
# Guidelines for GBA Based Certificate Provisioning

# Version 1.0

# 26 January 2022

*This is a Non-binding Permanent Reference Document of the GSMA*

## Security Classification: Non-confidential



## Copyright Notice



## Disclaimer



## Compliance Notice

The information contain herein is in full compliance with the GSM Association's antitrust compliance policy.

*[Note to editor: Include one of the following statements only; delete the other]:*

This Permanent Reference Document is classified by GSMA as an Industry Specification, as such it has been developed and is maintained by GSMA in accordance with the provisions set out in GSMA AA.35 - Procedures for Industry Specifications.

OR

This Permanent Reference Document has been developed and is maintained by GSMA in accordance with the provisions set out in GSMA AA.34 - Policy and Procedures for Official Documents.

# Table of Contents

# 1 Introduction

## 1.1 Background

Cellular based Vehicle-to-everything (C-V2X) communication, which integrates V2X and 4G/5G cellular mobile communication technology, accelerates the development of intelligent traffic management, intelligent dynamic information services and intelligent vehicle control and is one of a range of Internet of Things (IoT) applications that will be deployed in support of intelligent transportation.

By taking the advantage of short-range direct communication on the PC5 interface defined by 3GPP, V2X messages, such as Basic Safety Message (BSM), Signal Phase And Timing message (SPAT), Road Side Information (RSI), Road Safety Message (RSM), MAP message etc., can be interacted dynamically among OnBoard Unit (OBU), Road Side Unit (RSU) and pedestrian, to enable the future implementation of high-level automatic driving for connected and autonomous vehicles.

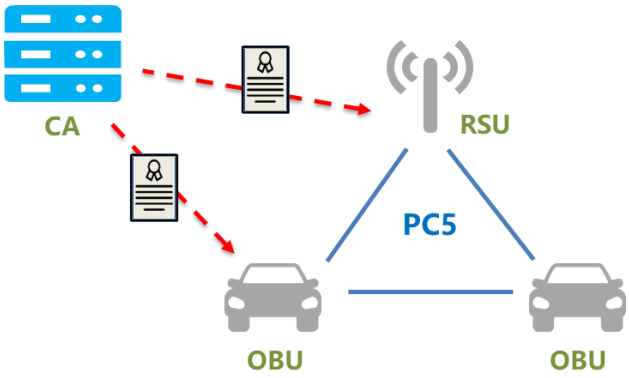


**Figure 1: C-V2X direct communication on the PC5 interface**

Safety and security are at the core of any traffic system so the entire C-V2X ecosystem needs to ensure trust and authenticity of communication between all entities involved. At present, Public Key Infrastructure (PKI) based certificate management systems are recognised by the global V2X industry to establish trust relationship between these entities and digital certificates are used to protect V2X messages transmitted directly on the PC5 interface (as shown in Figure.1). This requires C-V2X certificates and security-sensitive information to be provisioned on the vehicle Onboard Unit (OBU) and Roadside Unit (RSU) in a secure way to get devices into an operational state before a vehicle is on the road and the RSU is put into use.

How to implement certificate provisioning effectively and efficiently is a real challenge faced by the industry in the process of introducing and upgrading C-V2X technology. The GSMA provides solutions to this challenge through either GBA based solutions for C-V2X certificate defined in this document, or the GSMA IoT SAFE mechanism described in [13] and [14].

## 1.2 Scope

This document provides guidelines on how to use the GBA mechanism to practically implement online certificate provisioning for C-V2X, and other IoT and M2M scenarios. To achieve this, customer requirements and enhancements to the GBA system are considered, specifically with regard to GBA session key management and related aspects.

As GBA is access network agnostic and the 5G GBA standard that enhances the interface between BSF and 5G Unified Data Management (UDM) is still under discussion within 3GPP, it is considered that the recommendations proposed in this document are also suitable for adoption in 5G cellular networks.

GBA security implementations on C-V2X devices are also discussed to safeguard the security of the keys and information related to C-V2X certificates and device security recommendations are also provided.

The enhanced GBA mechanism recommended in this document has generic application with the result that the mechanism has the potential to be used in a range of vertical industry applications and is not just confined to C-V2X certificate provisioning.

Based on specific market and technical requirements, vertical industries can benefit from the implementation of either the enhanced GBA mechanism recommended in this document or the IoT SAFE mechanisms in [13] and [14].

## 1.3 Abbreviations

| Abbreviation | Description |
|---|---|
| 3GPP | The 3rd Generation Partnership Project |
| 5GAA | 5G Automotive Association |
| AC | Authorization Certificate |
| ACA | Authorization CA |
| AKA | Authentication and Key Agreement |
| AP | Authentication Proxy |
| APDU | Application Protocol Data Unit |
| AuC | Authentication Center |
| AUTN | Authentication Token |
| AV | Authentication Vector |
| B-TID | Bootstrapping Transaction IDentifier |
| BSF | Bootstrapping Server Function |
| BSM | Basic Safety Message |
| CCMS | C-ITS security Credential Management System |
| CK | Cipher Key |
| C-V2X | Cellular based Vehicle-to-everything |
| CA | Certificate Authority |
| DNS | Domain Name Server |
| EC | Enrolment Certificate |
| ECA | Enrolment CA |
| ESPS | Efficient Security Provisioning System |
| FQDN | Fully Qualified Domain Name |
| GBA | Generic Bootstrapping Architecture |
| GP | Global Platform |

| Abbreviation | Description |
|---|---|
| HMAC | Hash-based Message Authentication Code |
| HSS | Home Subscriber Server |
| HTTP | Hyper Text Transfer Protocol |
| IK | Integrity Key |
| IMPI | IP Multimedia Private Identity |
| IMSI | International Mobile Subscriber Identity |
| ISIM | IP multimedia Service Identity Module |
| IoT | Internet of Things |
| KDF | Key Derivation Function |
| MNO | Mobile Network Operator |
| MSISDN | Mobile Subscriber ISDN number |
| NAF | Network Application Function |
| OBU | OnBoard Unit |
| PC | Pseudonym Certificate |
| PCA | Pseudonym CA |
| PKI | Public Key Infrastructure |
| RES | authentication RESponse |
| RFC | Request For Comments |
| RNG | Random Number Generator |
| RSI | Road Side Information |
| RSM | Road Safety Message |
| RSU | Road Side Unit |
| RTE | Runtime Environment |
| SCMS | Security Credential Management System |
| SE | Security Element |
| SPAT | Signal Phase And Timing message |
| TLS | Transport Layer Security |
| UDM | Unified Data Management |
| UE | User Equipment |
| UICC | Universal Integrated Circuit Card |
| USIM | Universal Subscriber Identity Module |
| VIN | Vehicle Identification Number |
| XRES | eXpected authentication RESponse |

## 1.4 References

| Ref | Title |
|---|---|
| [1] | B. Brecht, D. Therriault, A. Weimerskirch, W. Whyte, V. Kumar, T. Hehn and R. Goudy, "A Security Credential Management System for V2X Communications," IEEE Transactions on Intelligent Transport Systems, vol. 19, no. 12, pp. 3850-3871, |

| Ref | Title |
|---|---|
| | December 2018. |
| [2] | Crash Avoidance Metrics Partners (CAMP) LLC, "Security Credential Management System Proof–of–Concept Implementation; EE Requirements and Specifications Supporting SCMS Software Release 1.2.2," [Online]. Available: https://wiki.campllc.org/display/SCP [Accessed 1 August 2019]. |
| [3] | ETSI, "TS 102 940 v1.3.1 (2018-04) Intelligent Transportation Systems (ITS); Security; ITS communications security architecture and security management," 2018. |
| [4] | 5GAA, "White paper "Efficient Security Provisioning System"" |
| [5] | 3GPP, "TS 33.220 "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA)"" |
| [6] | 3GPP, "TS 33.221 "Generic Authentication Architecture (GAA); Support for subscriber certificates"" |
| [7] | 3GPP, "TS 33.222 "Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS)"" |
| [8] | IETF, "RFC 2104 "HMAC: Keyed-Hashing for Message Authentication"" |
| [9] | 3GPP, "TS 33.101 "UICC-terminal interface; Physical and logical characteristics"" |
| [10] | 3GPP, "TS 33.102 "Characteristics of the Universal Subscriber Identity Module (USIM) application"" |
| [11] | Global Platform, "Global Platform Card Specification" |
| [12] | 3GPP, "TS 24.109 " Bootstrapping interface (Ub) and network application function interface (Ua); Protocol details"" |
| [13] | GSMA IoT.04 Common Implementation Guide to Using the SIM as a ‘Root of Trust’ to Secure IoT Applications |
| [14] | GSMA IoT.05 IoT Security Applet Interface Description |

# 2 C-V2X certificate provisioning application scenarios

Some typical application scenarios that could leverage GBA for C-V2X certificate provisioning are provided below. These are provided as examples and are not intended to constitute an exhaustive list of all possible usage scenarios.

