ETSI GS NFV-SEC 020 V1.1.1 (2026-04)

GROUP SPECIFICATION

Network Functions Virtualisation (NFV);
Security;
Identity Management and Security Specification

Disclaimer

The present document has been produced and approved by the Network Functions Virtualisation (NFV) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference
DGS/NFV-SEC020

Keywords
NFV, security

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI IPR online database.

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Network Functions Virtualisation (NFV).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document specifies requirements for secure VNF identity management and trust relationships in NFV. The present document specifies how identities are securely lifecycle managed, verified and trusted. The present document addresses both horizontal and vertical relationships and leverages existing work in ETSI GR NFV-SEC 005 [i.1], ETSI GR NFV-SEC 007 [i.2], ETSI GS NFV-SEC 009 [i.3], ETSI GS NFV-SEC 012 [1] and ETSI GS NFV-SEC 013 [i.4].

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found in the ETSI docbox.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are necessary for the application of the present document.

[1] ETSI GS NFV-SEC 012: "Network Functions Virtualisation (NFV) Release 5; Security; System architecture specification for execution of sensitive NFV components".

[2] IETF RFC 3986: "Uniform Resource Identifier (URI): Generic Syntax".

[3] IETF RFC 9334: "Remote ATtestation procedureS (RATS) Architecture".

[4] IETF RFC 5280: "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".

[5] IETF RFC 9711: "The Entity Attestation Token (EAT)".

[6] OpenID4VP: "OpenID for Verifiable Presentations".

[7] GlobalPlatform Card GPC_SPE_095: "Digital Letter of Approval".

[8] W3C® Recommendation: "Verifiable Credentials Data Model v2.0".

[9] OpenID4VCI: "OpenID for Verifiable Credential Issuance".

[10] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".

[11] DIF: "Presentation Exchange".

[12] IETF RFC 8414: "OAuth 2.0 Authorization Server Metadata".

[13] IETF RFC 7519: "JSON Web Token (JWT)".

[14] IETF RFC 7515: "JSON Web Signature (JWS)".

[15] IETF RFC 7516: "JSON Web Encryption (JWE)".

[16] IETF RFC 7517: "JSON Web Key (JWK)".

[17] SPIFFE Federation.

[18] ETSI GS NFV-SOL 003: "Network Functions Virtualisation (NFV) Release 5; Protocols and Data Models; RESTful protocols specification for the Or-Vnfm Reference Point".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

[i.1] ETSI GR NFV-SEC 005: "Network Functions Virtualisation (NFV); Trust; Report on Certificate Management".

[i.2] ETSI GR NFV-SEC 007: "Network Functions Virtualisation (NFV); Trust; Report on Attestation Technologies and Practices for Secure Deployments".

[i.3] ETSI GS NFV-SEC 009: "Network Functions Virtualisation (NFV); NFV Security; Report on use cases and technical approaches for multi-layer host administration".

[i.4] ETSI GS NFV-SEC 013: "Network Functions Virtualisation (NFV) Release 3; Security; Security Management and Monitoring specification".

[i.5] ETSI TR 119 460 (V1.1.1): "Electronic Signatures and Infrastructures (ESI); Survey of technologies and regulatory requirements for identity proofing for trust service subjects".

[i.6] ENISA Remote ID proofing report /2021-03: "Remote ID proofing; Analysis of methods to carry out identity proofing remotely ".

[i.7] ETSI GR NFV 003: "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

[i.8] FIPS PUB 199: "Standards for Security Categorization of Federal Information and Information Systems".

[i.9] Gartner: "Leading the IoT, Gartner Insights on How to Lead in a Connected World".

[i.10] FIPS PUB 202: "SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions".

[i.11] NIST Special Publication 800-90A Revision 1.

[i.12] ETSI GS NFV-SEC 023: "Network Functions Virtualisation (NFV) Release 5; Security; Container Security Specification".

[i.13] ETSI GR NFV-SEC 018: "Network Functions Virtualisation (NFV); Security; Report on NFV Remote Attestation Architecture".

[i.14] IETF RFC 9901: "Selective Disclosure for JSON Web Tokens".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GR NFV 003 [i.7] and the following apply:

qualified attestation of attributes: identity credentials as Verifiable Credentials including identity attributes that have been qualified and signed by a trusted provider, a qualified attestation of attributes provider

qualified attestation of attributes provider: trusted provider of qualified attestation of attributes

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GR NFV 003 [i.7] and the following apply:

3GPP	3rd Generation Partnership Project
ABAC	Attribute-Based Access Control
AUTHN	Authentication process
AUTHZ	Authorization process
CIA	Confidentiality, Integrity, Availability
CSR	Certificate Signing Request
DLOA	Digital Letter of Approval
ID	Identity
PVID	Primary Verifiable Identity Document
RBAC	Role-Based Access Control
SM	Security Manager
VIP	Verifiable Identity Presentation

4 Overview

4.1 Introduction

Identity (ID) is defined as "the fact of being who or what a person or thing is" and is usually used as a parameter to uniquely distinguish one being from a group of others.

In the realm of human society, where individuals are often unknown and untrusted, passports are relied on to establish our identities across the globe.

Passports are considered trustworthy documents because they are issued by governmental authorities, universally recognized as reliable and equipped with security features to ensure the integrity of the document.

A passport includes an identifier, the passport number, which indicates the country of issuance and a unique identifier for the individual. Additionally, it includes several attributes that inherently represent the person such as their first name, family name, photograph, a fingerprint, and other physical characteristics. These claimed attributes are verified by an authoritative body during the passport issuance process to confirm they correspond with the actual person. This is the proofing process. These attributes are utilized during the verification process to ascertain that the individual presenting the passport is indeed the person they claim to be.

Similarly in the realm of zero trust and distributed environments, such as NFV, entities shall substantiate their claims to establish trust. Each of these entities require an identity document akin to a passport, which should be trustable, interpretable and verifiable by all the entities that initiate communication with it.

Every element of a telecoms network and everything or person using the network needs an identity to determine the characteristics of that individual or component. For CSPs, the identity of NFVI components, SDN routing and VNFs are key to how CSPs design, manage and operate their networks.

Identities may be self-assigned, given, inherited, derived, acquired, allocated or obtained in a large number of ways.

If an attacker obtains access to a CSP network implemented with NFV then it needs to be possible, even months after the event, to retrace the attack to establish where they got into the network, what was accessed, for how long and, as far as possible, what identity they used to achieve this access. Similarly, if a customer reports a fault it needs to be possible to trace their current and past usage of services to resolve the issue.

Therefore, in an NFV environment, it needs to be possible for identities to be trusted, structured, unique, and immutable for a given period, if networks are to be operated securely and with a low risk of fraud.

The present document describes secure identity management in the context of NFV, in terms of what an identity is, what that identity is used for, how it is assigned, how it is discovered and how it is securely managed throughout the lifecycle of that identity.

4.2 Identity Definition Purposes and Uses of Identity

The present document defines an ID structure that vertically spans the NFV domain and the application domain above it. Information from both domains is necessary to implement effective, real-world security policies. The ID will contain information about both the TYPE and INSTANCE of a software process. The information about the TYPE of a running INSTANCE, which is available in a higher trust domain, shall not be available in any lower trust domain. Further, information about the TYPE of a running INSTANCE available in any trust domain shall not be available in any other trust domain of equal sensitivity, unless the process in the source trust domain explicitly intends it, as expressed in the appropriate security policies.

Identity is the foundation of networks, it enables distinction among individual instances and individual types, discovery of suitable partners of a process, attachment to such partners once discovered, and, as it is also the foundation of security, identity enables the assignment, tracking and evolution of trust.

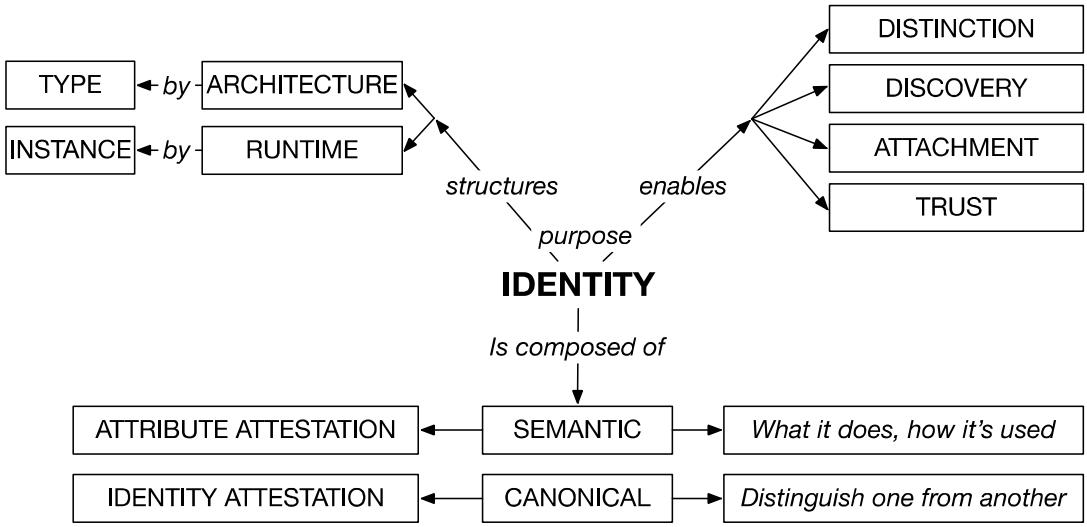


Figure 4.2-1: Purposes and Uses of Identity

There are two basic flavours of identity in a network: TYPE identity, through which architecture is structured, and INSTANCE identity, through which the runtime is structured.

Identity enables the development of ontologies, allowing the systems designer to make statements about an object's capabilities and uses, and based on this, allows the implementer to distinguish, manage and secure runtime instantiations of said objects.

A person's passport identifies them with a globally unique identifier, primarily defined by the passport number. It also contains information about the issuing country, allowing the assessment of trust and identifying the structure of the passport.

Similarly, an identity document for an entity in the virtualized world should possess a globally unique identifier, that includes:

- information about the issuing system (e.g. NFV system), allowing the identification of the scheme used for this identity;
- the trust domain associated with the entity; and
- a unique identifier.

This globally unique identifier is the canonical identity of figure 4.3-1, distinguishing one entity from another.

A person's passport includes attributes inherent to the person's identity verified during the proofing process before issuance. These attributes are crucial for confirming the person's identity during passport presentation, verification of fingerprint, picture, and physical characteristics such as eyes colour, height.

In the same way, an identity document for virtualized entities should include attributes inherent to the entity's identity, such as:

- Entity name (e.g. the 3GPP name of a virtual function)
- Hash of the software image
- Attestation result
- Trust domain where it is instantiated
- Identity of the NFV-MANO, and the NFVI that instantiate the entity
- ContainerID where it is instantiated
- Timestamp of instantiation
- Location instantiation
- Etc.

The list of attributes could depend on the service provider and the trust domain, as per policies.

These attributes shall undergo verification by a trusted authority during a proofing process which is synonymous with the attestation process.

Attributes, as described, correspond to the semantic identity outlined in figure 4.3-1, defining the entity's functionality and usage.

4.3 Hierarchy

IDs have usage and meaning that span domains. Figure 4.3-1 depicts the main elements that share usage and meaning of IDs in the larger context in which NFV exists. While the figure shows the relationships, the details of the actual interfaces themselves are outside the scope of the present document.

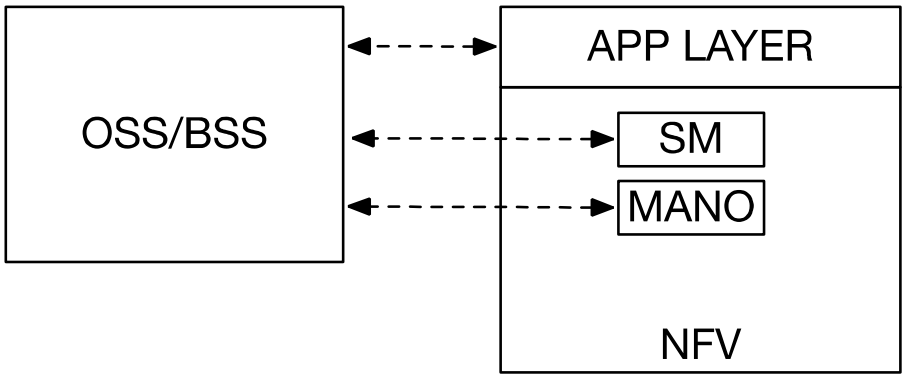


Figure 4.3-1: NFV Context

Operations Support Systems (OSS) and Business Support Systems (BSS) have existed, and will continue to exist, outside the NFV domain but are also intricately linked with the operation of virtualised functions. As described in the present document, some interaction will have to exist between the relationship of the OSS/BSS and the application layer function (e.g. 3GPP NF) implemented using NFV, with the relationships between OSS/BSS and both the Security Manager (SM) and MANO. The interaction, whether automated or manual, is intended to bridge the gap between the NFV infrastructure and the application layer.

The VNF instance communicates with entities at these different levels and may have different identity attributes for these different levels, even if these attributes are associated to the same VNF instance.

In the context of identity of human, depending on the domain where the identity of a person is used, the attributes presented could change. For example, an alumni could present his diploma as attribute, with the date of issuance, when he presents himself for a job. But in the context of health care, he will present an identity containing its identifier for the health care service.

In the same way, the identity of the virtual entity may include attributes that are relevant for the domain where the identity is used. For example, the identity of a VNF instance at the NFV level, communicating with the VNFM may include some attributes (e.g. role of the VNF) that are relevant at this level. The same VNF communicating with another VNF at 3GPP level may include other attributes such as the name of the VNF at 3GPP level and its role at 3GPP level.

Some attributes could be useful for both levels. This is the case for example of attestation result that is relevant at NFV level but also at 3GPP level to enable the trust.

The identity management of the VNF instance shall allow the bridge between NFV infrastructure and application layer, that could be 3GPP but also other application context.

5 Identity-Related Concepts and Definitions

5.1 General

5.1.1 TYPE and INSTANCE

The present document defines the software package TYPE category, which describes the particular functions a package is capable of fulfilling at the application layer (e.g. a firewall, a 3GPP-defined Serving Call Session Control Function (S-CSCF), etc.), and the VNF INSTANCE category, which identifies the running instance of a VNF and/or its component software process.

5.1.2 Lifecycle Events

Further, the present document defines two events in the lifetime of a software package: the ON-BOARDING event, when a software package is received from the vendor and is added to the software catalogue of the CSP, and the run-time events of INSTANTIATION, MODIFICATION, and TERMINATION, when the VNFI is executed, modified, and terminated, respectively.

5.1.3 Confidentiality, Integrity, and Availability

The US National Institute of Standards and Technology (NIST) in FIPS Publication 199 [i.8] defines the Confidentiality/Integrity/Availability (CIA) model, which defines LOW/MODERATE/HIGH impact to each of the three dimensions. This model has been adopted for the present document. The attributes can be applied at both ON-BOARDING time, to the TYPE of a package, and at INSTANTIATION time, to the running INSTANCE of the package.

For the purposes of the present document, the same three levels are defined (LOW, MODERATE, and HIGH) for each of the three dimensions. While all three dimensions play into ID management, the present document concentrates on the Confidentiality dimension. At the discretion of the CSP, the NFV system shall operate with at least two of the three confidentiality levels, with the Security Manager assigned to the higher confidentiality level of the two, and MANO assigned to the lower level of the two. If the CSP elects to implement all three levels, the SM shall be assigned to the HIGH confidentiality level, and MANO shall be assigned to the LOW confidentiality level.

For the purposes of the present document, the Availability dimension is expanded to encompass not only Availability but also Authorization and Authentication. Again, the same three levels are defined for these.

5.1.4 Trust Domains

A trust domain is defined as a set of processes running at the same sensitivity level due to the application of a common set of security policies. The CSP shall manage the flow of information across trust domains in such a way that information and attributes of a higher sensitivity trust domain shall not transfer to a lower sensitivity trust domain.

There may exist many separate trust domains of the same sensitivity level, but information contained in each is not necessarily available from any other. For example, administrators given access to the HIGH sensitivity level in one trust domain will not necessarily have access to another trust domain of HIGH sensitivity level, as dictated by the need-to-know principle. When the present document refers to a "HIGH trust zone", it is always to be read as meaning a HIGH confidentiality trust zone, unless the other two attributes are used explicitly. A consequence of this requirement is that data needs to be labelled as exportable or non-exportable.

