# Host parameterization via pass-through (PT)

## Use case guide

## About this document

### Scope and purpose

This document outlines the setup process of the OPTIGA™ Authenticate NBT to enable host parameterization using the pass-through mode. It demonstrates the steps through which the end user can operate and explore the use case 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 pass-through communication functionality of the OPTIGA™ Authenticate NBT.

# Table of contents

**About this document** . . . 1
**Table of contents** . . . 2
**List of tables** . . . 3
**List of figures** . . . 4
**1 Introduction** . . . 5
1.1 NFC I2C bridge tags . . . 5
**2 Use case overview** . . . 6
2.1 General information . . . 6
2.2 Methodology . . . 6
2.3 PT use case example . . . 6
**3 Use case integration** . . . 7
3.1 Prerequisites . . . 7
3.2 Operation modes . . . 7
3.3 Personalization . . . 8
3.3.1 Device target state . . . 8
3.3.2 Utilized interfaces . . . 9
3.3.3 Personalization procedure . . . 9
3.3.3.1 Interface configurations . . . 10
3.3.3.2 Type 4 Tag application's file configurations . . . 10
3.3.3.3 Activating the OPERATIONAL life cycle state . . . 12
3.4 Operational use case . . . 12
3.4.1 Operational flow example: Host configuration . . . 12
**A Appendix** . . . 14
A.1 Technical background . . . 14
A.1.1 OPTIGA™ Authenticate NBT system architecture . . . 14
A.1.2 Hardware configuration . . . 16
A.1.3 Interface description . . . 17
A.1.4 Command reference . . . 18
A.1.5 Life cycle states . . . 18
A.2 Device delivery condition . . . 18
A.2.1 Initial NDEF message . . . 20
**References** . . . 21
**Glossary** . . . 22
**Revision history** . . . 25
**Disclaimer** . . . 26

# List of tables

Table 1 EF.CC (relevant for access via the NFC interface) . . . . . 11
Table 2 EF.FAP (example FAP settings) . . . . . 11
Table 3 Supported applications of the OPTIGA™ Authenticate NBT . . . . . 15
Table 4 Type 4 Tag application and files . . . . . 16
Table 5 Command set of the OPTIGA™ Authenticate NBT . . . . . 18

# List of figures

Figure 1 Host parameterization via pass-through . . . . . 6
Figure 2 Embedded tag - pass-through mode . . . . . 7
Figure 3 Interface configurations for pass-through mode . . . . . 8
Figure 4 OPTIGA™ Authenticate NBT example target configuration for pass-through mode . . . . . 8
Figure 5 Standard personalization procedure . . . . . 9
Figure 6 Personalization procedure example via NFC interface using a mobile phone . . . . . 10
Figure 7 Host configuration via pass-through – operational flow . . . . . 13
Figure 8 OPTIGA™ Authenticate NBT product architecture . . . . . 14
Figure 9 Type 4 Tag file structure . . . . . 15
Figure 10 Embedded tag . . . . . 16
Figure 11 NFC-only tag . . . . . 17
Figure 12 Logical communication states of OPTIGA™ Authenticate NBT . . . . . 17
Figure 13 Delivery condition: Interface configuration . . . . . 19
Figure 14 Delivery condition: Application file content, access conditions (per-file, per-interface) . . . . . 19
Figure 15 URI record . . . . . 20
Figure 16 External record . . . . . 20

# 1 Introduction

This use case guide assists users in understanding the key features of the OPTIGA™ Authenticate NBT that enable the host parameterization via pass-through (PT) use case. It also provides a high-level overview of how the device needs to be configured for this use case and the steps required to realize real-world use case scenarios.

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

Chapter 3 describes how the 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 like 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 describes the OPTIGA™ Authenticate NBT's pass-through mode. This mode allows synchronous data transfer between a remote NFC device (for example, mobile phone) and a host MCU via the NFC and I2C interfaces to support use cases such as host parameterization or host card emulation.

## 2.1 General information

Host parameterization via pass-through is used in embedded environments, where the OPTIGA™ Authenticate NBT is connected to a host via the I2C. The device translates communication initiated by an NFC-enabled mobile phone to the host and vice versa. Pass-through means that the I2C and NFC interfaces are available at the same time. Requests from a remote NFC device to the OPTIGA™ Authenticate NBT are routed to the host MCU for processing via the I2C interface. Responses from the host are sent to the device via the I2C interface and translated into NFC responses for the remote device.

## 2.2 Methodology

The OPTIGA™ Authenticate NBT's specific connectivity options enable the device to serve as a translation component between the NFC and the I2C interface. By using this functionality, the device converts a host MCU into an NFC-enabled device.

## 2.3 PT use case example

The embedded system for the PT use case example, consisting of an OPTIGA™ Authenticate NBT and a host MCU, communicates with an NFC-enabled mobile phone. The device handles the NFC protocol with the mobile phone, which is used to transfer configuration data to the host. The NFC-enabled mobile phone will try to select an application using an AID that is not registered on the device, however emulated by the host MCU. When the device is configured for pass-through mode, this selection request is automatically routed to the host.

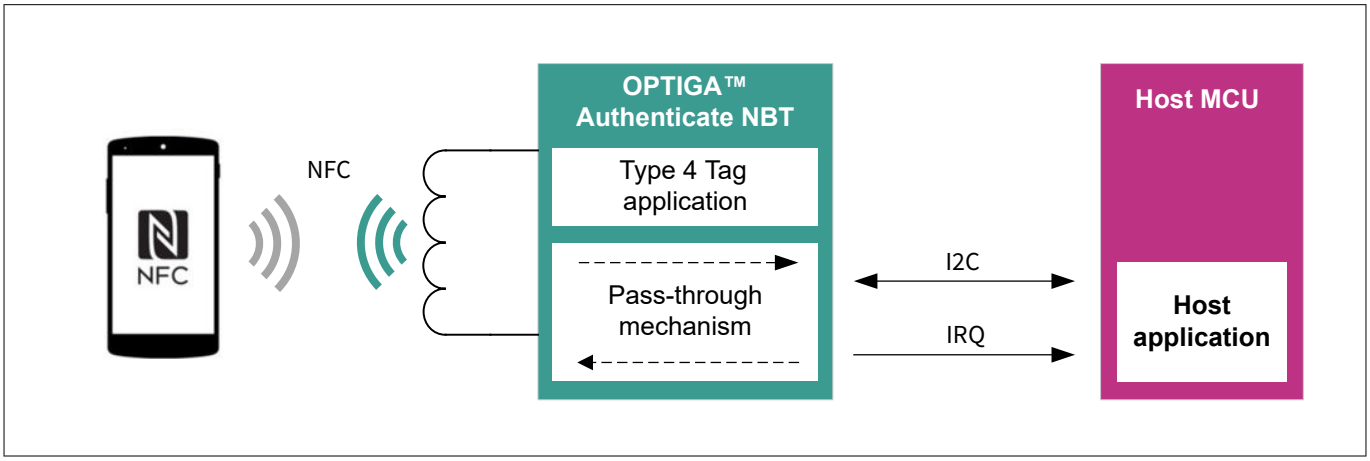



**Figure 1 Host parameterization via pass-through**

Possible usage scenarios include:

