Brand protection

Use case guide

About this document

Scope and purpose

This document outlines the setup process of the OPTIGA™ Authenticate NBT for offline and online brand protection use cases. It demonstrates the steps through which the end user can operate and explore the two brand protection use cases with the device.

Intended audience

This document is primarily intended for solution providers, system integrators, application developers, and product marketers who want to evaluate and test the brand protection (offline and online) functionality of the OPTIGA™ Authenticate NBT.

Table of contents

About this document . . . 1
Table of contents . . . 2
List of tables . . . 4
List of figures . . . 5
1 Introduction . . . 6
1.1 NFC I2C bridge tags . . . 6
2 Use case overview . . . 7
2.1 General information . . . 7
2.2 Methodology . . . 7
2.3 Brand Protection with offline authentication using PKI . . . 7
2.4 Brand Protection with online authentication using COTT . . . 8
3 Use case integration . . . 9
3.1 Prerequisites . . . 9
3.2 Operation modes . . . 9
3.3 Personalization . . . 10
3.3.1 Device target state . . . 10
3.3.2 Utilized interfaces . . . 11
3.3.3 Personalization procedure . . . 11
3.3.3.1 Interface configurations . . . 12
3.3.3.2 Type 4 Tag application's file configurations . . . 12
3.3.3.3 Brand protection with offline authentication . . . 14
3.3.3.4 Brand protection with online authentication . . . 15
3.3.3.5 Activating the OPERATIONAL life cycle state . . . 15
3.4 Operational use case . . . 15
3.4.1 Operational flow example: Brand protection with offline authentication . . . 15
3.4.2 Operational flow example: Brand protection with online authentication . . . 17
A Appendix . . . 19
A.1 Technical background . . . 19
A.1.1 OPTIGA™ Authenticate NBT system architecture . . . 19
A.1.2 Hardware configuration . . . 21
A.1.3 Interface description . . . 22
A.1.4 Command reference . . . 23
A.1.5 Life cycle states . . . 23
A.2 Device delivery condition . . . 23
A.2.1 Initial NDEF message . . . 25
References . . . 26
Glossary . . . 27

Table of contents

Revision history 30

Disclaimer 31

List of tables

Table 1 EF.CC (relevant for access via the NFC interface) 13
Table 2 EF.FAP (example FAP settings) 13
Table 3 Supported applications of the OPTIGA™ Authenticate NBT 20
Table 4 Type 4 Tag application and files 21
Table 5 Command set of the OPTIGA™ Authenticate NBT 23

List of figures

Figure 1 Offline brand protection components . . . 8
Figure 2 Online brand protection components . . . 8
Figure 3 NFC-only tag - brand protection . . . 9
Figure 4 Interface configuration for the brand protection use cases . . . 10
Figure 5 Target configuration for brand protection . . . 10
Figure 6 Standard personalization procedure . . . 11
Figure 7 Personalization procedure example via NFC interface using a mobile phone . . . 12
Figure 8 Infineon X.509v3 device certificate . . . 14
Figure 9 Offline brand protection – operational flow . . . 16
Figure 10 Online brand protection – operational flow . . . 17
Figure 11 OPTIGA™ Authenticate NBT product architecture . . . 19
Figure 12 Type 4 Tag file structure . . . 20
Figure 13 Embedded tag . . . 21
Figure 14 NFC-only tag . . . 22
Figure 15 Logical communication states of OPTIGA™ Authenticate NBT . . . 22
Figure 16 Delivery condition: Interface configuration . . . 24
Figure 17 Delivery condition: Application file content, access conditions (per-file, per-interface) . . . 24
Figure 18 URI record . . . 25
Figure 19 External record . . . 25

1 Introduction

This use case guide assists users to understand the key features of the OPTIGA™ Authenticate NBT that enable online and offline brand protection use cases. It also provides a high-level overview of how the device needs to be configured for these use cases and the steps required to realize real-world use case scenarios.

Chapter 2 describes the use cases in general and the specific features of the OPTIGA™ Authenticate NBT.

Chapter 3 describes how the described use case will be enabled on the OPTIGA™ Authenticate NBT, beginning with its personalization and guiding through the implementation of the use case.

The Appendix A section provides generic information about the OPTIGA™ Authenticate NBT such as its product architecture, the supported interfaces and the command set. Furthermore, this section contains a comprehensive description of the product delivery condition, which summarizes all the relevant details to enable the preparation of the device for its intended use.

***Note**: For a collection of all available support material for the product, refer to its product page [5].*

1.1 NFC I2C bridge tags

NFC Bridge Tags are dual-interface tags that enable contactless features for IoT devices via an I2C controller interface, allowing for a touch-and-go experience with a mobile phone. On one side, the NFC Bridge Tags include a contactless passive NFC interface and on the other side, a contact-based I2C target interface that connects to the MCU of the IoT device.

The OPTIGA™ Authenticate NBT harnesses the Integrity Guard 32 security architecture to provide an option for the end-user with symmetric and asymmetric cryptographic operations, as well as password-based data protection schemes. As a result, the device is ideal for security demanding applications.

This product includes device authentication, pass-through and asynchronous data transfer modes, which can be used for variety of applications such as:

- Keyless access and activation of shared mobility vehicles
- Controlled access to personal electronic devices such as HDD
- Theft prevention for electronic goods by authenticated activation

This tag can also be used in healthcare and industrial applications. The OPTIGA™ Authenticate NBT, in combination with healthcare sensors, enables access to information through an NFC-enabled mobile phone or reader. Furthermore, the device is an ideal product for industrial applications such as headless configuration and parametrization of devices, assembly line programming and fault diagnostics.

2 Use case overview

This chapter introduces the OPTIGA™ Authenticate NBT and describes how it can be utilized to facilitate brand protection use cases. The device offers customized brand protection features that can be leveraged in both online and offline scenarios.

2.1 General information

The purpose of brand protection is to maintain the integrity of a brand, prevent revenue loss due to counterfeit products, and maintain customer trust. Brand protection refers to the process of safeguarding a company's intellectual property. This includes taking action against unauthorized use, infringement or counterfeiting of a company's brand or products. By leveraging NFC technology, brands can simultaneously safeguard their intellectual property and interact with the end users. Additionally, it enhances the user experience by simplifying the product usage process and offering more information about the product.

Equipped with an NFC Forum compliant Type 4 Tag platform, the OPTIGA™ Authenticate NBT offers simple-to-use and easy-to-integrate capabilities to safeguard branded items and verifying product authenticity. Developers can also take advantage of the platform's integration capabilities to implement end user engagement strategies.

Designed to interact with mobile phone infrastructure, the OPTIGA™ Authenticate NBT is fully compatible with the NFC Forum's Type 4 Tag application. Leveraging its advanced capabilities can lead to more robust protective measures and a maximum overall user experience.

The OPTIGA™ Authenticate NBT supports two specific brand protection methods:

1. Brand Protection with offline authentication using public key infrastructure (PKI)
2. Brand Protection with online authentication using cryptographic one-time token (COTT)

The upcoming chapters elaborate details of these methods and provide a comprehensive guide on how to utilize the OPTIGA™ Authenticate NBT. Additionally, generic implementation concepts to facilitate understanding are introduced.

2.2 Methodology

The OPTIGA™ Authenticate NBT offers support for offline and online authentication methods, making it a valuable tool for brand protection use cases. OEMs and end users benefit from the features, as it helps to verify the authenticity of the product and prevent counterfeiting. Whether verifying the authenticity of a product in person or online, the device provides a robust and reliable solution for brand protection. With its NFC Forum Type 4 Tag compatibility, the device can seamlessly interact with mobile phone infrastructures, delivering enhanced interoperability.

2.3 Brand Protection with offline authentication using PKI

This method makes use of public key infrastructure, which eliminates the need for a cloud connection or additional online services.

End customers who want to validate the authenticity of a product can tap the OPTIGA™ Authenticate NBT-equipped NFC tag attached to the product with any off-the-shelf NFC enabled mobile phone (Android or iOS) executing the brand's product authentication application. The mobile application retrieves the public-key certificate from the NDEF message in the OPTIGA™ Authenticate NBT and validates it by utilizing the application's Root CA (certificate authority). Furthermore, the brand protection application may visually notify the end user of the result of the authentication process.

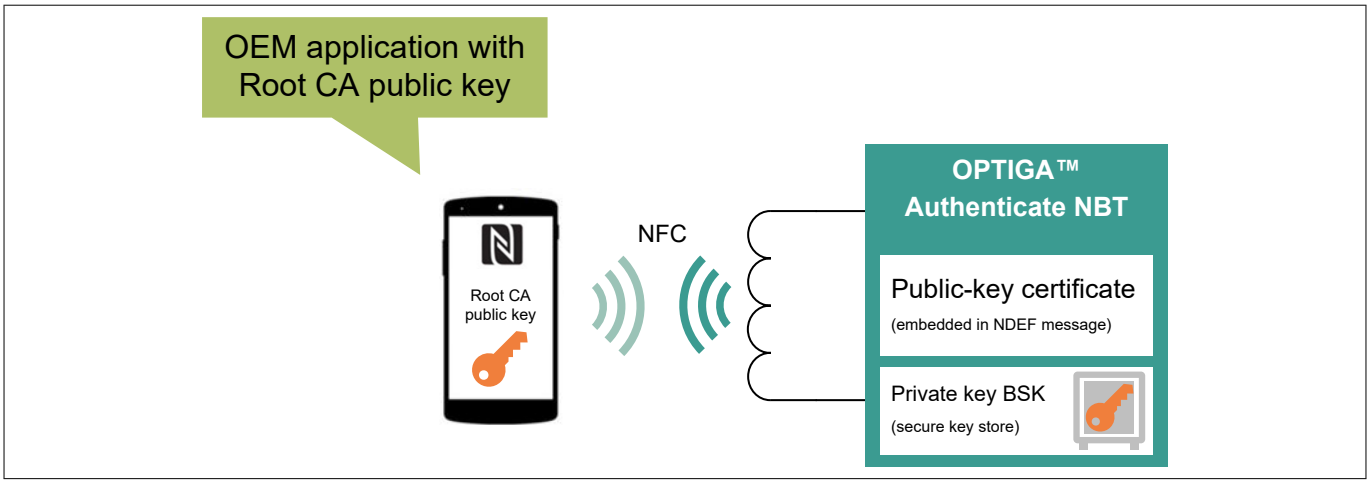


Figure 1 Offline brand protection components

As a second step, the application can transfer a challenge to the OPTIGA™ Authenticate NBT which then computes a signature based on the elliptic curve digital signature algorithm (ECDSA) using the private brand protection signing key (BSK) stored in its secure key store. The signature reverted to the OEM application is then used to verify product authenticity.