## 2.1 Scenario 1: Bootstrapping and Enrolment

Bootstrapping is the first step to prepare C-V2X devices (e.g., OBU, RSU) into a ready state for V2X communication operations before they can be used in V2X ecosystem. In this procedure, elementary security-critical information including Certificate Authority (CA) information pertaining to the C-V2X PKI system and their certificates, key pairs and corresponding Enrolment Certificates (EC) shall be provided to the Security Element (SE) of C-V2X devices to enable them to further apply for authorisation certificates.

Bootstrapping can be performed during the production process or in the field. C-V2X devices are equipped with USIM card/eSIM access to an Enrolment CA (ECA) server through 4G/5G cellular networks to obtain the security-critical information. As there are no pre-configured security credentials in C-V2X devices initially, ECA servers have no way to identify, authenticate and authorise C-V2X devices, so GBA, which is able to bootstrap and establish

application layer security associations based on cellular network identifiers (International Mobile Subscriber Identity (IMSI) and Mobile Subscriber ISDN number (MSISDN)) and authentication vectors derived by pre-existing shared root keys, can be leveraged by ECA to authenticate C-V2X devices.

After obtaining the information (e.g., Vehicle Identification Number (VIN), OBU/RSU ID) that is required to be bound to an enrolment certificate, a C-V2X device initiates a GBA procedure and is authenticated by the GBA platform in a cellular network through the Authentication and Key Agreement (AKA) mechanism in order to access the ECA server. If successful, the USIM and GBA platform will generate a specific GBA session key and provide it to the upper-layer certificate management application of the C-V2X device and the ECA server respectively. With this session key, the C-V2X device can set up an end-to-end secure communication tunnel in the application layer to the ECA server and apply for an enrolment certificate by sending an EC request message which offers the necessary information and enrolment certificate public key that is locally generated.

On receiving the EC request, the ECA server verifies the message and signs an enrolment certificate for the C-V2X device if the request is authorised, and then transmits the signed enrolment certificate and related security-critical information to the C-V2X device through the secure tunnel. After securely storing these credentials in the SE, the bootstrapping and enrolment procedure is complete.

### 2.2 Scenario 2: C-V2X Authorisation Certificate Application

An enrolled C-V2X device needs to apply for authorisation certificates with the request messages signed by the enrolment certificate before it can securely communicate with other C-V2X devices.

As specified by US Security Credential Management System (SCMS) [1] [2], EU C-ITS Security Credential Management System (CCMS) [3] and 5GAA Efficient Security Provisioning System (ESPS) [4], a typical authorisation certificate is a Pseudonym Certificate (PC), which is used to conceal the real identity of a C-V2X device with a temporary identity generated randomly to protect user privacy. Other types of authorisation certificates are application certificates, identification certificates or authorisation tickets, which are used to verify the rights of a user to access services.

To apply for authorisation certificates, C-V2X devices need to securely access and communicate with the Authorisation CA (ACA) server. One way to achieve this is to use the USIM and cellular network security GBA capability to provide symmetric session keys to establish end-to-end secure tunnels in the application layer. Afterwards, C-V2X devices can generate key pairs in the local SEs and interact with ACA servers to apply for and download authorisation certificates through these tunnels, which can ensure the security of messages.

## 3 Introduction of Standard GBA

GBA is a standardised mechanism specified by 3GPP for application layer security associations to be bootstrapped and established with the pre-existing 4G/5G mobile network authentication capabilities. The key 3GPP standards relating to GBA are as follows:

- TS 33.220: Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA);

- TS 33.221: Generic Authentication Architecture (GAA); Support for Subscriber Certificates;
- TS 33.222: Generic Authentication Architecture (GAA); Access to Network Application Functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS);
- TS 24.109: Bootstrapping Interface (Ub) and Network Application Function Interface (Ua); Protocol Details;
- TS 29.109: Generic Authentication Architecture (GAA); Zh and Zn Interfaces based on the Diameter Protocol; Stage 3.

Based on the root key pre-shared by the USIM and mobile network Authentication Center (AuC), mutual secure authentication and key agreement can be carried out between a C-V2X device and a mobile network, which can generate and provide a shared GBA session key to a C-V2X application client and a server to establish a secure and trusted connection in the application layer.

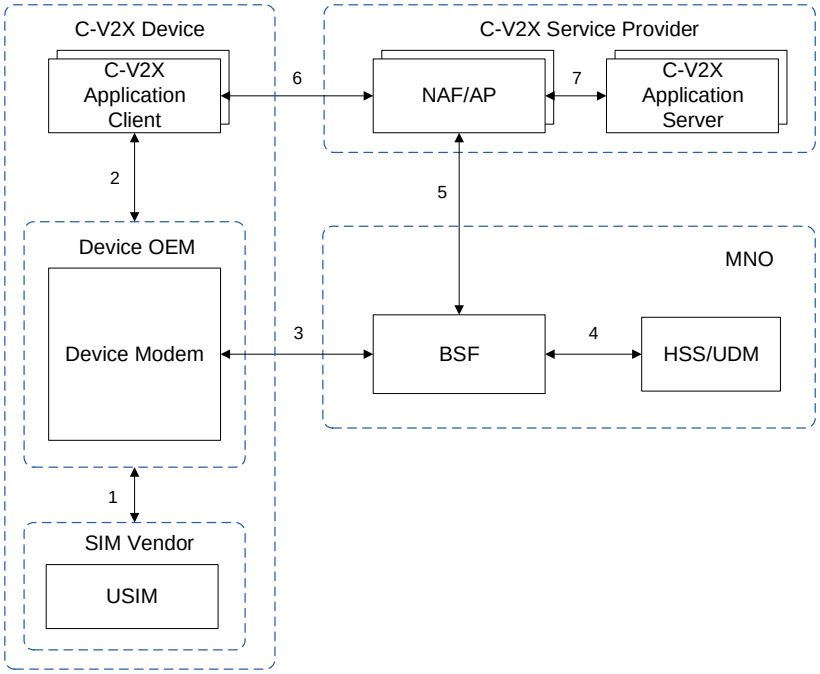


**Figure 2 General architecture of standard GBA for C-V2X application**

Figure 2 depicts a general GBA architecture for common C-V2X use. In the case of the C-V2X certificate provisioning scenario, the C-V2X application server would be instantiated as an Enrolment CA or Authorisation CA server, and the application client in the C-V2X device would be instantiated as a certificate management application client.

On the network side, the HSS/UDM and Bootstrapping Server Function (BSF) are 3GPP defined standard network elements under the control of the MNO. HSS/UDM are the fundamental databases for 4G and 5G networks, which support the AuC function and store the set of user security settings required by GBA. BSF is defined in 3GPP TS 33.220 [5]. With the authentication vectors retrieved from HSS/UDM, the BSF authenticates the C-V2X device (modem & USIM) through the 3GPP AKA mechanism and then negotiates a GBA session key for the Network Application Function/Authentication Proxy (NAF/AP) of the C-V2X service provider who subscribes to the GBA service from the MNO.

NAF/AP is another 3GPP standard network element defined in 3GPP TS 33.220/33.221/33.222 [5] [6] [7], which gets the GBA session key from the BSF and assists the C-V2X application server to take advantage of the GBA capability to secure communications to the application client on the C-V2X device. With the GBA session key, a secure tunnel (e.g., Transport Layer Security (TLS)) is established between the NAF/AP and the C-V2X device, and all messages to the application server can be assured as coming from an authorised C-V2X device of the MNO.

