ETSI GR NFV-SEC 018 V1.1.1 (2019-11)

Network Functions Virtualisation (NFV);
Security;
Report on NFV Remote Attestation Architecture

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
DGR/NFV-SEC018

Keywords
NFV; security: trust services

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 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, 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.

Foreword

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

Modal verbs terminology

In the present document "**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 identifies and studies Remote Attestation architectures applicable to NFV systems, including the definition of attestation scope, stakeholders, interfaces and protocols required to support them. Additionally the present document identifies and discusses functional and non-functional capabilities to be supported in an NFV system and provides a set of recommendations.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

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 are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GR NFV-SEC 007: "Network Functions Virtualisation (NFV); Trust; Report on Attestation Technologies and Practices for Secure Deployments".

[i.2] ETSI GS NFV-IFA 026: "Network Functions Virtualisation (NFV); Management and Orchestration; Architecture enhancement for Security Management Specification".

[i.3] ETSI GS NFV-REL 005: "Network Functions Virtualisation (NFV); Accountability; Report on Quality Accountability Framework".

3 Definition of terms, symbols and abbreviations

3.1 Terms

Void.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ACL Access Control List
AR Attestation Result
AS Attestation Server
BCA Blockchain of Certificate Authority
CPU Central Process Unit

CRTM	Core Root of Trust for Measurement
CSC	Cloud Service Customer
CSCA	Cloud Service Customer A
CSCB	Cloud Service Customer B
CSP	Cloud Service Provider
CSU	Cloud Service User
DLT	Distributed Ledger Technology
EM	Element Management
EMS	Element Management System
FC	Functional Component
GUID	Globally Unique IDentifier
HMEE	Hardware-Mediated Execution Enclave
HSM	Hardware Security Module
HW	Hardware
IAIS	Infrastructure Attestation Information Service
II	Second
LCP	Launch Control Policies
LoA	Level of Assurance
MAC	Message Authentication Code
MANO	MANagement and Orchestration
NFVI	Network Function Virtualisation Infrastructure
NP	Network Provider
PDLT	Permissioned Distributed Ledger Technology
PKI	Public Key Infrastructure
RA	Remote Attestation
RAIC	Remote Attestation Information Customer
RAIP	Remote Attestation Information Provider
RAS	Remote Attestation Server
RATP	Remote Attestation Trusted Party
RIAP	RA Information Provider
RoT	Root of Trust
RTM	Root of Trust for Measurement
RTR	Root of Trust for Reporting
RTS	Root of Trust for Storage
SE	Security Environment
SEMS	Security EMS
SM	Security Module
SSR	System State Report
SuE	System under Evaluation
TCB	Trusted Computing Base
TEE	Trusted Execution Environment
TTL	Time to Live
TTP	Trusted Third Party
UUID	Universally Unique Identifier
VM	Virtual Machine
VMI	Virtual Machine Introspection
VNF	Virtual Network Function
VNFCI	VNF Component Instance
VNFI	Virtual Network Function Instantiation
vRoT	virtual Root of Trust

4 Motivation and Problem Description

4.1 Overview

Today's deployed systems face a huge amount of threats that have the capability to compromise them partly or fully and, in many cases, involves that an attacker modifies a system such that malicious software is executed. Execution of code that was not intended to be executed on the system is expected to be detectable. One defensive measure that addresses the malicious software execution is Remote Attestation (RA).

Remote in this context is defined as the attestation taking place outside of the immediate trusted element by a Trusted Third Party (TTP). In contrast, for local attestation, a specific hardware module might use Launch Control Policies (LCP) which are capable of halting boot (or some other action) on that device if the policies are not satisfied by the gathered measurements.

Specifically, RA is a well-known concept that is used to determine the trustworthiness of systems. Hence it might be used to facilitate the detection of unintended/malicious software. The overall process during RA is:

1) accumulation of information on a system A;

2) reporting of the accumulated information to a different system B; and

3) evaluation on basis of a comparison between the reported and well-known reference information.