2.4 Brand Protection with online authentication using COTT

This method involves verifying the authenticity of a product by connecting to the brand's cloud service. This service is accessible through the web browser of an NFC-enabled mobile phone.

The end user taps the branded item, for example, a luxury hand bag, with the NFC-enabled mobile phone. The phone detects the OPTIGA™ Authenticate NBT's presence without the need of a dedicated mobile phone application. It automatically reads the NDEF message, which contains a dynamically generated URL pointing to the web service, and forwards the embedded COTT value to the Cloud Authentication Service. This Cloud Authentication Service verifies the authenticity of the tagged item and the result of the verification may be returned and displayed via the phone's web browser. The web service can provide additional product related information, for example, manufacturing or purchase information.

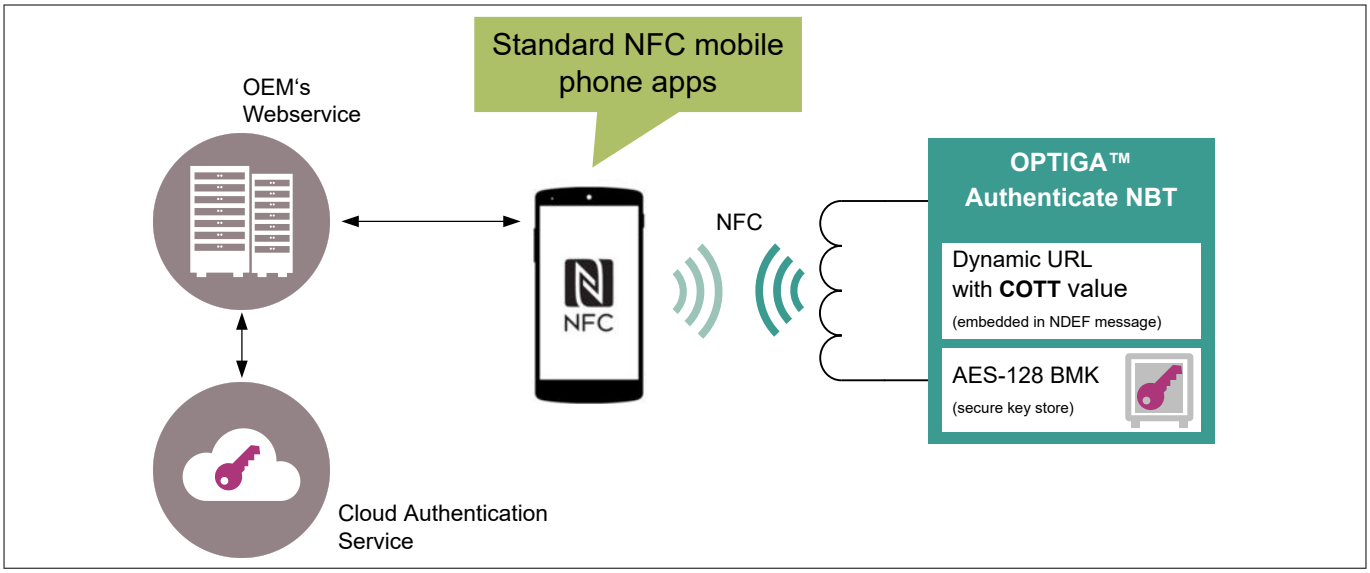


Figure 2 Online brand protection components

3 Use case integration

This chapter describes how to use the OPTIGA™ Authenticate NBT for offline and online brand protection applications. This includes the steps required to configure the product as well as interactions with the product during the OPERATIONAL state.

***Note**: Infineon Technologies provides host libraries to support the integration of OPTIGA™ Authenticate NBT into custom applications on different platforms. Multiple example applications demonstrate how these libraries can be utilized for interactions with the device during personalization and operation in different use cases. For more information, refer to product website [5] or the Software Integration Guide [8].*

The offline brand protection use case is demonstrated using a NFC-enabled phone to perform local verification on the phone. In the online brand protection scenario, the COTT value is validated in a Python framework. Both scenarios are making use of the NFC-only tag setup.

The examples described in this section cover the flow to configure the OPTIGA™ Authenticate NBT. These steps correspond to the example applications provided for this use case, which contain a basic example of how the device can be integrated for this document's use case.

A comprehensive summary of the OPTIGA™ Authenticate NBT's technical details, relevant for the implementation of this use case, is presented in Appendix A.1.

3.1 Prerequisites

The OPTIGA™ Authenticate NBT is shipped in its device delivery condition (see Appendix A.2).

The device must be physically connected to an NFC antenna via its L_A and L_B pins allowing to operate the OPTIGA™ Authenticate NBT from an NFC-enabled phone to obtain energy from the electromagnetic RF field generated by the phone. Optionally, it can be supplied via its power pins (V_{CC}, GND) from an external source (see Figure 3). In this configuration, the contactless communication distance of the device is enhanced.

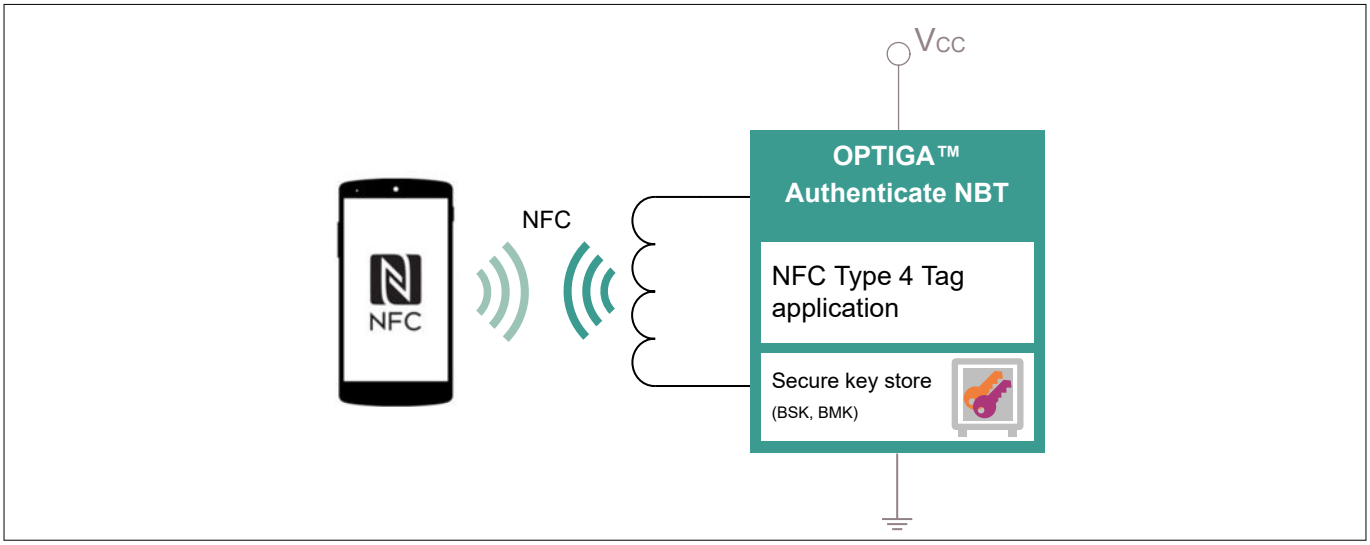


Figure 3 NFC-only tag - brand protection

In brand protection use cases, the device will typically operate as an NFC-only tag. However, using the functionality via the I2C interface in an embedded setup is possible and not restricted to the NFC interface.

The procedures to enable the device for these use cases and to update OPTIGA™ Authenticate NBT's NDEF message are explained in the following sections.

3.2 Operation modes

Brand protection methods encompass a range of strategies and techniques designed to safeguard for example a company's brand identity from misuse, counterfeiting, infringement, intellectual property, and market

presence. When using NFC-enabled phones, Near Field Communication technology is involved to authenticate and to protect products from counterfeiting and engages end consumers in brand interactions.

3.3 Personalization

The following chapter describes how to configure the OPTIGA™ Authenticate NBT for the brand protection (online and offline) use cases.

3.3.1 Device target state

In order to use the OPTIGA™ Authenticate NBT for brand protection applications, following interface configurations must be applied:

- The NFC interface needs to be enabled
- The I2C/IRQ settings can remain unchanged

Interface settings	I2C interface	Enabled
	NFC interface	Enabled
IRQ settings	I2C-IRQ	Disabled
	PT-IRQ	Disabled
	NFC-IRQ	Disabled

Figure 4 Interface configuration for the brand protection use cases

The Type 4 Tag application's NDEF file of the OPTIGA™ Authenticate NBT is utilized in brand protection use cases. NFC-enabled mobile phones, based on Android and iOS, are natively accessing and interpreting the content of the NDEF message file. This functionality is used for online and offline brand protection use cases.

***Note**: It is recommended to prevent the update of the NDEF message by the end users after the personalization (for both brand protection schemes) to avoid modification of the data in the field.*

At the same time, the proprietary files of the OPTIGA™ Authenticate NBT are not intended to be used in brand protection use cases. Thus, the access can be inhibited by mean of the File Access Policy.

Figure 5 shows an example device configuration for brand protection use cases. The following list summarizes the relevant settings for this example.

- The NDEF file can be read via the NFC interface
- After the personalization, the NDEF file cannot be written through the NFC interface anymore
- Access to the NDEF file from the I2C interface is blocked
- Proprietary files (FILE_1, FILE_2, FILE_3 and FILE_4) are not used, no reading or writing is permitted

Type 4 Tag application file	File Access Policy file				Capability Container file				NDEF message file				FILE_1				FILE_2				FILE_3				FILE_4			
File usage/content	Type 4 Tag application file				References to Type 4 Tage files				Infineon URL and certificate				<emtpy>				<emtpy>				<empty>				<empty>			
Operation	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update
Access condition value for BP	N	N	N	N	N	N	A	N	N	N	A	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N

Access conditions: A = ALWAYS; N = NEVER; P = PASSWORD REQUIRED

Figure 5 Target configuration for brand protection

3.3.2 Utilized interfaces

In PERSONALIZATION life cycle state, the OPTIGA™ Authenticate NBT can be configured via both interfaces, I2C and NFC.

3.3.3 Personalization procedure

Figure 6 depicts the standard personalization procedure.

