PROGRAMME OF THE EUROPEAN UNION

NAVIGATION MADE IN EUROPE

GALILEO OPEN SERVICE NAVIGATION MESSAGE AUTHENTICATION INTERNET DATA DISTRIBUTION INTERFACE CONTROL DOCUMENT (OSNMA IDD ICD)

Issue 1.0 | July 2023

#EUSpace

TERMS OF USE AND DISCLAIMERS

Authorised use and scope of use

The European GNSS (Galileo) Open Service Navigation Message Authentication Internet Data Distribution Interface Control Document Issue 1.0 (hereinafter referred to as OSNMA IDD ICD) and the information contained herein is made available to the public by the European Union (hereinafter referred to as Publishing Authority) for information, standardisation, research and development and commercial purposes for the benefit and the promotion of the European Global Navigation Satellite Systems programmes (European GNSS Programmes) and according to terms and conditions specified thereafter.

General Disclaimer of Liability

With respect to the OSNMA IDD ICD and any information contained in the OSNMA IDD ICD, neither the EU as the Publishing Authority nor the generator of such information make any warranty, express or implied, including the warranty of fitness for a particular purpose, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information hereby disclosed or for any product developed based on this information, or represents that the use of this information would not cause damages or would not infringe any intellectual property rights. No liability is hereby assumed for any direct, indirect, incidental, special or consequential damages, including but not limited to, damages for interruption of business, loss of profits, goodwill or other intangible losses, resulting from the use of the OSNMA IDD ICD or of the information contained herein. Liability is excluded as well for consequences of the use and / or abuse of the OSNMA IDD ICD or the information contained herein.

Copyright

The OSNMA IDD ICD is protected by copyright which belongs to the European Union. Any alteration or translation in any language of the OSNMA IDD ICD as a whole or parts of it is prohibited unless the Publishing Authority provides a specific written prior Permission.

The OSNMA IDD ICD may only be partly or wholly reproduced and/or transmitted to a third party in accordance with the herein described permitted use and under the following conditions: the present “Terms of Use and Disclaimers”, are accepted, reproduced and transmitted entirely and unmodified together with the reproduced and/or transmitted information; the copyright notice “© European Union 2023” is not removed from any page.

Miscellaneous

No failure or delay in exercising any right in relation to the OSNMA IDD ICD or the information contained therein shall operate as a waiver thereof, nor shall any single or partial exercise preclude any other or further exercise of such rights. The disclaimers contained in this document apply to the extent permitted by applicable law.

Reference is made in this OSNMA IDD ICD to documents, standards or other information from third parties, in particular the Internet Engineering Task Force (IETF). The use of these documents, standards or other information is under the sole responsibility of the users and such use may be subject to terms and conditions determined by these third parties.

Updates

The OSNMA IDD ICD could be subject to modification, update and variations. Those modifications, updates and variations will reflect, among others, the result of the execution of the Public Observation phase.

The publication of updates will be subject to the same terms as stated herein unless otherwise evidenced.

Although the Publishing Authority will deploy its efforts to give notice to the public for further updates of OSNMA IDD ICD, it does not assume any obligation to advise on further developments and updates of the OSNMA IDD ICD, nor to consider any inputs, comments proposed by interested persons or entities, involved in the updating process.

ISBN: 978-92-9206-073-2

DOI: 10.2878/325903

DOCUMENT CHANGE RECORD

REASON FOR CHANGE	ISSUE	REVISION	DATE
First version of the document	1	0	July 2023

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

1.1 Purpose and Scope of the document

The present Galileo Open Service Navigation Message Authentication Internet Data Distribution Interface Control Document (hereinafter referred to as OSNMA IDD ICD) aims at complementing the OSNMA Receiver Guidelines [1] and the OSNMA SIS ICD [2] by providing the users with the information required to access and retrieve the cryptographic data (Public Key and Merkle Tree) available via the EGNSS GNSS Service Centre (GSC) interfaces. The distribution of the mentioned cryptographic data is supported by the provision of Public Key Infrastructure (PKI) certificates to ensure that the data is coming from the Galileo System. A description of the PKI certificates is also given in this document together with information to the users on how they can be used.

This first version of the document will be updated before the service declaration with the inclusion of a PKI certificate signing the Merkle Tree.

The information provided in this document along with the OSNMA SIS ICD [2] and the OSNMA Receiver Guidelines [1] shall allow the full implementation of the OSNMA protocol including the authentication of the associated chain of trust.

1.2 Structure of the document

The document is organised as follows:

- Section 1: this section provides a brief introduction to the document.
- Section 2: provides a general overview of the OSNMA, the PKI and also the Internet Data Distribution.
- Section 3: describes how to access the GSC interface to retrieve the Public Key (PK) and the Merkle Tree (MT) products.
- Section 4: describes the PKI certificates and explains how to use them to verify the authenticity of the PK and the MT products.
- Annex A : lists the applicable and reference documents.
- Annex B : provides the list of acronyms.
- Annex C : contains the XSD files used in the generation of the xml files available to the users at the GSC interface.
- Annex D : contains the attributes of the different PKI certificates and Certificate Revocation Lists (CRL).

2 OSNMA, PKI AND THE INTERNET DATA DISTRIBURION OVERVIEW

2.1 What is OSNMA

The Open Service Navigation Message Authentication (OSNMA) is a data authentication function of the Galileo Open Service, allowing users to confirm that the Galileo OS Navigation Data has originated from the Galileo system and has not been modified.

The authentication concept is based on two main principles:

- The use of keys from a single one-way chain shared by the Galileo satellites through a Timed Efficient Stream Loss-tolerant Authentication (TESLA) protocol.
- The possibility to authenticate satellites which do not transmit OSNMA with the data retrieved from satellites transmitting OSNMA, referred to as cross-authentication.

Both principles reduce the computation and communication overhead, and increase the service availability and robustness to data loss (see [1] and [2] for full details on the OSNMA protocol).

From a receiver perspective, the processing of the OSNMA data can be described at a high level by the following steps, illustrated in Figure 1:

- The receiver retrieves the **navigation data** and the corresponding OSNMA data (**tag**, **TESLA chain key** and **TESLA root key**). The **tag** authenticates the **navigation data** and is received before its associated **TESLA chain key**.
- The **TESLA root key** is authenticated by means of its digital signature using a **Public Key**[1] that shall be available at the receiver.
- The receiver authenticates the **TESLA chain key** with the **TESLA root key** or with a previously authenticated key from the TESLA chain.
- The receiver re-generates locally the **tag** with the verified **TESLA chain key** and the **data**, and checks whether it coincides with the received **tag**.

In addition to this, in order to verify the **Public Key** (in case a new PK is provided or the OSNMA Alert Message is transmitted), the receiver must also store the **Partial Merkle Tree** containing the root and associated intermediate nodes[2]. PKI certificates will be made available to the users/manufacturers in order to verify that the root node of the Merkle Tree is coming from the Galileo system. These certificates may also be used for the verification of the Public Key. Refer to sections 4.3 and 4.4 for further details on how to verify the root node of the Merkle Tree and the PK.

If the result of all these steps is successful the user shall consider the **navigation data** as authentic.

[1] The Public Key is made available to the users via the SIS (see [1] and [2]) and via the GSC interfaces as per section 3.2.1.

[2] Full details on the use of the Merkle Tree for the verification of the PK and the concept of Partial Merkle Tree can be found in section 5.1 of [1].

