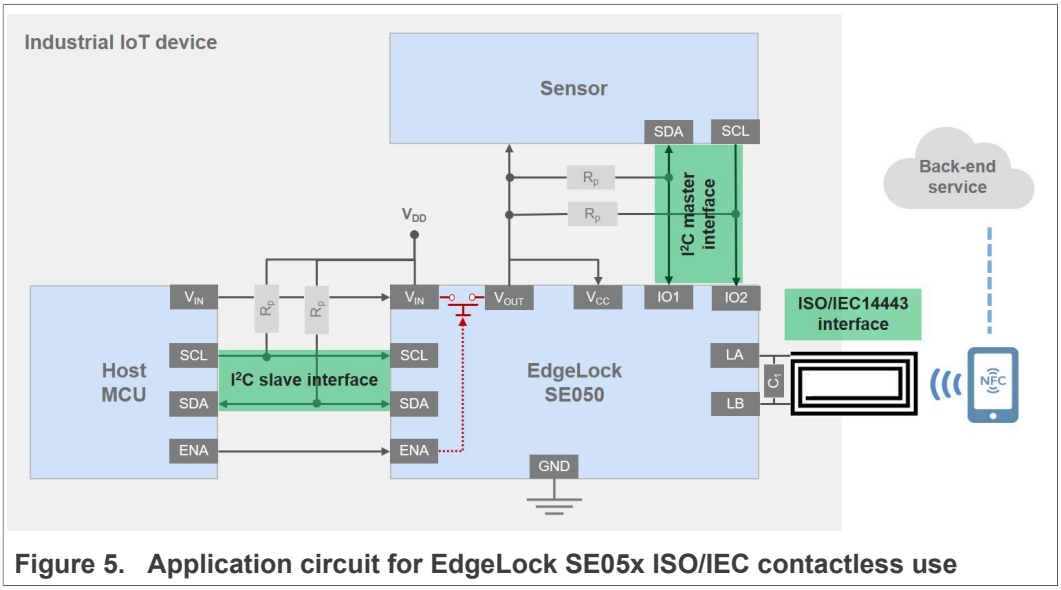

Figure 5. Application circuit for EdgeLock SE05x ISO/IEC contactless use

In this application circuit, the host MCU provides the clock and pull-up supply on the I²C bus and controls the supply switch via the ENA pin. The EdgeLock SE05x Vin pin is supplied by the main supply source of the system (VDD). Vcc is supplied directly by Vout and the sensor is connected to the IO1 and IO2 pins with external pull-up resistors, used as SDA and SCL lines towards EdgeLock SE05x. The antenna coil is connected to the LA and LB pins of EdgeLock SE05x. For further details on NFC antenna design, you can refer to NTAG antenna design guide support package.

Note: *EdgeLock SE05x IC capacitance is 56pF.*

3.3 Device enrollment

The device enrollment phase consists of securely onboarding a new device to an existing infrastructure. During enrollment, the device identity is verified and registered into the platform, it is programmed with cryptographic material, and policies are set to secure

infrastructure from unauthorized access. Once the enrollment is completed, the device can securely sign in to the network services and resources.

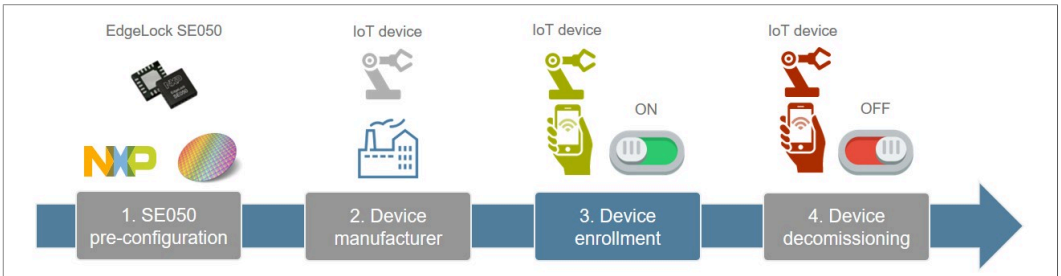

Figure 6. EdgeLock SE05x device enrollment phase

Based on the industrial setting described in Section 3, we assume that a new pressure sensor needs to be installed in the production line, as shown in Figure 7. This pressure sensor will immediately shutdown the equipment in case the measured pressure exceeds a certain threshold, considered dangerous for the engineers safety.