Implementing an access control system shall be mandatory. As described in ETSI GS NFV-SEC 012 [1], security maintenance has finer granularity under the implementation of Attribute-Based Access Control (ABAC), rather than the simpler Role-Based Access Control (RBAC). A difference between the two is that ABAC also takes into consideration the context (e.g. time-of-day access restrictions to certain resources) of a resource access event, not only the accessor and the resource itself. ABAC is comparable to concepts used in multi-factor authentication, RBAC is comparable to concepts used in single-factor authentication. Segregating access to a trust domain is therefore more robust under ABAC.

6 Management and Structure of Identity

6.1 Introduction - Purpose of Identity

The purpose of VNF instance identity is to uniquely identify the VNF instance and prove that the VNF instance is really what it claims to be.

A VNF instance identity has several usages:

- **Distinction:** A same program may be deployed and scale out to a large number of nodes, in different locations, within different infrastructures and may be updated with a new version. It is also associated with an issuing authority that manages it. The identity of the workload shall enable this distinction. A mechanism to ensure the uniqueness of identity across the CSP system at any given time shall be employed.
- **Discovery:** Once identities are issued for the workloads, they can be used for the discovery of the services in a catalogue, after their registration.
- **Authentication:** Identities can be used for authentication, proving that the service is what it claims to be, and enabling establishment of secure communication with it.
- **Authorization:** Once the services have authenticated to each other, they can control access to their services and data.
- **Confidentiality:** After authentication, a secure communication may be established between the services enabling the data exchanged being kept secret.
- **Trust:** The identity document issued for the service shall be trustworthy: issued by a trusted authority after a proofing process, verifiable, and containing attributes that can be checked at the authentication or authorization time.

This clause defines the two fundamental components of the Identity management:

- The Identifier: a decentralized unique identifier of the VNF instance.
- The Verifiable Identity document: a passport for the VNF instance, that carries the Identity.

The Verifiable Identity document shall be resistant to forgery and contain information that proves that it belongs to the VNF instance that presents it, and that proves its authenticity. In addition, this Verifiable Identity document supports verification of various identity attributes of the VNF instance that have been determined during the proofing process (i.e. attestation). The simple presentation of this Verifiable Identity document gives the relevant identity information to the other party, enabling the trust.

6.2 Structure of Identity

6.2.1 Introduction

The identity of a VNF/VNFC instance shall be uniquely defined and shall identify the instance across heterogeneous environments and organizations, within a global scope and shall be interpretable consistently regardless the context. Therefore, the NFV identity of the VNF instance shall be defined as a Uniform Resource Identifier (URI) as defined by IETF RFC 3986 [2].

6.2.2 Scheme

The scheme name defined for the URI of NFV instance identifiers, that refers to the present document is the following:

Scheme: nfvid

NOTE: The present document defines a specific scheme name for NFV, which is to be registered with IANA. It is possible to use a scheme name already defined and used in the cloud environment if it satisfies the requirements for identity management: example: "spiffe" identity name space.

6.2.3 Authority

In the context of a URI the authority identifies the domains. To avoid collisions in the identifiers and be able to identify the system in which the identity has been issued, the trust domain is included in the URI as a hierarchical element. With this hierarchical element, the remainder of the URI is delegated to the authority managing this trust domain.

The trust domain is a trust root of the system and is defined by the service provider, owner of the VNF instance. There could be a trust domain for e.g. operational or test instance. This trust domain name is self-registered by the service provider. There is no centralized authority for the registration of these trust domain names. To prevent collisions the service provider shall select the trust domain name that is highly likely to be globally unique (e.g. adding a service provider DNS name as a suffix of the trust domain name or using a randomly generated name such as UUID).

The trust domain is defined as the authority component of the URI where only the host part is present.

6.2.4 Path

The path component is used to uniquely identify a VNF instance within the scope of the "nfvid" scheme and the trust domain controlled by the service provider. The path definition is left open to the service provider. Path may be hierarchical with e.g. the name of the network service which the VNF instance is part of, the name of the VNF instance (e.g. udm) and the last path segment shall be the vnfInstanceID as defined in ETSI GS NFV-SOL 003 [18], which is an individual path segment, and issued during the creation of a VNF instance.

Example of an identity:

nfvId://test.operator.com/vnfInstanceId

NOTE 1: How the vnfcInstanceID is incorporated into the scheme is for future study.

NOTE 2: Some information contained in the VnfInfo could be used for an automatic identification of the VNF instance: e.g. vnfInstanceName, vnfProductName, or specific data in the metadata element. How this can be incorporated is for future study.

6.3 Properties and Attributes of Identity

6.3.1 Introduction

The VNF/VNFC instances have some properties or attributes that could be used as identity attributes.

Some of these attributes inherently identify the VNF/VNFC instance, some other attributes are identity attributes applicable to some context and have a meaning within this context only (e.g. the 3GPP entity name that has a meaning at the 3GPP layer only).

Attributes that inherently identify the VNF/VNFC instance are attributes that could be verified to prove that the VNF/VNFC instance is really what it claims to be. These attributes could be verified during a proofing process, an attestation process, before the issuance of the Identity document to the VNF/VNFC instance. The attestation verification is done by an authoritative entity in the trust domain of the VNF/VNFC instance. The Security Manager (SM) of the trust domain may include the attestation verification and identity management. The choice of attributes used to inherently identify the VNF/VNFC instance is controlled by the CSP associated with this trust domain and are further called selectors. These selectors are part of policies that could be registered in the SM during a registration process of the VNF at the time of instantiation.

The selectors included in the policies depend on the use case and the service provider. For example:

- Some simple use-cases could restrict the list of selectors to the software integrity attestation as described in ETSI GS NFV-SEC 023 [i.12], clause 6.5.1.4.
- Some use cases could use HMEE Attestation and additional selectors as instantiation locstamp: e.g. Lawful Interception.

6.3.2 Attributes bound to Identity

Table 6.3.2-1 lists potential attributes bound to identity that shall be able to be cryptographically bound to the identity. Some attributes are fixed upon the receipt of the VNF package from the vendor, end of testing, on-boarding of a package received from the vendor to the CSP catalogue, some other attributes are fixed upon the completion of the launch procedure of a VNF instance.

Table 6.3.2-1: List of potential attributes bound to identity

Attribute	Description	Fixed upon	Trust source of information	Comments	Use-case
	Attributes that could be included in the Identity URI as hierarchical paths				
Manufacturer		Package on-boarding	VnfInfo/SM		
ProductName		Package on-boarding	VnfInfo/SM		
Version		Package on-boarding	VnfInfo/SM		
	Inherent Identity Attributes for proofing process				
Integrity Attestation		Instantiation	Attestor	See ETSI GS NFV-SEC 023 [i.12], clause 6.5.1.4)	
HMEEAttestation		Instantiation	Quoting enclave/Attestor		LI
Run-time Attestation			Attestor		
Instantiation locstamp		Instantiation			LI
Instantiation timestamp		Instantiation			
LoA				(1, 2, 3, 4, 5a, 5b - see ETSI GR NFV-SEC 007 [i.2], clause 5)	
MANO IDs	Identification of MANO function instances (e.g. NFVO, VNFM, VIM) which effected the launch	Instantiation			
Security domain/namespace		Instantiation			
CGroup		Instantiation			
ContainerID		Instantiation			

NOTE 1: The present document presents a core set of potential attributes that can be cryptographically linked to an identity, as provided in table 6.3.2-1. How other attributes may be added in the list is for future study (e.g. 3GPP NFType, NF role). Furthermore, to improve practical use future editions may incorporate representative workflows, policies, and use cases. These additions will address further important security topics such as software supply chain assurance, software integrity validation, and certificate management practices. Common terminology and frameworks will support robust identity proofing and attestation in NFV environments, aligned with ETSI standards for interoperability.

NOTE 2: The use case column gives the use-cases for which an attribute is highly relevant. These are just examples and does not prevent the use of other attributes for the specific use case or use one attribute for a use-case that is not mentioned.

As the NFV system is used, it evolves, and new information is gathered and created that may change the setting or meaning of attributes bound at ON-BOARDING time or INSTANTIATION time. Once Identity (Instance ID) is fixed, the SM shall not change it. However, the values of the attributes bound to the ID may change and the SM may modify the bound attributes. Each SM, within its trust domain, shall ensure that for each change there is a clear and immutable record made within a cryptographically bound log file. These SM logs shall be available for auditing at least for the lifetime of the longest-lived element managed by the SM, or longer, as dictated by legal considerations or CSP policy.

6.4 Proof of Identity process: the attestation process

The Identity proofing process is an essential process before the issuance of the ID document to confirm identity attributes and ensure the entity is really what it claims to be.

For the human, this proofing process is a requirement of the eIDAS regulation and is done during a face-to-face or remotely with an authority that verifies some identity attributes (e.g. picture, fingerprint, height, eye colour) before issuing the ID document. ETSI TR 119 460 [i.5] presents a methodology based on three steps of the identity proofing process for natural persons or legal persons: attribute and evidence collection, attribute and evidence validation, and binding identity attributes to the applicant.

ETSI TR 119 460 [i.5] states that the process is finalized by issuance of the proof or assertion.

Figure 6.4-1 shows the general Identity proofing process for natural and legal persons proposed by ENISA in the report Remote ID proofing [i.6].

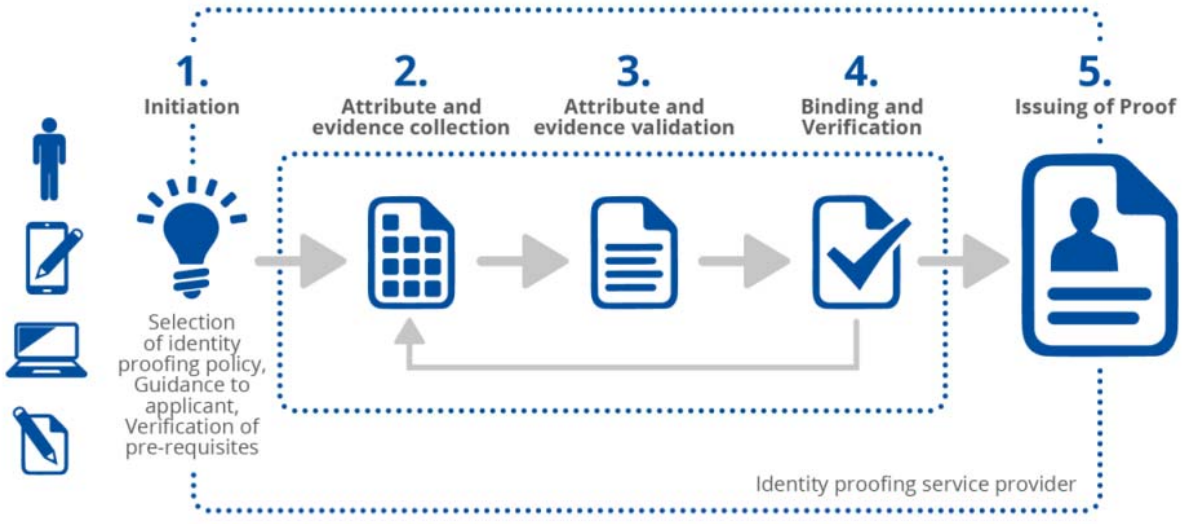


Figure 6.4-1: General identity proofing process for natural and legal persons-source ENISA report

The proofing process is based on the collection, validation, and verification of evidence.

For the VNFI/VNFCI case, this process is called the attestation process as defined by IETF RFC 9334 [3], for which all the steps visualised in figure 6.4-1 can be mapped. Table 6.4-1 gives this mapping.

Table 6.4-1: Mapping between Identity proofing process for human and attestation process for VNFI/VNFCI

Step number	Identity Proofing process steps for human	Attestation process steps as described in IETF RFC 9334 [3]	Comments
1	Initiation	Definition of policies, endorsements, attributes or claims to be included in measurement and associated golden or reference measurements as requisites.	Configuration of policies in the target environment and attester. Configuration of policies and reference values in the Verifier. Configuration of policies in the relying party.
2	Attribute and evidence collection	This step corresponds to the collection of claims, measurements, and provision of the corresponding evidence: a signed report.	This step is done at the attester. Evidence may be obtained from several different sources and shall be in a format such the verifier can process.
3	Attribute and evidence validation	Validation of the signed report (evidence) verifying that it really comes from the genuine attester.	This validation is done by the verifier.
4	Binding and verification	Association of the target environment to the security policies and reference values and check against these reference values.	This verification is done by the verifier.
5	Issuing of proof	This step corresponds to the issuance of the attestation report, that could be used further by the relying party.	This attestation report issuance is done by the verifier.

The step described above are represented in figure 6.4-2 in the conceptual data flow of the RATS Architecture described in IETF RFC 9334 [3], section 3 figure 1.

Furthermore, the remote attestation process is described in depth in ETSI GR NFV-SEC 018 [i.13] identifies and studies architectures applicable to NFV systems, including defining the attestation scope, stakeholders, interfaces, and protocols required to support them. It provides an architecture for remote attestation, covering measurement, reporting, and verification operations involved in attestation for NFV environments, being aligned with the RATS architecture.

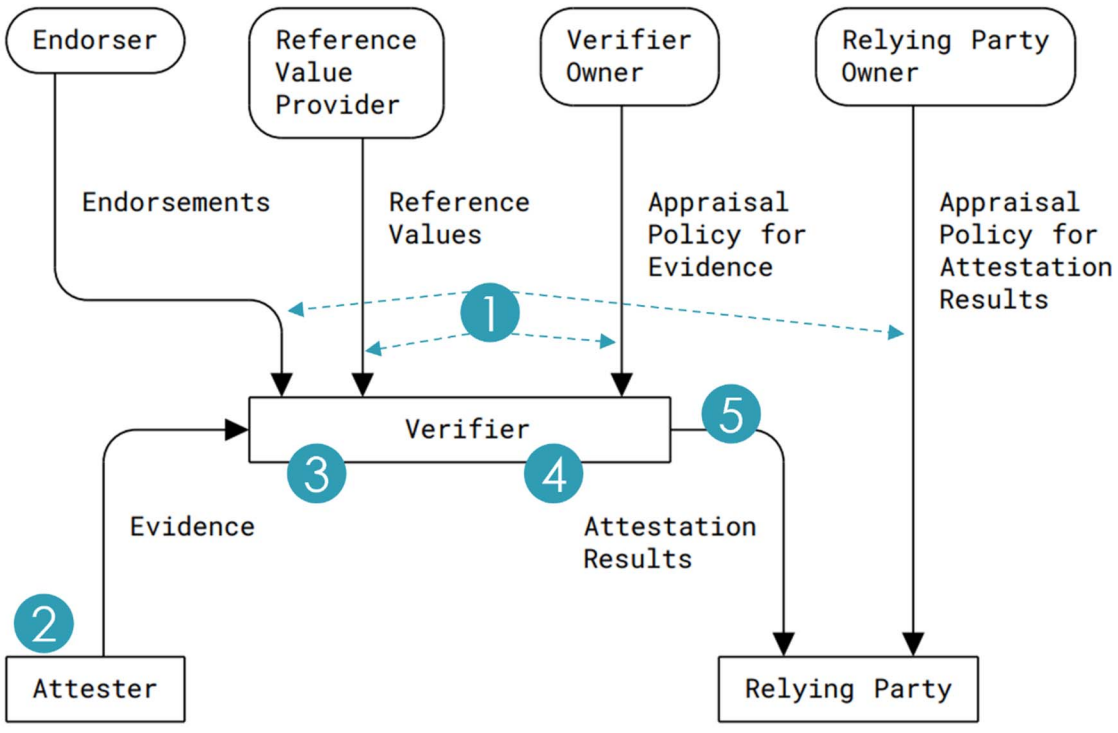


Figure 6.4-2: identity proofing process with RATS architecture

Before issuance of an identity document for VNFI/VNFCI, a proof of identity process, i.e. an attestation process, shall be performed as described in clause 8 of the present document describing the identity trust model.

7 Security constraints of identity

7.1 Usage and Consumption

7.1.1 Lifetime and uniqueness

Identities given to VNFs can last anywhere from seconds to years, depending on their specific needs, therefore specific use cases of identity require different security considerations. Security considerations for a short-term identity differ somewhat from those of a long-term identity, one of which is the level of uniqueness of the ID.