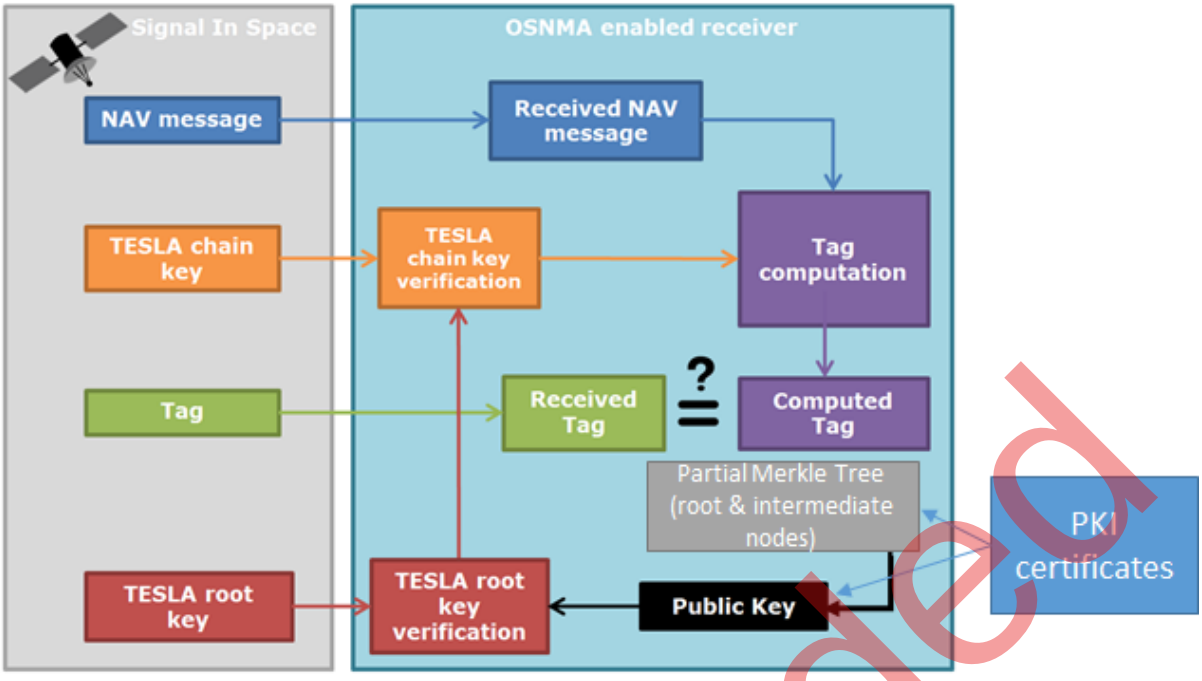

Figure 1: OSNMA processing logic including the PKI certificates

The retrieval of the data and operations required to perform these verification steps are further detailed in [1].

2.2 The Public Key Infrastructure

This section provides high level description of the Public Key Infrastructure (PKI) and definitions used for OSNMA.

A PKI is a set of hardware, software, people, policies and procedures needed to create, manage, distribute, use, store and revoke certificates. The purpose of a PKI is to make sure that the certificate can be trusted.

A digital certificate is an electronic data structure that binds an entity, being an institution, a person, a computer program, a web address etc. to its public key. Digital certificates are used for secure communication, using public key cryptography and digital signatures.

The general principles described above are depicted in Figure 2.

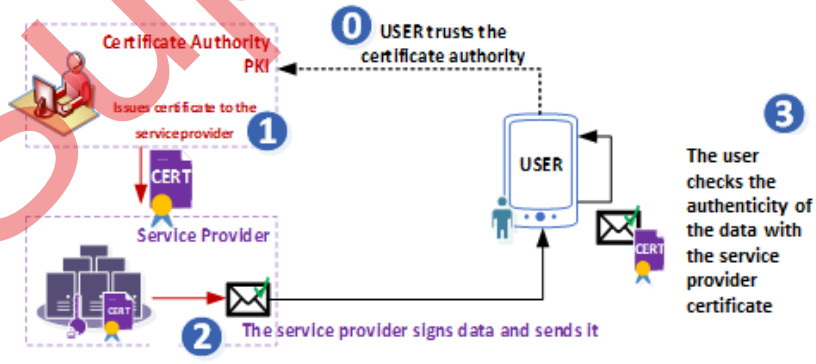

Figure 2: General principle of a certification process using a PKI

Within the scope of the OSNMA, the PKI provides the OSNMA users with digital certificates organised hierarchically that allow to verify the authenticity of the public cryptographic material provided through the GSC interface (see section 3) that are needed to authenticate the OSNMA data coming from the SiS. This is the so-called chain of trust and it is represented in Figure 3.

This chain of trust is organised as a 3-tier PKI hierarchy of certificate authorities. A Certificate Authority (CA) is an entity that stores, signs and issues digital certificates. The Root CA (RCA) in Figure 3 corresponds to the trusted certificate authority in Figure 2, and is the root of trust in the PKI. EUSPA manage the RCA on behalf of all the EU space programmes. The Subordinate CA (SCA) is an intermediate authority that is responsible for signing all certificates associated with the Galileo programme. Finally, the Issuing CA in this instance is the CA responsible for certificates associated with OSNMA and is maintained by the GSC. The EUSPA OSNMA ICA manages the OSNMA Public Key certificates corresponding to the End Entities (EEs) in Figure 3.

Another important element of the chain of trust is the Certificate Revocation List (CRL). The CRL is a list of digital certificates that have been revoked by each CA before their scheduled expiration date and should no longer be trusted. For further details on how users and manufacturers can use this chain of trust within the OSNMA context, please refer to section 4.

The Certificate Policy and Certification Practice Statement (CP/CPS) documents for each certification authority (RCA, SCA and ICA) provide details about the certification policy and practices that apply when issuing digital certificates. Also, the documents describe the general rules for providing certification services such as: registration, public key certification, key and certificates rekey and certificate revocation.

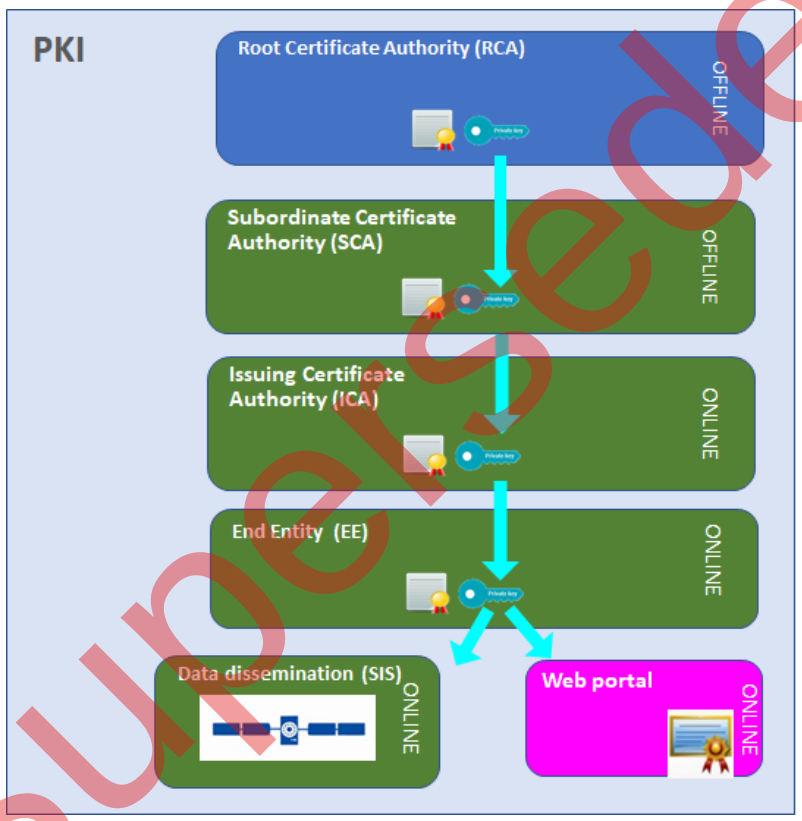

Figure 3: PKI chain of trust

2.3 The Galileo OSNMA Internet Data Distribution

The context of the user receiver interfaces is shown in Figure 4.