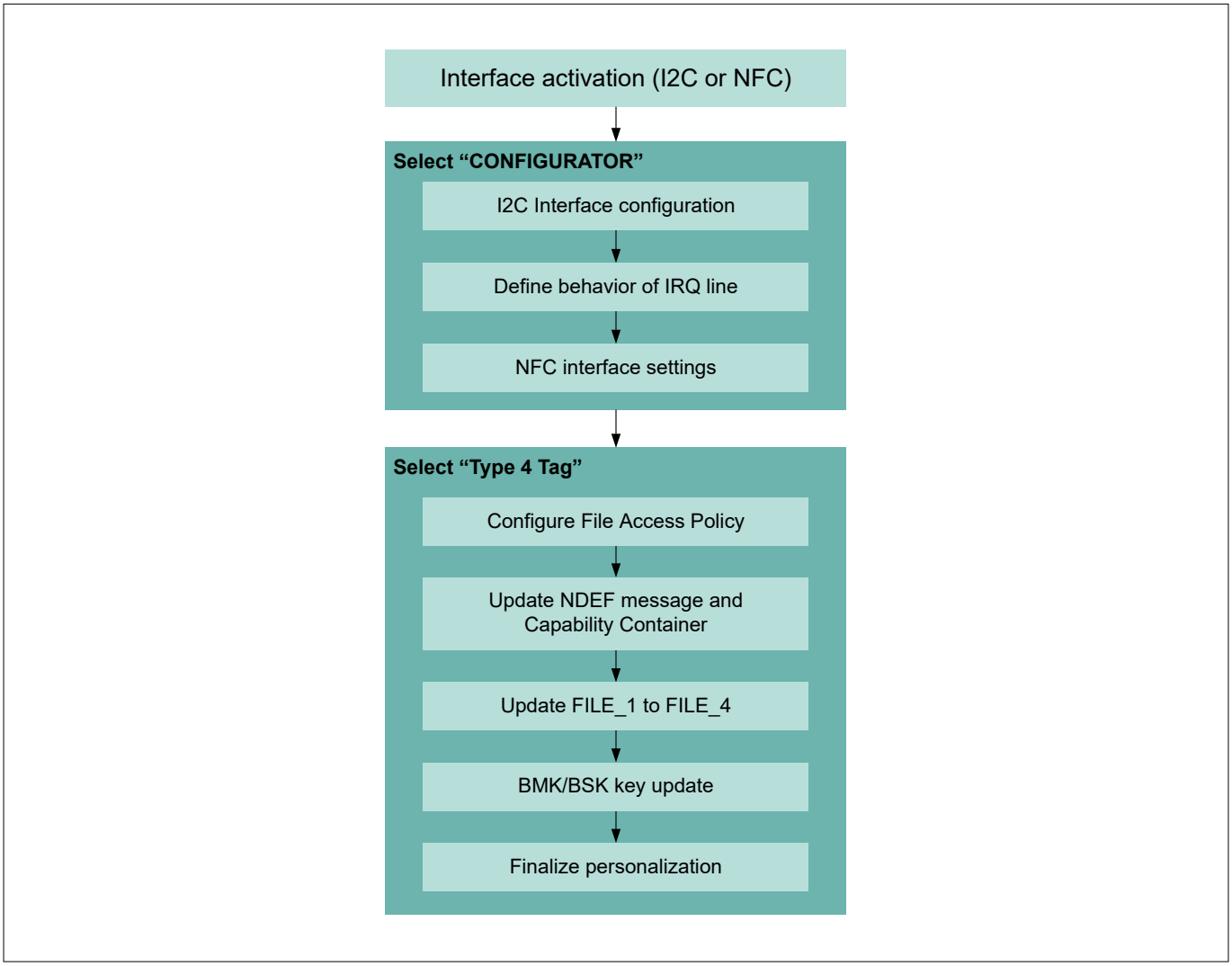


Figure 6 Standard personalization procedure

The personalization of OPTIGA™ Authenticate NBT can be executed via the I2C interface (from a host MCU) or the NFC interface (from an NFC-enabled mobile phone). It is recommended to perform interface-related configurations via the CONFIGURATOR application once the preferred interface is activated.

Subsequently, the Type 4 Tag application's file contents (for example, application-related data in the NDEF file) should be changed, file access conditions (in the EF.FAP file) can be updated accordingly and key values must be exchanged to application- and/or customer-specific values.

The last step in this sequence is to activate the OPERATIONAL state to finalize the preparation of OPTIGA™ Authenticate NBT for the usage in the field.

***Note**: Infineon Technologies provides the implementation of example applications for mobile phones (iOS and Android) to personalize the OPTIGA™ Authenticate NBT for certain use cases. As these applications are shared as full source code, they can be easily modified and extended to custom personalization schemes [5].*

The sequence utilizing a mobile phone, in an NFC-only tag setup, is illustrated in Figure 7.

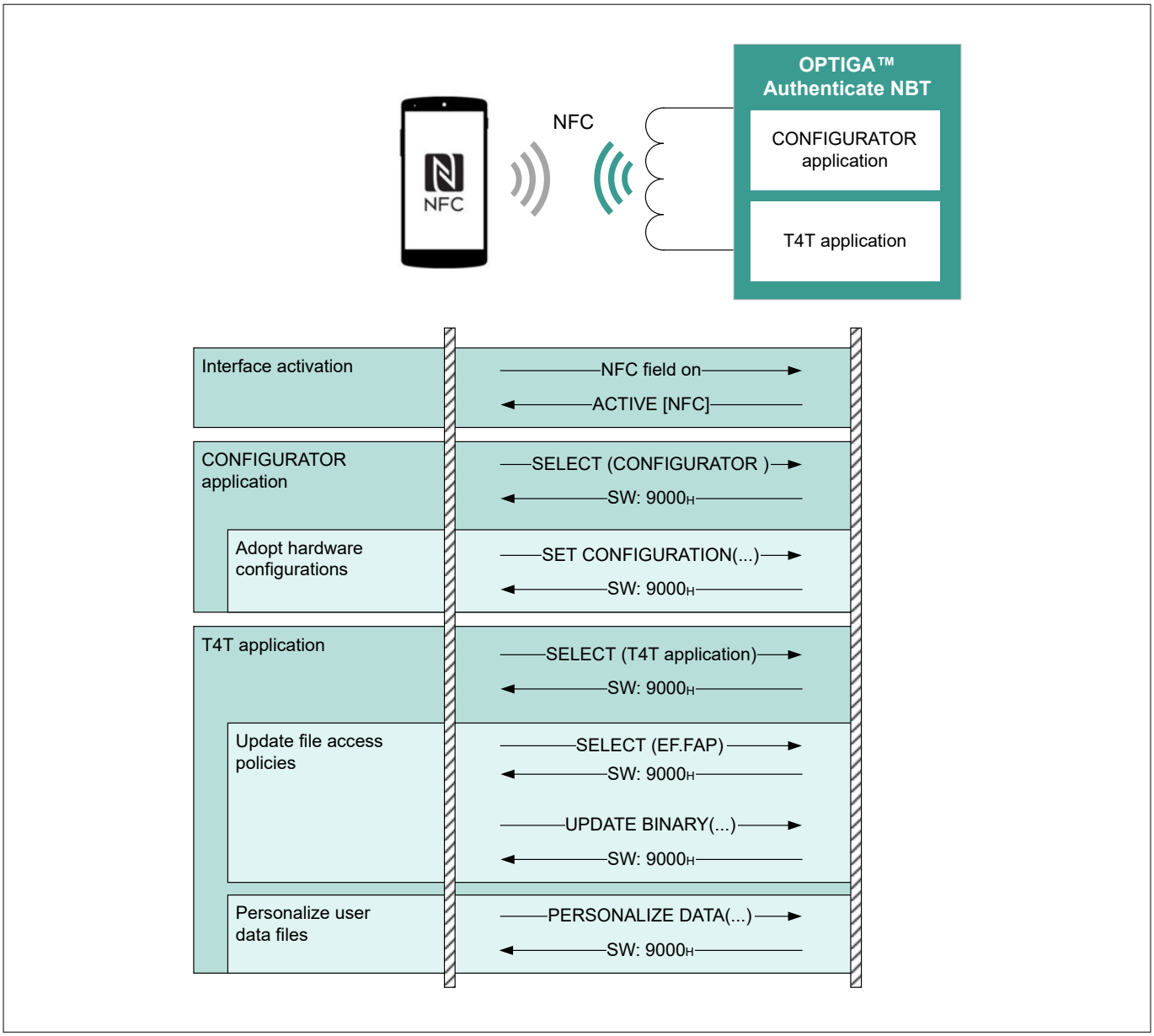


Figure 7 Personalization procedure example via NFC interface using a mobile phone

3.3.3.1 Interface configurations

The configuration of the OPTIGA™ Authenticate NBT at delivery provides appropriate interface settings to be used in the brand protection scenarios. The NFC interface is enabled and the other default interface settings are not interfering with the brand protection use cases.

3.3.3.2 Type 4 Tag application's file configurations

After selecting the Type 4 Tag application of the OPTIGA™ Authenticate NBT, there are two ways to personalize the application file content. The implementer may select the preferred method that is most efficient in the development and/or production environment.

1. Standardized method using the UPDATE BINARY command
- The targeted file needs to be selected before its content can be accessed (SELECT file command)
- Even in the PERSONALIZATION state, file access conditions as set in the EF.FAP must be satisfied
  - Setting of proper FAP may be required in advance, otherwise updating may be denied

- BMK and BSK keys cannot be updated with this method
- Updates of file access conditions within the EF.FAP file need to be done for each of the application's file separately

2. Proprietary method using the PERSONALIZE DATA command

- No dedicated file selection required
- Exclusive method for updating BMK/BSK keys
- The update of the file access conditions for all application files are possible with a single command

The file access policy should be updated to define the per-file and the per-interface access rights for the application files. It is essential that the access right settings for the NDEF file (via the NDEF-File_CTRL_TLV) and the proprietary files (via the Proprietary-File_CTRL_TLV(s)) in the EF.CC file match the FAP configuration for the respective files. The OPTIGA™ Authenticate NBT keeps these setting in sync for the NDEF-File_CTRL_TLV, but application developers need to update these values for the Proprietary-File_CTRL_TLVs. If the data is not matching, this may result in non-compliance with the NFC Forum T4T Specification [1].

***Note**: The EF.CC settings only impact access from the NFC interface, while the FAP settings affect access from both interfaces supported by OPTIGA™ Authenticate NBT.*

In the following example, the access rights for the NDEF file (FileID: $E104_H$) allow read access via the NFC interface, whereas the update of data is blocked. Access to the proprietary files (FileIDs: $E1A1_H$, $E1A2_H$,$E1A3_H$, and$E1A4_H$) is prohibited as they are not used in the application.