Uniqueness is defined as "the quality of being the only one of its kind", and for network functions, the scope of uniqueness can vary. There are generally two levels of uniqueness that can be given to an identifier, global and local, and the level of uniqueness is also granted a lifetime, either permanent, time-constrained, or instantaneous.

A Globally Unique Identifier (GUID) is a specific type of identifier which has the properties of, at a given moment in time, never being repeated at any point within the system. Within the NFV architecture, this could be used to assign a unique identifier to VNF packages, the VNF Descriptor (VNFD) and the VNF included within the package. This would ensure uniqueness across the NFV deployment at that point in time.

A Locally Unique Identifier (LUI) is one that is unique in its local area but may be repeated at other points in the system. Local addresses require an intermediary system if it wishes to communicate with a device outside of its local area.

A Locally Unique Identifier (LUI) in NFV can be created by a device number assigned to virtual network devices or virtual switches within a specific domain or NFV local environment. For instance, in NFV, each virtual switch and virtual network device in a domain can be assigned a unique device number (e.g. network@0, network@1) which is unique within that domain, but the same device number may be reused in another domain elsewhere in the system.

Unique identifiers enable the MANO to effectively manage the entire system, as each VNFI and VNFCI will be associated with a unique identifier, either globally or locally. In general, IDs given to VNFIs which are externally visible are often longer lived and their meanings are more human-comprehensible (i.e. attributes of the VNF can be gleaned from the ID). These VNFIs, which are longer lived are usually assigned a globally unique, human-comprehensible identifier (e.g. AMF 1 London). Whereas IDs given to VNFs that are not externally visible tend to be shorter lived and additionally these may or may not be human-comprehensible depending on the situation, for example individual component instances of a VNFI can be assigned a locally unique identifier that is not so easily human-comprehensible (e.g. 324121). These locally unique identifiers are bound to the globally unique identifier of the VNFI, allowing the MANO to differentiate VNFCIs between the various VNFIs.

The NFV architecture uses hierarchical layers of identifiers to manage the complexity of virtualized network functions. At the foundational level, the NFV layer assigns unique identifiers to Virtual Network Function Component Instances (VNFCIs), ensuring every component is distinguishable within the infrastructure. Moving upward, application layer and service layer identifiers aggregate and abstract these component-level identities into broader operational contexts. Through this method of ID layering, individual VNFCIs are easily tracked all through the management chain, from the top-level service level identifier through to the application layer identifier and then down to the NFV layer identifier. The ability to uniquely identify each VNFCI combined with a method of tracking when and where they are located provides the MANO a route for effective network management and post-event forensics.

7.1.2 Authentication

The administrator that performs the on-boarding step shall be authenticated before each on-boarding action, i.e. there will be a one-to-one relationship between an administrative login and a package on-boarding action.

7.1.3 Authorization

The administrator that performs the on-boarding step shall be specifically authorized to do so.

7.1.4 Accounting

The Security Manager shall maintain logs of all operations performed, who performed them, and where (infrastructure/host).

7.1.5 Integrity

Integrity of identities in NFV systems is maintained through robust mechanisms such as cryptographic hashing of VNF configurations, remote attestation of VNFCIs and NFVI hosts, and secure channels using encryption and certificate-based authentication to ensure trustworthiness and detect any unauthorized changes or anomalies in the network functions and their configurations as described in ETSI GR NFV-SEC 007 [i.2].

7.1.6 Replay Prevention

Replay attacks are performed by capturing messages being sent between two devices, then subsequently replaying that message to repeat the action for which the message was for. An example of this is if an attacker were to obtain the message from the MANO to the VIM which instructs the VIM to spin up a new virtual firewall. Without security defences from the system, an attacker can replay that message multiple times to repeat the command, which in this case would spawn multiple firewalls resulting in a protentional Denial of Service (DoS) by consuming too many resources.

Encryption alone will not prevent an attack if the attacker can guess the expected outcome of a message, to successfully defend against this attack vector, a combination of authentication and authorization of the sender and integrity protection of the message is key. Authentication and authorization ensure that the original sender is permitted to send the message and the integrity check ensures that the received message has not been modified in transit.

To mitigate replay attacks, messages can include a non-transmitted time-variant parameter like a counter or a timestamp as part of their cryptographic hash or encryption, requiring some synchronization mechanism. Time stamps primarily help with sequencing by allowing the system to discard messages that are out of sequence or outside an acceptable time window.

Other mitigations for this kind of attack include One-Time Passcodes (OTPs), unique session identities, and maximum number of attempts for a user to repeat a specific command within a time period (e.g. user1 may only manually delete 2 AMFs per day). Binding information, such as the timestamps and location stamps, to identities enhances security as proposed in the present document.

Regarding identity provision, the other factors may include: the size of the ID pool, variance in the time between allocation and de-allocation, and the time delay before a de-allocated ID is reallocated back into the system. These can all be used in combination with random ID selection to form a system with an entropy high enough to not be cracked through brute force, as the level of entropy is proportional to the time taken to crack a system.

This layered protection approach aligns with applying replay-resistant authentication mechanisms that incorporate counters, nonces, and timestamp validations to maintain synchronization and message uniqueness.

8 Identity Trust Model

8.1 Introduction

In new architecture, micro-services move and are deployed in different environments, across heterogeneous and hybrid infrastructures operating at various levels of trust and sensitivity, and in multi-vendor environments.

In ETSI-NFV infrastructures, it should be assumed that the security of such a complex network is always exposed to external and internal threats.

It shall be possible for a relying party to verify at any time an entities identity.

Establishing trust or confidence depends on the ability to bind unique attributes to a unique identity, and this binding shall be verifiable by the relying party. When a satisfactory level of confidence in the attributes provided by an entity is achieved, a trust relationship can be established.

The following clauses detail one particular method by which trust, rooted in an attestation process, can be achieved in the identity. In general, the described method is optional as other methods may exist, and hence, the process may be implemented at the discretion of the system designer or operator.

8.2 General Model

8.2.1 Introduction

The identity trust model defined in this clause is based on these processes that are essential to establish a trust relationship:

1) An identity proofing process, i.e. attestation process where some attributes of the entity (VNFI/VNFCI) are measured and verified against reference measurements: A proof that the entity is really what it claims to be.

2) A binding of these attributes with the unique identity consisting of the provisioning of a primary verifiable identity document that includes the proof of attestation, and credentials and is signed by the authority involved in the identity proofing process. Central to the trust model is the assurance that this binding is completely reliable.

3) An authentication process that occurs at the beginning of a trust relationship between entities, during the establishment of connection, wherein a relying party needs to verify who the entity is before accepting the connection (i.e. mutual TLS negotiation and handshake). This process uses the primary verifiable identity document of the entity, which a relying party can verify before starting the trust relationship.

4) An authorization process that allows a relying party to check further specific access privileges before enabling an entity to access to its service or resources. For this authorization process, the verification that these specific attributes match the security policies required for the access is performed. For this authorization process, a verifiable identity presentation, including specific attributes as verifiable credentials, is delivered by the entity to the relying party. For this process, the trust resides in the ability for the relying party to verify that the attributes are really linked to the entity.

8.2.2 Architecture

8.2.2.1 Architecture diagram

Figure 8.2.2.1-1 shows the functional architecture used for the Identity framework.

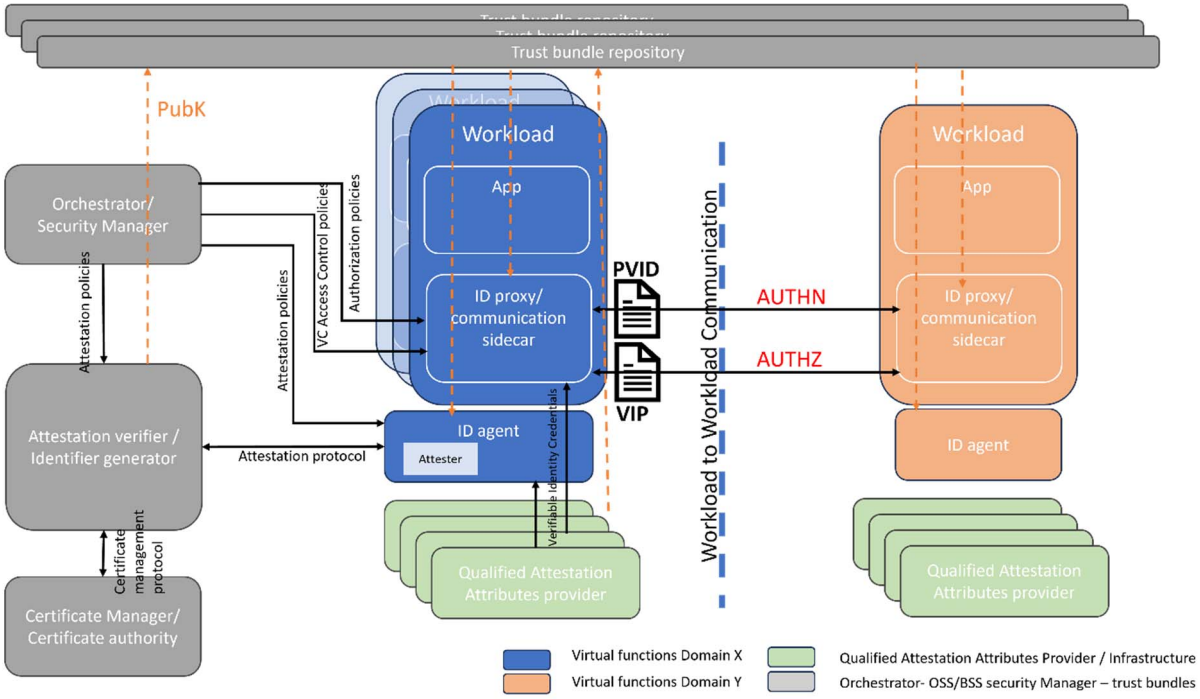


Figure 8.2.2.1-1: Identity trust model architecture

Primary verifiable identity document generation

The workload represented here, with its ID proxy and communication sidecar, obtains its primary identity during a bootstrapping process. This bootstrapping process is initialized by an ID agent and includes the proofing identity process and the binding with the unique identity as described in bullets 1 and 2 of clause 8.2.1.

The ID agent contains the attester and the description of the attestation policies with their selectors to get evidence from the operating system or hardware of the infrastructure. Evidences are signed measurements for each selector included in the attestation policies. The infrastructure and hardware providing these evidences are qualified attestation attributes providers.

The ID agent interact with a server, acting as an identifier generator and attestation verifier, to get an attestation result as an identity credential. This server is a qualified attestation attributes provider for this bootstrapping process. The ID agent with the attestation result, build the CSR and request the signing of the Primary Verifiable Identity Document including the attestation result to the Certificate Manager and CA directly or through the attestation server. This PVID is then provided to the communication sidecar to be used for the authentication process (AUTHN) of a workload to workload communication.

Workload to workload communication: AUTHN

The workload uses its PVID in X.509 format, during the TLS handshake. The counterpart entity, depending on its security policies may use the extension containing the attestation result to check if the workload complies with the security policies (e.g. integrity attestation successful, location correct, etc.) before accepting and establishing the secure channel with the workload.

Workload to workload communication: AUTHZ

Before giving access to the resource of the workload to a counterpart entity, the workload may check specific identity attributes of the counterpart entity using an attributes-based (ABAC) authorization process AUTHZ. These attributes depend on the context and security policies. The workload requests the specific identity attributes on the top of the authorization protocol (e.g. OAuth 2.0) using a protocol for requesting and presenting verifiable credentials as defined by OpenID4VP [6]. The counterpart entity creates its Verifiable Identity Presentation according to the request, an identity document as a JSON-based token including dedicated attributes, that the workload verifies against its authorization policies. For this verification the communication sidecar of the workload uses the trust bundle repository where the public keys of all qualified attestation attribute providers can be found.

The attributes included in the Verifiable Identity Presentation of the counterpart entity depend on access policies and on the trust domain of the workload that requests it. An attribute could be relevant inside the trust boundary but could expose sensitive information to the outside world. An access control to the attributes to be included in the VIP is an important feature and is included in the ID proxy.

8.2.2.2 Architecture entities

8.2.2.2.1 ID agent

The ID agent runs on every node on which a workload is running. The ID agent is responsible for providing the PVID of each workload after a successful attestation of the workload.

The ID agent's main components are:

- An attester which participates in the ID Agent attestation. The ID Agent attestation is the verification of the identity of the node the agent is running on providing node's evidence to the Attestation verifier. The evidence is provided after a request to the infrastructure for specific information relative to that node that proves its identity.
- A workload attester which verifies the identity of the workload through an attestation process: the workload attestation. The evidence is provided after a request to the infrastructure for specific information (selectors according to attestation policies) relative to that workload that proves its identity.
- A key manager to generate key pairs for the workloads and the PVID generation.

The ID agent exposes an interface to the ID proxy of the workload, an interface to the Attestation verifier/identifier generator, and an interface to the qualified attestation attributes providers, including the infrastructure to get evidence across multiple selectors for the attestation.

8.2.2.2.2 Workload and its ID proxy/communication sidecar

The workload is the application software included in the container and its ID proxy/communication sidecar.

The ID proxy/ communication sidecar is responsible for:

- Managing the communication with other external workloads within or outside the trust domain, establishing the communication channel (e.g. using mTLS) with the PVID provided by the ID agent.
- Requesting from the infrastructure and other qualified attestation attributes providers, the specific identity attributes of the workload (e.g. namespace, certification label, etc.) as verifiable credentials, i.e. the verifiable identity credentials.
- Storing the verifiable identity credentials.
- Generating the Verifiable Identity Presentation (VIP) of the workload to get authorization to access the other external workload API.
- Controlling the access to the verifiable identity credentials to be included in the VIP using the VC access control policies.
- Verifying the VIP of the other external workloads against authorization policies in order to authorize access to the workload API.

8.2.2.2.3 Qualified Attestation Attributes Provider / Infrastructure

The qualified attestation attributes provider is the provider of specific identity attributes as verifiable credentials as defined in W3C Verifiable Credentials Data Model v2.0 [8], i.e. verifiable identity credentials. This is a trusted party able to guarantee the veracity of the identity attributes and their binding to the workload. Some examples of such providers are:

- The infrastructure for some attributes such as the namespace (security domain) of the node, the geo-location coming from a trusted source, the container-ID, etc.

- The certification authority that delivers certification scheme certificates (certification labels), or a registrar that stores the Digital Letter of Approval as defined by GlobalPlatform in the Digital Letter of Approval specification [7].
- The RoT for Measurement (RTM) that delivers evidence for the attestation of integrity.

All the verifiable identity credentials are signed by the corresponding qualified attestation attributes provider.

NOTE: The Qualified Attestation Attributes Provider / Infrastructure plays the role of verifiable credential Issuer in the Verifiable Credential data model defined in W3C Verifiable Credentials Data Model v2.0 [8].

8.2.2.2.4 Attestation verifier / identity generator

The attestation verifier/identifier generator is responsible

- For managing and issuing the VNF Identity, i.e. the decentralized unique identifier, for the workloads of its trust domain.
- For storing the attestation policies, i.e. the selectors and the golden measurements that determine the conditions under which the PVID is issued to the workload.
- For providing the ID agent with the attestation policies of the workloads for which the ID Agent manages the identity.
- For authenticating the ID Agent by verifying the ID Agent attestation, and providing the PVID of the ID Agent.
- For verifying the workload attestation.
- For creating the signed PVID for workloads when requested by the ID Agent using a CSR with the collaboration of the certificate Manager/ Certificate Authority.

8.2.2.2.5 Certificate Manager and Certificate Authority

These are entities responsible for generating the signed PVID, the X.509-based ID document for the workload after the reception of the corresponding CSR.

8.2.2.2.6 Orchestrator / Security Manager

The orchestrator is involved in the registration of the ID Agent and the VNF/VNFC with the attestation verifier / Identifier generator.

The security manager or orchestrator provides the different security policies used in this identity framework:

- Attestation policies: the selectors and the golden measurements that determine the conditions under which the PVID is issued to the workload.
- The VC access control policies that determine which identity credentials can be included in the VIP depending on the trust domain of the counterpart workload that requests the Identity presentation.
- Authorization policies that determine the rules for authorizing access to the workload API, i.e. the Identity credentials type and their values.