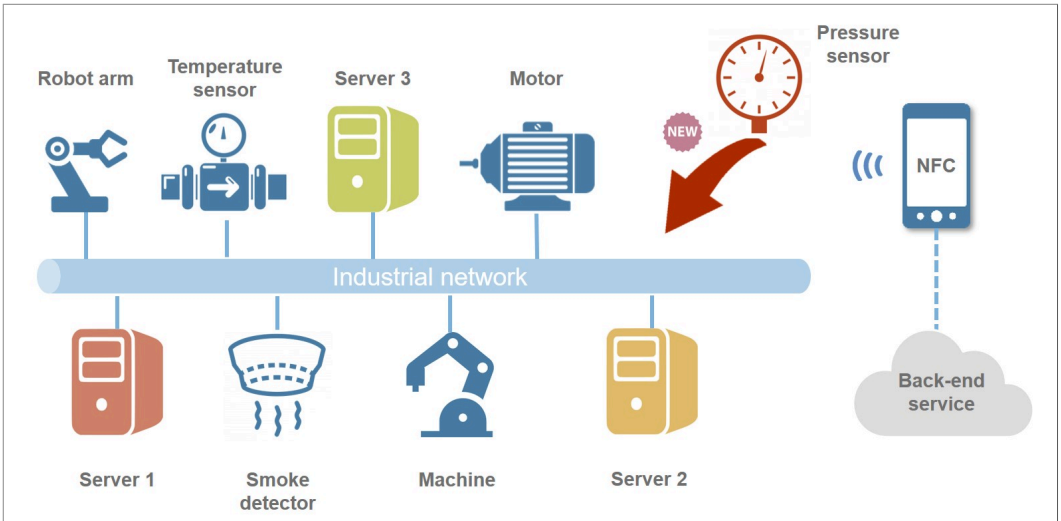

Figure 7. On-boarding of a new pressure sensor scenario

This pressure sensor is delivered with a default profile and settings. As such, it needs to be configured and enrolled into the factory infrastructure once it is installed in the production line. The pressure sensor embeds an EdgeLock SE05x, so that the technicians can use their NFC-capable phones to verify the device authenticity and provision, in the field, server-specific credentials on it.

3.3.1 EdgeLock SE05x supported functionality overview

In EdgeLock SE05x, these server-specific credentials are stored as *secure objects* with associated attributes. The secure objects supported are:

- ECKey: An elliptic curve key type (key pair / private key / public key).
- RSAKey: A RSA key type (key pair / private key / public key). The private key can be stored in CRT or in raw format.
- AESKey: An AES key (AES128, AES192 or AES256).
- DESKey: A DES key

- HMAC: A secret up to 256 bytes. Typically, it is used as input for MAC codes or key derivation functions when the key material is not 16 or 32 bytes long.
- BinaryFile: A file containing a byte array of a specific length (e.g. digital certificates, device IDs, Node IDs, strings, etc).
- Counter: A monotonic counter
- PCR: Platform configuration register of 32-byte that holds the value of a SHA256.
- UserID: A PIN code for authentication of 4 up to 16 bytes.

Each secure object has a number of attributes assigned to it, which include:

- Object ID: A unique ID of the secure object (4-bytes)
- Type: The type of the secure object (e.g. ECKey, BinaryFile, etc)
- Policy: A list of access and usage policies applicable for this secure object.
- Additional attributes: Other attributes such as number of failed authentications, maximum authentication attempts, etc

In addition, every secure object can have assigned a *policy set.* A *policy set* is a collecting of policies that restrict the usage of a specific object. Policies are assigned at object creation and cannot changed. The supported policies are:

- POLICY_OBJ_ALLOW_DELETE
- POLICY_OBJ_REQUIRE_SM
- POLICY_OBJ_REQUIRE_PCR_VALUE
- POLICY_OBJ_FORBID_ALL
- POLICY_OBJ_ALLOW_SIGN
- POLICY_OBJ_ALLOW_VERIFY
- POLICY_OBJ_ALLOW_ENC
- POLICY_OBJ_ALLOW_DEC
- POLICY_OBJ_ALLOW_KDF
- POLICY_OBJ_ALLOW_WRAP
- POLICY_OBJ_ALLOW_WRITE
- POLICY_OBJ_ALLOW_GEN
- POLICY_OBJ_ALLOW_KA
- POLICY_OBJ_ALLOW_READ
- POLICY_OBJ_ALLOW_ATTESTATION
- POLICY_OBJ_ALLOW_DESFIRE_AUTHENTICATION
- POLICY_OBJ_ALLOW_DESFIRE_DUMP_SESSION_KEYS
- POLICY_OBJ_ALLOW_IMPORT_EXPORT

Note that not every policy works with every object type. For further details on EdgeLock SE05x secure objects, attributes and policies use, you can refer to AN12514 - SE050 User guidelines and AN12413 SE050 APDU Specification.

3.3.2 Device and platform credentials

Based on the industrial setting described in Section 3, we assume that every piece of equipment must authenticate to one of the control servers before it comes online and is onboarded in the network. After a successful authentication, the device operation is managed by this server, data collected from it is secured and the communication is encrypted.

The smart factory owns, or subcontracts, a back-end service to secure communication between devices, services and users. This back-end service holds the factory root CA

certificate and is responsible for generating an intermediate CA certificate for each network server, and individual certificates for each device requiring a server connection.

The factory root CA is used to sign the intermediate CA certificate provisioned into each network server. Similarly, each server intermediate CA is used to sign the certificate provisioned into each device requiring connection to that particular network server. These certificates are used to provide trust and protect the entire industrial network through device authorization, authentication and data encryption.

The factory technicians have access to a user-friendly phone application, used to perform the device late-stage configuration and enroll new devices to the network in the field. The phone acts as "a bridge" between the back-end service and the device itself. On the one hand, the phone connects to the backend service using its wireless connectivity. On the other hand, the phone connects to the device using NFC and the EdgeLock SE05x's ISO/IEC 14443 interface.

As an example, we assume a phone application that allows technicians to enter the *device type*, the *device ID* and to configure the *server connection*. Figure 8 shows an example of this phone application UI as well as the credentials for this sample setup:

Figure 8. Example of phone application UI for device late-stage credential provisioning

Note: *For the sake of simplicity, private keys are not displayed in Figure 8, but are also needed and securely stored respectively in the PKI infrastructure, servers and in the device's EdgeLock SE05x together with each certificate.*

Optionally, the device certificate can include the *device ID* and *device type* as X.509 certificate parameters. In this case, the certificate generated and signed by back-end service will contain device-specific protected data (i.e. device ID and device type). In the Certificate below, you can see the contents of one the pre-provisioned certificates in EdgeLock SE05x.

```
Certificate:
 Data:
  Version: 3 (0x2)
  Serial Number:04:00:50:01:c0:2e:dc:e5:c8:c3:9e:
```

```
    04:25:05:59:55:00:00:00:00
  Signature Algorithm: ecdsa-with-SHA256
   Issuer: OU = Plug and Trust, O = NXP, CN = CloudConn-
Intermediate-040050010004A1F4-ECC
   Validity
    Not Before: Jun 18 00:00:00 2019 GMT
    Not After : Jun 15 00:00:00 2031 GMT
    Subject: OU = Plug and Trust, O = NXP, CN = CloudConn0-
ECC-04005001C02EDCE5C8C39E04250559550000
    Subject Public Key Info:
     Public Key Algorithm: id-ecPublicKey
       Public-Key: (256 bit)
       pub:
         04:3a:68:bf:ff:58:77:76:e2:dc:d3:63:1d:79:87:
         4a:cc:e4:17:ae:2b:57:80:d2:ed:bb:5d:b4:ea:c9:
         8c:fd:07:ee:7a:53:71:eb:db:0b:8a:ed:a2:21:7c:
         c9:2e:ac:13:c8:9a:1a:db:f3:15:f8:87:5c:b0:ae:
         e7:41:a8:84:36
       ASN1 OID: prime256v1
       NIST CURVE: P-256
    X509v3 extensions:
    X509v3 Basic Constraints:
      CA:FALSE
    X509v3 Key Usage:
      Digital Signature
        X509v3 Subject Alternative Name: email:d:NXP-SE050-EC-
D:04005001C02EDCE5C8C39E04250559550000
      Signature Algorithm: ecdsa-with-SHA256
         30:45:02:21:00:f2:28:5c:15:d0:3a:e3:e5:55:b2:73:26:24:
         19:54:60:af:25:e0:21:7a:d1:45:28:f9:fe:69:1b:7d:74:e9:
         ac:02:20:79:b6:ec:c4:39:13:d1:04:2b:fa:c4:81:ed:03:27:
         d9:a8:79:c3:43:21:81:3f:51:64:8c:ce:a3:24:f8:bc:be
```