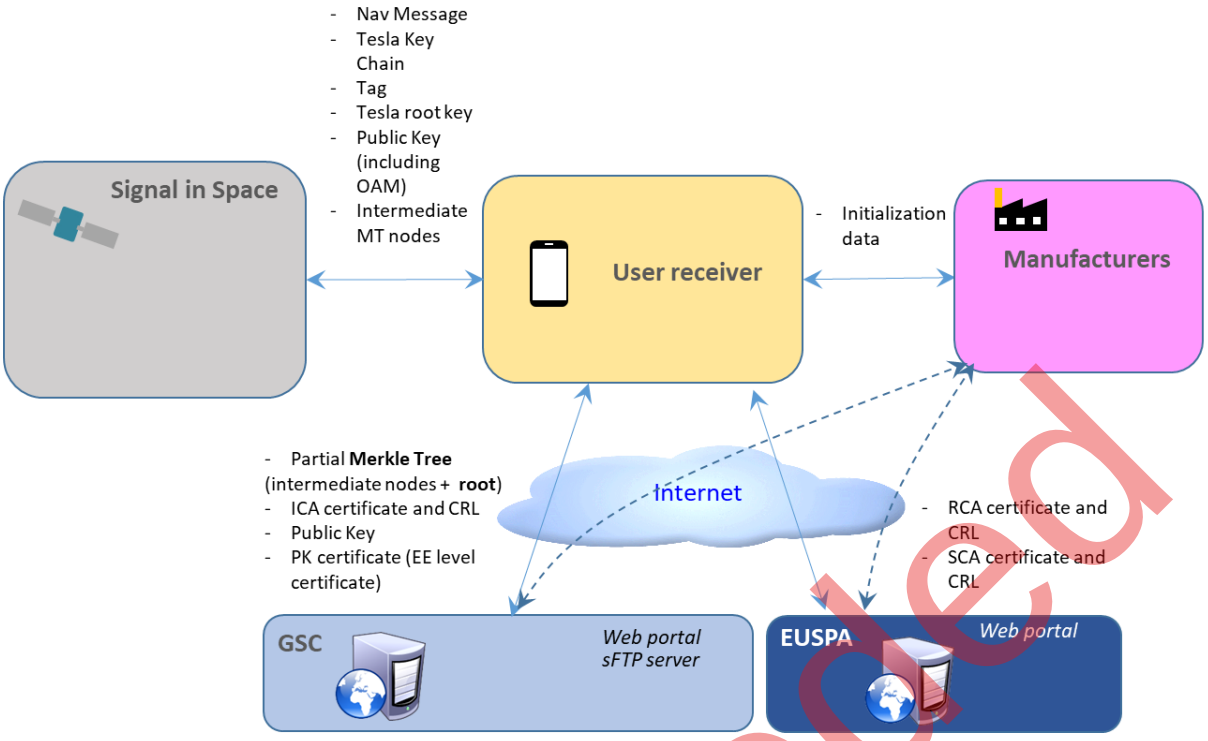

Figure 4: User receiver interface context

The SIS interface is fully described in [2]. The present document focuses in the products provided via the Internet and in the steps the users need to follow in order to retrieve the Partial Merkle Tree[3] and the Public Key from the interfaces provided at the GSC (see section 3) and how to retrieve and use the PKI certificates provided via the GSC interface and the EUSPA web portal in order to verify both the PK and the root node of the MT (see section 4). Section 4.3 provides guidelines of how the manufacturers should initialize the trust store[4] in the receivers.

[3] It is worth to remark that the root node of the Merkle Tree is the only cryptographic product that it is not provided through the SIS and can only be retrieved from the Internet Data Distribution Interface at the GSC.

[4] A trust store is a collection of cryptographical elements that are trusted by default.

3 OSNMA CRYPTOGRAPHIC PRODUCTS AT THE GSC INTERFACE

The aim of this section is to explain how the user can retrieve the OSNMA cryptographic material from the European GNSS Service Centre (GSC) interface[5].

The Galileo OSNMA cryptographic material is available by two means:

- Under the GSC products section of the **GSC website**, users can access the OSNMA cryptographic material provided via the web portal (Merkle Tree, Public Key and associated certificates) applicable, future and past versions.
- At the **Galileo OSNMA SFTP** server users can access the applicable and future versions of the OSNMA cryptographic material provided via this interface (Merkle Tree, Public Key and associated certificates).

The ICA CP/CPS document [5] is also available at this interface. See section 4 for further details on this document.
In order to access to the available Galileo OSNMA cryptographic material, users shall register in the GNSS Service Centre web portal and request access to the OSNMA Internet Data Distribution products of interest.

3.1 Registration to the GSC web portal and access request to the OSNMA products

Users can access the web portal via https://www.gsc-europa.eu/ and follow the steps for registration by clicking the "Register" link at the top of the home page. During the registration process, the user can select and request access to the OSNMA products from the GSC web portal and the SFTP server. Already registered users can modify their subscription by logging onto their account and clicking in the "My Account" link on top of the page and then clicking in the "Request Access Products"[6].

The user can also select if they want to subscribe to ad-hoc notifications in the following cases by email:

- When a new PK product is published in the web portal and SFTP server.
- When a new MT product is published in the web portal and SFTP server.
- When any of the PKI certificates is renewed or revoked (see section 4).

3.2 OSNMA Products at the GSC web portal

The OSNMA products available to the users via the GSC interface are the Public Key and the Merkle Tree.

3.2.1 OSNMA Public Key product

A user registered at the web portal and subscribed to the OSNMA products can check the applicable, future and historical Public Keys in *"GSC Products → OSNMA_PublicKey"*.

[5] As highlighted in Figure 4, the RCA and SCA PKI certificates and associated CRLs are not available at the GSC interface but at the EUSPA web portal. See section 4 for further details on where to find the different PKI certificates.

[6] During the Public Observation phase prior to the Initial Service Provision, the users shall follow the instructions in the web portal to register to the Public Observation Test phase.

3.2.1.1 Public Key currently in force

The Public Key currently in force can be found under *"GSC Products → OSNMA_PublicKey → Applicable"*. The different Galileo OSNMA Public Key products available with their respective file naming convention and format are provided in Table 1. For each file, its MD5 checksum is also available for download.

PRODUCT	FILE NAMING CONVENTION	FORMAT
OSNMA Public Key	OSNMA_PublicKey_YYYYMMDDhhmmss.xml	XML
OSNMA Public Key MD5	OSNMA_PublicKey_YYYYMMDDhhmmss.xml.md5	MD5[7]
Public Key Certificate	OSNMA_PublicKeyCRT_YYYYMMDDhhmmss_[newPKID]_X.crt	CRT
Certificate MD5	OSNMA_PublicKeyCRT_YYYYMMDDhhmmss_[newKPID]_X.crt.md5	MD5
Public Key Certificate Revocation list	OSNMA_PublicKeyCRL_YYYYMMDDhhmmss_[newKPID]_X.crl	CRL
Revocation list MD5	OSNMA_PublicKeyCRL_YYYYMMDDhhmmss_[newKPID]_X.crl.md5	MD5

Table 1: Public Key product selection

The Public Key can be downloaded in XML format with the *"Download product xml file"* link. The XML has the standard file structure:

- signalData: contains a header element and a body element.
- header: contains a standard "GAL-header" (see Annex C for the header schema).
- body: contains a single PublicKey element.

A PublicKey element contains:

- UID: a string. It is a unique ID of the Public Key.
- Applicability: applicability time.
- State: product availability State.
- i: an integer that indicates the Merkle Tree leaf associated to the PK.
- PKType: a string. It indicates the type of the Public Key.
- lengthInBits: length of the Public Key in bits.
- point: the compressed public point (PK) encoded in base16[8].
- Certificate: reference to Associated Certificate.
- CRL: reference to Associated Certificate Revocation List.
- PKID: an integer. It is the ID of the Public Key within the associated Merkle Tree.

Refer to Annex C for full details on the XSD grammar.

The *"Download product crt file"* option allows the user to download a PEM-encoded file with the X.509 certificate bundle for the Public Key. It is compatible with [6]. The certificate bundle contains the Public Key Certificate[9] along with the Issuing CA (ICA) certificate.

[7] MD5 is used only for compatibility reasons with existing standards/equipment.

[8] https://tools.ietf.org/html/rfc4648

[9] The PK certificate can also be referred to as End Entity (EE) PKR certificate.

The *"Download product crl file"* option allows the user to download a PEM-encoded file with the Certificate Revocation List (CRL) for the revoked Public Keys. It is compatible with [6].

Further information on the use of the certificates and CRL is provided in section 4.

3.2.1.2 Future Public Key

When the renewal of a Public Key is expected, the user can check the future Public Key in *"GSC Products → OSNMA_PublicKey → Future"*. The user can check and download the same information and products available for the current Public Key (see Table 1) for the future OSNMA Public Key. In nominal operations, when no Public Key renewal is expected, this page would appear empty.

3.2.1.3 Accessing past renewed or revoked Public Keys

The user can check past Public Keys in *"GSC Products → OSNMA_PublicKey → Historical"* and review the list of previous Public Keys in the historical records. The user needs to first select the product of interest by clicking on the date under the historical records to get re-directed to the products as shown in Table 1 for the specific date selected where it can be downloaded by clicking on the product .xml file.

3.2.2 OSNMA Merkle Tree product

A user registered at the web portal and subscribed to the OSNMA products can check the applicable, future and historical Public Keys in *"GSC Products → OSNMA_MerkleTree"*.