8.2.2.2.7 Trust bundle repository

The trust bundle repository is an object containing the cryptographic keys used for a specific trust domain, authoritative for the trust domain that the bundle represents and used to prove the validity of the Identity documents PVID and VIP for workloads and ID agents residing in this trust domain. It contains the cryptographic public keys of the qualified attestation attributes providers for the verification of the identity credentials, and the public keys used for the signature of the various identity documents used in the trust domain.

The trust bundles are used to federate several trust domains and give access to the public keys used across different trust domains. It enables a workload from a trust domain A to verify the Identity credentials and identity documents of another workload from trust domain B.

8.2.2.3 Architecture flows

8.2.2.3.1 High level flow

This clause gives a high-level description of the different processes of the identity trust model.

1) Registration of the workload in the attestation verifier / Identifier generator: The identity framework starts with the instantiation request of a virtual function (or workload). At this time, a registration process is done between the orchestrator and the attestation verifier / Identifier generator to provide the selectors and the golden measurements for the attestation.

2) Instantiation of ID Agent and PVID provisioning for this ID agent: Before the instantiation of the virtual function, an instantiation of an ID agent is done in the node where the virtual function is to be instantiated. This ID agent is attested, a PVID is issued to the ID agent.

3) Instantiation of the workload and PVID provisioning for the workload: ID agent is configured with the selectors and identifiers of the workloads that this ID agent will manage in the node. The workload is instantiated with its ID proxy and communication sidecar. After a successful attestation of the workload, a PVID including attestation attributes is provided to the ID proxy of the workload.

4) Adding a Verifiable Identity Credential in the ID proxy: A trusted provider of verifiable identity credentials, the qualified attestation attributes provider, issues the Verifiable credentials to the ID proxy at the request of the ID proxy or at the initiative of the qualified attestation attributes provider. For the verifiable identity credential issuance, the protocol OpenID4VCI [9] is used.

5) Interaction with the third party: After a mutual authentication using their PVIDs, two communicating workloads use an identity presentation of the counterpart workload with selected attributes to authorize access to their resources and API. For this interaction, the protocol OpenID4VP [6] implemented on top of OAuth2.0 defined by IETF RFC 6749 [10] and using the DIF presentation exchange [11] is used.

Figure 8.2.2.3.1-1 below depicts the high-level flow of the different processes.

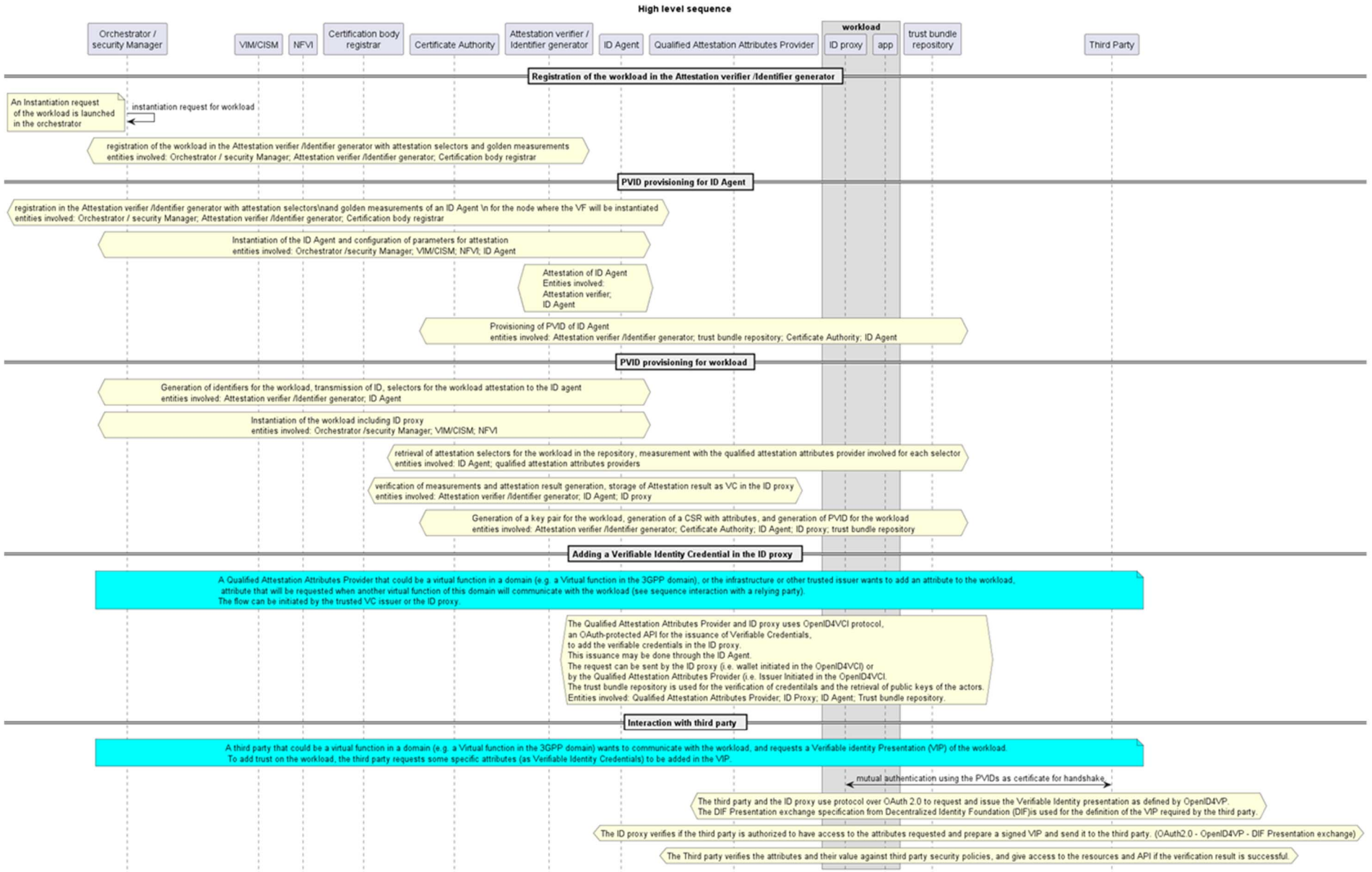


Figure 8.2.2.3.1-1: High level sequence

8.2.2.3.2 Attestation and PVID provisioning for ID Agent

This clause describes the sequence for the instantiation of the ID Agent on the node where the workload is to be instantiated, the attestation of the ID Agent and the provisioning of the PVID of the ID Agent by the Attestation verifier/ Identifier generator.

Figure 8.2.2.3.2-1 depicts the sequence for the instantiation, Attestation and PVID Provisioning of ID Agent

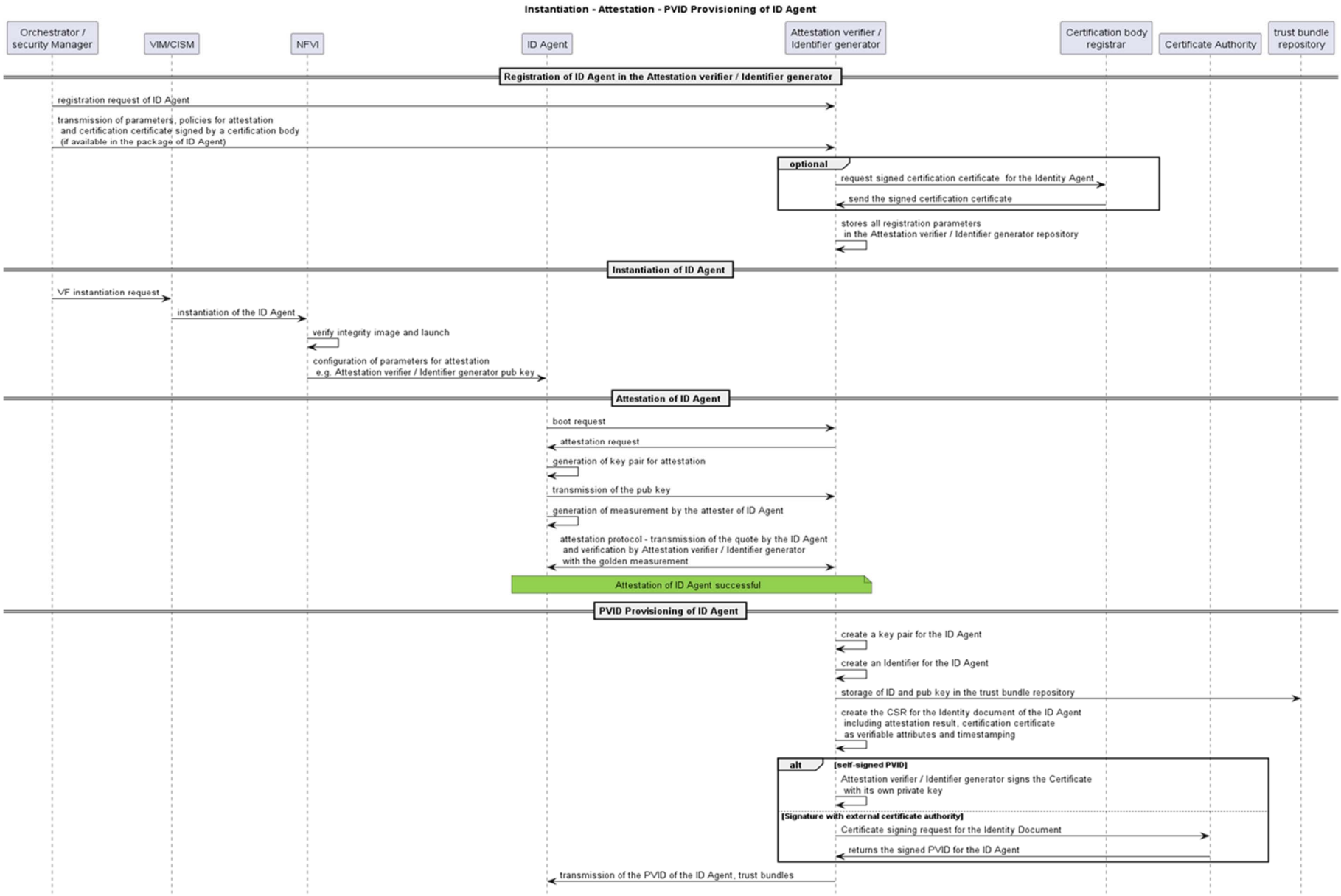


Figure 8.2.2.3.2-1: Instantiation - Attestation - PVID Provisioning of ID Age

8.2.2.3.3 Attestation and PVID provisioning for container-based workload

This clause describes the sequence for the registration in the verifier of the golden measurements and attestation selectors for the VNF/VNFC, the instantiation and attestation of the VNF/VNFC and the provisioning of the PVID for the VNF/VNFC.

The attestation is based on RATS, where the Attester is the Identity Agent, the verifier is the attestation verifier/identifier generator, and the relying party is the certificate authority that will deliver the signed certificate if the attestation result is successful.

The sequences are described in several figures presented below:

- Figure 8.2.2.3.3-1 describes the first phase: the registration of the VNF in the Attestation verifier / identifier generator.
- Figure 8.2.2.3.3-2 describes the second phase: the instantiation of each VNFC of VNF and their ID proxy, with the generation of the Identifiers, and the transmission of attestation selectors to the ID Agent for further attestation.
- Figure 8.2.2.3.3-3 describes the third phase: the measurement for each attestation selector.
- Figure 8.2.2.3.3-4 describes the fourth phase: the attestation result as a Verifiable Credential (VC) storage in the ID Proxy.
- Figure 8.2.2.3.3-5 describes the fifth phase: the generation of the PVID foreach identity of the VNFC and its storage in the ID proxy.

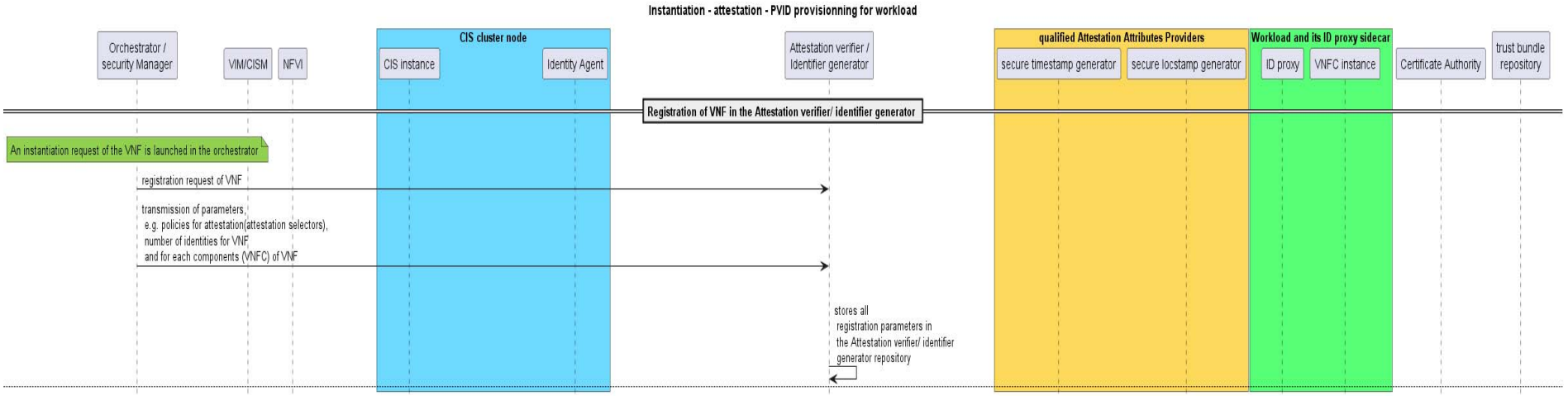


Figure 8.2.2.3.3-1: Registration of VNF in the Attestation verifier/ identifier generator