Table 1 provides details about the access rights settings for the application files of the OPTIGA™ Authenticate NBT in a brand protection application (online and offline).

Table 1 EF.CC (relevant for access via the NFC interface)

Tag	Length	FID	Size	READ	WRITE	Description
04_H	06_H	$E104_H$	1000_H	00_H	FF_H	NFC read: Yes; NFC write: No
05_H	06_H	$E1A1_H$	0400_H	FF_H	FF_H	No NFC access at all (no read, no write)
05_H	06_H	$E1A2_H$	0400_H	FF_H	FF_H	No NFC access at all (no read, no write)
05_H	06_H	$E1A3_H$	0400_H	FF_H	FF_H	No NFC access at all (no read, no write)
05_H	06_H	$E1A4_H$	0400_H	FF_H	FF_H	No NFC access at all (no read, no write)

The EF.FAP settings shown an Table 2 orchestrate the access from both interfaces. These FAP settings prevent any file access from the I2C interface.

Table 2 EF.FAP (example FAP settings)

FID	I2C_Read	I2C_Update	NFC_Read	NFC_Update	Description
$E103_H$	00_H	00_H	40_H	00_H	NFC read; no I2C access
$E104_H$	00_H	00_H	40_H	00_H	NFC read; no I2C access
$E1A1_H$	00_H	00_H	00_H	00_H	No NFC access, no I2C access
$E1A2_H$	00_H	00_H	00_H	00_H	No NFC access, no I2C access
$E1A3_H$	00_H	00_H	00_H	00_H	No NFC access; no I2C access
$E1A4_H$	00_H	00_H	00_H	00_H	No NFC access; no I2C access
$E1AF_H$	00_H	00_H	00_H	00_H	No NFC access; no I2C access

3.3.3.3 Brand protection with offline authentication

In its delivery condition, the OPTIGA™ Authenticate NBT includes an NDEF message containing an Infineon X.509v3 DER-encoded certificate. This device-individual certificate is issued by an Infineon PKI and it facilitates the originality check: the certificate contains the public key that corresponds to the device's pre-installed brand protection signing key (BSK, EC P-256 private key).

Henceforth, the public key can be used to validate a signature computed by the production-default BSK. The certificate subject common name (CN) contains the device's unique 7-byte NFC UID (TAG-UID), which is encoded as a hex string with no delimiters (for example, see Figure 8: "05848902954300"). This TAG-UID corresponds to the NFC UID used by the device during anti-collision.

```
        Certificate:
            Data:
                Version: 3 (0x2)
                Serial Number: 1836660215 (0x6d7935f7)
                Signature Algorithm: ecdsa-with-SHA384
                Issuer: C = DE, O = Infineon Technologies AG, OU = NFC bridge & CL tag devices,
                        CN = NFC bridge & CL tag devices manufacturing CA
                Validity
                    Not Before: Aug 28 13:38:43 2023 GMT
                    Not After : Aug 28 13:38:43 2043 GMT
                Subject: C = DE, O = Infineon Technologies AG, OU = NFC bridge & CL tag devices,
                         CN = 05848902954300
                Subject Public Key Info:
                    Public Key Algorithm: id-ecPublicKey
                        Public-Key: (256 bit)
                        pub:
                            04:d4:d2:da:90:1d:ea:42:37:3a:48:ec:a7:d9:f6:
                            d1:09:44:0e:a7:0a:24:20:6d:cd:76:24:56:60:ba:
                            77:83:1d:e3:8b:77:76:d4:b3:b9:30:37:e7:81:c9:
                            fd:f9:9f:c8:04:a8:47:97:d6:81:03:6b:bc:8e:b2:
                            04:39:c3:de:aa
                        ASN1 OID: prime256v1
                        NIST CURVE: P-256
                X509v3 extensions:
                    X509v3 Subject Key Identifier:
                        6C:55:2F:86:10:AB:DE:28:26:42:73:17:A0:E8:13:61:67:E0:F4:3A
                    X509v3 Key Usage: critical
                        Digital Signature
                    X509v3 Basic Constraints: critical
                        CA:FALSE
                    X509v3 Authority Key Identifier:
                        keyid:C4:3F:73:A2:1C:3B:66:BB:57:E2:89:ED:1B:72:03:7C:73:C0:58:68

            Signature Algorithm: ecdsa-with-SHA384
                 30:63:02:2f:50:cb:43:f1:36:fc:a2:4c:4b:73:df:1a:6e:84:
                 ec:d6:78:f0:2f:7d:cb:7d:20:2f:9a:fa:a0:cf:11:10:3d:b8:
                 13:a9:eb:d6:2f:4b:23:05:a7:56:75:9d:cc:42:bc:02:30:01:
                 e7:ae:66:8a:ff:0f:b5:0c:19:a6:e8:65:42:7a:7d:db:ca:1b:
                 84:34:ec:4c:55:98:0e:58:4d:95:bf:bb:58:59:dd:a3:f8:b5:
                 4e:0a:c7:38:e2:6f:14:58:71:4a:e5
```

Figure 8 Infineon X.509v3 device certificate

For demonstration purposes, this pre-installed certificate is used to implement an offline brand protection scheme. The scheme will then be based on the Infineon PKI and any Infineon OPTIGA™ Authenticate NBT with an original, pre-installed certificate will be seen as an authentic device.

In customer-specific brand protection schemes, it is recommended to load device-specific key pairs using the customer's own PKI. The customer-created X.509 certificate containing the device-specific public key needs to be embedded into an external record which gets loaded into the NDEF message. Additionally, the device-specific private key (BSK) needs to be loaded into OPTIGA™ Authenticate NBT's key store. For more information about the composition of this external record refer to the default configuration of the NDEF file in Figure 19.

***Note**: Infineon does not provide any recommendations to set up customer-specific root/intermediary certificate authorities.*

3.3.3.4 Brand protection with online authentication

In its delivery condition, the OPTIGA™ Authenticate NBT includes an NDEF message containing an URI record. This record contains an URL pointing to the Infineon website (https://www.infineon.com/?cott=) as well as the COTT placeholder string (refer to Figure 18).

The initial URI NDEF record, including the COTT placeholder string looks as follows:

https://www.infineon.com/?cott=PLACEHOLDERPLACEHOLDERPLACEHOLDERPLACEHOLDER

A new Cryptographic One-Time Token is computed each time the NDEF message file is selected on the OPTIGA™ Authenticate NBT. A Message Authentication Code (MAC) is computed by utilizing chip-individual information (TAG-UID), a header byte, and a random number from the tag (TAG-RANDOM, generated each time the NDEF file is selected) as inputs. The AES-128-CMAC calculation uses the AES-128 BMK from the device's secure key store. The input data and the calculated MAC over this data are then merged and a base64url-encoded value is inserted into the URL replacing the COTT placeholder string.

Example: https://www.infineon.com/?cott=AQWPP9nUsgBNpDlMyEa1kWZR1FEQ6mBuxXmASo261wU=

OEMs personalizing the OPTIGA™ Authenticate NBT to use the device in an online brand protection application, shall update the NDEF message by exchanging the URL to an OEM-specific address. This field needs to be encoded accordingly within the URI record (see Figure 18). Additionally, the initial MAC'ing key BMK needs to be exchanged in the device's secure key store to a device-specific, OEM-defined value.

3.3.3.5 Activating the OPERATIONAL life cycle state

The OPTIGA™ Authenticate NBT state transition from the PERSONALIZATION to the OPERATIONAL life cycle state is triggered by the following sequence:

- Selecting the CONFIGURATOR application
- Sending the DGI "FINALIZE PERSONALIZATION" embedded either into a SET CONFIGURATION or a PERSONALIZE DATA command (refer to Extended Datasheet [6])

***Note**: This step may be skipped during the development phase to allow the developer to make several optimization attempts.*

3.4 Operational use case

After activating the OPERATIONAL life cycle state, the OPTIGA™ Authenticate NBT is intended to be operated from the NFC interface to execute the brand protection scheme. For this purpose, the following components are required:

- An OPTIGA™ Authenticate NBT connected to an NFC antenna, either in the NFC-only tag setup or embedded into a host system with a connection to a host MCU
- An NFC reader device, for example, an NFC-enabled mobile phone executing a dedicated application (offline brand protection method) using the Root CA public key for validation
- For the online brand protection method, a web service covering a cloud authentication service is required

3.4.1 Operational flow example: Brand protection with offline authentication

When the OPTIGA™ Authenticate NBT is used in an offline brand protection application, a mobile phone application validates, for example, a branded luxury item using public-key cryptography and public key infrastructure. Using a trusted CA certificate, the PKI enables the mobile phone to perform local verification.