On the terminal side, the C-V2X device modem authenticates the BSF of the Mobile Network Operator (MNO) during the GBA procedure based on the authentication vectors provided by the USIM, and if authentication succeeds, the GBA session key will be negotiated for the upper layer. According to 3GPP TS 33.220, there are two ways to generate a GBA session key on the device. One is GBA_ME, where the GBA session key Ks_NAF is calculated by the modem, and the other is GBA_U, where the GBA session key Ks_ext_NAF (equals to Ks_NAF) and Ks_int_NAF are calculated by the Universal Integrated Circuit Card (UICC). In both methods, Ks_NAF/Ks_ext_NAF can be transmitted to and operated by the upper-layer application client, while Ks_int_NAF is always required to be kept and operated in the UICC for high security. The UICC is one kind of secure hardware with an EAL4+ security level in which the USIM function is implemented, so it can ensure the security of Ks_int_NAF.

## 3.1 Motivation

As shown above, the GBA mechanism can identify and authenticate C-V2X devices and establish application layer secure connections between C-V2X devices and the C-V2X application server without pre-configuration of any security credentials. Thus, GBA can be leveraged to solve the C-V2X certificate provisioning problem faced by industry, which can save money and time for a range of stakeholders including the automotive enterprises, C-V2X device vendors, C-V2X certificate service providers, etc.

However, when GBA is deployed in practice for C-V2X, the following problems make it complex to use, which limit its attractiveness:

- According to the current GBA architecture, each C-V2X application server needs one NAF/AP network element to access the GBA service. This means many NAF/APs have to be deployed and maintained for different application servers, if a MNO wants to extensively open up and promote GBA capabilities and services. This will bring significant investment challenges and costs for the MNO and its customers with the result that the C-V2X customers require a more economically efficient way to adopt the GBA service.
- As the shared GBA session key is kept by NAF/AP, the application-layer messages, especially the security sensitive messages related to C-V2X certificates, need to be secured by the NAF/AP. Nevertheless, NAF/AP is a general network element implemented by vendors, it is difficult for them to support various security mechanisms that are required by C-V2X industry regulations. This challenge has the potential to limit the application of GBA, so it is necessary to find a practical way to make GBA solutions more flexible but at the same time allow C-V2X customers protect the transmitted messages in accordance with regulatory requirements.

The guidelines contained in this document are designed to make GBA services better meet C-V2X customer requirements and promote a more feasible GBA network security capability exposure method.

# 4 Technical Recommendations for Architecture

To enable the GBA service to be applied more easily, the enhanced GBA architecture recommended in this document is shown in Figure 3 below. With regard to C-V2X certificate provisioning application scenarios, the C-V2X application servers in this architecture are the Enrolment CA and the Authorisation CA servers, which are deployed by MNOs or the third parties that offer GBA services to C-V2X devices.

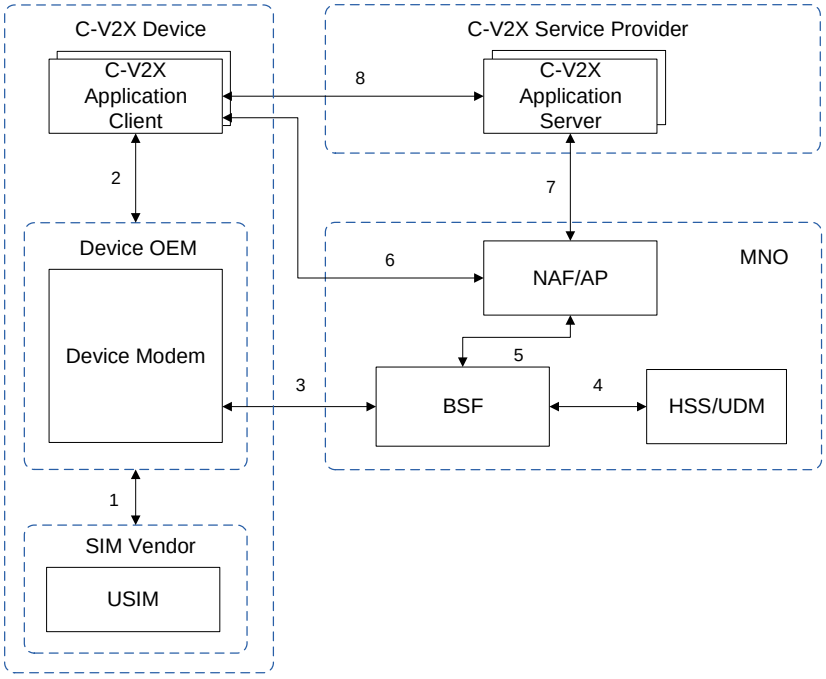


**Figure 3 Enhanced architecture of GBA for C-V2X application**

Compared with the standard GBA architecture, the main proposed enhancement lies in the GBA core network element, the NAF/AP, which is moved from the C-V2X service provider domain into the MNO domain. The objective is to multiplex the NAF/AP and to let one NAF/AP serve more than one C-V2X application server. As a result, only one or very few NAF/APs need to be deployed to enable MNOs to provide GBA services for more C-V2X applications.

In the enhanced architecture, the NAF/AP is shared by a few C-V2X application servers. Based on the GBA session keys (Ks_NAF or Ks_int_NAF) provided by BSF during the GBA bootstrapped security association procedure, the NAF/AP further derives GBA application session keys (denoted by K*) for each C-V2X application and shares the K* to the application server. Meanwhile, the C-V2X device also derives the same K* locally during the GBA bootstrapping procedure, and then provides the K* to the upper layer applications for use.

In this way, the C-V2X application server and the C-V2X device share the same GBA application session keys K*. They can use the K* to implement mutual authentication,

protect transmitted messages, establish a secure tunnel, etc. and then accomplish C-V2X certificate provisioning with the shared GBA application session keys.

On the terminal side, the GBA application session keys K* can be derived in two ways. In the case of GBA_ME, K* is derived by the modem or the C-V2X application client based on the Ks_NAF and in the case of GBA_U, K* is derived by the USIM based on the Ks_int_NAF and always kept in the UICC SE for high security.

# 5 Technical Recommendations for Service Flow

## 5.1 General

Based on the enhanced GBA architecture, the recommended service flow for online C-V2X certificate provisioning is depicted in Figure.4. The procedure is composed of 4 steps: initiation of bootstrapping, bootstrapping, bootstrapping security association usage and application security association usage.

The first 3 steps follow the standard 3-step GBA procedure specified in 3GPP TS 33.220 in order to minimise the impact of the enhancements on the commercial equipment. The outline is given here and on the terminal side, the interactions between C-V2X application client, device modem and USIM, which are not specified by 3GPP but are implemented in practice, are featured. For the detailed procedure of these 3 steps, please refer to 3GPP TS 33.220.

The fourth and final step is the enhanced procedure for GBA. Some necessary, but minor, enhancements on the NAF/AP are required to enable the NAF/AP to provide specific GBA application session keys K* to the C-V2X application server to help establish the end-to-end security association in the application layer by using the bootstrapped keys K* to apply for the enrolment certificate and authorisation certificate. The detailed procedure is described in Section 5.5.