3.2.2.1 Merkle Tree currently in force

The current Merkle Tree in force can be checked under "*GSC Products → OSNMA_MerkleTree → Applicable"*. The different Galileo OSNMA Merkle Tree products available with their respective file naming convention and format are provided in Table 2. For each file, its MD5 checksum is also available for download.

PRODUCT	FILE NAMING CONVENTION	FORMAT
OSNMA Merkle Tree	OSNMA_ MerkleTree _YYYYMMDDhhmmss.xml	XML
OSNMA Merkle Tree MD5	OSNMA_ MerkleTree _YYYYMMDDhhmmss.xml.md5	MD5[10]

Table 2: Merkle Tree available product selection

Users can download the Merkle Tree in XML format with the *"Download product xml file"* link.

The Merkle Tree XML has the standard structure:

- signalData: contains a header element and a body element.
- header: contains a standard "GAL-header" (see Annex C for the header schema).
- body: contains a single MerkleTree element.

A MerkleTree element contains:

- UID: a string. It is a unique ID of the Merkle Tree.
- Applicability: applicability time.
- State: product availability State.

[10] MD5 is used only for compatibility reasons with existing standards/equipment.

- N: an integer. It is the number of Public Keys in the base of the Merkle Tree.
- HashFunction: a string defining which hash function was used to compute the Merkle Tree nodes.
- PublicKey elements (refer to section 3.2.1). The number of keys depends on the number of PK elements already published.
- The necessary TreeNodes required for the verification of the PK[11]. A TreeNode contains:
    - j: an integer. It is the height of the node in the Merkle Tree according to [2].
    - i: an integer. It is the position of the node in the Merkle Tree level according to [2].
    - lengthInBits: the length in bits of the hash in the x_ji element.
    - x_ji: a string with the base16 encoded Merkle Tree node.

Refer to Annex C for full details on the XSD grammar.

3.2.2.2 Future Merkle Tree

When the renewal of a Merkle Tree is expected, the user can check the future Merkle Tree in "*GSC Products → OSNMA_MerkleTree → Future*". The user can check and download the same information and products available in Table 2 for the future OSNMA Merkle Tree.

It is to be noted that the renewal of the MT is expected to take place very rarely, typically after more than 10 years as stated in [2] and that **the future Merkle Tree root is expected to be available on the GSC user interface at least two years before the planned renewal[12]** . In nominal operations, when no Merkle Tree renewal is expected, this page would appear empty.

3.2.2.3 Accessing past Merkle Trees

The user can check past Merkle Trees in "*GSC Products → OSNMA_MerkleTree → Historical*" and review the list of previous Merkle Trees in the historical records. The user needs to first select the product of interest by clicking on the date under the historical records to get re-directed to the products as shown in Table 2 for the specific date selected where it can be downloaded by clicking on the product .xml file.

3.3 SFTP Server interface

The GSC SFTP server has a dedicated directory to publish OSNMA cryptographic material. In order to retrieve the OSNMA products by means of the SFTP protocol, the user should register in the GNSS Service Centre and request access to the products as described in section 3.1.

The connection details of the GSC OSNMA SFTP are provided in Table 3.

Host	osnma.gsc-europa.eu
User	%gsc web portal username%
Pwd	%gsc web portal password%
Port	2222

Table 3: SFTP server connection details

The Galileo OSNMA products that can be retrieved from the SFTP server are the same as in the web portal: OSNMA_MerkleTree, OSNMA_PublicKey and the associated certificates.

The Galileo OSNMA products are structured in a folder tree, containing all the products and additional information related to these products. There is a dedicated folder for each product: OSNMA_MerkleTree and OSNMA_PublicKey.

[11] Full details on the use of the Merkle Tree for the verification of the PK including the need of intermediate nodes can be found in section 5.1 of [1].

[12] This applies to the Service Provision phase but not to the Public Observation Test phase.

Within each product folder, the user can find the Applicable and Future folders where the products and the relevant PKI certificates are stored.

The description of the products provided in sections 3.2.1 and 3.2.2 is also applicable to the ones provided via the SFTP server.

4 MERKLE TREE ROOT[13] AND PUBLIC KEY VERIFICATION USING PKI CERTIFICATES

As stated in section 2.2, the PKI certificates are part of a 3-tier PKI hierarchy:

- 1st: Root CA (RCA),
- 2nd: Subordinate CA (SCA),
- 3rd: Issuing CA (ICA)

Each certificate should be verified against the CRL published by their certificate authority such as:

- RCA CRL manages SCA certificates,
- SCA CRL manages ICA certificates,
- ICA CRL manages EE certificates.

Manufacturers and users should also download the CP/CPS documents for each certification authority ([3], [4] and [5]). These documents provide details about the certification policy and practices that apply when issuing digital certificates. Also, the documents describe the general rules for providing certification services such as: registration, public key certification, key and certificates rekey and certificate revocation.

4.1 PKI elements format and access

4.1.1 RCA elements

The RCA elements described in the next subsections can be found and retrieved from the EUSPA web portal https://euspa.europa.eu/about/how-we-work/pki.

4.1.1.1 RCA certificate

The RCA Certificate is a X509v3 certificate in a '.crt' file, (Base64 encoded – PEM – ASN.1 standard [7]).

As stated in the standard [6], the integrity of the certificate is provided by the electronic signature (using active RCA private key) of the data contained in the certificate.

The attributes of this certificate are described in Annex D .

Note: the RCA Certificate includes RCA Public Key.

4.1.1.2 RCA CP/CPS

The RCA CP/CPS is a pdf document described in [3] that defines for the RCA:

- Requirements and standards imposed by the PKI with respect to the various topics.
- How a CA and other participants in a given domain implement procedures and controls to meet the requirements stated in the CP.

4.1.1.3 RCA CRL

The RCA CRL is a X509v2 CRL in a '.crl' file, (Base64 encoded – PEM – ASN.1 standard [7]).

[13] The Merkle Tree verification using PKI certificates is not available during the Public Observation test phase but it will be available as of the Service Declaration.

As stated in the standard [6], the integrity of the CRL is provided by the electronic signature (using the active RCA private key) upon the data contained in the CRL. No other integrity protection mechanisms are required.

The attributes of this CRL are described in Annex D .

Note: RCA CRL contains a list of IDs of revoked SCA certificates.

4.1.2 SCA elements

The SCA elements described in the next subsections can be found and retrieved from the EUSPA web portal https://euspa.europa.eu/about/how-we-work/pki.

4.1.2.1 SCA certificate

SCA Certificate is a X509v3 certificate in a '.crt' file, (Base64 encoded – PEM – ASN.1 standard [7]).

As stated in the standard [6], the integrity of the certificate is provided by the electronic signature (using active RCA private key) upon the data contained in the certificate.

The attributes of this certificate are described in Annex D .

Note: SCA Certificate includes SCA Public Key.

4.1.2.2 SCA CP/CPS

SCA CP/CPS is a pdf document described in [4] that defines for SCA:

- Requirements and standards imposed by the PKI with respect to the various topics,
- How a CA and other participants in a given domain implement procedures and controls to meet the requirements stated in the CP.

4.1.2.3 SCA CRL

SCA CRL is a X509v2 CRL in a '.crl' file, (Base64 encoded – PEM – ASN.1 standard [7]).

As stated in the standard [6], the integrity of the CRL is provided by the electronic signature (using active SCA private key) upon the data contained in the CRL. No other integrity protection mechanisms are required.

The attributes of this CRL are described in Annex D

Note: SCA CRL includes list of ID of revoked ICA certificates.

4.1.3 ICA elements

The ICA elements are available at the GSC interface as described in sections 3.2.1 and 3.3.

4.1.3.1 ICA certificate

The ICA Certificate is a X509v3 certificate bundled with the EE certificate in the same .crt file (Base64 encoded – PEM – ASN.1 standard [7]) as indicated in section 3.2.1.1.

As stated in the standard [6], the integrity of the certificate is provided by the electronic signature (using the active SCA private key) of the data contained in the certificate.

Attributes of this certificate are described in Annex D .

Note: the ICA Certificate includes the ICA Public Key.

4.1.3.2 ICA CP/CPS

The ICA CP/CPS is a pdf document described in [5] that defines for ICA:

- Requirements and standards imposed by the PKI with respect to the various topics.

- How a CA and other participants in a given domain implement procedures and controls to meet the requirements stated in the CP.