Note: *This certificate template it is provided for illustrative purposes.*

3.3.3 Device credential provisioning

During the provisioning phase, the device is personalized with customized credentials besides the ones already injected as part of the pre-configuration for ease of use (see Section 3.1). As described in Section 3.3.1, custom credentials are stored in EdgeLock SE05x by creating new secure objects and its associated attributes (identifier, type, policy set, etc). For this trust provisioning in the field, an applet level SCP or secure object import functions is required to ensure end-to-end confidentiality and integrity.

Based on the industrial setting described in Section 3, we assume that the device provisioning process requires of: an ECC key pair and its corresponding digital certificate, and an other device configuration parameters. During this process, the phone acts as a bridge between the factory back-end service and the device to be configured.

The device provisioning flow is depicted in Figure 9. It consists of these phases:

- The technician authenticates to the back-end service and submits the device configuration using the mobile application.
- The back-end service sets up an SCP03 secure messaging with the EdgeLock SE05x embedded in the pressure sensor.
- The back-end request to the EdgeLock SE05x to create a keypair, and send back the corresponding public key. With the received public key, the back-end service creates

a certificate for the device. This certificate is signed with the intermediate CA of the server the device needs connect to.

- Using SCP03 secure messaging, the back-end service injects the signed device certificate, and any other device configuration parameters into EdgeLock SE05x.