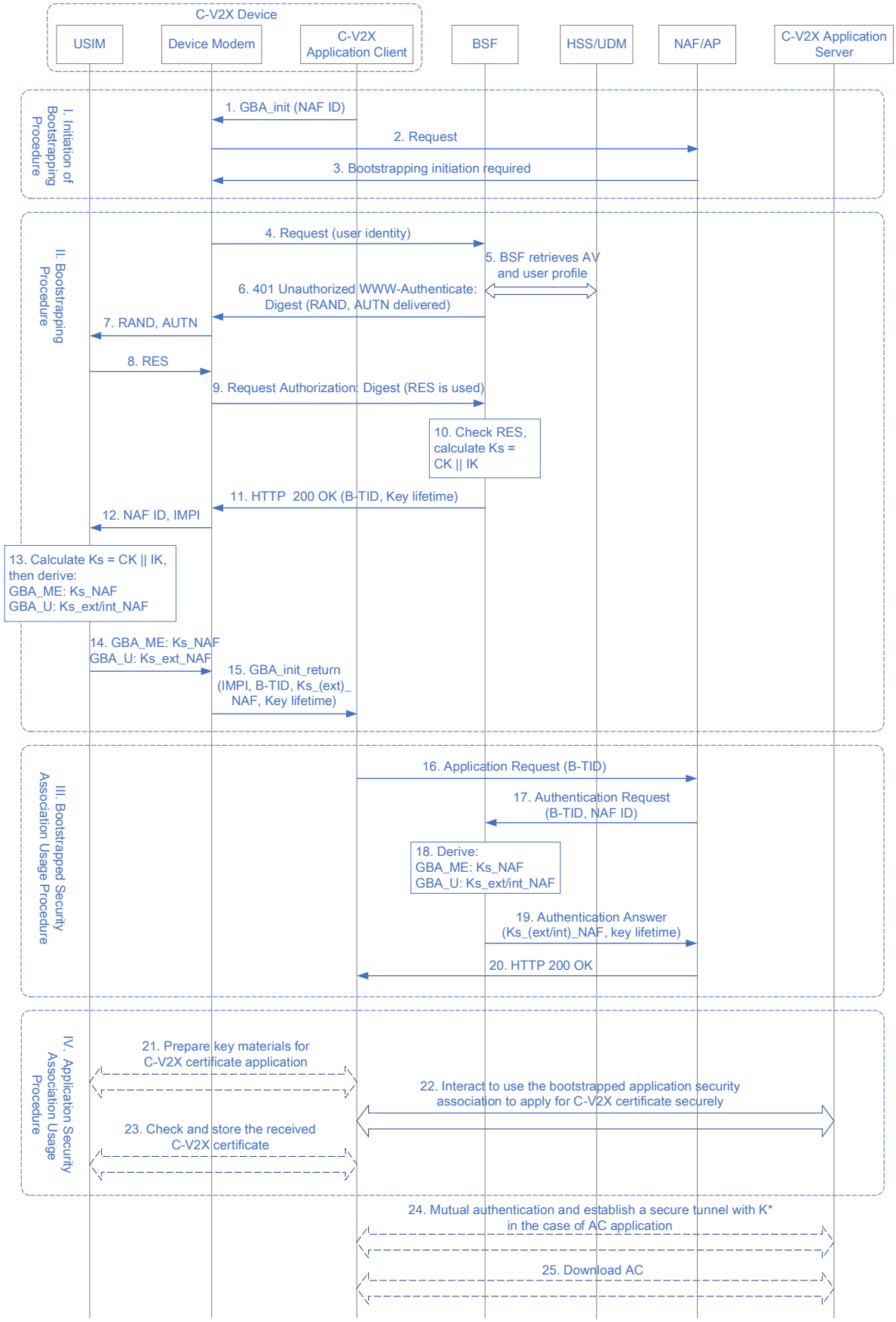


**Figure 4 Enhanced GBA service flow for C-V2X certificate provisioning**

## 5.2 Step I: Initiation of Bootstrapping

Before the C-V2X application client, which is responsible for C-V2X certificate management, communicates with the C-V2X application server to apply for C-V2X certificates, they first need to agree whether to use the GBA mechanism. When the C-V2X application client (User) wants to interact with the C-V2X application server, but it does not know whether the C-V2X application server requires the use of the shared keys obtained by means of the GBA, the C-V2X application client may contact the NAF/AP of the C-V2X application server for further instructions.

1. If the C-V2X application client starts to apply for C-V2X enrolment or authorisation certificates, it triggers the device modem to initiate GBA bootstrapping by invoking GBA_init API, with NAF ID as an input. The NAF ID includes the FQDN of the C-V2X application server and security protocol identifier.
2. According to the NAF ID, the device modem finds the NAF/AP of the C-V2X application server by DNS and initiates communication with the NAF/AP.
3. As configured in advance, the NAF/AP, on behalf of the C-V2X application server, replies with a bootstrapping initiation message to require the use of shared keys obtained by means of the enhanced GBA.

## 5.3 Step II: Bootstrapping

When the device modem knows the bootstrapping procedure is needed, it shall perform a bootstrapping authentication first, and then return the result to the C-V2X application client.

4. The modem of the C-V2X device sends an HTTP request towards the BSF.
5. The BSF retrieves the complete set of GBA user security settings and one Authentication Vector (AV, AV=RAND||AUTN||XRES||CK||IK) from the HSS.
6. BSF forwards the RAND and AUTN to the device modem in HTTP 401 message (without the CK, IK and XRES).
7. The device modem forwards RAND and AUTN to the USIM via APDU interface.
8. The USIM checks AUTN to verify that the challenge is from an authorised mobile network, calculates CK, IK and RES, and then returns RES to the device modem.
9. The device modem sends another HTTP request, containing the Digest AKA response (calculated using RES), to the BSF.
10. The BSF authenticates the device modem by verifying the Digest AKA response and calculates key material Ks by concatenating CK and IK. In order to bind Ks with the subscriber, the BSF also generates a B-TID, in format of base64encode(RAND) @ BSF_servers_domain_name, where RAND value comes from step 6.
11. The BSF sends a HTTP 200 OK message, including a B-TID, to the device modem to indicate successful authentication. In addition, in the HTTP 200 OK message, the BSF supplies the lifetime of Ks.
12. The device modem forwards NAF ID and IMPI to the USIM via APDU interface.

Note: As specified in the Section 7.1.2.4 of 3GPP TS 31.102 for the GBA interface, the device modem must transmit the IMPI to the USIM, no matter how the IMPI is obtained. IMPI can be obtained by the device modem by deriving it from the IMSI, or it can also be obtained from the IP Multimedia Service Identity Module (ISIM) which has stored the IMPI.

13. The USIM calculates the key material Ks by concatenating CK and IK and derives the GBA session key Ks_ext_NAF and Ks_int_NAF from the Ks in the case of GBA_U, and then stores the Ks and Ks_int_NAF, which will not leave the UICC SE. In the case of GBA_ME, the USIM derives the Ks_NAF from the Ks.
14. The USIM returns Ks_(ext)_NAF to the device modem.
15. The device modem gives a GBA_init_return to the C-V2X application client, with IMPI, B-TID, Ks_(ext)_NAF, Key lifetime as outputs, in order to enable the C-V2X application client to use the already bootstrapped security association.

## 5.4 Step III: Bootstrapped Security Association Usage

After initiation of the bootstrapping procedures, the C-V2X application client knows the GBA mechanism is required to be used when it applies for the C-V2X certificates, and if the bootstrapping authentication succeeds, the C-V2X application client shall execute the following operations to use the already bootstrapped security association.

16. The C-V2X application client sends an application HTTP request to the NAF/AP, including B-TID.
17. The NAF/AP sends the authentication request to retrieve the GBA session key Ks_ext_NAF and Ks_int_NAF corresponding to the B-TID from the BSF in the case of GBA_U. In the case of GBA_ME, the NAF/AP retrieves Ks_NAF instead.
18. The BSF derives Ks_(ext)_NAF and Ks_int_NAF from the Ks corresponding to the B-TID.
19. The BSF sends the authentication answer to the NAF/AP, supplying the requested GBA session keys and their lifetime.
20. The NAF/AP sends a HTTP 200 OK message to the C-V2X application client to indicate the success of Step III.

## 5.5 Step IV: Application Security Association Usage

When Step III is finished, the C-V2X application client and the NAF/AP share the GBA session keys obtained by the means of the GBA and then need to use the GBA bootstrapped application security association to bootstrap the C-V2X device and provision the C-V2X certificates.

To implement this, the C-V2X device needs to prepare related key materials for the C-V2X certificate application, interact with the C-V2X application server to take the advantage of the bootstrapped application security association to apply for the C-V2X certificate securely and then check and store the received C-V2X certificate. Steps 21~23 in Figure 4 describe the process.

In the case of GBA_ME, these 3 steps can be done by the C-V2X application client itself, as it can use the Ks_NAF returned by the device modem to securely interact with the C-V2X application server. In the case of GBA_U, step 21 and step 23 shall be done by the client, the device modem and the USIM cooperatively, so that the Ks_int_NAF can be kept in the UICC SE and the hardware security capabilities of the UICC can be used to achieve high security levels.

Based on the difference in how to provide the K* to the C-V2X application server, there are 2 solutions for step IV to implement C-V2X certificate application. Both solutions can be used to bootstrap the C-V2X device and apply for the EC and AC by the means of the GBA as for

scenarios 1 and 2, and the C-V2X application will be ECA or ACA. Figures 5 and 6 below show these procedures, which specifically highlight the GBA_U case to show the interactions on the terminal side.