- Download a firmware update to a host system
- Configuring or controlling a host system ("headless")
- Card application emulation on the host

The above-mentioned scenarios are listed and described in more detail in the Extended Datasheet [6].

The following chapters discuss the technical environment requirements, as well as the interface availability and configurations of the OPTIGA™ Authenticate NBT, in order to use it most efficiently in the pass-through mode.

# 3 Use case integration

This chapter describes how to use the OPTIGA™ Authenticate NBT for the pass-through use case. 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 pass-through functionality is demonstrated using the OPTIGA™ Authenticate NBT connected to a PSoC microcontroller. The example shown in this section covers the flow to configure the device. 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 pass-through mode requires the embedded tag setup, where the device 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 for the communication via the I2C interface
- I2C host interface is using via SDA, SCL, and IRQ to communicate

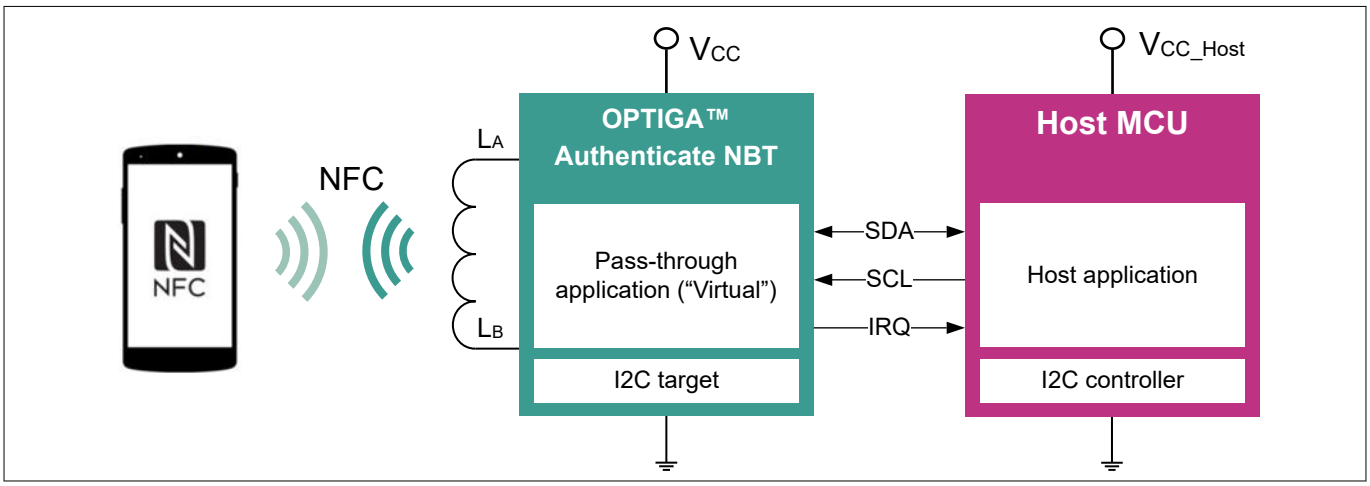



**Figure 2 Embedded tag - pass-through mode**

## 3.2 Operation modes

In pass-through mode, the NFC communication takes place between the NFC-enabled mobile phone (acting as an NFC reader) and a host MCU via the OPTIGA™ Authenticate NBT (acting as an NFC listener). The device can be powered via $V_{CC}$, the I2C interface is active, and the device is simultaneously connected to an NFC antenna via the $L_A$ and $L_B$ pads. This enables use cases such as headless device configuration or firmware update of a host device without involving the T4T application installed on the OPTIGA™ Authenticate NBT. Alternatively, the device can be configured for asynchronous data transfer mode [9].

For more information on pass-through, refer to Extended Datasheet [6].

## 3.3 Personalization

The following chapter describes how to configure the OPTIGA™ Authenticate NBT for the pass-through use case.

### 3.3.1 Device target state

The following configurations can be used to operate the OPTIGA™ Authenticate NBT in the pass-through mode:

- Both, the I2C and NFC interfaces need to be enabled
- The I2C interface supports the GP T=1' protocol
- Pass-through mode is enabled by setting the IRQ function to PT-IRQ

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

**Figure 3 Interface configurations for pass-through mode**

Once the PT mode is enabled, the SELECT command with an unregistered Application IDentifier (AID) from the NFC interface activates the PT mode. After activating this PT mode, the OPTIGA™ Authenticate NBT will automatically transfer and forward commands received from NFC to the I2C interface.

While the PT mode operation is active, the Type 4 Tag application is bypassed. Nevertheless, the Type 4 Tag application coexists when the PT mode is enabled and may be used to take advantage of an NDEF message for improved user experience. An OEM that provides a specific application for mobile phone as well as a card emulation functionality on the host system, can also use dedicated NDEF records embedded in the NDEF message file. These records may include a direct link to the OEM's application in the iOS App Store or Android Play store, or a URL pointing to the OEM's website. For example, NFC-enabled mobile phones natively read and record NDEF messages, making this scenario possible.