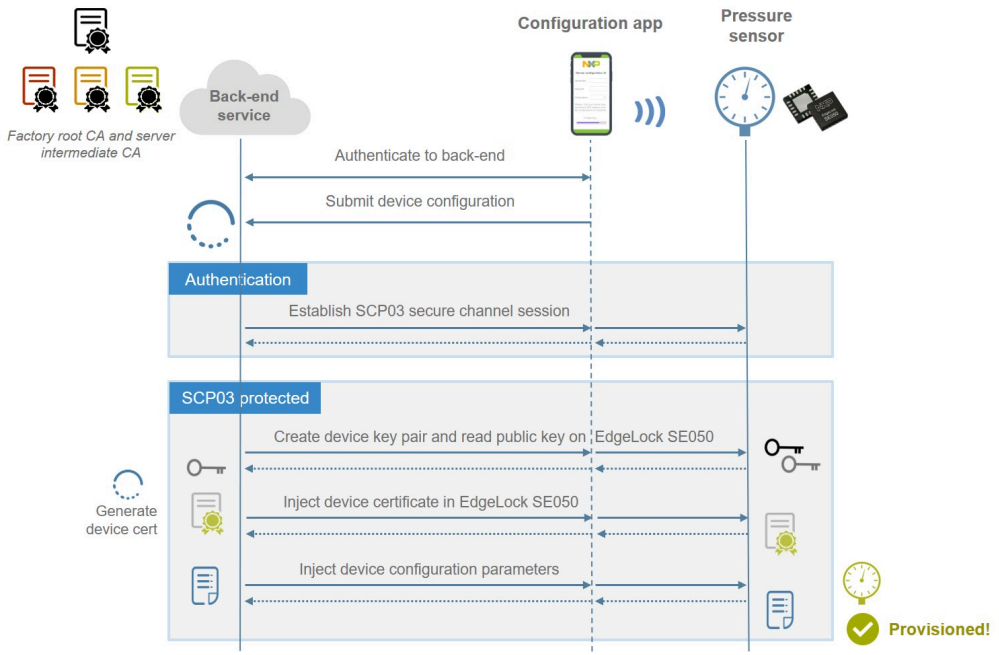

Figure 9. Device credential provisioning flow

Alternatively, users can also safely import credentials using *ECKey* protocol or *ImportExternalObject* command. The ECKey session allows us to establish a secure channel session with EdgeLock SE05x based on asymmetric cryptography (ECC key authentication object). The *ImportExternalObject* command uses a mechanism based on ECC cryptography to protect the object content. It allows us to import objects into EdgeLock SE05x through a secure channel which does not require the user to open a session with EdgeLock SE05x.

The AN12413 - SE050 APDU specification contains the description of the APDU commands accepted by EdgeLock SE05x. Refer to this document for the format, fields and content specification of the commands mentioned in this section.

3.3.4 Device authentication

With the credentials provisioned in the network servers and devices signed into a certificate chain of trust, what remains is to authenticate the device when it attempts to connect to the network server. Based on the industrial setting described in Section 3, we assume that the server forces a previous authentication to any device requiring connection to it. If this authentication is successful, network access is granted and communication is encrypted using the dedicated network key provisioned during Section 3.3.3.

The authentication of the device consists of two steps: the *certificate validation* and the *private key proof of possession* as shown in Figure 10:

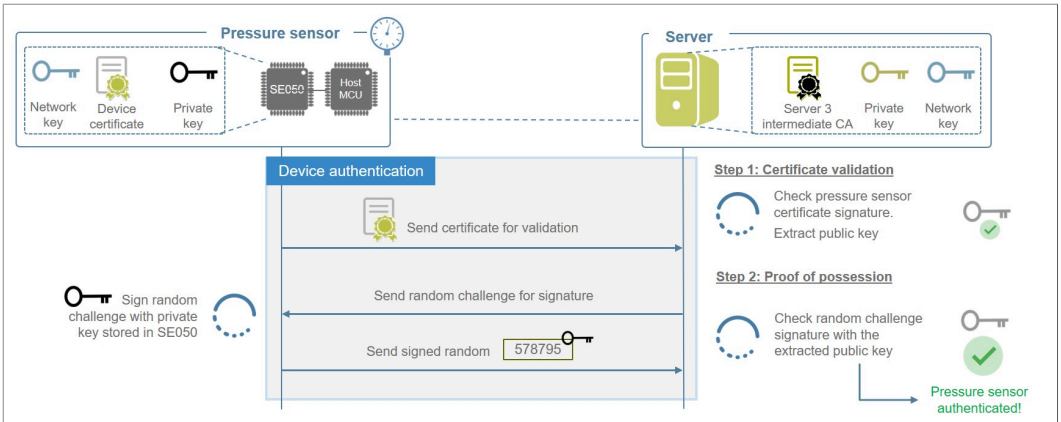

Figure 10. Pressure sensor authentication flow to connect to the network

The *certificate validation* step verifies the integrity and proves that it was signed by a trusted root (i.e. factory root CA). In this case, the pressure sensor sends its device certificate together with its certificate chain of trust. If the certificate is validated, the server can trust the public key belongs to that device. Note that, for the sake of simplicity, Figure 10 does not depict the entire chain of certificates. Chain of trust is depicted in Figure 8.

The *proof of possession* step involves a cryptographic challenge and response process between the two parties to ensure that the uploader of the certificate also knows the associated private key. It consists of generating a random challenge that must be signed with the device private key and verified with the public key in possession by the server. As the private key is kept secret and protected in EdgeLock SE05x, only this device possesses the knowledge to complete this step.