In the case of the AC application, the ACA server will return a waiting time to notify the C-V2X application client when the AC is ready instead of sending back the issued certificate directly like EC. This requires the C-V2X application client to access the ACA server to download the AC after Step IV when the timer expires. The C-V2X application client can choose to use the shared K* to implement mutual authentication and establish a secure tunnel with the ACA server to provide secondary protection of the downloaded AC (as shown in step 24 and 25 in Figure 4), no matter whether the AC is protected by the ACA server's certificate or not.

### 5.5.1 Solution 1: C-V2X Application Server Request

In solution 1, on receiving the C-V2X Application Request message forwarded by the NAF/AP, the C-V2X application server sends a User Information Request message with B-TID as the index to request NAF/AP to derive the K* and then gets the K* and related information back.

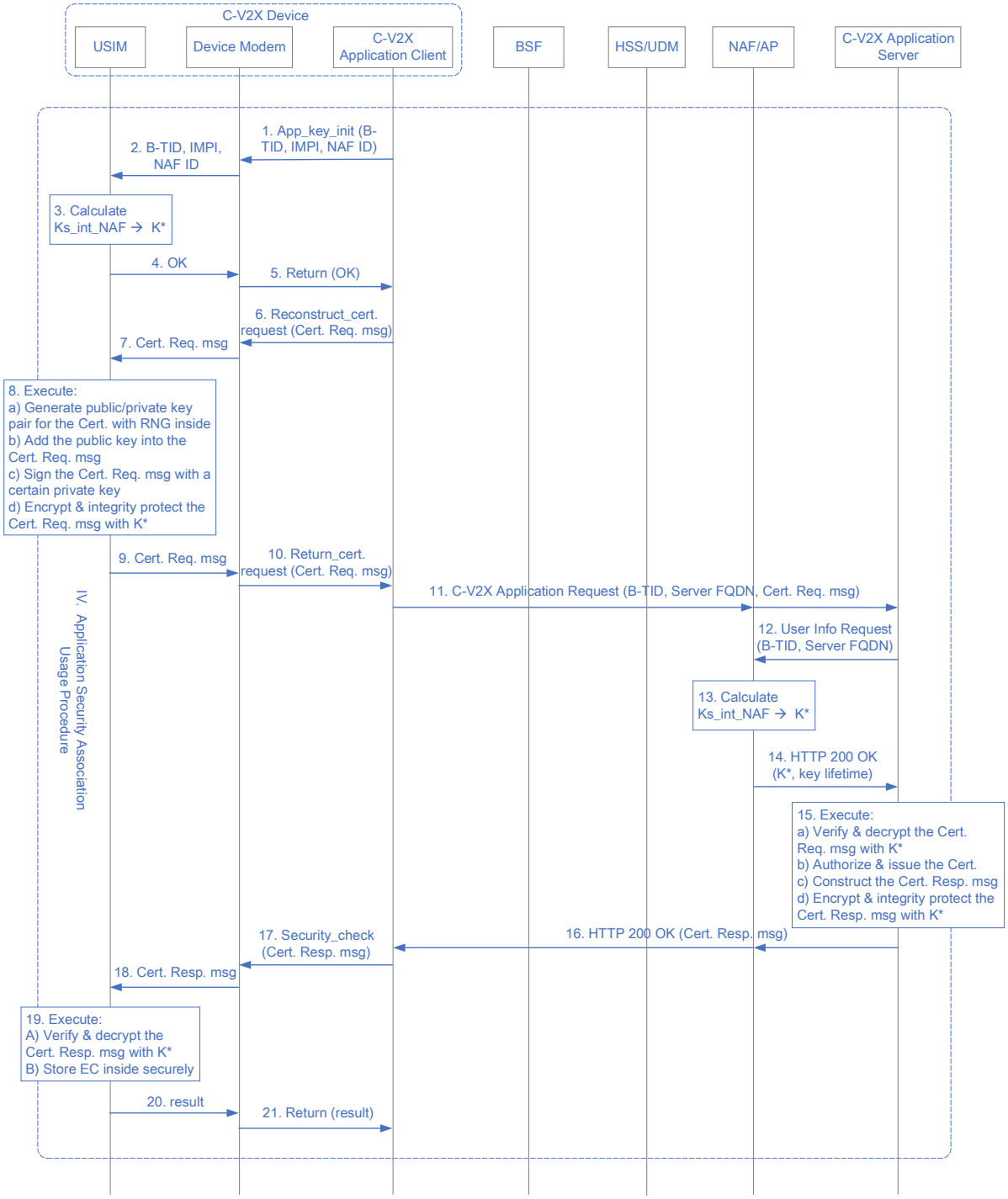


**Figure 5 Solution 1 procedure for Step IV**

1. The C-V2X application client invokes APP_key_init API opened up by the device modem to trigger the USIM to derive GBA application session key K*, with B-TID, IMPI and NAF ID as inputs.
2. The device modem forwards B-TID, IMPI and NAF ID to the USIM via APDU interface.
3. Based on the GBA session key Ks_int_NAF kept inside, the USIM further derives GBA application session keys K*, which are used to protect the upcoming C-V2X application messages.

K* = KDF (Ks_int_NAF, String, B-TID, UE ID, Service ID). For derivation details please see Section 4.6.

4. If successful, the USIM returns OK.
5. The device modem forwards the result of OK to the C-V2X application client.
6. The C-V2X application client constructs a Certificate Request message according to the regulations and/or standards of the C-V2X industry to apply for EC or AC. In the message, the information related to EC/AC public/private key pair should be kept absent. Then, the C-V2X application client sends the message to the USIM by invoking Reconstruct_cert. request API.
7. The device modem forwards the Certificate Request message to the USIM.
8. The USIM executes the following operations to reconstruct the Certificate Request message. The security algorithm used here should meet the requirements of the C-V2X industry.

   In the case of EC application (scenario 1), the USIM should:

   a) Generate a public/private key pair for the EC by the Random Number Generator (RNG) inside the UICC SE.
   b) Add the EC public key into the outgoing Certificate Request message of EC.
   c) Sign the Certificate Request message of EC with the EC private key and add the self-signature into the message.
   d) Encrypt and integrity protect the Certificate Request message with the K* derived in step 3 and add the HMAC to the message.

   In the case of AC application (scenario 2), the USIM should:

   a) Generate a public/private key pair for the AC by the Random Number Generator (RNG) inside of the UICC SE.
   e) Add the AC public key into the outgoing Certificate Request message of AC.
   f) Sign the Certificate Request message of AC with the EC private key and add the signature and EC into the message.
   g) Encrypt and integrity protect the Certificate Request message with the K* derived in step 3 and add the HMAC to the message.

9. The USIM returns the reconstructed Certificate Request message.
10. The device modem forwards the Certificate Request message to the C-V2X application client.
11. The C-V2X application client sends the C-V2X Application Request message to the corresponding C-V2X application server, which includes B-TID, Server FQDN and the Certificate Request message. The whole message is protected by HTTP Digest with the Ks_ext_NAF key corresponding to the B-TID and forwarded by the NAF/AP according to the Server FQDN. Before forwarding, NAF/AP authenticates the message through HTTP Digest to make ensure its authenticity.
12. The C-V2X application server sends User Information Request message to obtain the GBA application session key K* corresponding to the B-TID from the NAF/AP.
13. Based on the GBA session key Ks_int_NAF obtained from the BSF before, the NAF/AP also derives the GBA application session key K*.
14. The NAF/AP returns HTTP 200 OK to supply the K* and its lifetime to the C-V2X application server.

15. The C-V2X application server executes the following operations to verify the Certificate Request message and issue the EC or AC.

    a) Verify the HMAC of the Certificate Request message and decrypt the message with the K*.
    b) Check the authorisation of the EC/AC application request. If authorised, issue the EC/AC for the C-V2X device.
    c) Construct a Certificate Response message, according to the C-V2X industry regulations and/or standards, which contains the issued EC or a download waiting time of AC.
    Taking PC (one kind of AC) application for example, it will take the PCA server several minutes to generate at least 20 pseudonym certificates, so a method adopted by the C-V2X industry is to return a waiting time to the C-V2X device to notify the PCs ready time. When the time expires, the C-V2X application client responsible for certificate management accesses the PCA server again to download the issued PCs.
    d) Encrypt and integrity protect the Certificate Response message with K* and add HMAC to the message.