Accordingly, the evaluation result is either system A is in a trusted or an untrusted state.

A TTP, i.e. the verifier, in the context of RA is the entity that holds known good values, acquires measurement reports of system state and makes the decision whether a given system, element, component etc. is trusted. What trusted is not defined means other than stating that the given system meets some a priori criteria, for example, but not limited to, that the system only loaded and executed software that is well known. How information that an element is trusted is not defined is interpreted by other elements either. Consequently, RA facilitates to assess whether a remote service is provided by a trustworthy environment. Such trust establishment is the fundamental step prior to the remote entity engaging in further interaction such as consuming services or to deliver sensitive/secret data to the remote service. For example, tenants might use RA to assess if the overall infrastructure (NFVI) is trustworthy, datacenters might use RA to assess trustworthiness of subsystems they use, and management entities might use RA to assess the trustworthiness of individual infrastructural components. Furthermore, tenants might offer RA services to its remote users and thus offer an overall assurance assessment of the end service or a service for proving compliance. For example, to demonstrate that data is stored at a correct geographical location. Hence, there are numerous use-cases and scenarios that might be considered where attestation is a fundamental step of creating an overall trustworthy system.

A trustworthy element is the entity which has a component that provides a unique identifier, certification (e.g. through cryptographic signing) and which is able to store measurements and data about the state of that element (including related sub-elements or dependent elements if necessary) in a tamperproof and verifiable form. For example, the TPM2.0 quoting mechanism using the TPMS_ATTEST data structure is an example of this.

4.2 Problems and Challenges

However, the classical RA concept and architecture was designed on basis of individual systems with clearly defined roles and assumptions. Thus, this traditional approach is not directly applicable in modern system architectures that rely on virtualisation, since it does not consider such systems from an architectural point of view. One approach that could simply overcome these problems would be to ignore the virtualisation altogether and treat each system individually. In this case, however, important information that might not be established at a later time easily gets lost and, thus, would not result in an acceptable evaluation result. Apart from the virtualisation, the NFV architecture introduces different other characteristics and constraints RA needs to adhere, for instance different roles, components, responsibilities and even visibility within the deployed systems. For these reasons, it is necessary to adopt all of the NFV related characteristics and constraints in order to derive a meaningful and applicable RA solution for NFV.

More specifically, when speaking about the attestation procedures, there are two important aspects to consider. One is the attestation protocol itself and the other is, how the information that the appraiser gets via the attestation protocol is transformed or interpreted into a statement of being trustworthy. It might be the case that this interpretation is simple but one might easily define use cases where the task of interpreting attested data is hugely complex. In any case, what is important here is that the appraiser has the knowledge how to interpret the attested data. Such knowledge is easier to arrange when the attester and appraiser are close and are, for example, aware of their environment. This leads also to the question where the appraisal takes place. One extreme is that the one that wants the information about the trustworthiness also performs the appraisal. Another extreme is that the appraisal comes from an a priori Trusted Third Party (TTP). In the latter case the one that wants to establish trustworthiness could only get a binary decision from the TTP: trusted vs not-trusted. Alternatively more complex information is provides such as levels of assurance.

The attestation protocol consists of the messages and procedures through which the attester interacts with the appraiser. The details of the protocol are coupled to the technical environment of the attester. On the other hand the purpose of the attestation protocol is to securely deliver attestation information to the appraiser, whereas the information is securely gathered in the attester's environment. This secure acquisition is necessary, so the appraiser is able to deliver a statement on the trustworthiness-state.

When using a TTP Attestation Server (AS), the semantics what trustworthiness means is hidden, and is coupled to an agreement by which one is allowed to talk to the AS, but there is typically no explicit data transferred, e.g. data that would detail what trustworthiness means for an Openstack Controller. Again, where the attestation server is located is not defined. Encapsulating the trustworthiness allows for a simpler way to adopt different implementation technologies, but it might also cause that for certain technologies fully leveraging the features of the remote attestation functions in that technology becomes limited. These consideration are the main reason why this present document distinguishes between the high level use-cases in clause 4.5.