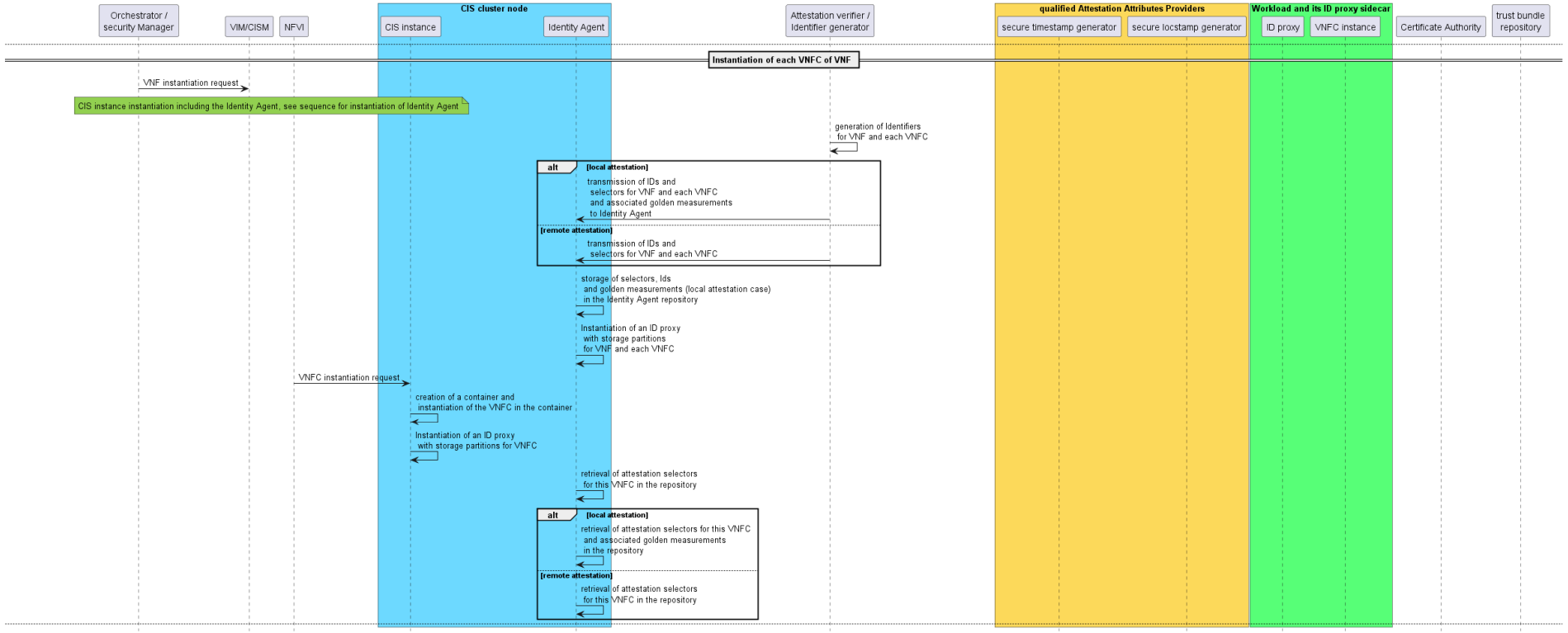


Figure 8.2.2.3.3-2: Instantiation of each VNFC of VNF

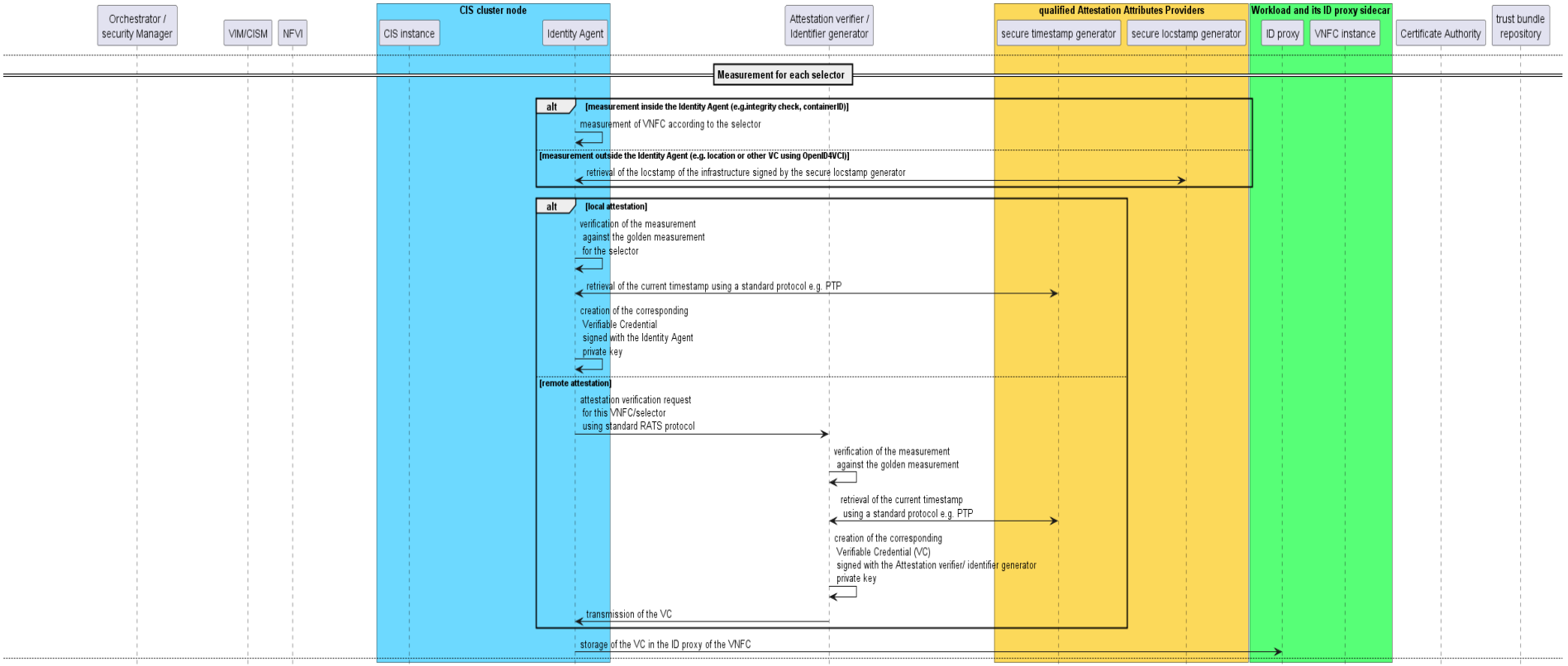


Figure 8.2.2.3.3-3: Measurement for each selector

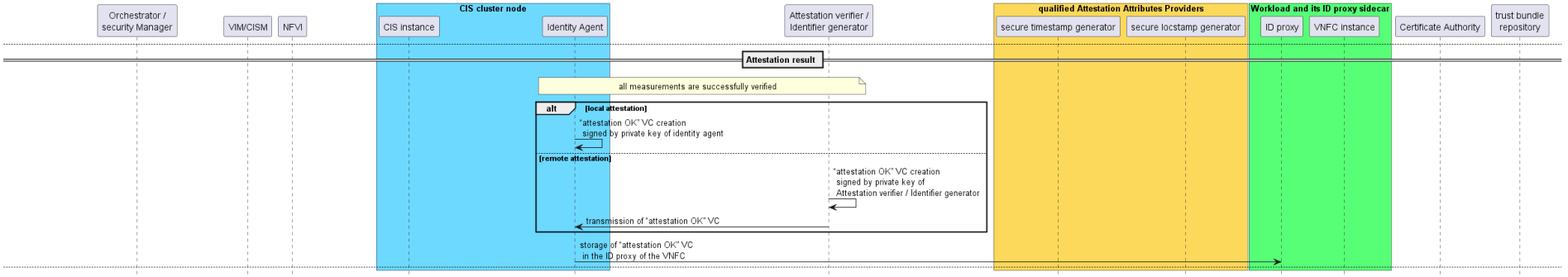


Figure 8.2.2.3.3-4: Attestation result

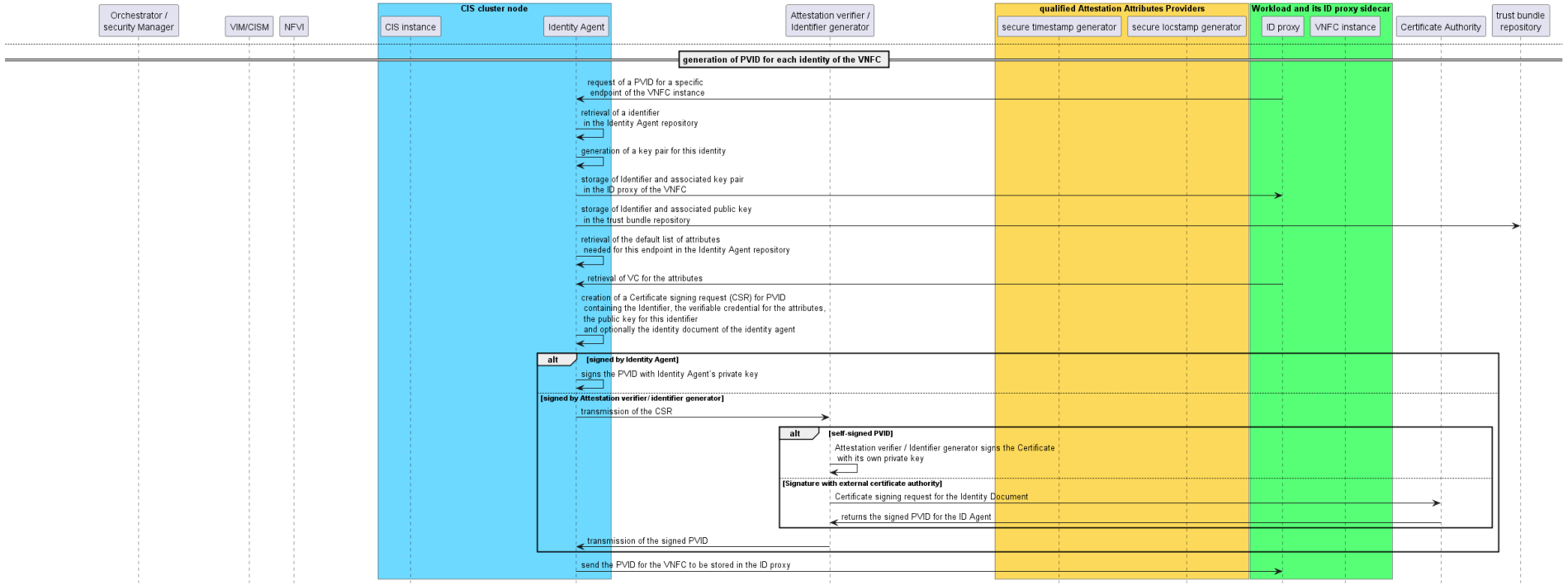


Figure 8.2.2.3.3-5: generation of PVID for each identity of the VNFC

After the provisioning of the PVID, the VNFC (e.g. VNFC1) will use its PVID containing the attestation result and other attributes as a certificate for mutual authentication when communicating with another VNF/VNFC (e.g. VNFC2). The VNFC2 will then be able to check if the VNFC1 has been attested for its integrity and the result of this attestation and the location of the infrastructure where the VNFC1 is instantiated, before establishing a secure channel with the VNFC1. The VNFC1 will be able to do the same with the PVID of the VNFC2.

NOTE: The local attestation case has security implications, especially around the transmission of the golden measurements and their storage within the Identity Agent.

8.2.2.3.4 Adding a Verifiable Identity Credential in the ID proxy

This clause describes how a verifiable identity credential is issued by a qualified attestation attributes provider, a trusted provider of verifiable identity credentials.

The Qualified Attestation Attributes Provider and ID proxy use OpenID4VCI [9] protocol, an OAuth-protected API for the issuance of Verifiable Credentials, to add the verifiable identity credentials in the ID proxy. This issuance may be done through the ID Agent. The request can be sent by the ID proxy (i.e. wallet initiated in the OpenID4VCI [9]) or by the Qualified Attestation Attributes Provider (i.e. Issuer Initiated in the OpenID4VCI [9]). The trust bundle repository is used for the verification of credentials and the retrieval of public keys of the actors.

The verifiable identity credential issued should be cryptographically bound to the identifier of the VNFC that possesses the identity credential. This cryptographic binding allows the entity receiving this verifiable identity credential in a Verifiable Identity Presentation (VIP) to verify that the VNFC presenting the credential is the same to whom that credential was issued. OpenID4VCI [9] is used for this process, where the credential issuer is the trusted VC issuer of this current specification. Some restrictive clauses are provided in the present document.

Discovery of trusted VC issuers and the verifiable identity credentials they provide is done through the trust bundle repository. The discovery endpoint is provided to the ID agent by the Attestation verifier / identifier generator when the ID agent is instantiated.

After the discovery of the trusted VC issuers, The ID proxy or ID Agent if the Verifiable Identity credentials are provisioned through the ID Agent, retrieves the trusted VC issuer metadata. These metadata are published in a JSON document available at the path formed by concatenating the string (specific for this nfvid scheme, differs from the OpenID4VCI [9] specification) /**.well-known/nfvid-credential-issuer** to the trusted VC issuer identifier (as defined in clause 6.2 of the present document).

The path formed shall point to a JSON document compliant the OpenID4VCI [9] and shall be returned using the application/json media type.

The trusted VC issuer metadata parameters are specified in OpenID4VCI [9], section 10.2.3:

- credential_issuer: the identifier of the trusted VC issuer as defined in 6.2
- authorization_servers: identifier of the OAuth 2.0 authorization server as defined in IETF RFC 8414 [12], the trusted VC issuer relies on for authorization. This parameter is used only if a specific authorization server is used. If this parameter is omitted, the trusted VC issuer is also acting as an authorization server, and the trusted VC issuer identifier is used as OAuth 2.0 issuer value.

Other metadata parameters specify the URL of the credential issuer endpoints, the encryption algorithms, if supported, and the list of credentials that the trusted VC issuer supports the issuance of. This list is a name/value pairs differs from the OpenID4VCI [9] examples that are relative to persons. The list of credentials is defined in clause 6.3.2 of the present document.

Figure 8.2.2.3.4-1 below depicts the sequence for the issuance of a Verifiable identity credential in the ID proxy initiated by the ID proxy.

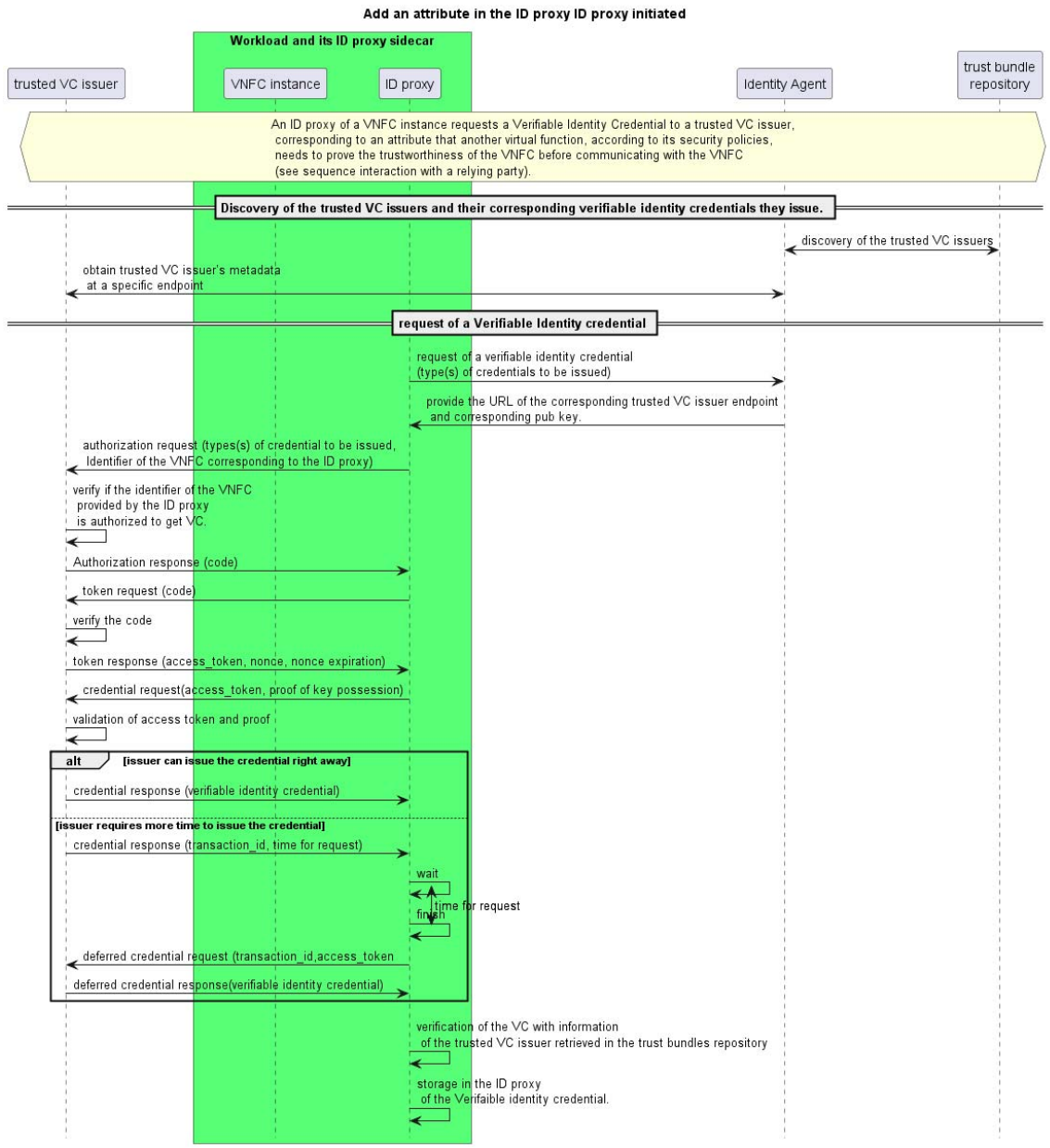


Figure 8.2.2.3.4-1: Sequence for the issuance of a Verifiable identity credential in the ID proxy initiated by the ID proxy

8.2.2.3.5 Interaction with third party

The present clause describes the sequence for a fine-grain authorization process, where a VNFC request the authorization to access to resource of a third-party suing OAuth 2.0. The third-party uses a fine-grain authorization process.

For this aim, the third-party requests to a VNFC a Verifiable Identity Presentation with specific verifiable identity credentials, according to its security policies.

The protocol used for this process is based on OpenID4VP [6] specification. OpenID4VP [6] defines a mechanism on top of OAuth 2.0 IETF RFC 6749 [10] that enables presentation of Verifiable Credentials as Verifiable Presentations.

The authorization request sent by the third-party contains the definition of the verifiable identity presentation, with the attributes needed for the third-party to have trust in the VNFC, according to its security policies. The verifiable identity presentation definition follows the DIF Presentation Exchange [11].

The third-party, with the VIP of the VNFC, can assess if it can open its API to the VNFC, delivering an access token to the VNFC for this access.