16. The C-V2X application server returns HTTP 200 OK to send the Certificate Response message to the C-V2X application client.
17. The C-V2X application client sends the Certificate Response message to the USIM by invoking Security_check API.
18. The device modem forwards the Certificate Response message to the USIM.
19. The USIM verifies the HMAC of the Certificate Response message and decrypts the message with the K*. If successful, then:

    In the case of EC application, the USIM should store the EC in the UICC SE, and return OK to indicate the C-V2X application client that the EC has already been applied securely.

    In the case of AC application, the USIM should return the plain Certificate Response message back to the C-V2X application client, from which it can obtain the waiting time for downloading the AC later.

20. The USIM returns the result to the device modem.
21. The device modem forwards the result to the C-V2X application client.

### 5.5.2 Solution 2: NAF/AP Push

In solution 2, before forwarding the C-V2X Application Request message, the NAF/AP uses the B-TID as the index to find corresponding Ks_int_NAF in the locally maintained security contexts and derives the K* first, then pushes the K* and related information to the C-V2X application server by sending them along with the C-V2X Application Request message.

The whole procedure is shown in Figure 6. Most steps of solution 2 are the same as solution 1, the only differences are highlighted in bold for convenience.

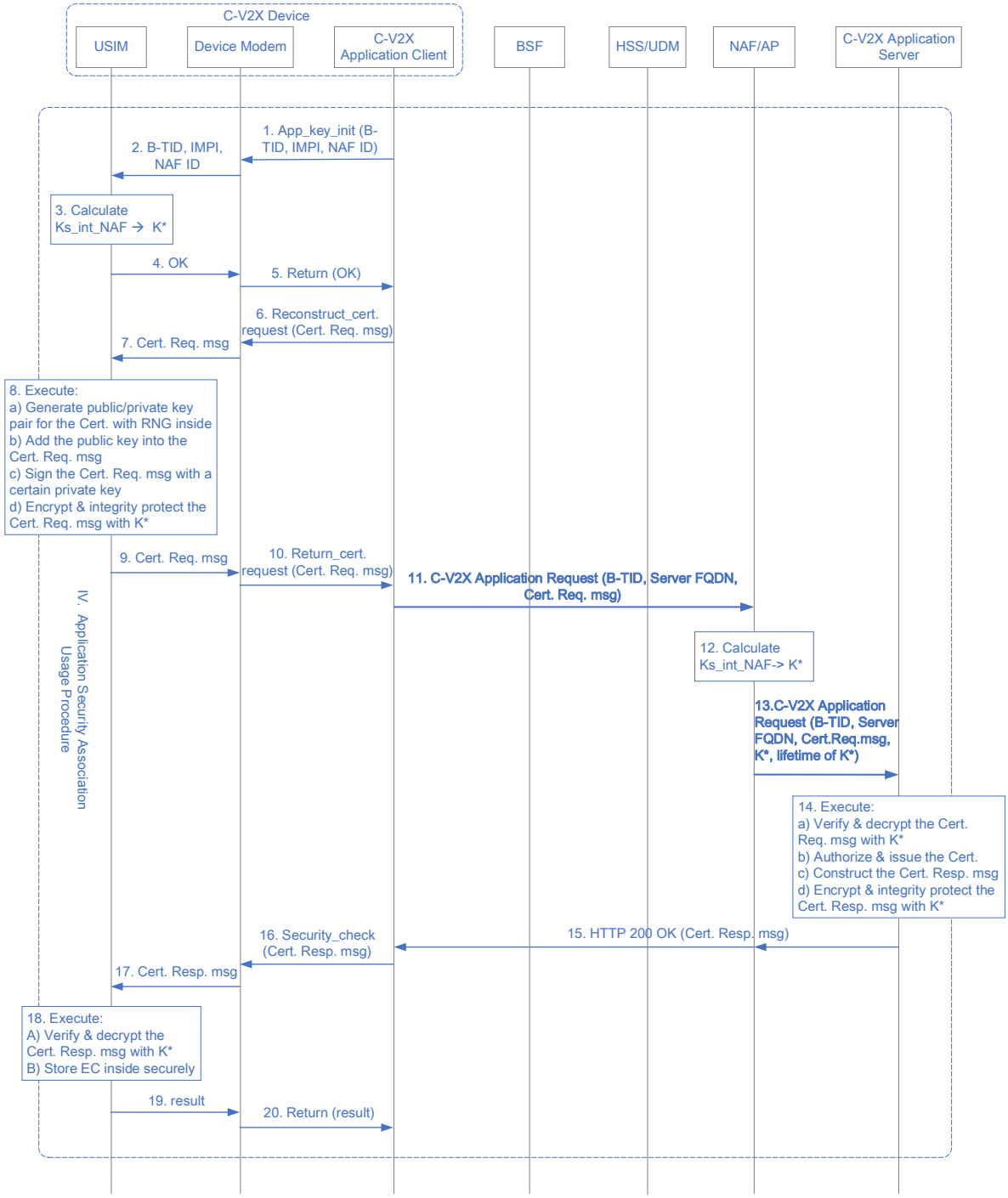


**Figure 6 Solution 2 procedure for Step IV**

1. In step 1~5, the C-V2X application client triggers the USIM to derive GBA application session key K*.
2. In step 6~7, the C-V2X application client constructs a Certificate Request message according to the C-V2X industry regulations and/or standards to apply for EC/AC and sends the message to the USIM by invoking Reconstruct_cert. request API through the device modem.
3. In step 8~10, the USIM generates a public/private key pair for the EC/AC, adds the information of EC/AC public key into the outgoing Certificate Request message, signs the message with EC private key, adds the signature and EC (in the case of AC

applications) into the message and encrypts and integrity protects the message with the K*. Afterwards, it returns the reconstructed Certificate Request message back to the C-V2X application client.

4. In step 11~13, the C-V2X application client sends the C-V2X Application Request message to the NAF/AP. Before forwarding it, the NAF/AP retrieves the Ks_int_NAF with the received B-TID and derives the K*, then pushes the K* and related information along with the sent C-V2X Application Request message.
5. In step 14~15, the C-V2X application server verifies and decrypts the Certificate Request message, applies and issues the EC/AC and sends the Certificate Response message protected by the K* back.
6. In step 16~20, the C-V2X application client invokes the USIM to verify and decrypt the Certificate Response message with the K* and returns the result to the C-V2X application client. The same as step 17~21 of solution 1.

## 5.6 Derivation of K*

K* is the specific application session key bootstrapped by the enhanced GBA mechanism and used to protect the end-to-end communication in the application layer. It contains a series of keys used for various purposes, which forms a secure communication path between the C-V2X device and the C-V2X application server.

In the case of GBA_U, K* = KDF (Ks_int_NAF, String, B-TID, UE ID, Service ID). In the case of GBA_ME, K* = KDF (Ks_NAF, String, B-TID, UE ID, Service ID), where Ks_NAF is used instead.

KDF is the key derivation function which adopts the HMAC algorithm specified in RFC 2104 [8] and returns a key value of 128 bits in length. Ks_int_NAF/Ks_NAF is the session key bootstrapped by the standard GBA. UE_ID is the IMPI of the C-V2X device, Service ID is the FQDN of the C-V2X application server which should be globally unique. String is used to generate the specific keys for different purposes and recommended as follows.

| K* | String | Purpose |
|---|---|---|
| K1 | C-V2X_Enc | Used to encrypt the certificate request/response message transmitted between the C-V2X device and the C-V2X application server for EC/AC application. |
| K2 | C-V2X_Int | Used to integrity protect the certificate request/response message transmitted between the C-V2X device and the C-V2X application server for EC/AC application. |
| K3 | C-V2X_Auth | Optionally used. It can be derived and provided to the C-V2X application to enable mutual authentication between the C-V2X device and the C-V2X application server. |
| K4 | C-V2X_E2E_Sec | Optionally used. It can be derived and provided to the C-V2X application to enable the C-V2X device and the C-V2X application server to establish an end-to-end secure tunnel for further communication. |

**Table 1 String value recommendation for various K***

# 6 Technical Recommendations for Network Elements

## 6.1 C-V2X Application Client

The C-V2X application client takes responsibility for certificate management for the C-V2X device. It chooses the GBA type, i.e., GBA_ME or GBA_U mechanism, based on the security requirements of the user and runs the logic as specified by the C-V2X industry to implement EC/AC certificate application.