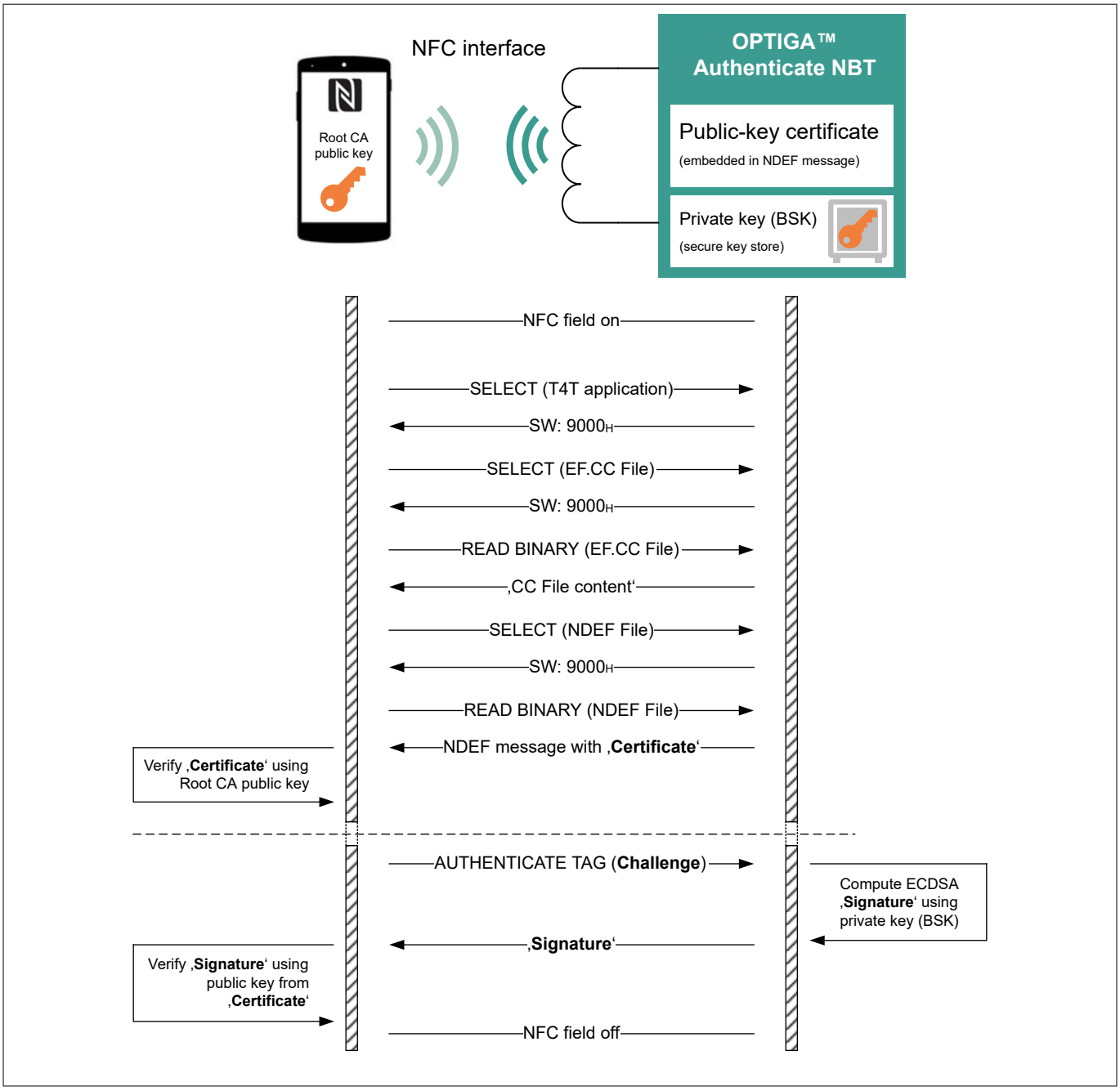


Figure 9 Offline brand protection – operational flow

The high-level flow of the use case is as follows:

1. The end user launches the mobile phone application (with the offline brand protection method implemented) and taps it to the OPTIGA™ Authenticate NBT equipped NFC tag
2. The application on the mobile phone reads the tag's ID and extracts the public-key certificate from the NDEF message
3. Using the available Root CA public key, the phone validates the public-key certificate
4. The phone authenticates an active tag by sending a challenge and retrieving the public-key signature
5. The phone displays the authentication results and turns the NFC field off

3.4.2 Operational flow example: Brand protection with online authentication

The operational flow of the online authentication does not require a dedicated application on the mobile phone. When an end user taps the mobile phone to the tag attached to the branded luxury item, the mobile phone selects and reads the NDEF message of the OPTIGA™ Authenticate NBT's Type 4 Tag application. The NDEF message contains an URL, pointing to the brand's web portal for remote verification via a cloud authentication service. The phone's web browser uses this link to connect to this online service.

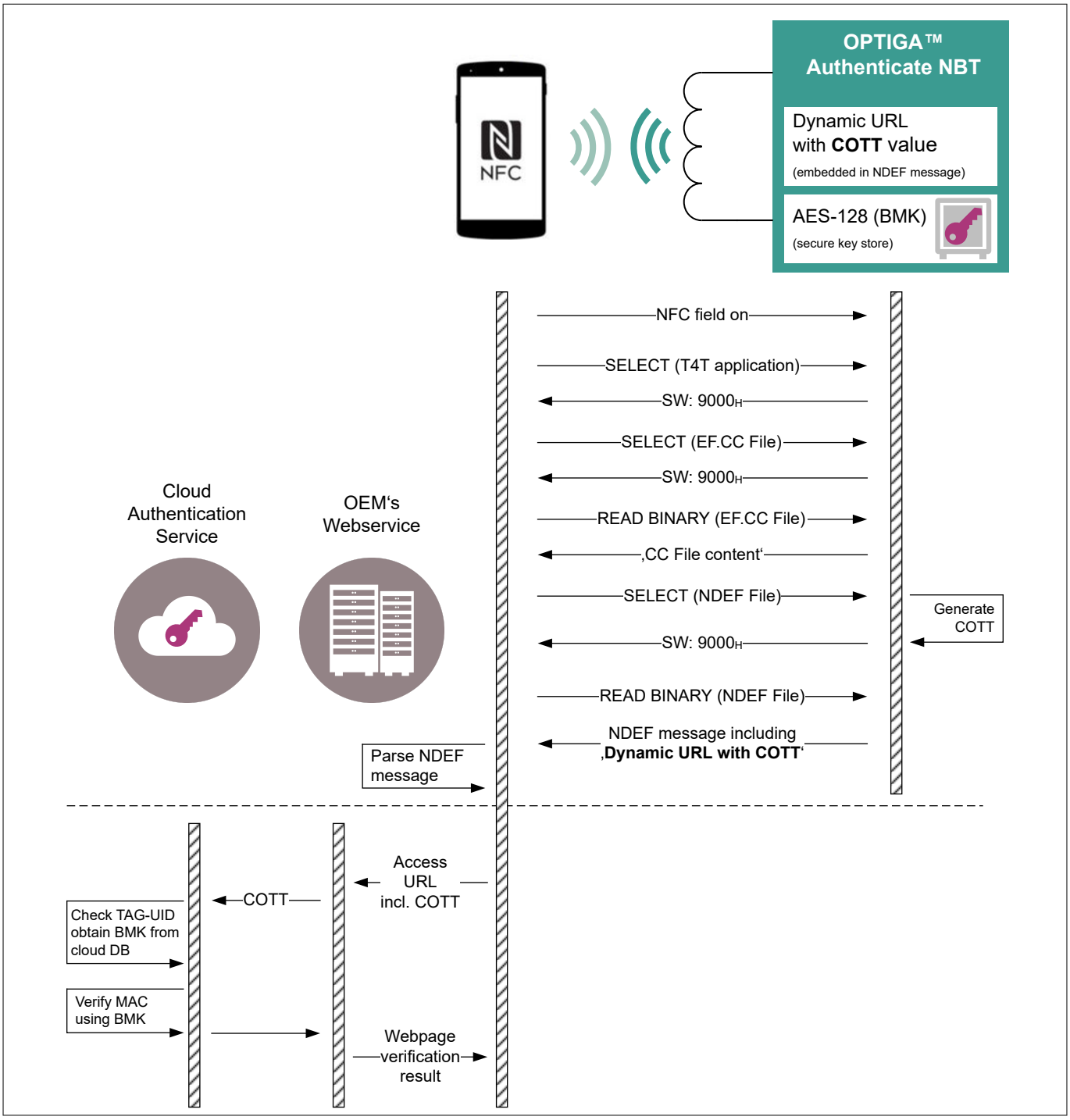


Figure 10 Online brand protection – operational flow

The high-level flow of the use case is as follows:

1. The end user taps the NFC tag with the mobile phone

2. The mobile phone accesses the tag's NFC application. Therein, it selects the NDEF message, which triggers the OPTIGA™ Authenticate NBT to generate a new COTT value using the tag's BMK. The device then returns this dynamically generated URL, containing a 44 bytes base64url-encoded COTT value as part of this URI record when the NDEF message is read
3. Using the phone's web browser, the phone connects to the URL, including the COTT
4. The OEM's web service at the specified URL receives the request, parses the response, extracts the COTT value from the URL and forwards this to the OEM's cloud authentication service
5. The authentication service executes the following steps:
   a. Decoding the base64url-encoded COTT
   b. Parse the COTT information (extract header, TAG-UID, TAG-RANDOM and MAC)
   c. To avoid replay attacks, at this point, the web service may check if the combination of COTT information is already used before
   d. Based on the TAG-UID, the web service retrieves the corresponding BMK from its database
   e. Calculation of the AES-128-CMAC over header, TAG-UID and TAG-RANDOM using the BMK
   f. Compare the MAC from the COTT value of the request URL with the calculated AES-128-CMAC
6. The OEM's web service displays a web page and reports the verification result on the mobile phone's web browser to the end user. Additional information about the tagged item may be shared to the customer

A Appendix

The following section cover technical information about the OPTIGA™ Authenticate NBT, including its features and specifics relating to the product delivery condition. This summary can serve as a starting point to prepare the device for its intended use case.

A.1 Technical background

A brief overview of the OPTIGA™ Authenticate NBT features can be found in following sections. This covers basic information on hardware interconnection scenarios, descriptions of the available communication interfaces, a short introduction of the product architecture including important functional blocks as well as a command reference which is used to personalize and to operate the device.

A.1.1 OPTIGA™ Authenticate NBT system architecture

The OPTIGA™ Authenticate NBT is delivered with the following selectable applications:

- CONFIGURATOR application: Used to modify the device's hardware-related settings or configuration such as interface settings, IRQ behavior, life cycle state, and additional settings
- Type 4 Tag application: Contains the EF.CC (Capability Container file), the NDEF file, proprietary "mailbox" files, and the EF.FAP (File Access Policy file)
- Pass-through application: This "virtual" application allows to transfer bigger amount of data between an NFC reader device and a host. The device manages the NFC protocol in terms of framing, timing, and waiting time extensions during the exchange of application commands

The OPTIGA™ Authenticate NBT utilizes a protected key storage to store the BSK (Brand Protection Signing Key) and the BMK (Brand Protection MAC'ing Key). Furthermore, the passwords used to manage the access to the application files are saved in a secured memory area.