4.1.3.3 ICA CRL

The ICA CRL is a X509v2 CRL in a '.crl' file (Base64 encoded – PEM – ASN.1 standard [7]).

As stated in the standard [6], the integrity of the CRL is provided by the electronic signature (using the active ICA private key) of the data contained in the CRL. No other integrity protection mechanisms are required.

The attributes of this CRL are described in Annex D .

Note: the ICA CRL contains a list of IDs of revoked EE certificates.

4.1.4 EE PKR Elements

4.1.4.1 EE PKR certificate

An EE PKR certificate is a X509v3 certificate in a '.crt' file (Base64 encoded – PEM – ASN.1 standard [7]) which also includes the ICA certificate (see section 3.2.1.1).

The EE PKR certificate is available at the GSC interface as described in sections 3.2.1 and 3.3.

The integrity of this certificate is provided by the electronic signature (using the active ICA private key) of the data contained in the certificate.

Attributes of the certificate are described in Annex D

4.2 PKI certificates verification

4.2.1 EE PKR certificate validity

An EE PKR certificate is considered valid when compliant to the certificate path validation described in section §6 of [6]. Every attribute of the file should be checked with:

- ⇨ The issuer of the EE PKR certificate is the subject of the active and valid ICA certificate,
- ⇨ The certificate validity period includes the current time,
- ⇨ The signature of the EE PKR certificate is valid (using the ICA public key algorithm, the ICA public key, and the ICA key parameters),
- ⇨ At the current time, the certificate is not revoked in the valid ICA CRL (section §6.3 of [6]).

4.2.2 ICA certificate validity

An ICA certificate is considered valid when compliant to certificate path validation described in section §6 of [6]. Every attribute of the file should be checked with:

- ⇨ The issuer of the ICA certificate is the subject of the active and valid SCA certificate,
- ⇨ The certificate validity period includes the current time,
- ⇨ The signature of the ICA certificate is valid (using the SCA public key algorithm, the SCA public key, and the SCA key parameters).
- ⇨ At the current time, the certificate is not revoked in the valid SCA CRL (section §6.3 of [6]).

4.2.3 ICA CRL validity

An ICA CRL is considered valid when compliant to complete CRL validation described in section §6.3 of [6]. Every attribute of the file should be checked with:

- ⇨ The issuer of the ICA CRL is the subject of the active and valid ICA certificate,

- ⇨ The CRL validity period may include the current time: in case of current time higher than next update, a warning should be raised but the CRL remains valid,
- ⇨ The signature of the ICA CRL is valid (using the ICA public key algorithm, the ICA public key, and the ICA key parameters).

4.2.4 SCA certificate validity

An SCA certificate is considered valid when compliant to certificate path validation described in section §6 of [6]. Every attribute of the file should be checked with:

- ⇨ The issuer of the SCA certificate is the subject of the active and valid RCA certificate,
- ⇨ The certificate validity period includes the current time,
- ⇨ The signature of the SCA certificate is valid (using the RCA public key algorithm, the RCA public key, and the RCA key parameters),
- ⇨ At the current time, the certificate is not revoked in the valid RCA CRL (section §6.3 of [6]).

4.2.5 SCA CRL validity

An SCA CRL is considered valid when compliant to complete CRL validation described in section §6.3 of [6]. Every attribute of the file should be checked with:

- ⇨ The issuer of the SCA CRL is the subject of the/an active and valid SCA certificate,
- ⇨ The CRL validity period may include the current time: in case of current time higher than next update, a warning should be raised but the CRL remains valid,
- ⇨ The signature of the SCA CRL is valid (using the SCA public key algorithm, the SCA public key, and the SCA key parameters).

4.2.6 RCA certificate validity

An RCA certificate is considered valid when compliant to certificate path validation described in section §6 of [6]. Every attribute of the file should be checked with:

- ⇨ The issuer of RCA certificate is the subject of the RCA certificate,
- ⇨ The certificate validity period includes the current time,
- ⇨ The signature of the RCA certificate is valid (using the RCA public key algorithm, the RCA public key, and the RCA key parameters).

4.2.7 RCA CRL validity

An RCA CRL is considered valid when compliant to complete CRL validation described in section §6.3 of [6]. Every attribute of the file should be checked with:

- ⇨ The issuer of the RCA CRL is the subject of the active and valid RCA certificate,
- ⇨ The CRL validity period may include the current time: in case of current time higher than next update, a warning should be raised but the CRL remains valid,
- ⇨ The signature of the RCA CRL is valid (using the RCA public key algorithm, the RCA public key, and the RCA key parameters).

4.3 Merkle Tree root verification and receiver initialization by the manufacturers

The verification of the Merkle Tree root is not available at the time of publication of this document but, during the service phase the root node of the Merkle Tree will be signed by a PKI certificate that will allow its verification.

Once this is available, receiver manufacturers will have to:

- Check the authenticity of the Merkle Tree file signed by the related PKI certificate through its associated chain of trust.

- Initialize the trust store of the receivers with the authenticated and valid root node of the Merkle tree that is available at the GSC interface (see section 3).

Once the Merkle Tree root is introduced in the trust store, there will be no need to check its authenticity again.

4.4 Public Key verification

Once the root node of the Merkle Tree is placed in the Trust Store during receiver initialization, the Public Key can be verified using the data provided via the SIS following the steps as described in [1].

However, the user also has the option to verify the PK using the PKI certificates through the following chain of trust:

- ⇨ EE PKR certificate (section 4.2.1) with ICA CRL (section 4.2.3),
- ⇨ ICA certificate (section 4.2.2) with SCA CRL (section 4.2.5),
- ⇨ SCA certificate (section 4.2.4) with RCA CRL (section 4.2.7),
- ⇨ RCA certificate (section 4.2.6).

4.5 Validity period of the certificates

For continuity of service purposes, when renewing the certificate at each PKI level, there will be an overlap (validity period of each certificate) between the active and future certificates as described in Figure 5 (in the example the validity period of xCA2 starts before validity period of xCA1 expires). This overlap is driven by notification to registered users when a new future CRL/certificate is published as described in section 3.1.

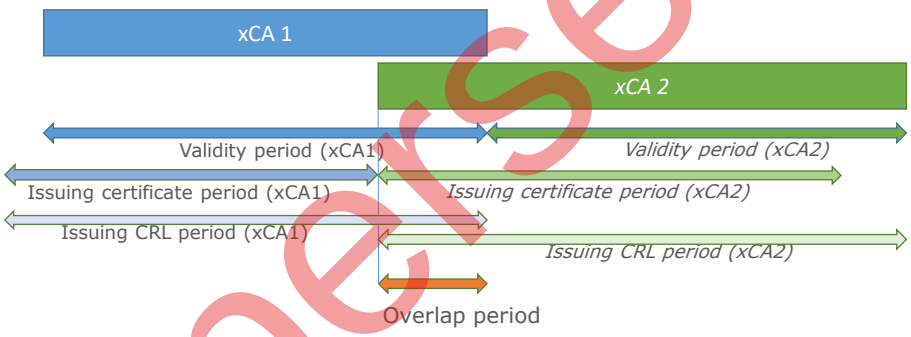

Figure 5: Overlap period of the certificates

Annex A Applicable and Reference Documents

A.1. Applicable Documents

[1] Galileo OSNMA Receiver Guidelines, Issue 1.0, European Union, December 2022.
[2] Galileo OSNMA SIS Interface Control Document, Issue 1.0, European Union, December 2022.
[3] PKI System Certificate Policy and Certification Practice Statement for ROOT CA-001, European Union, July 2023.
[4] PKI System Certificate Policy and Certification Practice Statement for SUB CA-001, European Union, July 2023.
[5] PKI System Certificate Policy and Certification Practice Statement for ICA-001, European Union, July 2023.

A.2. Reference Documents

[6] RFC 5280 - Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, IETF, May 2008.
[7] Abstract Syntax Notation One (ASN.1), ITU-T, February 2021.