### 6.1.1 General functional requirements

| Functional requirements for the general function | |
|---|---|
| **A1** | The C-V2X application client shall be able to choose which mechanism, GBA_ME or GBA_U, shall be used according to the security requirements of the C-V2X industry. |
| **A2** | The C-V2X application client shall be able to invoke the GBA_init API to initiate GBA bootstrapping procedure and indicate GBA_ME/GBA_U to use. |
| **A3** | The C-V2X application client shall be able to set a timer for the K* based on its lifetime. |
| **A4** | The C-V2X application client shall be able to interact with the ECA/ACA server to apply for EC/AC. |
| **A5** | The C-V2X application client shall be able to set a timer to wait for AC downloading in the case of AC application. |

### 6.1.2 Functional requirements for the GBA_ME

| Functional requirements for the GBA_ME mechanism | |
|---|---|
| **B1** | The C-V2X application client shall be able to derive the K* based on the Ks_NAF returned by the GBA_init API to protect the messages for C-V2X certificate application. |
| **B2** | The C-V2X application client shall be able to use the Ks_NAF to calculate the HTTP Digest of the message transmitted between the C-V2X device and the NAF/AP to ensure its authenticity. |
| **B3** | The C-V2X application client shall be able to construct the Certificate Request message and use the K* to protect the message. |
| **B4** | The C-V2X application client shall be able to verify and decrypt the Certificate Response message returned from the ECA/ACA server with the K* and handle the message locally, e.g., storing the EC, waiting for downloading the AC. |

### 6.1.3 Functional requirements for the GBA_U

| Functional requirements for the GBA_U mechanism | |
|---|---|
| **C1** | The C-V2X application client shall be able to trigger the USIM to derive the K* based on the Ks_int_NAF to protect the messages for C-V2X certificate application. |
| **C2** | The C-V2X application client shall be able to use the Ks_ext_NAF to calculate the HTTP Digest of the message transmitted between the C-V2X device and the NAF/AP to ensure its authenticity. |
| **C3** | The C-V2X application client shall be able to construct the Certificate Request message and invoke the USIM to reconstruct the message and use the K* to protect the message. |
| **C4** | The C-V2X application client shall be able to invoke the USIM to verify and decrypt the Certificate Response message returned from the ECA/ACA server with the K* and get the handling result from the USIM. |

## 6.2 USIM Card

The USIM card is the basic SE that supports the generic functionalities and provides the local security environment for the C-V2X device. Its architecture is illustrated in Figure 7.

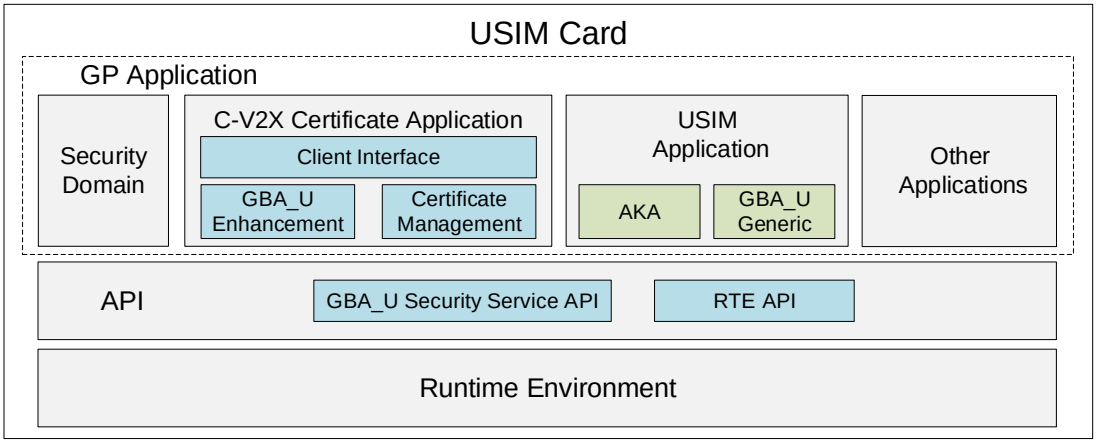


**Figure 7 USIM Card Architecture**

In the case of the GBA_ME mechanism, the USIM card only needs to support and provide the generic application functionality of the AKA authentication to the C-V2X device, as the C-V2X certificate management and K* derivation functions are implemented by the C-V2X application client outside the USIM.

In the case of the GBA_U mechanism, besides the generic application functionalities of AKA and GBA_U Generic in the USIM Application shown in green, the USIM card also needs to support the functionalities in the C-V2X Certificate Application and internal APIs highlighted in blue. The C-V2X Certificate Application implements the functionalities of the GBA_U enhancement, certificate management and client interface which provides service interfaces

to the C-V2X application client and the external user. The internal APIs include the GBA_U security service API, which implements cryptography operation based on Ks_int_NAF and the Runtime Environment (RTE) API, which implements general cryptography operation and secure storage capabilities. In this way the USIM card can always store and keep the cryptography parameters (including K*, EC and AC private keys corresponding to their certificates, at least) in its SE.

To guarantee the security of sensitive data on the C-V2X device, it is recommended to adopt GBA_U in preference to GBA_ME.

### 6.2.1 General functional requirements

| Functional requirements for the general function | |
|---|---|
| **D1** | The USIM card shall be able to support the UICC function, as specified in 3GPP TS 31.101 [9]. |
| **D2** | The USIM card shall be able to support the USIM application function, including generic functionality such as AKA, as specified in 3GPP TS 31.102 [10]. |
| **D3** | The USIM card shall be able to support the GBA_U generic function for GBA_U in USIM application, as specified in 3GPP TS 33.220 [5]. |
| **D4** | The USIM card shall be able to support GP application management to implement application loading, installing and deleting, as specified in Global Platform Card Specification [11]. |

### 6.2.2 Functional requirements for the GBA_U enhancement

| Functional requirements for the GBA_U enhancement | |
|---|---|
| **E1** | The USIM card shall be able to derive the GBA application session key K* from Ks_int_NAF. |

### 6.2.3 Functional requirements for the certificate management

| Functional requirements for the certificate management | |
|---|---|
| **F1** | The USIM card shall be able to reconstruct the Certificate Request message for EC, including:<br>• generate asymmetrical key pairs for EC;<br>• add the EC public key into the Certificate Request message;<br>• self-sign the Certificate Request message with the EC private key;<br>• encrypt and integrity protect the Certificate Request message with the K*. |
| **F2** | The USIM card shall be able to verify and decrypt the Certificate Response message for EC with the K* and store the EC certificate securely if the verification is successful. |
| **F3** | The USIM card shall be able to reconstruct the Certificate Request message for AC, including:<br>• generate asymmetrical key pairs for AC;<br>• add the AC public key into the Certificate Request message;<br>• sign the Certificate Request message with the EC private key; |

| | |
|---|---|
| | • encrypt and integrity protect the Certificate Request message with the K*. |
| **F4** | The USIM card shall be able to verify and decrypt the Certificate Response message for AC with the K* and return the plain message back to the C-V2X application client if the verification is successful. |

### 6.2.4 Functional requirements for the client interface

| **Functional requirements for the client interface** | |
|---|---|
| **G1** | The USIM card shall be able to support the interface to enable the C-V2X application client to initiate K* derivation. |
| **G2** | The USIM card shall be able to support the interface to enable the C-V2X application client to reconstruct Certificate Request message and verify the Certificate Response message for EC. |
| **G3** | The USIM card shall be able to support the interface to enable the C-V2X application client to reconstruct Certificate Request message and verify the Certificate Response message for AC. |
| **G4** | The USIM card shall be able to support the interface to enable the C-V2X application client to sign the C-V2X PC5 messages with the private key of AC. |

### 6.2.5 Functional requirements for the internal API

| **Functional requirements for the internal API** | |
|---|---|
| **H1** | The USIM card shall be able to support the GBA_U security service API to enable GBA_U enhancement functionality to derive K* from Ks_int_NAF according to Section 5.6. |
| **H2** | The USIM card shall be able to support the RTE API to enable certificate management functionality to implement asymmetrical key pair generation, secure storage, encryption/decryption, integrity protection/verification, digital signature, etc.. |

## 6.3 NAF/AP

The NAF/AP is the core network element to support the enhanced GBA mechanism. It shall derive the K* from Ks_(int)_NAF and other parameters and provide the K* to the C-V2X application server for use.