Figure 8.2.2.3.5-1 below describes this interaction with the third-party.

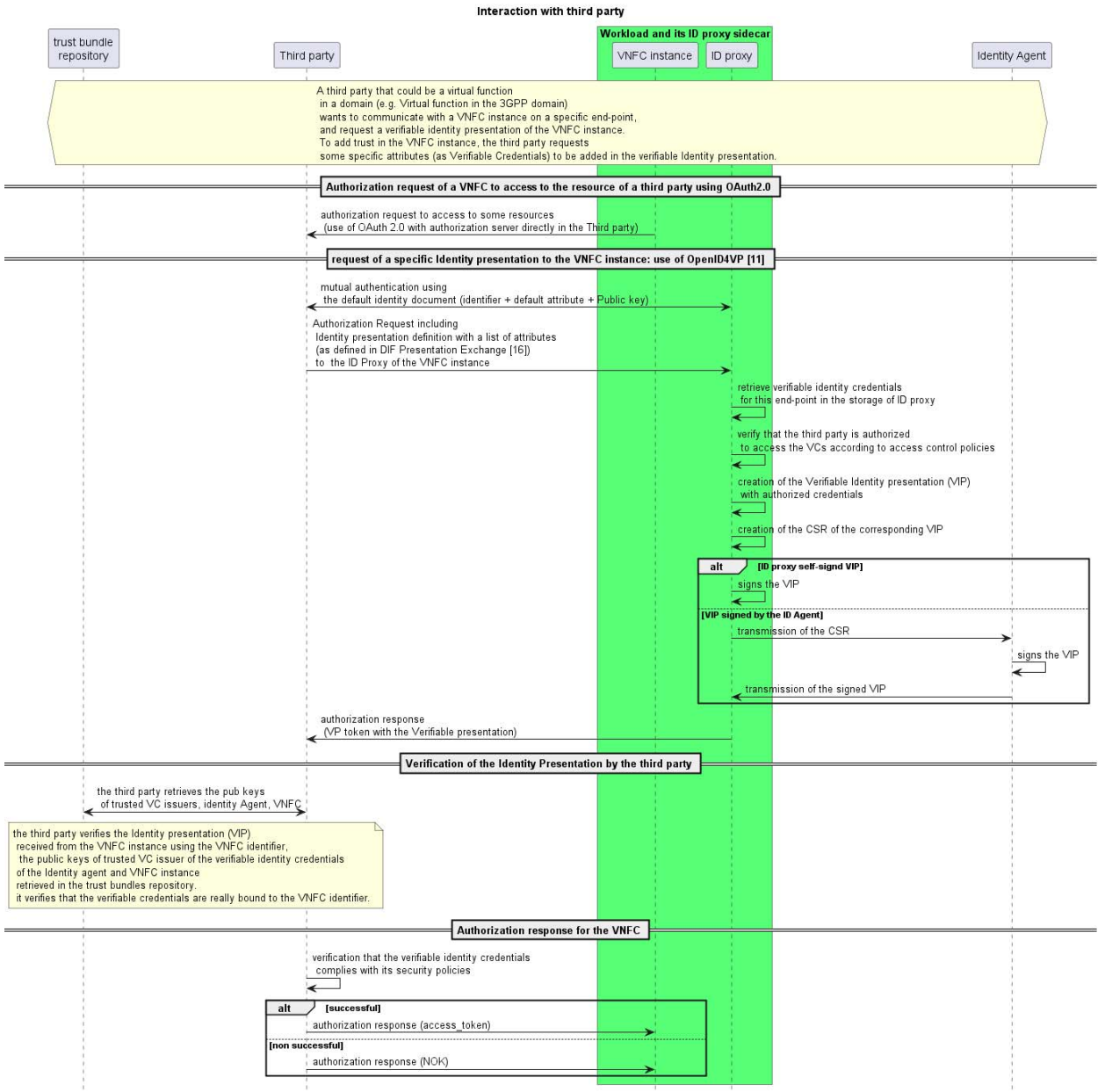


Figure 8.2.2.3.5-1: Sequence for interaction with third-party

8.2.3 VNFI/VNFCI Verifiable Identity Documents

8.2.3.1 Introduction

The VNFI/VNFCI Verifiable Identity Document (VID) is a document that the VNFI/VNFCI uses to prove its identity to a communicating party. The present document contains the Identity of the VNFI/VNFCI and is signed by an authority within the VNFI/VNFCI trust domain.

The VNFI/VNFCI VID is used for two different processes:

- Authentication to secure the process-to-process communication and establish a secure communication channel.
- Authorization to control the access to resources and API.

For the authentication process, as the secure communication channel is established using a TLS or mTLS, an identity document based on an X.509 certificate is well adapted, enabling the authentication system to be interoperable, platform agnostic and enabling legacy usage. In the following of the present document, the corresponding ID document is named Primary Verifiable Identity Document (PVID). This is the first identity document used for exchanges between VNFIs/VNFCIs.

For the authorization process, there is a need for an extensible identity document, adaptable to the VNFI/VNFCI counterpart policies. A JSON-based document including claims describing the identity attributes of the VNFI/VNFCI is more adapted for a cross-domain communication. The present document provides a specific identity presentation of the VNFI/VNFCI, with attributes that are dynamically learned and updated, and those attributes are used as match criteria to enable trust and give access. The present document enables an access control based on attributes (ABAC). In a changing environment, this dynamicity in the policy change and update of attributes enable a continuous verification in the environment and enable applying the zero-trust principle. In the following of the present document, the corresponding verifiable ID document is named Verifiable Identity Presentation (VIP).

8.2.3.2 Primary Verifiable Identity Document

8.2.3.2.0 Introduction

In clause 8.2.3.2, an EAT is included in X.509 to create the PVID to be presented to another workload. It is acknowledged that the EAT is used outside its intended scope, but for the time being, neither RATS nor LAMPS have defined attributes suitable for including an EAT in a certificate. Hence, for the time being, EAT will be kept and the present document updated once IETF or RATS has progressed on said topics.

8.2.3.2.1 PVID: X.509-based document

The Primary Verifiable Identity Document (PVID) is an X.509-based VID following the IETF RFC 5280 [4]. The PVID SHALL include the identity of the subject and MAY include in extensions identity attributes associated to the subject. This clause addresses the encoding of the PVID information into an X.509 version 3 certificate, the constraints which are set, and how to validate the X.509 PVID.

The VNFI/VNFCI PVID is an end entity certificate as defined in IETF RFC 5280 [4]. The CA field in the Basic constraints extension shall be set to false, as defined in clause 4.2.1.9 of IETF RFC 5280 [4].

8.2.3.2.2 Identity

The Identity of a VNF/VNFC instance is defined in clause 6.2 of the present document and is defined as an URI. This identity shall be included in the X.509 PVID in the Subject Alternative Name extension (SAN extension) as defined in clause 4.2.1.6 of the IETF RFC 5280 [4]. The PVID shall contain exactly one URI SAN and then exactly one Identity.

A PVID containing more than one URI SAN shall be rejected during the validation of the PVID.

The PVID may contain any number of other SAN field types e.g. DNS SANs.

The Subject field is not required, however the URI SAN shall be marked as critical if the Subject field is omitted, as defined in clause 4.1.2.6 of IETF RFC 5280 [4].

8.2.3.2.3 Key Usage and Extended Key Usage

The Key Usage extension defines the purpose of the key contained in the certificate. The key usage extension is defined in clause 4.2.1.3 of IETF RFC 5280 [4]. The Key Usage Extension shall be set and shall be marked as critical. The VNFI/VNFCI PVID shall set digitalSignature and may set keyEncipherment or keyAgreement. The keyCertSign or cRLSign shall not be set.

The Extended Key Usage extension indicates one or more purposes for which the key contained in the certificate may be used, in addition to or in place of the basic purposes indicated in the key usage extension. It is defined in clause 4.2.1.12 of IETF RFC 5280 [4]. The VNFI/VNFCI PVID should include this extension, and it may be marked as critical. When included, fields id-kp-serverAuth and id-kp-clientAuth shall be set.

8.2.3.2.4 Identity attributes

Identity attributes as defined in clause 6.3.2 of the present document may be included in the PVID. The attributes included in the PVID are chosen by the CSP. It is recommended to include in the PVID only identity attributes that change infrequently, so as to minimize the potential administrative overhead involved in re-issuing certificates due to changes in non-essential information.

The identity attributes are included in the Subject Directory Attributes extension (SDA extension) as defined in clause 4.2.1.8 of IETF RFC 5280 [4]. This extension shall be marked as non-critical.

The extension is defined as a sequence of one or more attributes:

```
id-ce-subjectDirectoryAttributes OBJECT IDENTIFIER ::= { id-ce 9 }

SubjectDirectoryAttributes ::= SEQUENCE SIZE (1..MAX) OF Attribute
```

8.2.3.2.5 Identity attributes type and Value

8.2.3.2.5.1 Attestation result: Entity Attestation Token

The Entity Attestation Token as defined in the IETF RATS Entity Attestation Token (EAT) [5], is a CBOR (CWT) web token or a JSON Web Token (JWT) containing attestation-oriented claims. The EAT JWT may be included in the Subject Directory Attributes extension (SDA extension) of the PVID to add attestation result, location, timestamp and certification results related to the VNFI/VNFCI.

The claims that could be included in the EAT are the followings shown in table 8.2.3.2.5.1-1.

Table 8.2.3.2.5.1-1: Claims in the EAT for attestation result

Claims	Description
iat	The "iat" claim defined in CWT and JWT is used to indicate the date-of-creation of the token, the time at which the claims are collected and the token is composed and signed.
measres	The "measres" claim is a general-purpose structure for reporting comparison of measurements to expected reference values. This claim provides a simple standard way to report the result of a comparison as success, failure, fail to run, and absence.
location	The "location" claim gives the geographic position of the entity from which the attestation originates. Latitude, longitude, altitude, accuracy, altitude-accuracy, heading and speed shall be as defined in the W3C Geolocation API.
dloas	The "dloas" claim conveys one or more Digital Letters of Approval (DLOAs). This claim is issued by a verifier. See note.
NOTE: A DLOA is a document that describes a certification that an entity has received. Examples of certifications represented by a DLOA include those issued by Global Platform and those based on Common Criteria. The DLOA is unspecific to any particular certification type or those issued by any particular organization.	

8.2.3.2.6 X.509 PVID

Table 8.2.3.2.6-1 below describes the specific fields of the X.509 certificate defining the PVID in complement of the description given in other sub-clauses of clause 8.2.3.2.

Table 8.2.3.2.6-1: X.509 PVID

Field	M/O	Value description
Subject	O	
Subject Alternative Name extension (IETF RFC 5280 [4], clause 4.2.1.6)	M	ExtnID = id-ce-subjectAltName (id-ce 9) Critical = true if subject field is not present uniformResourceIdentifier = identity URI as defined in clause 6.2. as a IA5String
Key Usage Extension (IETF RFC 5280 [4], clause 4.2.1.3)	M	extnID = id-ce-keyUsage (id-ce 15) critical = True extnValue = {digitalSignature (M), keyEncipherment (O) or keyAgreement (O)} keyCertSignn or cRLSign shall not be set
Extended Key Usage extension (IETF RFC 5280 [4], clause 4.2.1.12)	O	ExtnID = id-ce-extKeyUsage (id-ce 37) Critical = may be true, otherwise false extnValue = { id-kp-serverAuth (M), id-kp-clientAuth (M)}
Subject Directory Attributes extension (IETF RFC 5280 [4], clause 4.2.1.8)	O	ExtnID = id-ce-subjectDirectoryAttributes (id-ce 9) critical = false extValue = {list of Attributes}as defined in clause 8.2.3.2.5

8.2.3.3 Verifiable Identity Presentation

8.2.3.3.1 Introduction

As described in clause 8.2.2.3.5, the third-party that needs to check if it can trust the VNFC according to its own security policies, sends an authorization request to the ID proxy of the VNFC with the definition of the verifiable identity presentation (Presentation Definition as defined in DIF Presentation Exchange [11]), with requested attributes needed for the third-party to assess trust in the VNFC.

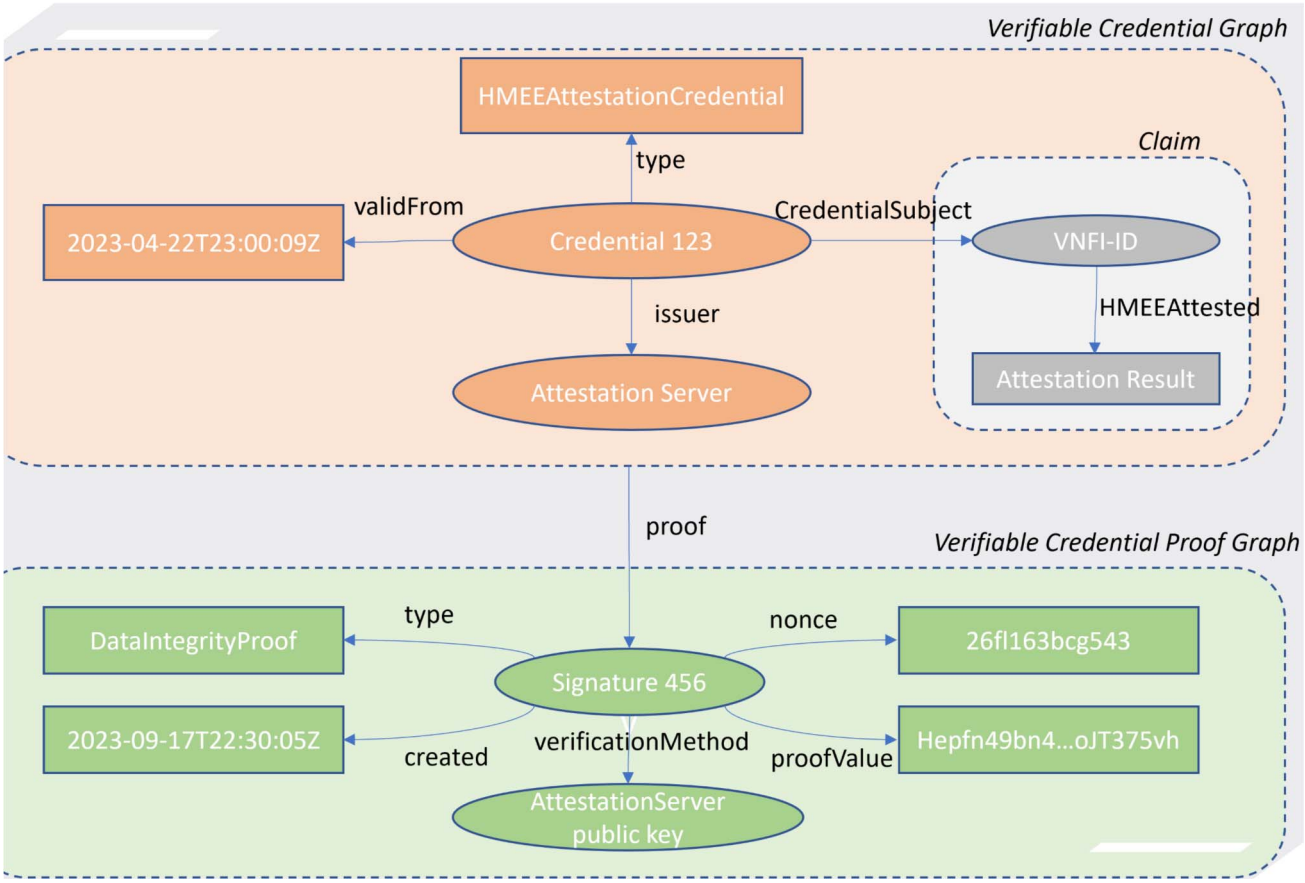
The ID proxy of the VNFC sends back a response to this authorization request containing the VIP document including the verifiable credentials that have been requested in the presentation definition and a presentation submission (as defined in DIF Presentation Exchange [11] that includes a mapping between the claims in the VIP and the requested attributes in the presentation definition. The VIP document is the vp_token describes in the OpenID4VP [6] specification.

8.2.3.3.2 Verifiable Credentials and Verifiable Presentation data model

Several data models for the verifiable credentials may be used.

For this version of the present document, the data model for the verifiable credentials and the verifiable identity presentation is as defined in W3C Verifiable Credentials Data Model v2.0 [8].