Annex B Acronyms

Acronym	Definition
AD	Applicable Document
CA	Certification Authority
CP	Certificate Policy
CPS	Certificate Practice Statement
CRL	Certificate Revocation List
EC	European Commission
ECDSA	Elliptic Curve Digital Signature Algorithm
EE	End Entity
EU	European Union
EUSPA	European Union Space Programme Agency
GNSS	Global Navigation Satellite System (e.g. GPS, Galileo, GLONASS etc.)
GSC	EGNSS Service Centre
ICA	Issuing Certificate Authority
ICD	Interface Control Document
IDD	Internet Data Distribution
MT	Merkle Tree
OAM	OSNMA Alert Message
OS	Open Service
OSNMA	Open Service Navigation Message Authentication
PK	Public Key
PKI	Public Key Infrastructure
PKR	Public Key Renewal
RCA	Root Certificate Authority
RD	Reference Document
SCA	Subordinate Certificate Authority
SFTP	Secure File Transfer Protocol
SIS	Signal In Space
TESLA	Timed Efficient Stream Loss-tolerant Authentication
XML	Extensible Mark-up Language
XSD	XML Schema Definition

Annex C OSNMA Product Schemas

The following sections provide the XSD schemas used to generate the xml files available to the users at the GSC interface.

C.1. Public Key schema

```
<?xml version="1.0" encoding="UTF-8"?>
-<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
	<xs:include schemaLocation="OSNMA_Common_Types_V01.01.xsd"/>
	<xs:element substitutionGroup="GAL-body" type="PublicKeyType" name="PublicKey"/>
	-<xs:complexType name="PublicKeyType">
		-<xs:complexContent>
			-<xs:extension base="GAL-body-Type">
				-<xs:all>
					<!-- Unique ID of the Public Key. Not present in Merkle Tree. -->
					<xs:element type="xs:string" name="UID" minOccurs="0"/>
					<!-- Applicability time. Not present in Merkle Tree. -->
					<xs:element type="ApplicabilityType" name="Applicability" minOccurs="0"/>
					<!-- Product State according to SiS. Not present in Merkle Tree. -->
					<xs:element type="ProductState" name="State" minOccurs="0"/>
					-<xs:element type="xs:integer" name="i">
						-<xs:annotation>
							<xs:documentation>Position in the MerkleTree. If -1, then it is not present in the MerkleTree</xs:documentation>
						</xs:annotation>
					</xs:element>
```

```
                    <xs:element type="PKType" name="PKType"/>
                    <xs:element type="xs:nonNegativeInteger" name="lengthInBits"/>
                    <xs:element type="xs:hexBinary" name="point"/>
                        <!-- Associated Certificate. Not present in Merkle Tree. -->
                    <xs:element type="xs:anyURI" name="Certificate" minOccurs="0" maxOccurs="1"/>
                        <!-- Associated Certificate Revocation List. Not present in Merkle Tree. -->
                    <xs:element type="xs:anyURI" name="CRL" minOccurs="0" maxOccurs="1"/>
                    -<xs:element type="PKIDType" name="PKID">
                        -<xs:annotation>
                            <xs:documentation>Public Key ID, as per MerkleTree. If -1, then it is not present in the
                        MerkleTree</xs:documentation>
                        </xs:annotation>
                    </xs:element>
                </xs:all>
            </xs:extension>
        </xs:complexContent>
</xs:complexType>
-<xs:simpleType name="PKType">
        -<xs:restriction base="xs:string">
            <xs:enumeration value="ECDSA P-256/SHA-256"/>
            <xs:enumeration value="ECDSA P-521/SHA-512"/>
            <xs:enumeration value="Alert Message"/>
        </xs:restriction>
</xs:simpleType>
-<xs:simpleType name="PKIDType">
        -<xs:restriction base="xs:integer">
```

```
                    <xs:minInclusive value="-1"/>
                    <xs:maxInclusive value="15"/>
                </xs:restriction>
            </xs:simpleType>
</xs:schema>
```

C.2. Merkle Tree schema

```
<?xml version="1.0" encoding="UTF-8"?>
-<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
	<xs:include schemaLocation="OSNMA_Common_Types_V01.01.xsd"/>
	<xs:include schemaLocation="OSNMA_PublicKey_V01.01.xsd"/>
	<xs:element substitutionGroup="GAL-body" type="MerkleTreeType" name="MerkleTree"/>
	-<xs:complexType name="MerkleTreeType">
		-<xs:complexContent>
			-<xs:extension base="GAL-body-Type">
				-<xs:sequence>
					<!-- Unique ID of the Merkle Tree -->
					<xs:element type="xs:string" name="UID"/>
					<!-- Applicability time -->
					<xs:element type="ApplicabilityType" name="Applicability"/>
					<!-- Product State according to SiS -->
					<xs:element type="ProductState" name="State"/>
					<!-- Number of leaves/Public Keys in the Merkle Tree -->
					<xs:element type="powerOfTwo" name="N"/>
					<!-- Optional applicability time -->
					<xs:element type="MerkleTreeHash" name="HashFunction"/>
					<!-- Public Keys included in the Merkle Tree (1 to N) -->
					<xs:element type="PublicKeyType" name="PublicKey" maxOccurs="unbounded" minOccurs="1"/>
					<!-- Merkle Tree internal nodes and root node (5 to 2*N-1) -->
					<xs:element type="MerkleTreeNodeType" name="TreeNode" maxOccurs="unbounded" minOccurs="5"/>
				</xs:sequence>
```

```
                    </xs:extension>
                </xs:complexContent>
            </xs:complexType>
            -<xs:simpleType name="MerkleTreeHash">
                -<xs:restriction base="xs:string">
                    <xs:enumeration value="SHA-256"/>
                </xs:restriction>
            </xs:simpleType>
            -<xs:complexType name="MerkleTreeNodeType">
                -<xs:all>
                    <xs:element type="xs:nonNegativeInteger" name="j"/>
                    <xs:element type="xs:hexBinary" name="x_ji"/>
                    <xs:element type="xs:nonNegativeInteger" name="lengthInBits"/>
                    <xs:element type="xs:nonNegativeInteger" name="i"/>
                </xs:all>
            </xs:complexType>
</xs:schema>
```

C.3. OSNMA Common types schema

```
<?xml version="1.0" encoding="UTF-8"?>
-<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified">
	<xs:include schemaLocation="GAL-EXT_common_header_V01.00.xsd"/>
	<xs:include schemaLocation="GAL_common_types_V01.00.xsd"/>
	-<xs:simpleType name="powerOfTwo">
		-<xs:restriction base="xs:nonNegativeInteger">
			<xs:enumeration value="0"/>
			<xs:enumeration value="1"/>
			<xs:enumeration value="2"/>
			<xs:enumeration value="4"/>
			<xs:enumeration value="8"/>
			<xs:enumeration value="16"/>
			<xs:enumeration value="32"/>
			<xs:enumeration value="64"/>
			<xs:enumeration value="128"/>
			<xs:enumeration value="256"/>
			<xs:enumeration value="512"/>
			<xs:enumeration value="1024"/>
			<xs:enumeration value="2048"/>
			<xs:enumeration value="4096"/>
			<xs:enumeration value="8192"/>
			<xs:enumeration value="16384"/>
			<xs:enumeration value="32768"/>
			<xs:enumeration value="65536"/>
```

```
                    <xs:enumeration value="131072"/>
                    <xs:enumeration value="262144"/>
                    <xs:enumeration value="524288"/>
                    <xs:enumeration value="1048576"/>
                </xs:restriction>
            </xs:simpleType>
            -<xs:complexType name="ApplicabilityType">
                -<xs:sequence>
                    <xs:element name="Begin" type="DateAndTime"/>
                    <xs:element minOccurs="0" name="End" type="DateAndTime"/>
                </xs:sequence>
            </xs:complexType>
            -<xs:simpleType name="ProductState">
                -<xs:restriction base="xs:string">
                    <xs:enumeration value="Renewed"/>
                    <xs:enumeration value="Revoked"/>
                    <xs:enumeration value="Applicable"/>
                    <xs:enumeration value="Future"/>
                </xs:restriction>
            </xs:simpleType>
</xs:schema>
```