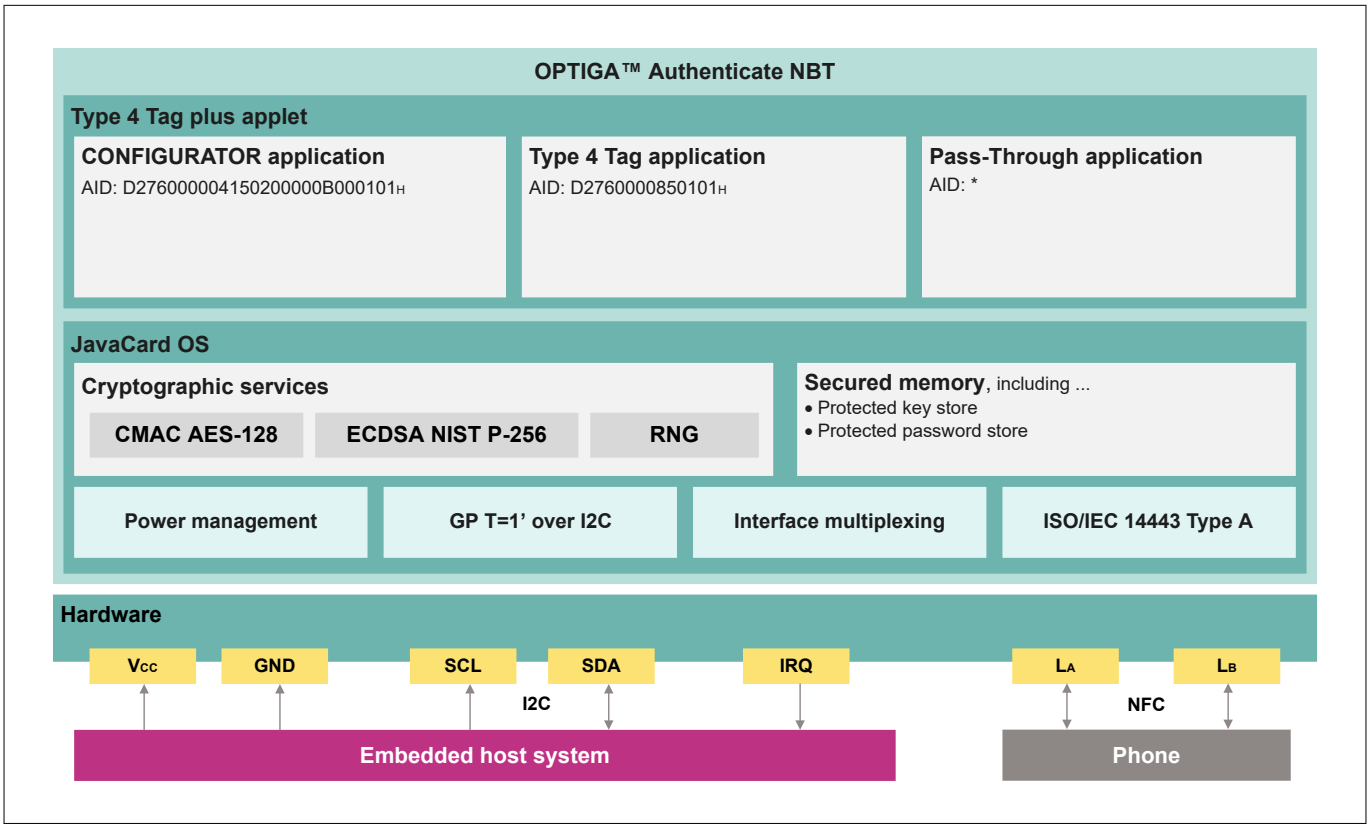


Figure 11 OPTIGA™ Authenticate NBT product architecture

Table 3 Supported applications of the OPTIGA™ Authenticate NBT

Application ID (AID)	Application	Functionality
D2 76 00 00 04 15 02 00 00 0B 00 01 01_H	CONFIGURATOR	Interface configurations
D2 76 00 00 85 01 01_H	Type 4 Tag	NFC Forum Type 4 Tag
Any other (length: 5 to 16 Bytes)	Pass-through	NFC to I2C Bridge Tag

The CONFIGURATOR application controls the OPTIGA™ Authenticate NBT hardware configuration as described in Chapter 4 of the Extended Datasheet [6]. The pass-through application is a "virtual" application that can be activated by attempting to select an application with an AID, which is not used by the CONFIGURATOR or the Type 4 Tag application.

The Type 4 Tag application adheres to the NFC Forum T4T Specification [1]. In addition, the OPTIGA™ Authenticate NBT's Type 4 Tag application contains four proprietary files (FILE_1 to FILE_4) as well as the File Access Policy file (EF.FAP).

All files in the Type 4 Tag application are accessible from both interfaces (NFC and I2C). Password-based file access rights can be configured to restrict access per-file and per-interface basis. This is accomplished by updating the relevant fields in the EF.FAP file during personalization. Furthermore, the Type 4 Tag application supports the management of each file's content as well as the secure key store. Before reading or modifying file contents, the corresponding application data file must be selected by its FileID using the SELECT file command.

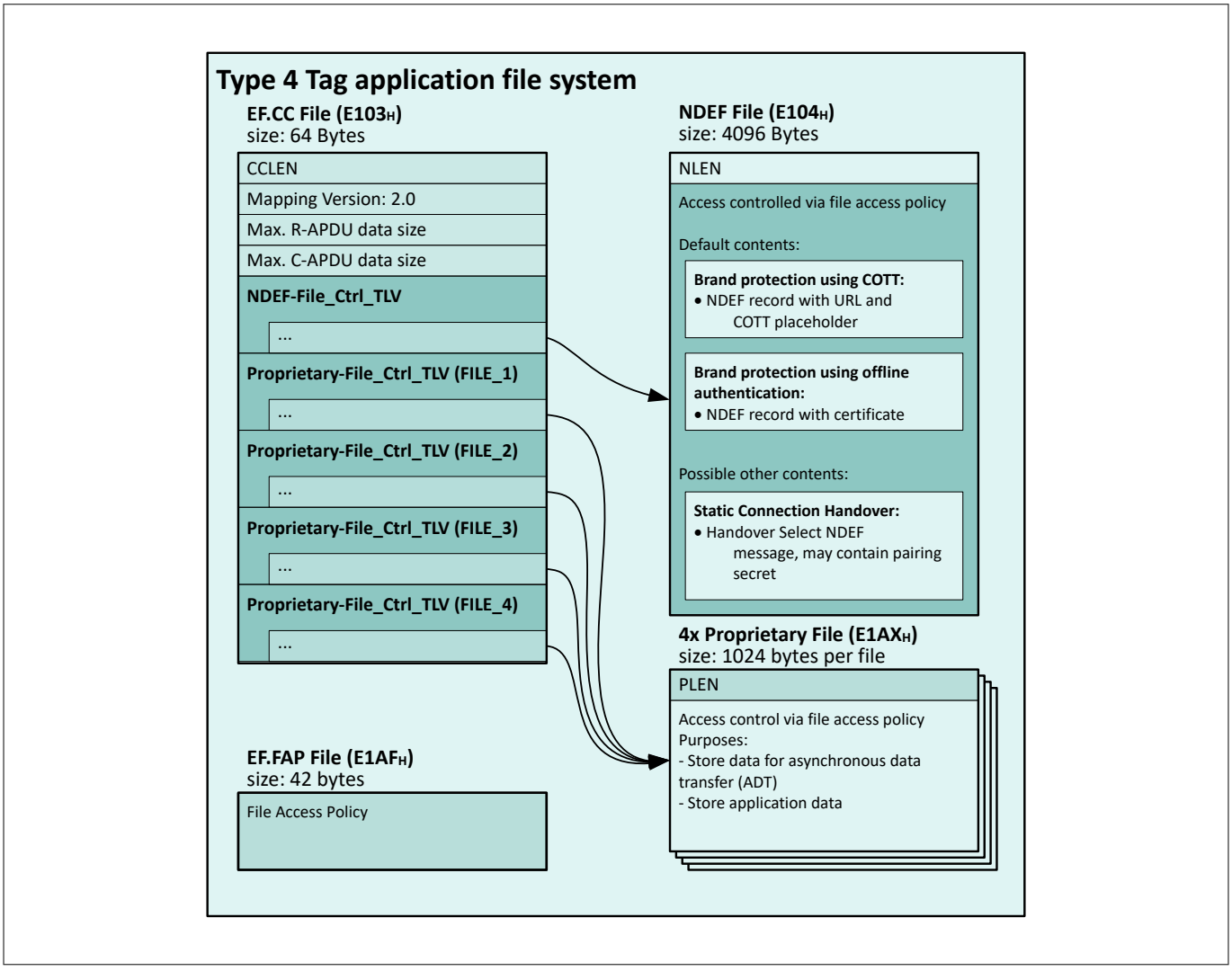


Figure 12 Type 4 Tag file structure

Table 4 Type 4 Tag application and files

File	FileID	Size [bytes]	Content
EF.CC	$E103_H$	64	Size and access policy of • NDEF file • FILE_1 to FILE_4
NDEF	$E104_H$	4096	NDEF message
FILE_1	$E1A1_H$	1024	Proprietary
FILE_2	$E1A2_H$	1024	Proprietary
FILE_3	$E1A3_H$	1024	Proprietary
FILE_4	$E1A4_H$	1024	Proprietary
EF.FAP	$E1AF_H$	42	Definition of access rights to • EF.CC file • NDEF file • FILE_1 to FILE_4 • EF.FAP file

A.1.2 Hardware configuration

In an embedded tag setup, the OPTIGA™ Authenticate NBT is integrated into the system via the following external connections:

- L_A and L_B are connected to an NFC antenna
- The device is externally supplied via V_{CC} and GND pins
- I2C host interface via SDA, SCL, and IRQ

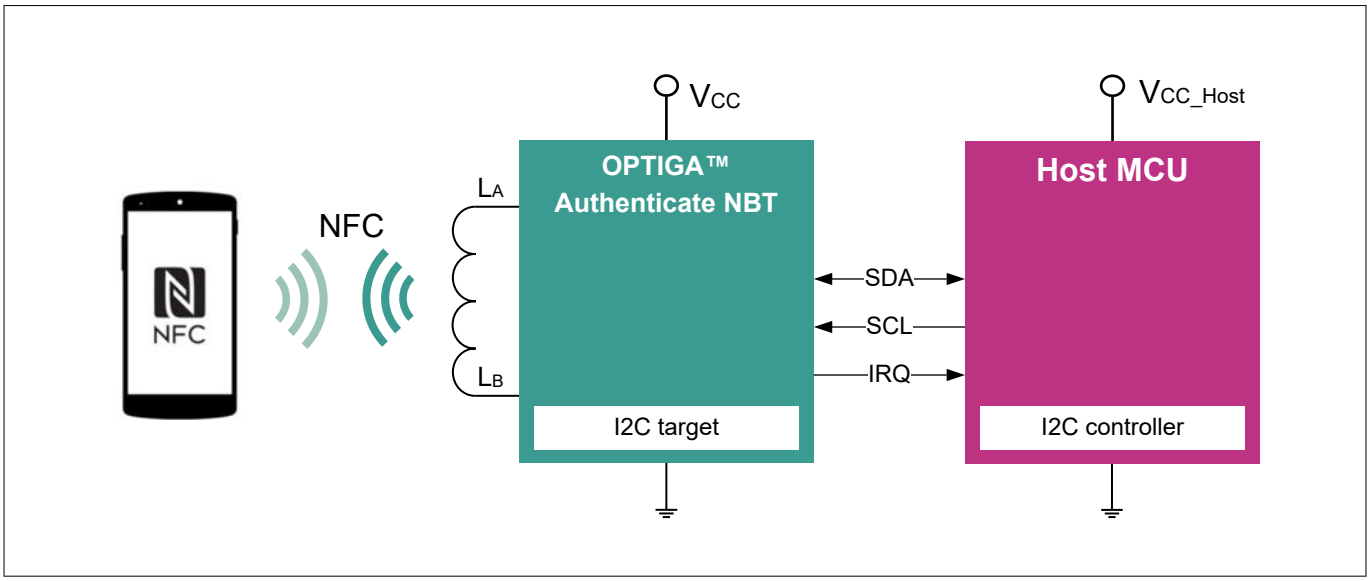


Figure 13 Embedded tag

Alternatively, the OPTIGA™ Authenticate NBT can be used as a stand-alone NFC-only tag, where the NFC-enabled phone may retrieve the connection handover message from the NDEF file. In this configuration, the device is connected to an NFC antenna via its L_A and L_B pins. Optionally, the device may be powered through its V_{CC} and GND pins, which extends the contactless communication distance.