### 6.3.1 General functional requirements

| **Functional requirements for the general function** | |
|---|---|
| **I1** | The NAF/AP shall be able to support authentication proxy function, as specified in 3GPP TS 24.109 [12]. |
| **I2** | The NAF/AP shall be able to support network application function (NAF), as specified in 3GPP TS 33.220. |
| **I3** | The NAF/AP shall be able to establish TLS secure tunnel with the C-V2X application server in advance to guarantee the security of the |

| | |
|---|---|
| | messages transmitted between them. |

### 6.3.2 Functional requirements for Solution 1

| Functional requirements for the Solution 1 | |
|---|---|
| **J1** | The NAF/AP shall be able to forward the C-V2X Application Request message with the B-TID included to the C-V2X application server. |
| **J2** | The NAF/AP shall be able to get the B-TID and the FQDN of the C-V2X application server from the User Information Request message sent by the C-V2X application server. |
| **J3** | The NAF/AP shall be able to derive the GBA application session key K* corresponding to the B-TID and the FQDN of the C-V2X application server and set its lifetime. |
| **J4** | The NAF/AP shall be able to send the GBA application session keys K* and its lifetime back to the C-V2X application server. |

### 6.3.3 Functional requirements for Solution 2

| Functional requirements for the Solution 2 | |
|---|---|
| **K1** | The NAF/AP shall be able to derive the GBA application session key K* corresponding to the B-TID and the FQDN of the C-V2X application server and set its lifetime. |
| **K2** | The NAF/AP shall be able to send the B-TID, GBA application session key K* and its lifetime to the C-V2X application server when forwarding the C-V2X Application Request message. |

## 6.4 C-V2X application server

The C-V2X application server issues EC/AC certificates for the C-V2X device. It shall get the K* from the NAF/AP and use it to protect the C-V2X certificate management messages. Optionally, it can use the K* to mutually authenticate the C-V2X device and establish an end-to-end secure tunnel in the application layer for further communication.

### 6.4.1 General functional requirements

| Functional requirements for the general function | |
|---|---|
| **L1** | The C-V2X application server shall be able to use the K* to verify and decrypt the Certificate Request message sent by the C-V2X device. |
| **L2** | The C-V2X application server shall be able to use the K* to protect the Certificate Response message sent back to the C-V2X device. |
| **L3** | The C-V2X application server shall be able to establish TLS secure tunnel with the NAF/AP in advance to ensure the security of the messages transmitted between them. |
| **L4** | The C-V2X application server may be able to use the K* to authenticate the C-V2X device mutually and establish an end-to-end secure tunnel in the application layer for further communication. |

### 6.4.2 Functional requirements for Solution 1

| Functional Requirements for the Solution 1 | |
|---|---|
| **M1** | The C-V2X application server shall be able to interact with the NAF/AP to request the GBA application session key K* and its lifetime with the B-TID obtained from the C-V2X Application Request message and its FQDN. |

### 6.4.3 Functional requirements for Solution 2

| Functional Requirements for the Solution 2 | |
|---|---|
| **N1** | The C-V2X application server shall be able to get the B-TID, GBA application session key K* and its lifetime from the C-V2X Application Request message forwarded by the NAF/AP. |

## 7 Benefit Analysis

The GBA mechanism recommended in this guidelines document seeks to make C-V2X certificate provisioning operation easier to implement. Taking the traditional GBA mechanism as a basis, the enhancement features can be summarised in three points: NAF/AP multiplexing, GBA application session key provisioning and GBA_U priority. The table below provides an analysis of the benefits the enhancements bring for the service provider (e.g., C-V2X certificate service provider), end customer (e.g., automotive enterprises and C-V2X device vendors) and network operator.

| Feature | Description | Benefits for the C-V2X industry | | |
|---|---|---|---|---|
| | | For service provider | For end customer | For MNO |
| NAF/AP multiplexing | Use one or more NAF/AP deployed in MNO domain to serve several C-V2X application servers. | Does not need to deploy and maintain the NAF/AP network element in the C-V2X service provider domain, which realises cost savings. | Reduce the service cost when subscribing to a GBA service from an MNO. | Reduce the number of NAF/AP network elements that need to be deployed, which helps to reduce the cost of deploying the GBA service extensively. |
| GBA application session key provisioning | Bootstrap the C-V2X device with AKA authentication and derive the K*; Provide the K* to the application server/client for end-to-end security | Able to identify and authenticate the C-V2X devices, and establish application layer secure connections between the C-V2X device and the application server without pre- | Able to implement identification, authentication and certificate provisioning for the C-V2X devices online, which improves production efficiency, reduces | Enable the service provider and end customer to bootstrap the C-V2X device and use the GBA application session key freely, which helps them to accept and adopt |

| | | | | |
|---|---|---|---|---|
| | communications in the application layer. | configuration of any security credentials in advance, which helps reduce costs and brings flexibility for C-V2X certificate provisioning;<br>Able to implement the required security mechanisms with the flexible K*, which helps to meet the C-V2X industry security regulatory requirements. | costs and brings flexibility for C-V2X certificate provisioning;<br>Able to implement the required security mechanisms regulated by the C-V2X industry. | network operator's GBA service more easily. |
| GBA_U priority | Prefer GBA_U mechanism to GBA_ME for security reasons;<br>Take the USIM as the SE on the terminal side to ensure local and communication security of the C-V2X device. | Promoting security levels on the terminal side by ensuring that sensitive data (e.g., certificate private keys) is stored securely and the secure communication tunnel in the application layer is terminated in the SE, which helps to realise safe production and management for C-V2X certificate provisioning. | Get local and communication security assurance of the C-V2X device, which helps to avoid system-wide production line updates for C-V2X certificate provisioning, including the establishment of the physical security environment, network security systems, personnel training and improvement of management processes, and save costs. | Able to provide security assured GBA service with the USIM as the security root for various service providers and end customers, which helps MNOs to explore new value-added services in the C-V2X industry. |

In general, this enhanced GBA mechanism can securely implement C-V2X certificate provisioning online through 4G/5G cellular networks. The mechanism greatly reduces the deployment, operation and maintenance costs for MNOs and saves the production and management costs for their customers (including both service providers and end customers), thus the enhanced mechanism is considered to be secure, effective, low-cost and quicker to implement.

# 8 Security Considerations

This document describes a mechanism to provision the certificates in the C-V2X scenarios online. Without loss of generality, this mechanism can also be used in other scenarios that need end-to-end secure connections between terminal devices and the application servers to implement communication security.

The security of such a connection mainly depends on the GBA type, the cryptography keys derived based on the enhanced GBA and the cryptography algorithm deployed in the entire system. Therefore, careful consideration should be given by the administrator to decide the GBA type, the parameter values of the key length and the key lifetime, and to select the cryptography algorithms used to derive the keys and protect the connections.

As different countries/regions may have specific regulations on the use of cryptography technologies, no recommendation on cryptography algorithms is given here. It should be noted that the selection of cryptography algorithms should be in accordance with local laws and regulations where the enhance GBA mechanism is used.

# Annex A Document Management

## A.1 Document History

| Version | Date | Brief Description of Change | Approval Authority | Editor / Company |
|---|---|---|---|---|
| 1.0 | 26th Jan 2022 | New PRD presented for ISAG approval | GSMA ISAG | Jie Ma, China Mobile Limited |

## A.2 Other Information

| Type | Description |
|---|---|
| Document Owner | GSMA, Fraud and Security Group |
| Editor / Company | Jie Ma, China Mobile Limited |
| Contributors | Ye Tian (China Mobile Limited)<br>Li Su (China Mobile Limited)<br>Hui Liu (China Mobile Limited)<br>Rong Zhang (China Telecom)<br>Louis Lou (Huawei)<br>Rujun Li (ZTE)<br>Chris Mulley (ZTE)<br>Hervé Collet (Thales) |

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at prd@gsma.com

Your comments or suggestions & questions are always welcome.