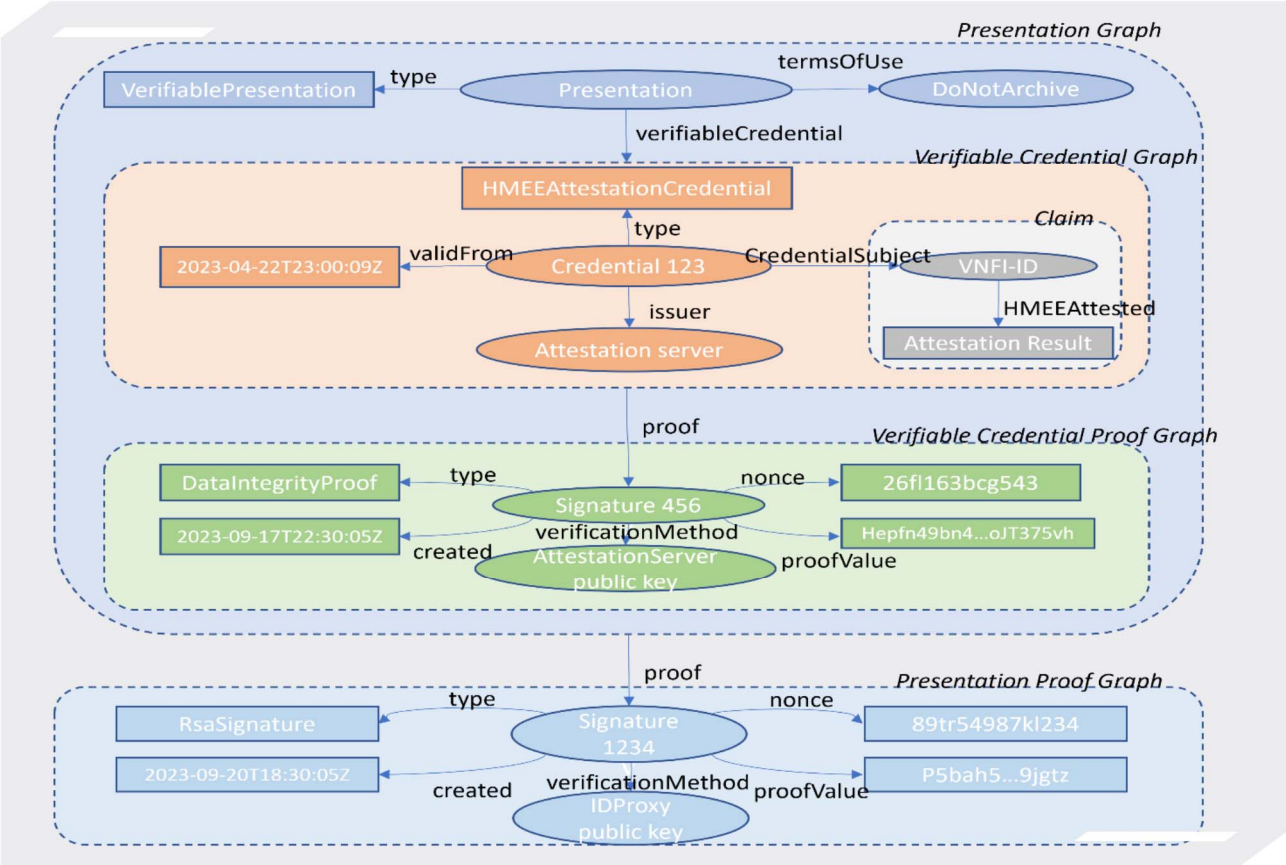
Figure 8.2.3.3.2-1 below shows an example of a verifiable credential graph as defined by W3C Verifiable Credentials Data Model v2.0 [8] representing an HMEEAttestationCredential.



NOTE: This figure is derived from figure 9 of W3C Verifiable Credentials Data Model v2.0 [8]. Copyright© 2025 World Wide Web Consortium. https://www.w3.org/copyright/software-license-2023/.

Figure 8.2.3.3.2-1: W3C verifiable credential graph

Figure 8.2.3.3.2-2 below shows an example of a verifiable presentation graph as defined by W3C Verifiable Credentials Data Model v2.0 [8] that includes a single verifiable credential, the HMEEAttestationCredential represented in figure 8.2.3.3.2-1 above.



NOTE: This figure is derived from figure 9 of W3C Verifiable Credentials Data Model v2.0 [8]. Copyright© 2025 World Wide Web Consortium. https://www.w3.org/copyright/software-license-2023/.

Figure 8.2.3.3.2-2: W3C verifiable presentation graph

8.2.3.3.3 VIP: JSON-based document

8.2.3.3.3.1 Description

The Verifiable Identity Presentation is a token-based Identity document.

For this version of the present document, the JWT technology as defined in IETF RFC 7519 [13] is the token-based document used for the Verifiable Identity Presentation (VIP). This type of document is very well adapted to the Verifiable Identity Presentation (VIP) which shall be extensible and compatible with existing applications and libraries. Additionally, this JWT is a compact format for representing claims and may be signed using a JSON Web Signature (JWS) structure as defined in IETF RFC 7515 [14] and/or encrypted using JSON Web Encryption (JWE) structure as defined in IETF RFC 7516 [15].

Other formats of JWT technology may be used such as JSON-LD as defined by W3C, or IETF RFC 9901 [i.14] allowing a selective disclosure of JWT claims.

8.2.3.3.3.2 VIP header

8.2.3.3.3.2.0 General

The VIP header is the JOSE header describing the cryptographic operation applied to the VIP and additional properties.

8.2.3.3.3.2.1 Algorithm header

"alg" header: defines the algorithm used for the VIP. The "alg" values supported are the following.

Table 8.2.3.3.3.2.1-1

Alg Param Value	Digital Signature Algorithm
RS256	RSASSA-PKCS1-v1_5 using SHA-256
RS384	RSASSA-PKCS1-v1_5 using SHA-384
RS512	RSASSA-PKCS1-v1_5 using SHA-512
ES256	ECDSA using P-256 and SHA-256
ES384	ECDSA using P-384 and SHA-384
ES512	ECDSA using P-521 and SHA-512
PS256	RSASSA-PSS using SHA-256 and MGF1 with SHA-256
PS384	RSASSA-PSS using SHA-384 and MGF1 with SHA-384
PS512	RSASSA-PSS using SHA-512 and MGF1 with SHA-512

8.2.3.3.3.2.2 Key ID header

The "kid" header is optional

8.2.3.3.3.2.3 Type header

The "typ" header is optional and if set, its value shall be either JWT or JOSE.

8.2.3.3.3.3 VIP claims for verifiable presentation

8.2.3.3.3.3.1 Generic claims

Table 8.2.3.3.3.3.1-1

Claim	Optional/Mandatory	Description	Value
issuer claim: "iss"	Optional	This claim identifies the issuer of the VIP.	Identity of the IDproxy of the workload that issues the VIP as defined in clause 6.2.
Subject: "sub"	Mandatory	This claim identifies the subject of the VIP.	identity of the workload as defined in clause 6.2.
Audience: "aud"	Mandatory	This claim identifies the entity that the VIP is intended for.	client-id sent in the presentation request.
Expiration time: "exp"	Mandatory	This claim identifies the expiration time after which the VIP shall not be accepted for processing.	The value is a number containing a NumericDate value. The value shall take into account the clock skew between the emitter of the VIP and the verifier of the VIP.
Not Before: "nbf"	Optional	This claim identifies the time before which the VIP shall not be accepted for processing.	The value is a number containing a NumericDate value. The value shall take into account clock skew between the emitter of the VIP and the verifier of the VIP.
Issued At: "iat"	Optional	This claim identifies the time at which the VIP was issued.	The value is a number containing a NumericDate value.

Claim	Optional/Mandatory	Description	Value
JWT ID: "jti"	Optional	This claim provides a unique identifier for the VIP. The "jti" claim can be used to prevent the JWT from being replayed. Collisions among values produced by this issuer or by different issuers shall be prevented.	Unique identifier (globally unique).
"nonce"	Mandatory	This claim is used to securely bind the VIP to a particular transaction.	The value of the nonce sent in the presentation request. This nonce is a case-sensitive String .

8.2.3.3.3.3.2 VIP Additional claims already registered in IANA

Table 8.2.3.3.3.3.2-1

Consortium who defines the claim	Claim	Optional/Mandatory	Description	Value
W3C Verifiable Credentials Data Model	**"vp"**	Optional	This claim is used to include the verifiable credentials or attributes.	This claim is defined by W3C Verifiable Credentials Data Model v2.0 [8]
	"vc"	Optional	This claim is used to define the verifiable credentials in the verifiable presentation.	This claim is defined by W3C Verifiable Credentials Data Model v2.0 [8]
	"proof"	Optional	This claim gives a digital proof that make the verifiable credential or the verifiable presentation tamper-evident. the proof is usually a digital signature.	This claim is defined by W3C Verifiable Credentials Data Model v2.0 [8] but not registered in IANA
RATS eat	**"iat"**	Optional	This claim identifies the time at which the eat was issued, i.e. the time of the attestation of the VNFC.	IETF RFC 9711 [5]
	"measres"	Optional	This claim is a general-purpose structure for reporting comparison of measurements to expected reference values. This claim provides a simple standard way to report the result of a comparison as success, failure, fail to run, and absence.	IETF RFC 9711 [5]
	"location"	Optional	This claim gives the geographic position of the VNFC from which the attestation originates. Latitude and longitude shall be provided is the location claim is included in the VIP. This claim may be used for the instantiation locstamp.	IETF RFC 9711 [5]
	"dloas"	Optional	This claim conveys one or more Digital Letters of Approval (DLOAs). A DLOA as defined by GlobalPlatform Card GPC_SPE_095 [7] is a document that describes a certification that an entity has received.	IETF RFC 9711 [5]
	"swname"	Optional	This claim contains a very simple free-form text value for naming the software used by the entity.	IETF RFC 9711 [5]

Consortium who defines the claim	Claim	Optional/Mandatory	Description	Value
	"swversion"	Optional	This claim gives a simple version for the software. This claim shall only be present if a "swname" claim is present.	IETF RFC 9711 [5]
	"hwmodel"	Optional	This claim differentiates hardware models, products and variants manufactured by a particular OEM, identified by oemid claim, and is present only if "oemid" claim is present.	IETF RFC 9711 [5]
	"hwversion"	Optional	This claim is a text string the format of which is set by each manufacturer, and is present only if the "hwmodel" claim is present.	IETF RFC 9711 [5]
	"uptime"	Optional	This claim contains the number of seconds that have elapsed since the VNFC was last booted.	IETF RFC 9711 [5]
	"bootcount"	Optional	This claim contains a count of the number times the VNFC has been booted. Support for this claim requires a persistent storage on the device.	IETF RFC 9711 [5]
	"oemid"	Optional	This claim identifies the Original Equipment Manufacturer (OEM) of the hardware.	IETF RFC 9711 [5]

8.2.3.3.3.3.3 VIP specific claims not registered in IANA

Table 8.2.3.3.3.3.3-1

Claim	Optional/Mandatory	Description	Value
"inst_timestamp"	Optional	This claim gives the instantiation time of the VNFC as a timestamp.	Number containing a NumericDate value.
"loa"	Optional	This claim gives the Level of Assurance for the VNFC.	Value from 0 to 5 including 5a and 5b as defined in ETSI GR NFV-SEC 007 [i.2].
"nfv_mano_id"	Optional	This claim identifies the NFV_MANO function instances (e.g. NFVO, VNFM, VIM) which effected the launch of VNFC.	
"namespace"	Optional	This claim identifies the security domain or namespace of the VNFC.	
"cgroup"	Optional	This claim identifies the cgroup of the container where the VNFC is instantiated.	
"containerid"	Optional	This claim identifies the container ID of the container where the VNFC is instantiated.	
"3gpp_nf_name"	Optional	This claim provides the name of the 3GPP network function the VNFI implements.	
"3gpp_nf_type"	Optional	This claim provides the type of the 3GPP network function the VNFI implements.	
"3gpp_nf_role"	Optional	This claim provides the role of the 3GPP network function the VNFI implements.	

8.2.3.3.3.4 Verification of the VIP

The verifiable presentations shall be verified against replay attacks.

The cryptographic proof of possession in the verifiable presentation shall be bound to the intended audience of the verifiable presentation, i.e. the identifier of the entity that has requested the verifiable presentation and to the respective transaction identified by the nonce sent in the authorization request. The verifier of the verifiable presentation shall verify this binding to the values of client-id and nonce value it had used in the authorization request. If the nonce value is not correct, the response with the VIP shall be rejected.

The verification of the VIP is done using information from the trust bundle repository as defined in clause 8.2.3.4. In this registry, the public key material needed to authenticate credentials from a particular trust domain is stored, i.e. the public key material of all the verifiable credential issuers, the ID agents and ID proxy.

In a distributed and zero-trust environment, there is a need to enable communication across boundaries, and to allow a workload in one trust domain to securely authenticate and verify the VIP of a workload in a foreign trust domain.

For this aim, a federation of trust domains is defined, each trust domain using its own trust bundle repository. The federation consists of the exchange of trust bundles between trust domains. This is defined in clause 8.3.

8.2.3.4 Trust Bundles

8.2.3.4.1 Introduction

The concept of the trust domain is introduced in clause 5.1.4. In the Identity Trust Model defined in the present document, the trust domain is the basis by which the Identity is qualified, a root of trust for the identity, and indicates the authority that has issued the Identity. The issuing authority manages the issuance of identities within its respective trust domain. The trust domain is indicated in the identifier as defined in clause 6.2, in the authority part of the URI.

To enable other trust domains to validate the identities of a given trust domain, the cryptographic keys used by the issuing authority of this given trust domain are available in a trust bundle associated to the trust domain. The trust bundle includes all public keys (JWK set) per qualified attestation attributes provider that issues the verifiable credentials included in the PVID or VIP.

Summarizing, the trust domain is an identity namespace, backed by an issuing authority with a set of cryptographic keys (as trust bundle), that serve as anchor for all the identities managed in this specific trust domain.

8.2.3.4.2 Trust Bundle format

8.2.3.4.2.1 Introduction

A way to implement the trust bundle is the use of a JSON-encoded structure which includes an array of JSON Web Key set (JWK set compliant to IETF RFC 7517 [16]) per qualified attestation attribute provider.

A JWK set as defined in IETF RFC 7517 [16] is a data structure that represents a set of JWKs (JSON Web Keys). This format is widely supported and is used for inter-domain federation.

Figure 8.2.3.4.2.1-1 below depicts the structure of the trust bundle.

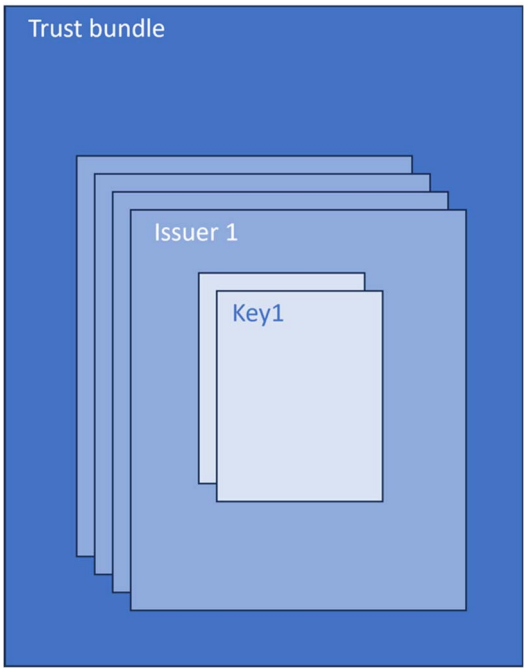


Figure 8.2.3.4.2.1-1: Trust bundle structure

8.2.3.4.2.2 Trust bundle parameters

The trust bundle parameters are listed in table 8.2.3.4.2.2-1 below.

Table 8.2.3.4.2.2-1: Trust bundle parameters

Parameter	Value	Mandatory/Optional	Comments	Defined by
Trust domain	This parameter identifies the trust domain associated with the trust bundle	Mandatory	The value is as defined in clause 6.2. The authority part of the URI	ETSI-NFV
issuers	Array of qualified attestation provider containing the corresponding JWK set	Mandatory		ETSI-NFV

8.2.3.4.2.3 Issuers parameters

The issuers parameters are listed in table 8.2.3.4.2.3-1 below.