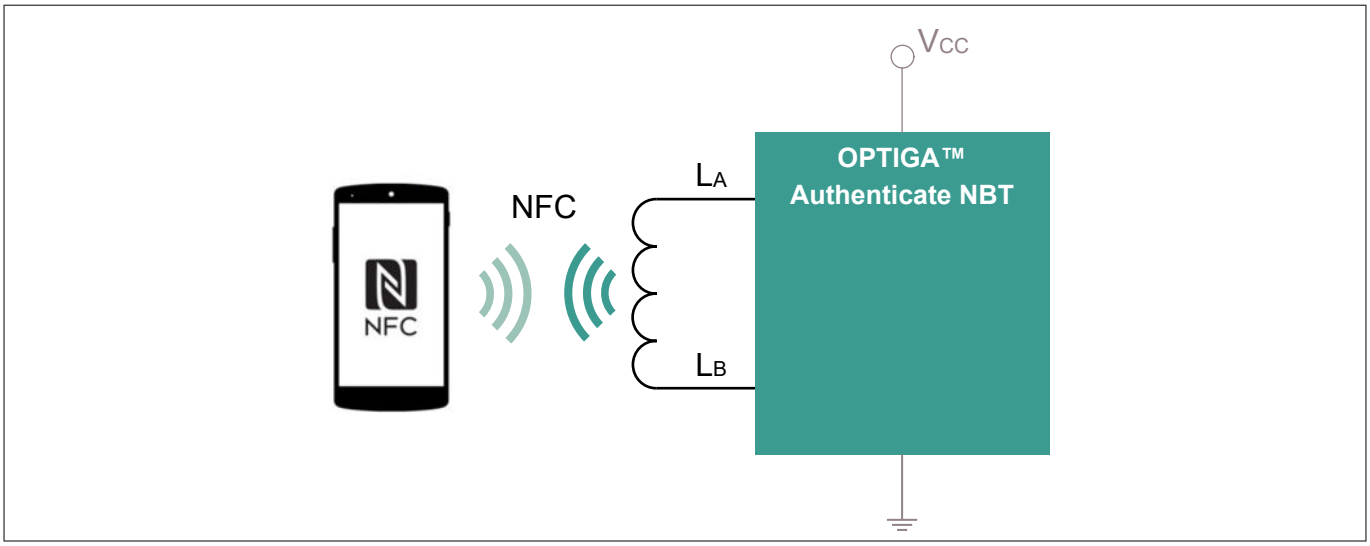


Figure 14 NFC-only tag

A.1.3 Interface description

The OPTIGA™ Authenticate NBT includes an NFC interface as well as an I2C target interface. In the NFC-only tag scenario, for example, the NFC interface of device is physically connected to an external antenna. In an embedded tag hardware setup, the device is powered through its V_{CC} and GND pins from an external source. In this setup, the SCL and SDA lines can also be connected to an host MCU to exchange data via the I2C interface.

The IRQ line of OPTIGA™ Authenticate NBT can directly be connected to one GPIO of the I2C controller MCU (host MCU). Then it must be configured as interrupt pin to support the implementation of the protocol according to Global Platform T=1' I2C specification.

Figure 15 depicts the logical communication states of the OPTIGA™ Authenticate NBT, including state transitions and the events triggering these. Once an interface is activated (either NFC or I2C), the device is locked into that interface until it is released (by field off or a timeout).

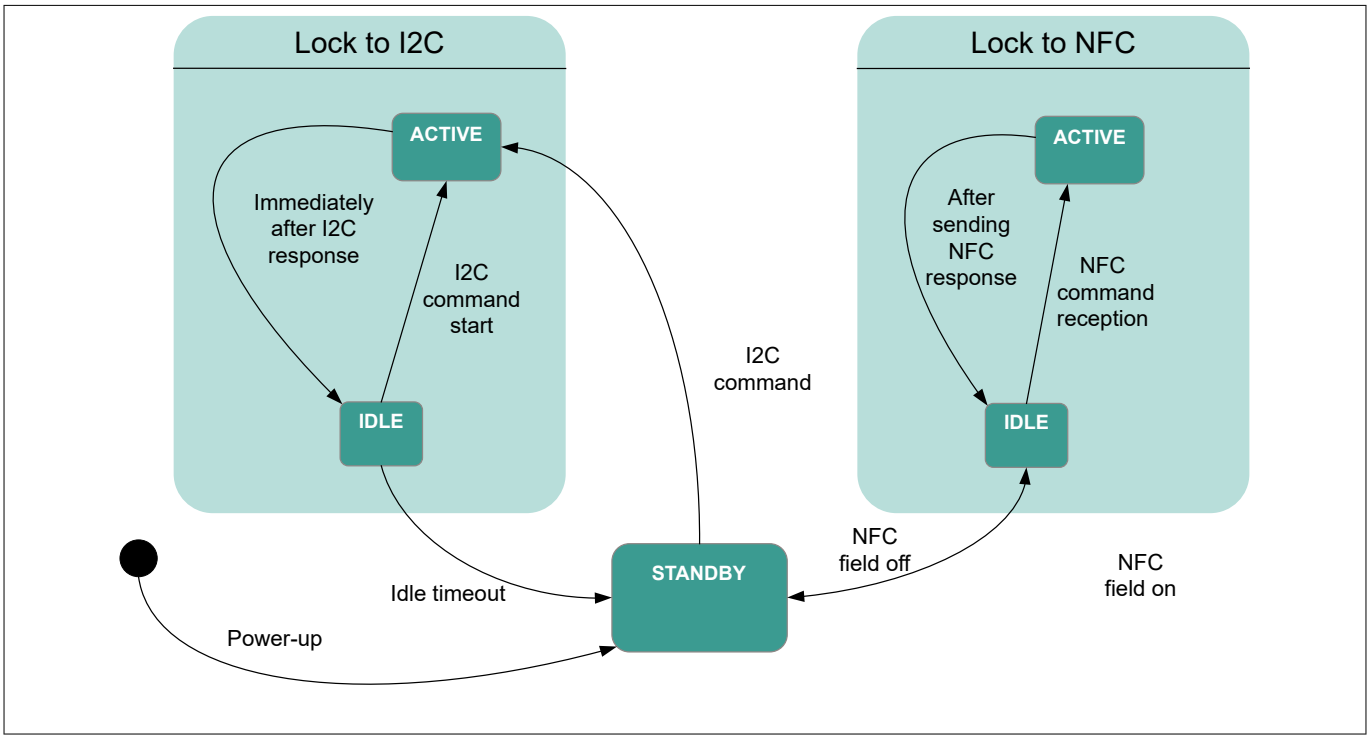


Figure 15 Logical communication states of OPTIGA™ Authenticate NBT

A.1.4 Command reference

The list of commands to personalize the OPTIGA™ Authenticate NBT for a use case and to operate the device in this application is provided in Table 5. Moreover, the table specifies the acceptance of each command, depending on the product life cycle state.

Table 5 Command set of the OPTIGA™ Authenticate NBT

Command	CLA	INS	Application	PERSONALIZATION	OPERATIONAL
SELECT (application)	00_H	$A4_H$	Type 4 Tag CONFIGURATOR	✓	✓
SELECT (file)	00_H	$A4_H$	Type 4 Tag	✓	✓
READ BINARY	00_H	$B0_H$	Type 4 Tag	✓	✓
UPDATE BINARY	00_H	$D6_H$	Type 4 Tag	✓	✓
PERSONALIZE DATA	00_H	$E2_H$	Type 4 Tag	✓	x
CHANGE/UNBLOCK PASSWORD	00_H	24_H	Type 4 Tag	✓	✓
AUTHENTICATE TAG	00_H	88_H	Type 4 Tag	✓	✓
GET CONFIGURATION	20_H	30_H	CONFIGURATOR	✓	x
SET CONFIGURATION	20_H	20_H	CONFIGURATOR	✓	x

A.1.5 Life cycle states

The OPTIGA™ Authenticate NBT supports two life cycle states as described in the Extended Datasheet [6]:

- PERSONALIZATION state: The product will be in the PERSONALIZATION state at the time of delivery. In this life cycle state, application developers can unconditionally modify the specific settings to prepare the device for the targeted use case. This covers:
  - Interface configurations
  - File access conditions and passwords
  - File content
  - Cryptographic keys

  ***Note**: When the product configuration and the data personalization steps are finished, it is recommended to switch the OPTIGA™ Authenticate NBT to the OPERATIONAL life cycle state to prevent unintended changes during the usage*

- OPERATIONAL state: In this state, the device is ready to be operated in the target application scenario. Product configuration functions are disabled. Configured file access policies prevent unverified operations on the file (based on the use case configuration)

  ***Note**: After the activation of the OPERATIONAL state on the OPTIGA™ Authenticate NBT, the life cycle cannot be restored to PERSONALIZATION state*

A.2 Device delivery condition

The OPTIGA™ Authenticate NBT comes with preloaded CONFIGURATOR and the Type 4 Tag applications. At delivery, the product is set to PERSONALIZATION state and the default configuration of the applications allow unconditional access to the following:

- CONFIGURATOR application
  - To adopt interface settings
  - To set life cycle state to OPERATIONAL
- Type 4 Tag application
  - To modify the File Access Policy (FAP)
  - To modify file content of user data files
  - Execute key exchange of the BSK or BMK

The OPTIGA™ Authenticate NBT is configured with I2C and NFC interfaces enabled. Refer to Extended Datasheet [6] for more details.