In addition to the mandatory interface configuration, for the pass-through mode, the T4T application can be configured as preferred. As an example, the application may be configured for NDEF only with the following settings (see Figure 4):

- The NDEF file can be read via the NFC and the I2C interface
- The NDEF file can only be written via the I2C interface. This enables the host to update the data contained within it after the device is fully operational in the field
- Since the proprietary files (FILE_1, FILE_2, FILE_3 and FILE_4) are not used, no reading or writing is permitted

Type 4 Tag application: **Target configuration of files and access conditions, PT use case**

| 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 PT | N | N | N | N | A | N | A | N | A | A | 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 4 OPTIGA™ Authenticate NBT example target configuration for pass-through mode**

### 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 5 depicts the standard personalization procedure.

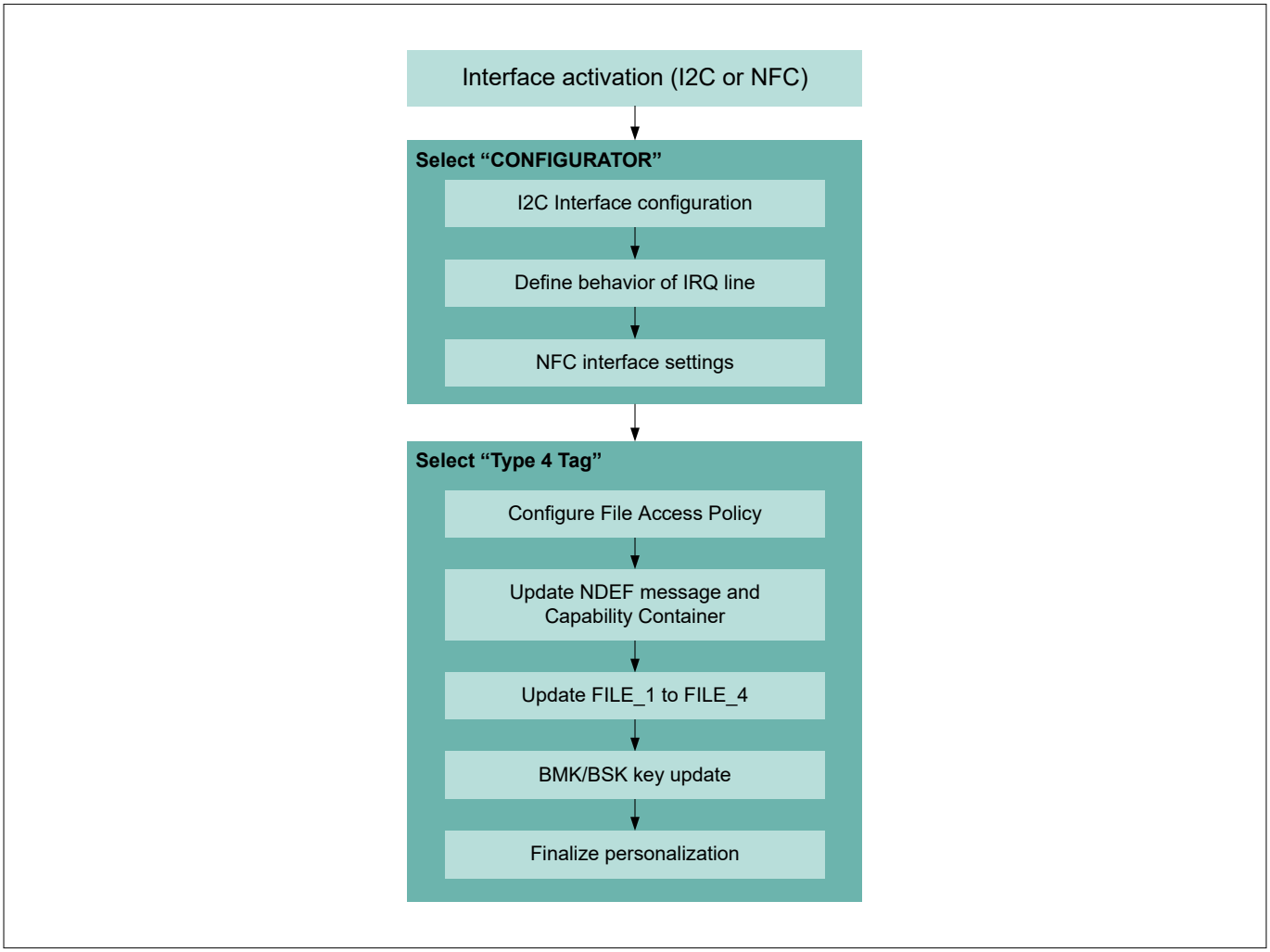



**Figure 5 Standard personalization procedure**

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 6.