Table 8.2.3.4.2.3-1: Issuers parameters

Parameter	Value	Mandatory/Optional	Comments	Defined by
Sequence number (spiffe_sequence)	Monotonically increasing number. the number shall be changed whenever the contents of the bundle are updated	Optional	This parameter is used for update ordering/supersession.	SPIFFE
Refresh Hint (spiffe_refresh_hint)	Integer representing the suggested refresh interval in seconds	Optional	This parameter is used to indicate how often the qualified attestation attribute provider key bundle should be checked for updates, and is representative of the key rotation frequency.	SPIFFE
iss	Identifies the principal that issued the JWT, i.e. the qualified attestation attribute provider	Mandatory	This parameter identifies the qualified attestation attribute provider of the verifiable credential or identifies the authority that sign the PVID or the VIP.	IETF
sub	Identifies the owner of the keys, i.e. the qualified attestation attribute provider	Mandatory	Should be the same as the "iss".	IETF
JWK set		Mandatory	See table 8.2.3.4.2.4-1.	IETF

8.2.3.4.2.4 JWK set parameters

The JWK set parameters are listed in table 8.2.3.4.2.4-1 below.

Table 8.2.3.4.2.4-1: JWK set parameters

Parameter	Value	Mandatory/Optional	Comments	Defined by
Keys	Array of JWK values	Mandatory	Order of the JWK values does not imply an order of preference among them. If key types or uses of a JWK element are unknown the JWK element shall be ignored. If the keys parameter contains empty array (e.g. revocation of all keys) or with no usable keys the workloads shall treat all ID documents from the trust domain as invalid and untrusted.	IETF

8.2.3.4.2.5 JWK elements

A JWK element represents a single cryptographic key, to authenticate a single type of ID document (e.g. PVID or VIP) or to authenticate a verifiable credential.

The JWK parameters are listed in table 8.2.3.4.2.5-1 below.

Table 8.2.3.4.2.5-1: JWK parameters

Parameter	Description	Value	Mandatory/Optional	Comments	Defined by
kty	Key type: Identifies the cryptographic algorithm family used with the key	As defined in clause 4.1 of IETF RFC 7517 [16]	Mandatory	If contains an unknown key type value, the JWK element shall be ignored.	IETF
use	Public key use	pvid; vip; vc	Mandatory	This parameter indicates the type of document that the key is authoritative for and signed: identity document (PVID or VIP) or verifiable credential.	IETF but new values defined by ETSI NFV that give information on the type of document that was signed: PVID, VIP or a VC
kid	This parameter is used to choose among a set of keys within the JWK set	Case sensitive string	Optional		IETF
iat	Identifies the time when this JWK was issued	Number containing NumericDate value per IETF RFC 7519 [13]	Optional		Open ID federation 1.0
exp	Identifies the expiration time on or after which the JWK shall not be accepted for processing	Number containing NumericDate value per IETF RFC 7519 [13]	Optional		Open ID federation 1.0

8.3 Validating Trust between Multiple Domains

8.3.1 Introduction

Existence of multiple trust domains and their distinct separation for security reasons is a fundamental NFV deployment aspect and requirement. NFV deployments inherently involve multiple distinct trust domains, such as infrastructure providers, tenants, and operators. To establish trustworthiness and validate trust between multiple domains is correspondingly essential for NFV deployments. Each trust domain acts in its own capacity, under its own authority, and is isolated from systems residing in other trust domains. Therefore, it is necessary to define a mechanism by which a workload can communicate with another workload from a foreign trust domain, allowing it to authenticate its credentials, issued by its own trust domain i.e. a "different" authority, than the authority in the foreign domain and hence allowing workloads from one trust domain to securely authenticate in a foreign trust domain.

The trust bundles as defined in clause 8.2.3.4 containing the public key materials needed to authenticate credentials from a particular trust domain and its use, as in the SPIFFE federation document [17], can be used to federate trust between different trust domains. To achieve a federation of trust, trust domains expose and consume these endpoints to share bundles between themselves. The operator of a trust domain may introduce or remove public keys of the Qualified Attestation Attributes Provider (or VC trusted issuer) that it trusts within its trust domain in or from the trust bundle repository. These keys are needed for the verification of the verifiable credentials in the PVID and VIP. The public keys of the ID agents and workloads are also included in the trust bundle repository during the process of PVID provisioning as described in clause 8.2.2.3.3 and are used for the verification of the signature of the PVID and VIP.

The semantics of the trust bundle endpoint are similar to the jwks_uri mechanism defined in the OpenID4VP [6].

specification, containing one or more public cryptographic keys used by a trust domain.

To retrieve trust bundle information from the trust bundle repository, the ID Agent maybe configured with the URL of the bundle endpoint, the endpoint profile and the trust domain name associated with the trust bundle endpoint (trust bundle repository). For the communication itself standard TLS-protected HTTP (i.e. HTTPS) is utilized when communicating with the trust bundle repository. A trust domain can keep a local copy of the foreign trust domain's bundle and poll the bundle endpoint periodically for updates. The corresponding Life-Cycle diagram for federating trust between different trust domains is shown in figure 8.3.1-1.

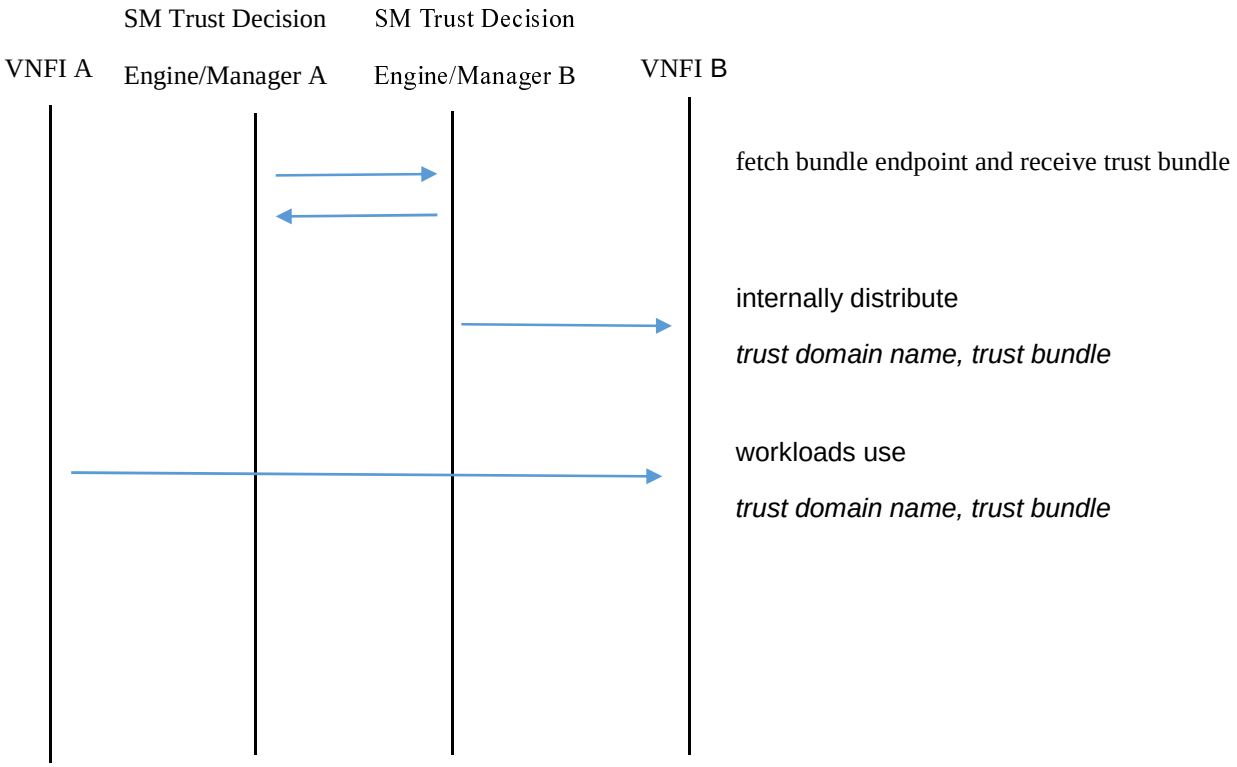


Figure 8.3.1-1: Federation of trust between different domains

8.3.2 Trust establishment process between workloads of different trust domains

The present clause describes the establishment of trust between workloads of different trust domains by use of a trust bundle repository.

For establishing trust between entities of different trust domains, i.e. at the beginning, a zero-trust environment, it is important that each workload or microservice has a PVID (Primary Verifiable Identity Document as defined in clause 8.2.3.2) that was attested, as in a first phase, identity management will be used for mutual authentication and setting up secure communication. This authentication is described in step 5 of clause 8.2.2.3.1 (figure 8.2.2.3.5-1).

Means each microservice has, attested by a Qualified Attestation Attributes Provider of its trust domain, an attested verifiable identity document (PVID and VIP) stored in its ID proxy with the public key of the Qualified Attestation Attributes Provider retrieved in the trust bundle repository.

The PVID is used for authentication between the two microservices of the different trust domains. A handshake is launched, and each micro-service requests the PVID of the other. They exchange their PVID and they retrieve the public keys as used by the Qualified Attestation Attributes Provider and the ID Agent in the trust bundle repository. They are then able to verify the PVID of the other and check the attestation result that proves that the micro-service is really what it claims to be.

They can then establish a secure channel with their key pairs. This is a classic authentication with certificates that include the attestation result.

This authentication is a first level of trust.

The VIP is used for a fine-grain authorization between two microservices of different trust domains.

Each of the trust domains the microservices belong to may have its own security policy concerning access. Hence the micro service requesting access to sensitive data of the other workload may need to proof that it complies to the security policy of the workload it wants to access. The workload that shall be accessed, requests the proof of certain aspects according to its security policy (i.e. location and certificates). It is named in the following: the requesting workload. The requested microservice requests the corresponding proofs (location, certificates), which are stored in its ID proxy. Then it requests its certification label from a certification registrar signed by the certification authority and stores the result in the ID proxy

Then the ID proxy generates the VIP (Verifiable ID Presentation) containing all these proofs and sends this to the requesting workload. The requesting workload checks all the proofs using the corresponding public keys of the issuer of proof in the trust bundle repository of the requested workload. It is then able to verify that these proofs comply with its security policies. After a successful security policy evaluation, an access token that will be used to access the sensitive resources is sent back to the requesting microservice as trustworthiness was established and the corresponding security policy is fulfilled.

This fine-grain authorization process is used to obtain a high-level of trust in a microservice.

Annex A (informative): Hash Constraints

The main reason to consider hashing efficiency is the security need to hide actual IDs and perform network management as much as possible on hashes of real IDs. SHA-2-256 is faster on fast-disappearing 32-bit hardware, due to the 32-bit internal state chunks, while SHA-2-512 is more efficient on modern 64-bit hardware, for any length input. Therefore, 512 bits is a sweet spot. The question now becomes, for how long?

Gartner [i.9] predicts twenty billion Internet of Things (IoT) devices will be in use by 2020. An identifier size of 35 bits (2^35 = 34 359 738 368) will cover that. It can be expected that the number of VNFs to be at least two orders of magnitude below this, therefore 29 bits should suffice for the year 2020. In conclusion, an INSTANCE ID size of 128 bits (a space of 3,4 x 1 038) should be sufficient for a good long time to cover the space.

The next requirement to consider is collision resistance, and pre-image resistance. In annex A.1 of NIST FIPS PUB 202 [i.10] NIST publishes a table that characterizes the security strength of various hashing functions.

Collision resistance is easy to define: it is the probability that there exist any two objects that have the same hash. Pre-image collisions are a special type of collision, one in which an object is chosen, then another object is sought such that it has the same hash as the chosen one. It is intuitive that the second proposition is harder, or probabilistically less likely. As expected, the table follows this pattern.

Security strength again, according to NIST SP800-90A [i.11], is "a number associated with the amount of work (that is, the number of operations of some sort) that is required to break a cryptographic algorithm or system in some way". If the security strength associated with an algorithm or system is S bits, then it is expected that (roughly) 2S basic operations are required to break it.

A design choice is made therefore to achieve a minimum of 128 bits of collision resistance, and 256 bits of pre-image resistance. SHA-384 satisfies both, with 192 bits of collision resistance and 384 bits of pre-image resistance. The only reason not to use 512, is to use the remaining 128 bits (512 - 384) for the TYPE part of the ID. This results in a total size of 512 bits, which, as stated before, is perfectly fit to hash on the 64-bit machines of today.

Annex B (informative): Change history

Date	Version	Information about changes
05/12/2017	0.0.1	First draft based on NFVSEC(17)000166r1
17/05/2018	0.0.1a	Output of drafting session at NFV#22 / NFV SEC#124 F2F
20/09/2018	0.0.2	Output to NFV#23 / NFV SEC#131 F2F
22/02/2019	0.0.3	Output of NFV#25 / NFV SEC#141 Beijing meeting
23/05/2019	0.0.4	Output of NFV#26 / NFV SEC#147 Sophia Antipolis meeting
24/11/2019	0.0.5	Output of NFV#27 / NFV SEC#152 Paris France meeting and clean-up-of references.
10/06/2021	0.0.6	Output of NFVSEC#188 and contribution NFVSEC(21)000048_SEC020v005_Review
10/03/2023	0.0.7	Included comments from Scott Cadzow and members of BT
05/07/2023	0.0.8	NFVSEC(23)000123 added to section 4.1 NFVSEC(23)000130 added to section 4.2 NFV(23)000106r1 making NFV SEC020 release independent
01/10/2023	0.0.9	NFVSEC(23)000190r1 added to section 4.1 and 4.2 NFVSEC(23)000171 change to section 4.2 and added to section 4.3
19/11/2023	0.0.10	Incorporating contributions NFVSEC(23)000199r2_SEC020_section_6_1 NFVSEC(23)000200r1_SEC020_section_6_2 NFVSEC(23)000201r4_SEC020_section_6_3
30/04/2024	0.0.11	Incorporating contributions: NFVSEC(24)000038 NFVSEC(24)000041r1
21/06/2024	0.0.12	Incorporating contributions: NFVSEC(24)000096r2 NFVSEC(24)000097r1 NFVSEC(24)000098r1
08/10/2024	0.0.13	Implementation of the following contributions accepted during the SEC#270 NFVSEC(24)000160_SEC020_section_8_2_1_clarification NFVSEC(24)000161_SEC020_section_8_Identity_trust_model_architecture NFVSEC(24)000162_SEC020_section_8_Identity_trust_model_architecture_entities NFVSEC(24)000163_SEC020_section_8_Identity_trust_model_architecture_high_leve NFVSEC(24)000164_SEC020_section_8_PVID_provisioning_for_ID_agent_flow
11/03/2025	0.0.14	Implementation of the following contributions accepted during the SEC#283 NFVSEC(25)000024r1_SEC020_section_8_PVID_provisioning_for_Workload NFVSEC(25)000026r1_SEC020_section_8_issuing_Identity_credential_in_ID_proxy NFVSEC(25)000027_SEC020_section_8_interaction_with_third_party
22/05/2025	0.0.15	Implementation of the following contributions accepted during the SEC#288 NFVSEC(25)000062r1_SEC020_section_8_VIP
05/06/2025	0.0.16	Implementation of the following contributions accepted during the SEC#289 - NFVSEC(25)000080_SEC020_section_8_Trust_Bundle setting the "issuers" parameter in the trust bundle parameters table as mandatory parameter as agreed during the SEC#289 meeting. Numbering of the tables has been added: - NFVSEC(25)000078_SEC020_section_8_3_Validating_Trust_between_Multiple_Domains as agreed during the SEC#289 meeting, Figure 8.3.1-1 updated with more NFV / Security Manager specific naming (e.g. workload = VNFI, Control Plane = SM Trust Decision Engine/Manager) Adding the link to SPIFFE federation in normative references
01/08/2025	0.0.17	Implementation of contribution NFVSEC(25)000091
23/10/2025	0.0.18	Implementation of contribution:NFVSEC(25)000131r1_NFVSEC(25)000134r1_SEC020_section_6_5_and_section_8
06/11/2025	0.0.19	Implementation of contribution: NFVSEC(25)000147_SEC020_section_7_8
12/11/2025	0.0.20	Transferred draft into new ETSI GS skeleton format
18/11/2025	0.0.21	Implementation of contribution: NFVSEC(25)000157_SEC020_clean_up

History

Version	Date	Status
V1.1.1	April 2026	Publication