The server declares the device authentic on a successful proof-of-possession response from the device. In that case, network access is granted. Figure 8 only shows the one-way authentication flow (pressure sensor authentication). In case a mutual authentication is required (server authentication), the same procedure can be extrapolated. You can find more details about how to implement it in AN12399- EdgeLock SE050 for device-to-device authentication.

3.4 Device decommissioning

Device security should be considerated through the product's entire lifecycle. In this respect, at the end-of-life of the device's lifecycle, it is key to remove critical data from the device and remove the device from production in a controlled manner. As such, operators need to be able to securely remove a device from service, ensuring that it is truly "dead" and does not expose a vulnerability that intruders could exploit.

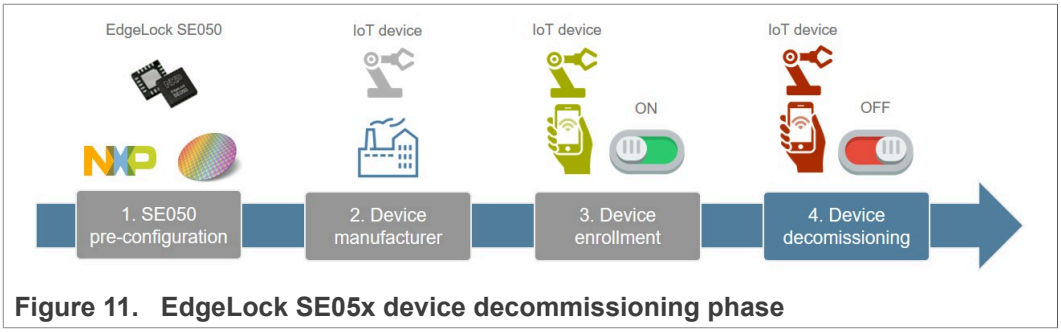

Figure 11. EdgeLock SE05x device decommissioning phase

Based on the late-stage device configuration described in Section 3.3.3, we created an *ECC key pair*, a *certificate* and some *configuration parameters*. Therefore, the decommissioning process for this device consists of removing these credentials from the EdgeLock SE05x storage. During this process, the NFC reader acts as a bridge between the factory back-end serivce and the device to be decommissioned. The device decommissioning flow is depicted in Figure 12. It includes these phases:

- The technician authenticates to the back-end service and submits the device decommissioning request using the mobile application.
- The back-end service sets up an SCP03 secure messaging with the EdgeLock SE05x embedded in the pressure sensor.
- Using SCP03 secure messaging, the back-end service deletes the *device key pair*, the certificate, and any configuration parameters previously injected in Section 3.3.3. These secure objects can be deleted by sending the *Delete ()* APDU command, passing the secure object ID as reference for its deletion.