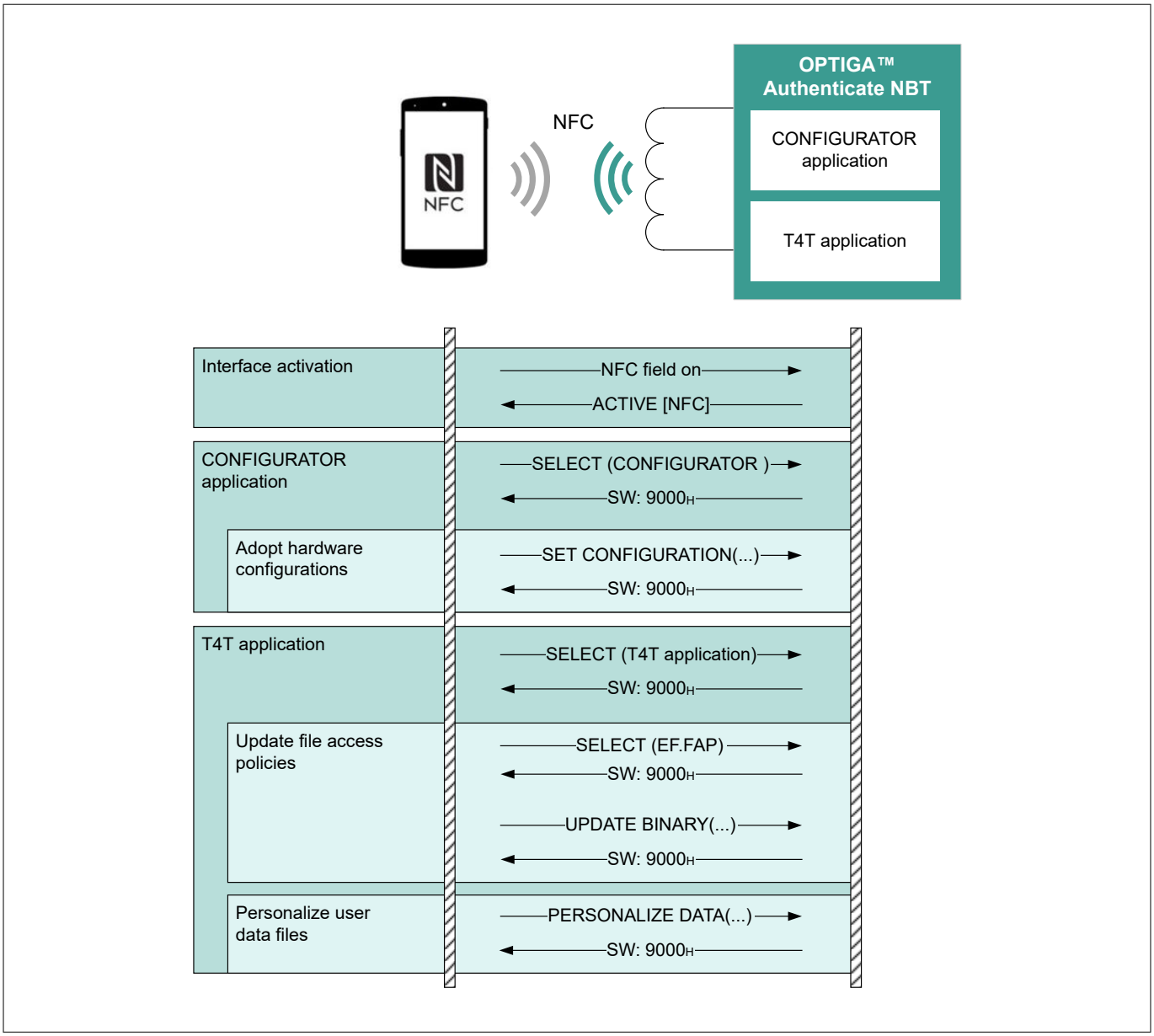



**Figure 6 Personalization procedure example via NFC interface using a mobile phone**

### 3.3.3.1 Interface configurations

Upon delivery, the OPTIGA™ Authenticate NBT is configured with interface settings that are suitable for the asynchronous data transfer mode, as the NFC and I2C interfaces are enabled by default.

The implementation of the GP T=1' I2C in the device relies on an interrupt line to notify the host that data is available. Hence, the PT-IRQ option should be configured for the OPTIGA™ Authenticate NBT's IRQ, since the IRQ functionality is disabled by default upon delivery.

### 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 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, and developers need to update this value for the Proprietary-File_CTRL_TLV. If the data is not matching, this may result in noncompliance with the NFC Forum T4T Specification [1]

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. In addition, all four proprietary files ($E1AX_H$)are registered in the Capability Container. The access to these files is blocked for reading and updating (see Table 1). Furthermore, the FAP settings prevent any file access for all files from the I2C interface (see Table 2).

***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.*

The following tables provide details about the access rights settings for the application files of OPTIGA™ Authenticate NBT in the embedded setup for the pass-through mode. Table 1 contains the access right settings an NFC reader (for example, NFC-enabled mobile phone) needs to consider.

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

| Tag | Length | FileID | 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 settings shown in Table 5 govern access to all application files for the example. In this case, full access to the NDEF file (FileID: $E1A2_H$) is permitted through the I2C interface. From the NFC interface, read access is granted; however, 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 2 EF.FAP (example FAP settings)**

| FileID | I2C_Read | I2C_Update | NFC_Read | NFC_Update | Description |
|---|---|---|---|---|---|
| $E103_H$ | $40_H$ | $00_H$ | $40_H$ | $00_H$ | No NFC update; no I2C update |
| $E104_H$ | $40_H$ | $40_H$ | $40_H$ | $00_H$ | No NFC update; full I2C |
| $E1A1_H$ | $00_H$ | $00_H$ | $00_H$ | $00_H$ | 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 |

***Note**: In the pass-through use case, modifying the T4T application's file content is mainly relevant if the NDEF features of the mobile phones are intended to be used. The same applies for updating BMK/BSK. However, if updating the BMK/BSK key is required, the PERSONALIZE DATA command should be used.*

#### 3.3.3.3 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

In the OPERATIONAL life cycle state, the OPTIGA™ Authenticate NBT supports both interfaces: NFC and I2C. In PT mode, the NFC and the I2C interface are active at the same time. An NFC-enabled mobile phone can communicate synchronously with a host. The OPTIGA™ Authenticate NBT serves as an NFC to I2C bridge device. The following components are required for the operational use:

- An NFC antenna connected to the device
- An NFC reader device, for example, an NFC-enabled mobile phone running a dedicated application
- Embedded setup: The device is additionally interconnected into host system via I2C interface

***Note**: Optionally, the NDEF message can be personalized with URL to OEM website and/or link to iOS App Store or Android Play Store.*

### 3.4.1 Operational flow example: Host configuration

In this example, an NFC-enabled mobile phone uses a dedicated mobile application to exchange data (for example, the device configuration) with a host. The OPTIGA™ Authenticate NBT is connected to the host via I2C as well as an NFC antenna, forming a passive NFC interface that manages the NFC communication initiated by the mobile phone.

An NFC-enabled mobile phone with the OEM's mobile device configuration app is trying to select an OEM's dedicated host application by its AID when approaching the NFC antenna of the device. After command reception, the OPTIGA™ Authenticate NBT determines whether this host application is available within its predefined file system. If the selected application is not found, the device activates pass-through mode and forwards the SELECT command and all subsequent commands from the NFC interface to the host system via the I2C interface. For each command, the host application that receives each command, processes it and prepares a corresponding response. The host sends the response back to the device via the I2C interface, and the device forwards the frame to the mobile phone via the NFC interface.

***Note**: The Type 4 Tag application on the OPTIGA™ Authenticate NBT is not used for the pass-through communication flow.*

Figure 7 illustrates the communication process to update data (for example, the device configuration). This update is initiated by an NFC-enabled mobile phone, running a specific application that contains and conveys configuration data.