C.4. GAL-EXT common header schema

```
<?xml version="1.0" encoding="UTF-8"?>
-<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
	<xs:include schemaLocation="GAL_common_header_V01.00.xsd"/>
	-<xs:simpleType name="GAL-EXT-element-Type">
		-<xs:annotation>
			<xs:documentation>List of External Elements.</xs:documentation>
		</xs:annotation>
		-<xs:restriction base="xs:string">
			<xs:enumeration value="GOC-SC"/>
		</xs:restriction>
	</xs:simpleType>
	-<xs:element name="GAL-EXT-GOC-SC-GLAd" substitutionGroup="GLAd">
		-<xs:annotation>
			<xs:documentation>Galileo Operating Company Service Centre (GOC-SC) GLAd.</xs:documentation>
		</xs:annotation>
		-<xs:complexType>
			-<xs:complexContent>
				-<xs:extension base="GLAd-GAL-Type">
					-<xs:sequence>
						<xs:element name="segment" fixed="EXT" type="GAL-segment-Type"/>
						<xs:element name="element" fixed="GOC-SC" type="GAL-EXT-element-Type"/>
					</xs:sequence>
				</xs:extension>
			</xs:complexContent>
```

```
            </xs:complexType>
        </xs:element>
</xs:schema>
```

C.5. GAL common header schema

```
<?xml version="1.0" encoding="UTF-8"?>
-<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
	-<xs:complexType name="GAL-header-Type">
		-<xs:annotation>
			<xs:documentation>Base header type.</xs:documentation>
		</xs:annotation>
	-<xs:sequence>
		-<xs:element name="source">
			-<xs:complexType>
				-<xs:sequence>
					<xs:element ref="GLAd"/>
				</xs:sequence>
			</xs:complexType>
		</xs:element>
		-<xs:element name="destination">
			-<xs:complexType>
				-<xs:sequence>
					<xs:element ref="GLAd"/>
				</xs:sequence>
			</xs:complexType>
		</xs:element>
		-<xs:element name="issueDate" type="xs:dateTime">
			-<xs:annotation>
				<xs:documentation>Date and time of production of the body data by the originator.</xs:documentation>
```

```
                </xs:annotation>
            </xs:element>
            -<xs:element name="signalVersion" type="version-Type">
                -<xs:annotation>
                    <xs:documentation>This field can be checked by the receiving Element interface to insure it is able to handle the attached data correctly.</xs:documentation>
                </xs:annotation>
            </xs:element>
            -<xs:element name="dataVersion" type="dataVersion-Type" minOccurs="0">
                -<xs:annotation>
                    <xs:documentation>Version of the data. This field must be setup by the source of the data whenever the data is  versioned.</xs:documentation>
                </xs:annotation>
            </xs:element>
        </xs:sequence>
</xs:schema>
```

Annex D Attributes of the PKI certificates and CRLs

This Annex provides the full list of attributes for all level of PKI certificates.

D.1. RCA certificate attributes

Certificate attributes	Value	Comment
Version	3 (0x2)	
Serial Number	Random and unique value	Value not to be checked by receiver
Signature Algorithm ID	ecdsa-with-SHA256	
Issuer	CN (Common name)=EUSPA ROOT CA O (Organization) =EUSPA C (Country) = ES	Subject of this certificate
Validity - NotBefore - NotAfter	 YYMMDDhhmmssZ YYMMDDhhmmssZ	
Subject	CN (Common name)=EUSPA ROOT CA O (Organization) =EUSPA C (Country) = ES	
Subject Public Key Info - Public Key Algorithm - SubjectPublicKey	id-ecPublicKey namedcurve : ASN1 OID: prime256v1 / NIST CURVE: P-256 random value for public key (according to curve)	Note : namedcurve is consistent with signature algorithm SubjectPublicKey should not be checked by receiver

Certificate extensions	OID	Include	Criticity	Value	Comment

Subject Key Identifier	{id-ce 14}	X	false	*SHA-1 of SubjectPublicKey**	Only presence of extension should be verified (not value)
Basic Constraints	{id-ce 19}	X	true		
CA				TRUE	
Maximum Path Length				Missing	This extension should be missing
Certificate Policies	{id-ce 32}			Missing	This extension should be missing
PolicyIdentifiers				Missing	This extension should be missing
CPS				Missing	This extension should be missing
CRL Distribution Points	{id-ce 31}	X*	false		
DistributionPointName					This value should be empty
Authority Information Access	{id-pe 1}			Missing	This extension should be missing
Extended key usage	{id-ce 37}			Missing	This extension should be missing
Key Usage	{id-ce 15}	X	true		
digital Signature				0	
contentCommitment				0	
key Encipherment				0	
data Encipherment				0	
key Agreement				0	
keyCertSign				1	
cRLSign				1	
encipherOnly				0	
decipherOnly				0	

Certificate trailer	**Value**	**Comment**
Signature Algorithm ID	ecdsa-with-SHA256	
Signature	Random*	Signature of TBSCertifacte (Certificate attributes + extension as defined in RFC5280) to be verified with valid issuer Public Key (RCA)

D.2. RCA CRL attributes

CRL attributes	Value			Comment	
Version	2 (0x1)				
Signature Algorithm ID	ecdsa-with-SHA256				
Issuer Name	CN (Common name)=EUSPA ROOT CA O (Organization) =EUSPA C (Country) = ES			Issuer of the CRL	
This Update	YYMMDDhhmmssZ				
Next Update	YYMMDDhhmmssZ				
Revoked certificates	OID			Value	
Serial Number	Serial number of revoked certificate				
Revocation date	YYMMDDhhmmssZ				
Reason code	{id-ce 21}	X	false	either keyCompromise (1), cACompromise(2), or superseded(4),	unspecified (0), affiliationChanged(3), cessationOfOperation(5), certificateHold(6), removeFromCRL(8), privilegeWithdrawn(9), aACompromise(10) are not expected
Certificate extensions	**OID**	**Include**	**Criticity**	**Value**	**Comment**
Authority Key Identifier	{id-ce 35}	X	false	*subject Key Identifier Of RCA**	Only presence of extension should be verified (not value)
Issuer alternative name	{id-ce 18}				
CRL Number	{id-ce 20}	X	false	*Counter*	When new CRL, value should be greater from the last CRL number when generating by the same RCA certificate.
Delta CRL Indicator	{id-ce 27}			Missing	This extension should be missing
Freshest CRL	{id-ce 46}			Missing	This extension should be missing

CRL trailer	Value	Comment
Signature Algorithm ID	ecdsa-with-SHA256	
Signature	Random*	Signature of tbsCertList CRL attributes + extension as defined in RFC5280) to be verified with valid issuer Public Key (RCA)

D.3. SCA certificate attributes

Certificate attributes	Value	Comment
Version	3 (0x2)	
Serial Number	Random and unique value	Value not to be checked by receiver
Signature Algorithm ID	ecdsa-with-SHA256	
Issuer	CN (Common name)=EUSPA ROOT CA O (Organization) =EUSPA C (Country) = ES	Subject of active and valide RCA certificate
Validity - NotBefore - NotAfter	 YYMMDDhhmmssZ YYMMDDhhmmssZ	
Subject	CN (Common name)=EUSPA GALILEO SCA O (Organization) =EUSPA C (Country) = ES	
Subject Public Key Info - Public Key Algorithm - SubjectPublicKey	id-ecPublicKey namedcurve : ASN1 OID: prime256v1 / NIST CURVE: P-256 random value for public key (according to curve)	Note : namedcurve is consistent with signature algorithm SubjectPublicKey should not be checked by receiver

Certificate extensions	OID	Include	Criticity	Value	Comment
Authority Key Identifier	{id-ce 35}	X	false	*subject Key Identifier Of RCA**	Only presence of extension should be verified (not value)
Subject Key Identifier	{id-ce 14}	X	false	*SHA-1 of SubjectPublic Key*	Only presence of extension should be verified (not value)
Basic Constraints	{id-ce 19}	X	true		
CA				TRUE	
Maximum Path Length				Missing	This extension should be missing

Certificate Policies	{id-ce 32}	X	false		
PolicyIdentifiers				1.3.6.1.4.1.60049.1	
CPS				https://www.euspa.europa.eu/about/how-we-work/pki/policy	Only presence of extension should be verified (not value)
CRL Distribution Points	{id-ce 31}	X*	false		
DistributionPointName				https://www.euspa.europa.eu/about/how-we-work/pki/products	
Authority Information Access	{id-pe 1}			Missing	This extension should be missing
Extended key usage	{id-ce 37}			Missing	This extension should be missing
Key Usage	{id-ce 15}	X	true		
digital Signature				0	
contentCommitment				0	
key Encipherment				0	
data Encipherment				0	
key Agreement				0	
keyCertSign				1	
cRLSign				1	
encipherOnly				0	
decipherOnly				0	

Certificate trailer	**Value**	**Comment**
Signature Algorithm ID	ecdsa-with-SHA256	
Signature	Random*	Signature of TBSCertifacte (Certificate attributes + extension as defined in RFC5280) to be verified with valid issuer Public Key (RCA)