Interface settings	I2C interface	Enabled
	NFC interface	Enabled
IRQ settings	I2C-IRQ	Disabled
	PT-IRQ	Disabled
	NFC-IRQ	Disabled

Figure 16 Delivery condition: Interface configuration

The Type 4 Tag application consists of the following seven files:

- Capability Container file (EF.CC)
- NDEF file
- Four proprietary files (FILE_1 to FILE_4)
- File Access Policy file (EF.FAP)

The EF.CC File contains meta information such as the FileID, file size, and access conditions of the NDEF file, and the proprietary files FILE_1 to FILE_4 in the File_CTRL_TLVs. The content is set to the default values described in the Extended Datasheet [6]

The FAP is used to manage the file access conditions on a per-file and per-interface basis. The initial file access conditions are set as shown in Figure 17. The FAP can be updated while the OPTIGA™ Authenticate NBT is in the PERSONALIZATION state.

***Note**: The access conditions for the NFC interface configured in the FAP overrule the FILE_CTRL_TLV settings in the Capability Container. When access conditions defined in the FAP get modified, access conditions in the EF.CC for in the NDEF-File_CTRL_TLV are automatically synchronized by the OPTIGA™ Authenticate NBT, while Proprietary-File_CTRL_TLVs need to be updated by the implementer.*

The NDEF file contains the initial NDEF message, which is described in detail in the following chapter.

Type 4 Tag application file	File Access Policy file				Capability Container file				NDEF message file				FILE_1				FILE_2				FILE_3				FILE_4			
File usage / content	Type 4 Tag application file access settings				References to Type 4 Tag files				Infineon URL and certificate				<empty>				<empty>				<empty>				<empty>			
Operation	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update	I2C_Read	I2C_Update	NFC_Read	NFC_Update
Access condition at delivery	A	A	A	A	A	N	A	N	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A

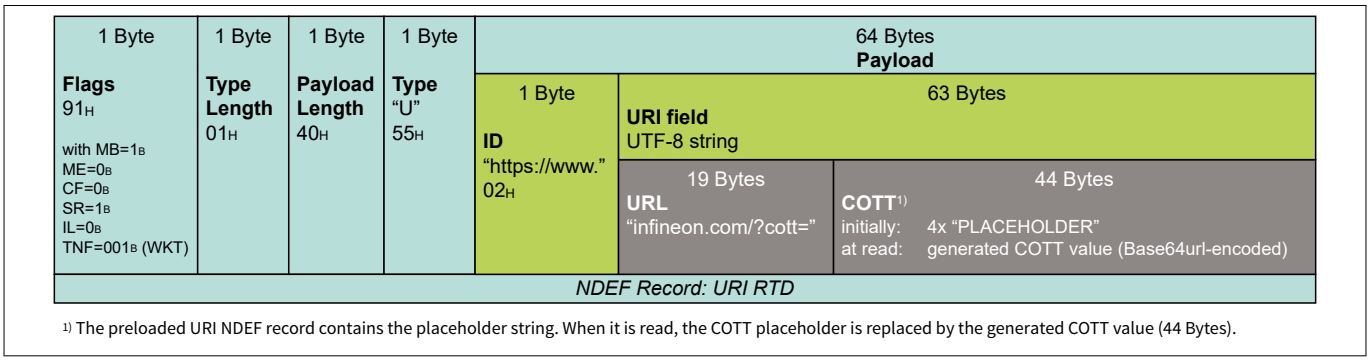
Access conditions: A = ALWAYS; N = NEVER; P = PASSWORD REQUIRED

Figure 17 Delivery condition: Application file content, access conditions (per-file, per-interface)

An AES-128-CMAC key (BMK) is preloaded to support online brand protection applications that use cryptographic one-time tokens. In addition, the OPTIGA™ Authenticate NBT's key store contains a private key for NIST P-256-based one-way authentication (BSK). The corresponding public key is stored inside an X.509 certificate, allowing the chip's authenticity to be checked. The NDEF record containing this certificate is also stored inside the NDEF file.

A.2.1 Initial NDEF message

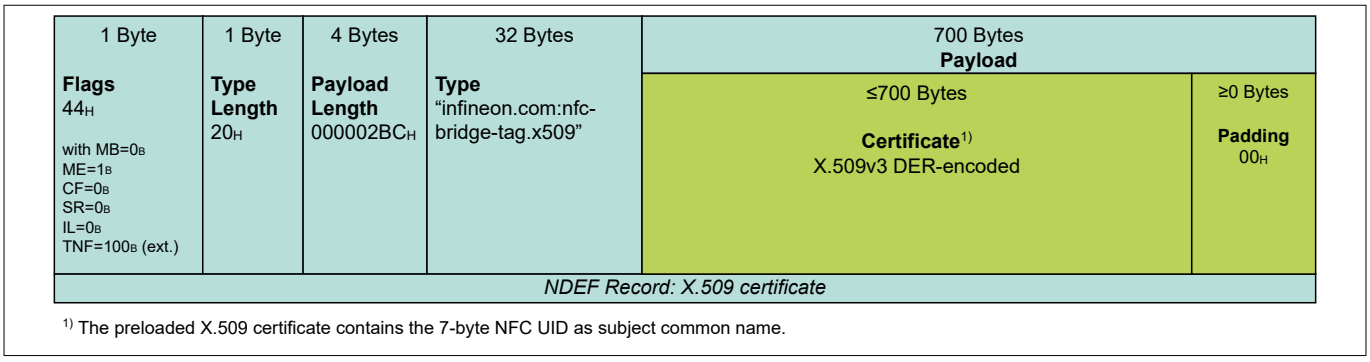
The OPTIGA™ Authenticate NBT is delivered with a preloaded NDEF message in the NDEF file. This NDEF message contains two NDEF records: the first is an URI record followed by an external record. Initially, the URI record contains a link to https://www.infineon.com/, followed by the COTT placeholder string used for online brand protection use cases.



1) The preloaded URI NDEF record contains the placeholder string. When it is read, the COTT placeholder is replaced by the generated COTT value (44 Bytes).

Figure 18 URI record

The external record is essential for the offline brand protection scheme supported by the OPTIGA™ Authenticate NBT. The record includes an X.509v3 DER-encoded public-key certificate generated by Infineon Technologies. During the manufacturing process of the chip, a certificate is created. This certificate contains each chip's individual UID and is generated during wafer-level personalization. It is embedded into the NDEF message's external record. For more information about the certificate, refer to the Appendix in [6].



1) The preloaded X.509 certificate contains the 7-byte NFC UID as subject common name.

Figure 19 External record

References

NFC Forum

[1] NFC Forum: *Type 4 Tag Technical Specification (Version 1.2)*; 2022-08-16
[2] NFC Forum: *NFC Data Exchange Format (NDEF) Technical Specification (Version 1.0);* 2006-07-24
[3] NFC Forum: *Activity Technical Specification (Version 2.3)*; 2023-02-03

GlobalPlatform

[4] GlobalPlatform: *APDU Transport over SPI/I2C (Version 1.0);* 2020-01

Infineon

[5] Infineon Technologies AG: *OPTIGA™ Authenticate NBT, product website* - https://www.infineon.com/OPTIGA-Authenticate-NBT
[6] Infineon Technologies AG: *OPTIGA™ Authenticate NBT, Extended Datasheet (latest revision)*
[7] Infineon Technologies AG: *OPTIGA™ Authenticate NBT, Release Notes (latest revision)*
[8] Infineon Technologies AG: *OPTIGA™ Authenticate NBT, Software Integration Guide (latest revision)*

Glossary

AES

Advanced Encryption Standard (AES)

The standard for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001. The algorithm described by AES is a symmetric-key algorithm (the same key is used for both encryption and decryption).

AID

application identifier (AID)

Used to reference an application.

APDU

application protocol data unit (APDU)
The communication unit between a smart card reader and a smart card.

BMK

brand protection MAC'ing key (BMK)

BSK

brand protection signing key (BSK)

BT

Bluetooth (BT)
A short-range wireless technology standard that is used for exchanging data between fixed and mobile devices over short distances.

C-MAC

command MAC (C-MAC)

CA

certificate authority (CA)

CC

capability container (CC)

CLA

class byte (CLA)

COTT

cryptographic one-time token (COTT)

DGI

data group identifier (DGI)

ECDSA

elliptic curve digital signature algorithm (ECDSA)

FAP

file access policy (FAP)

FID
file identifier (FID)
Used to reference an elementary file.

GND
ground (GND)

GP
GlobalPlatform (GP)

GPIO
general purpose input/output (GPIO)

I2C
inter-integrated circuit (I2C)

ID
identification (ID)

INS
instruction byte (INS)

IRQ
interrupt request (IRQ)
A type of exception that breaks the linear flow of a program. The requesting module needs a software service routine to evaluate its current state and take the necessary actions.

ISO
International Organization for Standardization (ISO)

MCU
microcontroller unit (MCU)
One or more processor cores along with memory and programmable input/output peripherals.

NBT
NFC bridge tags (NBT)

NDEF
NFC data exchange format (NDEF)
A standardized data format specification by the NFC Forum to describe how a set of actions are to be encoded onto a NFC tag or to be exchanged between two active NFC devices.

NFC
near field communication (NFC)

NFCT4T
NFC Type 4 Tag (NFCT4T)

NLEN
NDEF length (NLEN)
A field in the NDEF message that indicates the size of the NDEF message.

OEM

original equipment manufacturer (OEM)

PKI

public key infrastructure (PKI)
A set of roles, policies, hardware, software, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates and manage public-key encryption.

RNG

random number generator (RNG)

RTD

record type definition (RTD)

SCL

serial clock line (SCL)

SDA

serial data line (SDA)

T4T

Type 4 Tag (T4T)

TLV

tag length value (TLV)

UID

unique identifier (UID)

URI

uniform resource identifier (URI)
A string of characters that uniquely identify a name or a resource on a network, such as the Internet.

URL

uniform resource locator (URL)
A unique identifier used to locate a resource on the Internet (also referred to as a web address).

Revision history

Reference	Description
Revision 1.1, 2024-04-30	
All	Editorial changes
Revision 1.0, 2024-03-28	
All	Initial release

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2024-04-30

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2024 Infineon Technologies AG

All Rights Reserved.

Do you have a question about any aspect of this document?

Email:
CSSCustomerService@infineon.com

Document reference
IFX-jni1693584061137

Important notice

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie").

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

Warnings

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.