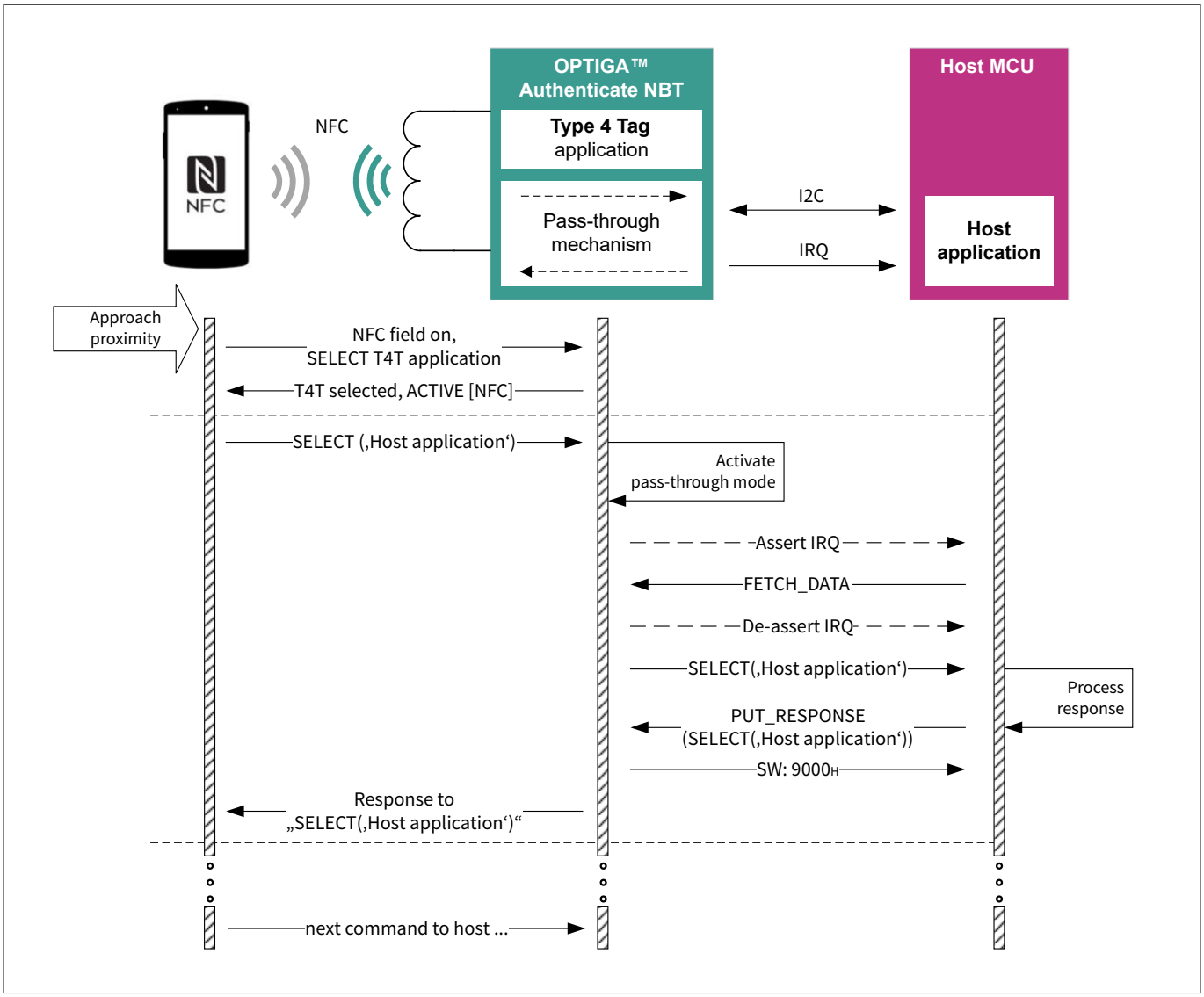



**Figure 7 Host configuration via pass-through – operational flow**

When the NFC-enabled mobile phone approaches the device antenna, it will automatically detect and select the T4T application. The mobile phone reads the URL present in the NDEF message (configured during personalization) and directs the users to the associated URL destination, for example, to download the OEM's host configuration application.

Once the OEM's host configuration application is installed on the mobile phone, it can be invoked to interact with the host application. After receiving the SELECT('Host application') command (determined by the Application IDentifier (AID)), the OPTIGA™ Authenticate NBT activates the pass-through mode. Therefore, the device activates the IRQ (configured to PT-IRQ during personalization) to indicate to the host that the data is available. The host fetches the data using the PASS-THROUGH FETCH DATA command and processes it. Once the results of the processing are available, the host transfers this response data to the device using the PASS-THROUGH PUT RESPONSE command. The response is forwarded to the mobile phone via the NFC interface. Each NFC command is automatically forwarded to the host using the described procedure, notifying the host by pulling up the IRQ line.

***Note**: Infineon Technologies provides example applications for iOS, Android, and Modus Toolbox to assist developers create applications swiftly and integrate them instantly. Infineon Technologies provides OPTIGA™ Authenticate NBT specific libraries for C++, JAVA and SWIFT.*