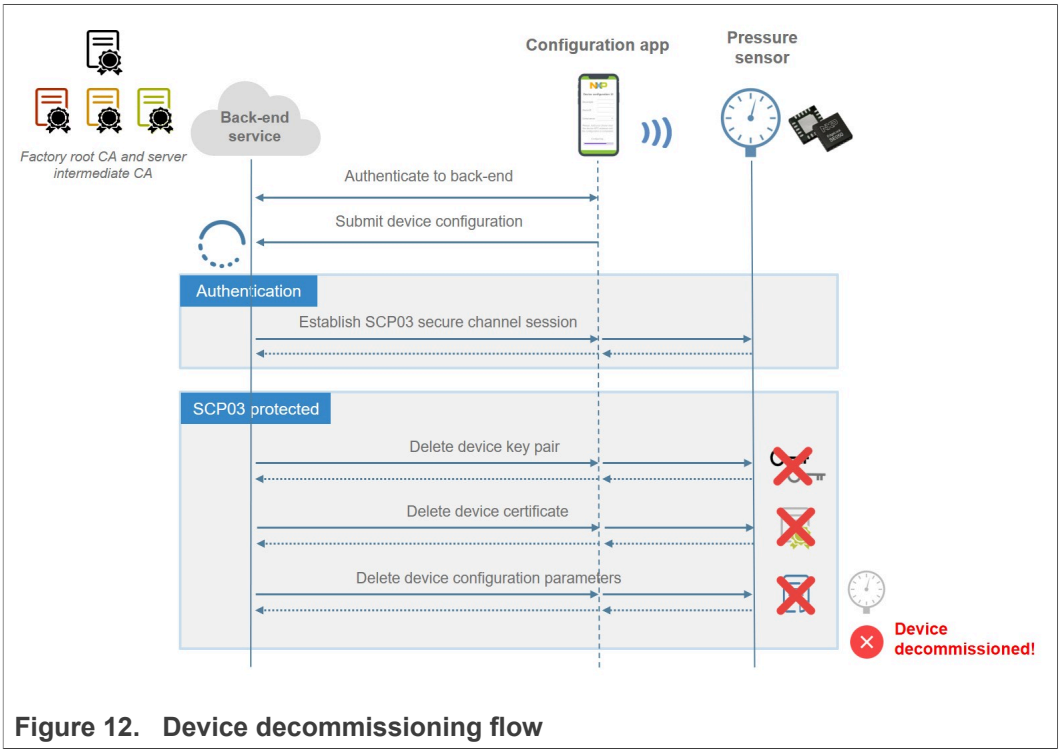

Figure 12. Device decommissioning flow

Note: *The secure object must have set the POLICY_OBJ_ALLOW_DELETE policy to allow deletion.*

The AN12413 - SE050 APDU specification contains the description of the APDU commands accepted by EdgeLock SE05x. Refer to this document for the format, fields and content specification of the commands mentioned in this section.

4 Annex A: Android NFC mobile application considerations

The EdgeLock SE05x ISO/IEC 14443 contactless interface allows any NFC-equipped Android device to send data to EdgeLock SE05x using this short-range wireless connectivity, available in most of the Android phones in the market.

The development of an NFC mobile application that interacts with EdgeLock SE05x does not impose any specific requirement. As such, it can be implemented as any other conventional application on the Android API for NFC.

As part of the Android API, the *IsoDep class* provides access to ISO-DEP (ISO/IEC 14443-4) properties and read / write operations on these type of NFC devices. The primary ISO-DEP operation is called *transceive (byte[ ])*, which allows developers to send raw commands to EdgeLock SE05x and receive the corresponding response.

At the application level, data exchanged between the Android device and the EdgeLock SE05x is conducted using the ISO/IEC7816-4 APDU standard and TLV format. In this case, developers need to build the specific EdgeLock SE05x APDUs as specified in AN12413 - SE050 APDU specification and use the *transceive ( )* command to send and receive data from EdgeLock SE05x over the ISO/IEC 14443 air interface.

As an example, this code snippet shows how to send the *GetVersion* command to EdgeLock SE05x. The overall process is as follows:

- The *onNewIntent* method is triggered each time the phone finds an NFC tag nearby (e.g. in this case, EdgeLock SE05x).
- Create an *IsoDep* object with the tag detected using the *IsoDep.get()* method.
- Connect to the tag with the *connect()* method.
- Specify the APDU to be sent.
- Send the APDU using the *transceive ()* method.

```
@Override
protected void onNewIntent(Intent intent) {
    setIntent(intent);
    readFromIntent(intent);
    if (NfcAdapter.ACTION_TAG_DISCOVERED.equals(intent.getAction())){
        myTag = intent.getParcelableExtra(NfcAdapter.EXTRA_TAG);
        final IsoDep nfcA = IsoDep.get(myTag);
        nfcA.setTimeout(5000);
        try {
            if (nfcA != null) {
                nfcA.connect();
                byte[] GetVersion_APDU =
 {0x80,0x04,0x00,0x20,0x00,0x09};
                nfcA.transceive(GetVersion_APDU);
            }
        } catch (IOException e) {
            e.printStackTrace();
        } catch (ClassNotFoundException e) {
            e.printStackTrace();
        }
    }
}
```

5 Legal information

5.1 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.

5.2 Disclaimers

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

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

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

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

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

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer. In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out 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.

5.3 Trademarks

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

Tables

Figures

Contents

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

Date of release: 7 December 2020
Document identifier: AN12664
Document number: 583210