D.4. SCA CRL attributes

CRL attributes	Value				Comment
Version	2 (0x1)				
Signature Algorithm ID	ecdsa-with-SHA256				
Issuer Name	CN (Common name)=EUSPA GALILEO SCA O (Organization) =EUSPA C (Country) = ES				Issuer of the CRL
This Update	YYMMDDhhmmssZ				
Next Update	YYMMDDhhmmssZ				
Revoked certificates					
Serial Number	Serial number of revoked certificate				
Revocation date	YYMMDDhhmmssZ				
Reason code	{id-ce 21}	X	false	either keyCompromise (1), cACompromise(2), or superseded(4),	unspecified (0), affiliationChanged(3), cessationOfOperation(5), certificateHold(6), removeFromCRL(8), privilegeWithdrawn(9), aACompromise(10) are not expected
Certificate extensions	**OID**	**Include**	**Criticity**	**Value**	**Comment**
Authority Key Identifier	{id-ce 35}	X	false	*subject Key Identifier Of SCA**	Only presence of extension should be verified (not value)
Issuer alternative name	{id-ce 18}				
CRL Number	{id-ce 20}	X	false	*Counter*	When new CRL, value should be greater from the last CRL number when generating by the same SCA certificate.
Delta CRL Indicator	{id-ce 27}			Missing	This extension should be missing
Freshest CRL	{id-ce 46}			Missing	This extension should be missing

CRL trailer	Value	Comment
Signature Algorithm ID	ecdsa-with-SHA256	
Signature	Random*	Signature of tbsCertList CRL attributes + extension as defined in RFC5280) to be verified with valid issuer Public Key (SCA)

D.5. ICA certificate attributes

Certificate attributes	Value	Comment
Version	3 (0x2)	
Serial Number	Random and unique value	Value not to be checked by receiver
Signature Algorithm ID	ecdsa-with-SHA256	
Issuer	CN (Common name)=EUSPA GALILEO SCA O (Organization) =EUSPA C (Country) = ES	Subject of active and valide SCA certificate
Validity - NotBefore - NotAfter	 YYMMDDhhmmssZ YYMMDDhhmmssZ	
Subject	CN (Common name)=EUSPA OSNMA ICA O (Organization) =EUSPA C (Country) = ES	
Subject Public Key Info - Public Key Algorithm - SubjectPublicKey	id-ecPublicKey namedcurve : ASN1 OID: prime256v1 / NIST CURVE: P-256 random value for public key (according to curve)	Note : namedcurve is consistent with signature algorithm SubjectPublicKey should not be checked by receiver

Certificate extensions	OID	Include	Criticity	Value	Comment
Authority Key Identifier	{id-ce 35}	X	false	*subject Key Identifier Of SCA**	Only presence of extension should be verified (not value)
Subject Key Identifier	{id-ce 14}	X	false	*SHA-1 of SubjectPublicKey*	Only presence of extension should be verified (not value)
Basic Constraints	{id-ce 19}	X	true		
CA				TRUE	

Maximum Path Length				0	
Certificate Policies	{id-ce 32}	X	false		
PolicyIdentifiers				1.3.6.1.4.1.60049.1.1	
CPS				https://www.euspa.europa.eu/about/how-we-work/pki/policy	Only presence of extension should be verified (not value)
CRL Distribution Points	{id-ce 31}	X*	false		
DistributionPointName				https://www.euspa.europa.eu/about/how-we-work/pki/products	
Authority Information Access	{id-pe 1}			Missing	This extension should be missing
Extended key usage	{id-ce 37}			Missing	This extension should be missing
Key Usage	{id-ce 15}	X	true		
digital Signature				0	
contentCommitment				0	
key Encipherment				0	
data Encipherment				0	
key Agreement				0	
keyCertSign				1	
cRLSign				1	
encipherOnly				0	
decipherOnly				0	

Certificate trailer	**Value**	**Comment**
Signature Algorithm ID	ecdsa-with-SHA256	
Signature	Random and unique value	Signature of TBSCertifacte (Certificate attributes + extension as defined in RFC5280) to be verified with valid issuer Public Key (ICA)

D.6. ICA CRL attributes

CRL attributes	Value				Comment
Version	2 (0x1)				
Signature Algorithm ID	ecdsa-with-SHA256				
Issuer Name	CN (Common name)=EUSPA OSNMA ICA O (Organization) =EUSPA C (Country) = ES				Issuer of the CRL
This Update	YYMMDDhhmmssZ				
Next Update	YYMMDDhhmmssZ				
Revoked certificates					
Serial Number	Serial number of revoked certificate				
Revocation date	YYMMDDhhmmssZ				
Reason code	{id-ce 21}	X	false	either keyCompromise (1), cACompromise(2), or superseded(4),	unspecified (0), affiliationChanged(3), cessationOfOperation(5), certificateHold(6), removeFromCRL(8), privilegeWithdrawn(9), aACompromise(10) are not expected
Certificate extensions	**OID**	**Include**	**Criticity**	**Value**	**Comment**
Authority Key Identifier	{id-ce 35}	X	false	*subject Key Identifier Of ICA**	Only presence of extension should be verified (not value)
Issuer alternative name	{id-ce 18}				
CRL Number	{id-ce 20}	X	false	*Counter*	When new CRL, value should be greater from the last CRL number when generating by the same ICA certificate.
Delta CRL Indicator	{id-ce 27}			Missing	This extension should be missing

Freshest CRL	{id-ce 46}			Missing	This extension should be missing
CRL trailer	**Value**			**Comment**	
Signature Algorithm ID	ecdsa-with-SHA256				
Signature	Random*			Signature of tbsCertList CRL attributes + extension as defined in RFC5280) to be verified with valid issuer Public Key (ICA)	

D.7. EE PKR certificate attributes

Certificate attributes	Value	Comment
Version	3 (0x2)	
Serial Number	Random and unique value	Value not to be checked by receiver
Signature Algorithm ID	ecdsa-with-SHA256	
Issuer	CN (Common name)=EUSPA OSNMA ICA O (Organization) =EUSPA C (Country) = ES	Subject of active and valide ICA certificate
Validity - NotBefore - NotAfter	 YYMMDDhhmmssZ YYMMDDhhmmssZ	
Subject	CN (Common name)=EUSPA OSNMA EE PKR O (Organization) =EUSPA C (Country) = ES	
Subject Public Key Info - Public Key Algorithm - SubjectPublicKey	id-ecPublicKey namedcurve : ASN1 OID: prime256v1 / NIST CURVE: P-256 random value for public key (according to curve)	Note : namedcurve is consistent with signature algorithm SubjectPublicKey should not be checked by receiver

Certificate extensions	OID	Include	Criticity	Value	Comment
Authority Key Identifier	{id-ce 35}	X	false	*subject Key Identifier Of ICA**	Only presence of extension should be verified (not value)
Subject Key Identifier	{id-ce 14}	X	false	*SHA-1 of SubjectPublicKey*	Only presence of extension should be verified (not value)
Basic Constraints				Missing	This extension should be missing

CA					
Maximum Path Length					
Certificate Policies	{id-ce 32}	X	false		
PolicyIdentifiers				1.3.6.1.4.1.60049.1.1.1 0.4.0.2042.1.2	NCP+
CPS				https://www.gsc-europa.eu/gsc-products/OSNMA/PKI/	Only presence of extension should be verified (not value)
CRL Distribution Points	{id-ce 31}	X*	false		
DistributionPointName				https://www.gsc-europa.eu/gsc-products/OSNMA/PKI	
Extended key usage	{id-ce 37}			Missing	This extension should be missing
Key Usage	{id-ce 15}	X	true		
digital Signature				1	
contentCommitment				0	
key Encipherment				0	
data Encipherment				0	
key Agreement				0	
keyCertSign				0	
cRLSign				0	
encipherOnly				0	
decipherOnly				0	

Certificate trailer	**Value**	**Comment**
Signature Algorithm ID	ecdsa-with-SHA256	
Signature	Random*	Signature of TBSCertifacte (Certificate attributes + extension as defined in RFC5280) to be verified with valid issuer Public Key (ICA)

TS-03-23-374-EN-N

LINKING SPACE TO USER NEEDS

www.euspa.europa.eu

@EU4Space

@EU4Space

EUSPA

@space4eu

EUSPA