# 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.

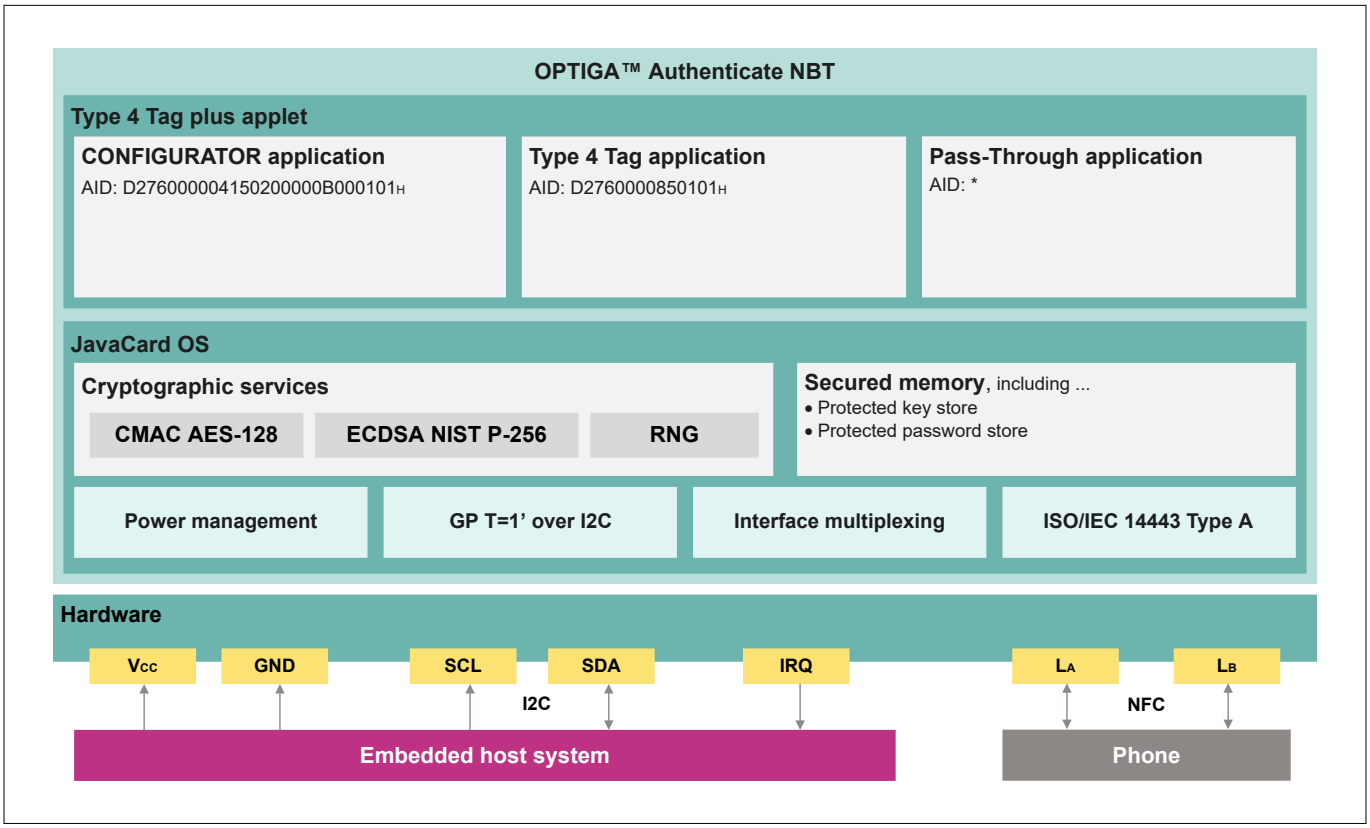



**Figure 8 OPTIGA™ Authenticate NBT product architecture**

## A Appendix

**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.

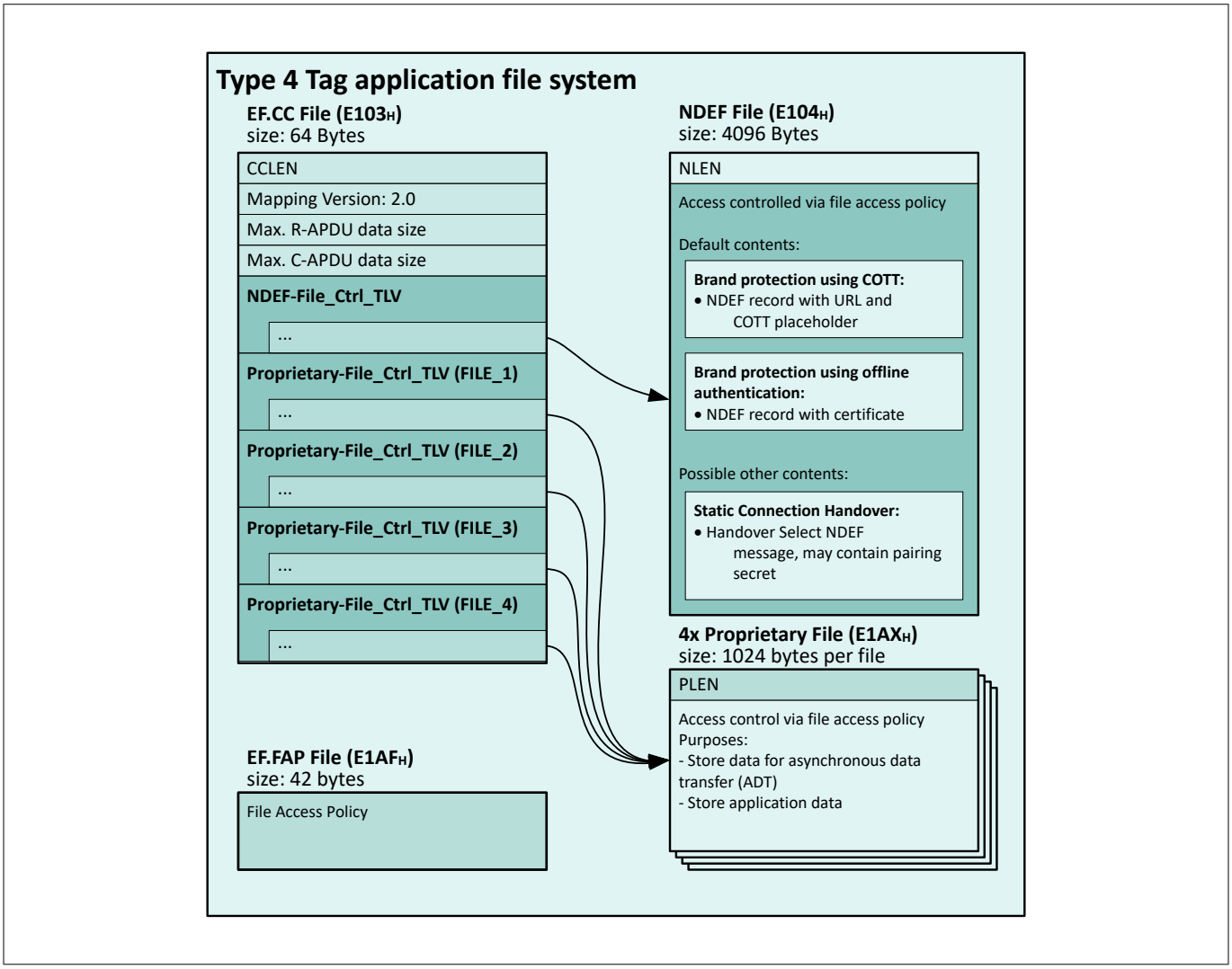



**Figure 9 Type 4 Tag file structure**

**Table 4 Type 4 Tag application and files**

| File | FileID | Size [bytes] | Content |
|---|---|---|---|
| EF.CC | $E103_H$ | 64 | Size and access policy of<br>• NDEF file<br>• 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<br>• EF.CC file<br>• NDEF file<br>• FILE_1 to FILE_4<br>• 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

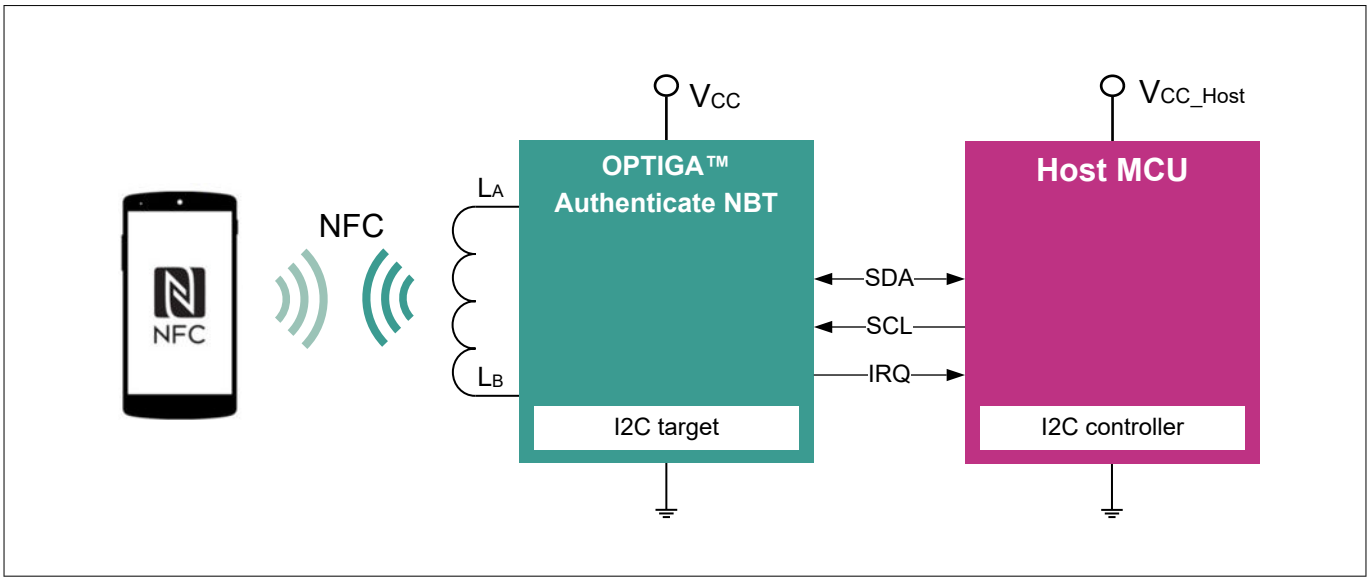



**Figure 10 Embedded tag**

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.

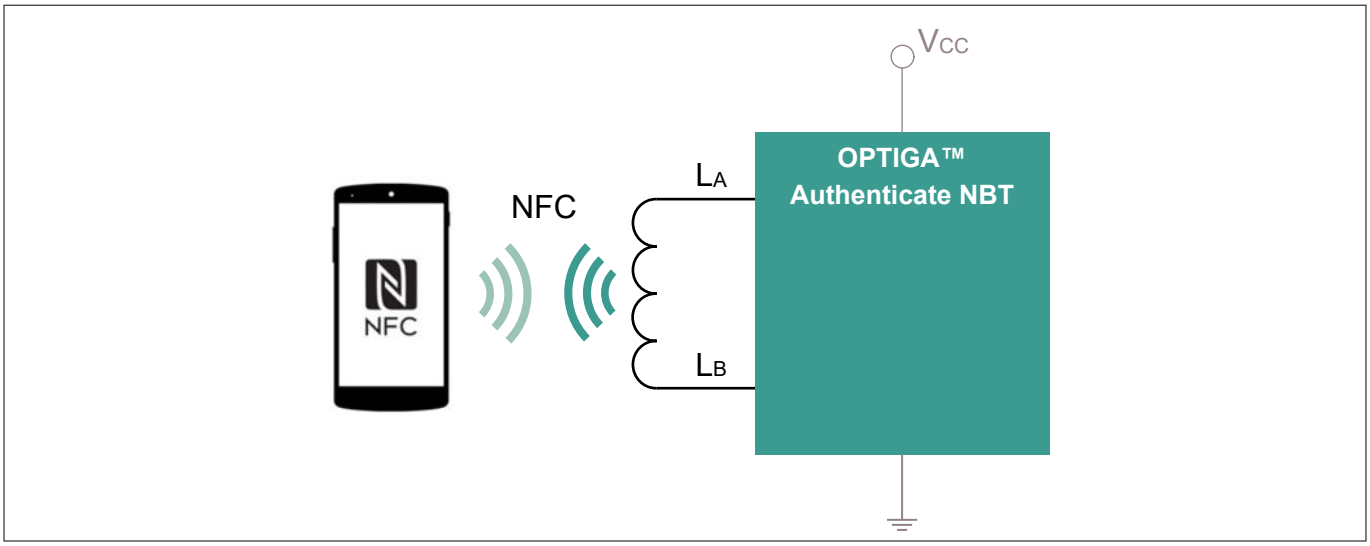



**Figure 11 NFC-only tag**

## 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 12 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).

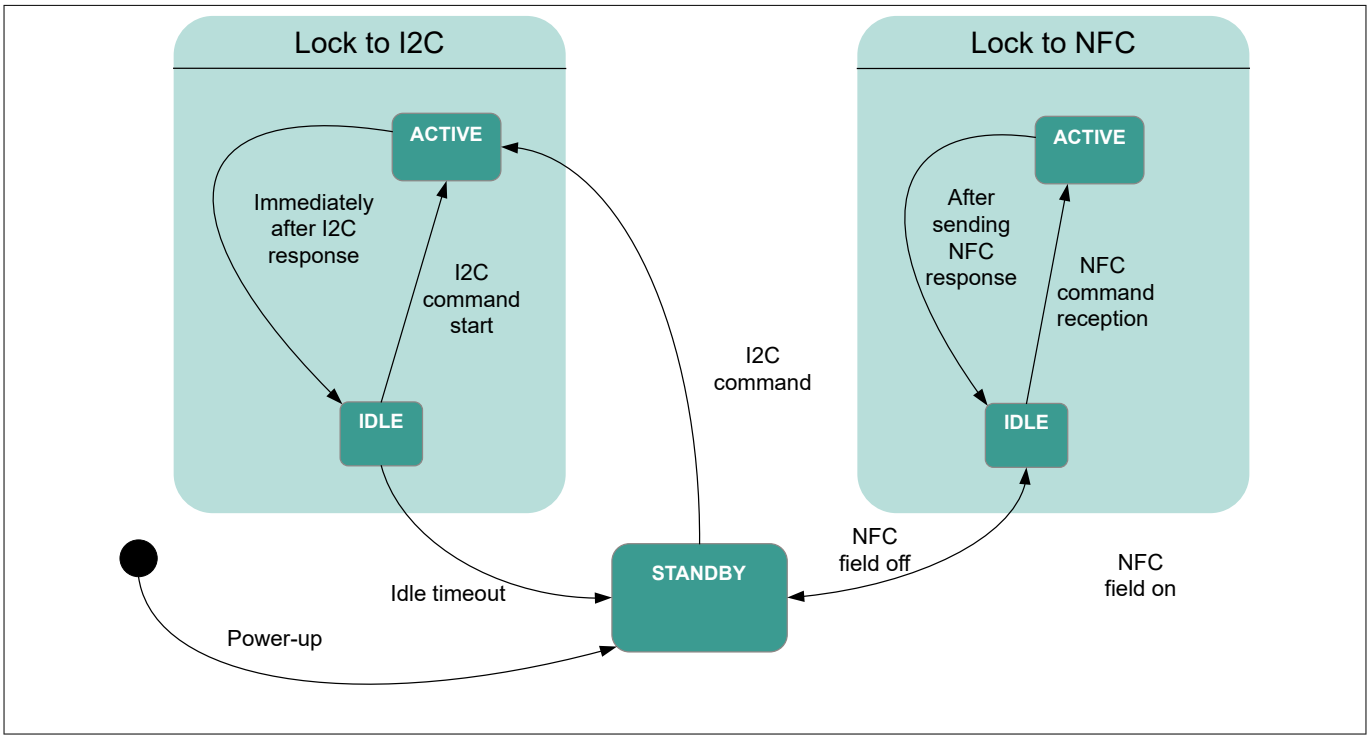



**Figure 12 Logical communication states of OPTIGA™ Authenticate NBT**

### 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 13 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 14. 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 |

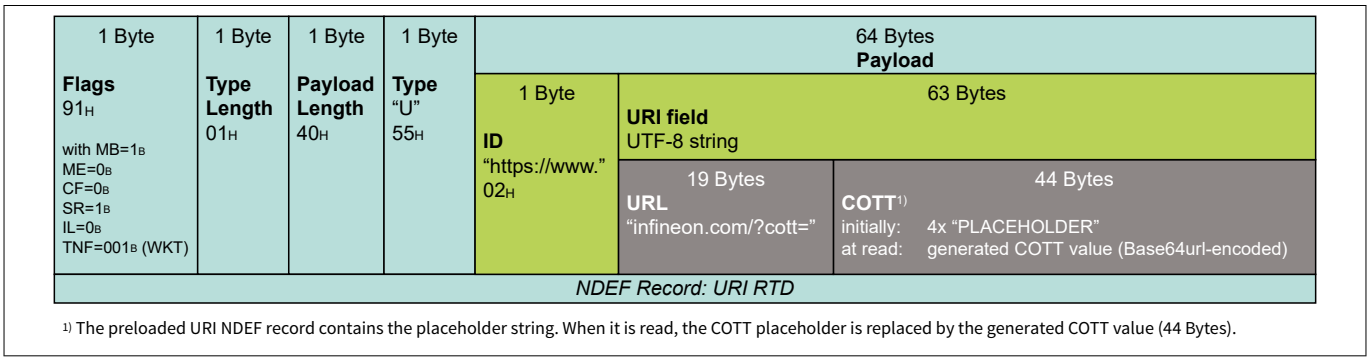
Access conditions: A = ALWAYS; N = NEVER; P = PASSWORD REQUIRED

**Figure 14 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

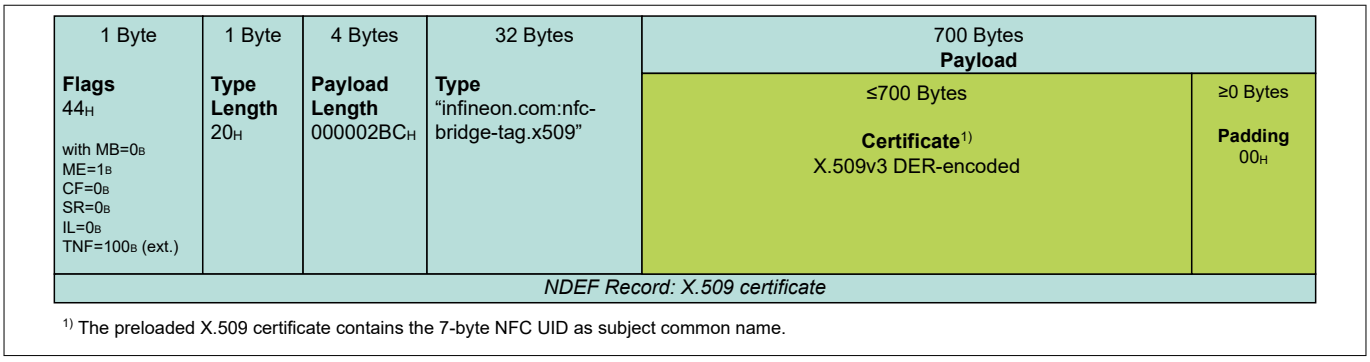
The OPTIGA™ Authenticate NBT is delivered with a preloaded NDEF message in the NDEF file. This NDEF message contains two NDEF records: the first is an URI record followed by an external record. Initially, the URI record contains a link to https://www.infineon.com/, followed by the COTT placeholder string 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 15 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 16 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)*
**[9]** Infineon Technologies AG: *Host parameterization via asynchronous data transfer (ADT), Use Case 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)*

## Glossary

**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-nmu1692874678063**

**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.