One example of RA is that one in Openstack known as trusted compute pools. Here the launch of a VM only occurs when Openstack Controller gets the confirmation that the compute node is trustworthy from a so-called Attestation Server (AS) which performs the appraisal of the trusted compute pool that Openstack Controller wants to use for the VM.

4.3 NFV Attestation Scope

The overall NFV attestation scope comprises multiple related systems and components. From a simplified top-down view, a NFV provides a particular service to a customer. Typically, the basis of this service is provided by software running inside virtualised systems that, in turn, are instantiated on top of hypervisors. This means, ideally, the overall attestation scope comprises all of the corresponding systems and components involved, i.e. one or many hypervisors, instantiating one or multiple VMs that execute one or many different application processes, schematically depicted in Figure 4.3-1.

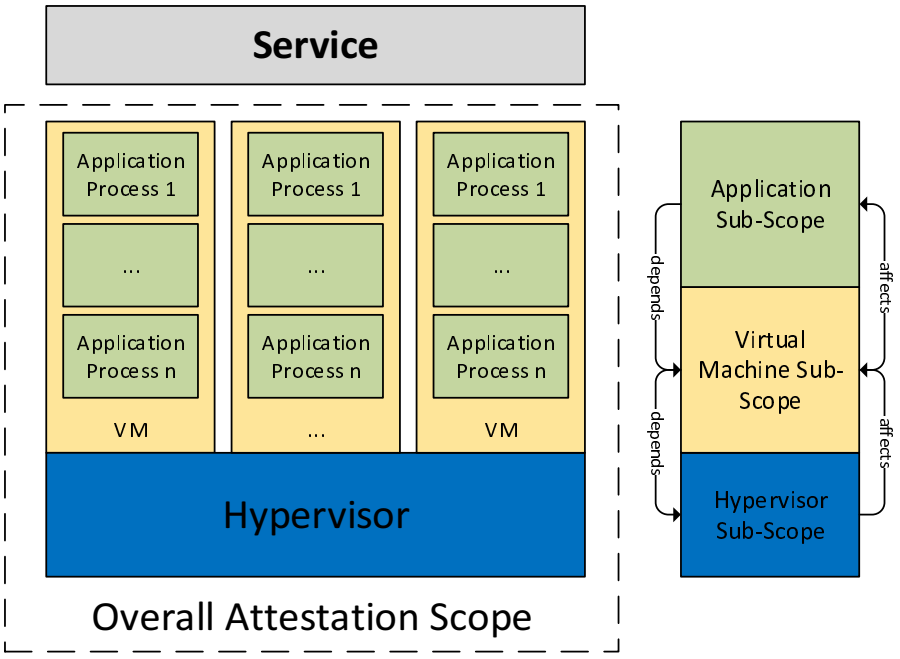

Figure 4.3-1: NFV Overall Attestation Scope

Specifically in NFV, the overall attestation scope is a composition of the described individual systems and components under the control of different roles and organizations with presumably limited visibility. Hence, the NFV attestation scope needs to be divided into multiple sub-scopes that aligns with the actual system architecture and, in addition to that, consider the mentioned additional roles, architectural components and characteristics introduced by the NFV high level architecture. These specifics are to be analysed and discussed in clause 5.3 in more detail.

In addition to the aforementioned aspects, the overall attestation scope depends also on the exact use case and, most importantly, on the agreed Level-of-Assurance (LoA) [i.1]. In particular, the LoAs define the sets of systems and components to be considered during attestation procedures and, thus, facilitate the determination of the overall attestation scope. An overview of the defined LoAs in relation to the attestation scope is depicted in Table 4.3-1.

Table 4.3-1

LoA Level	LoA defined set of attested Systems and Components	Type	Affected Attestation Sub-Scope(s)	Attestation
0	all components	None	None	None
1	Hardware and Virtualisation Platform	Loadtime	Hypervisor + Virtual Machine	Local
2	Hardware and Virtualisation Platform	Loadtime	Hypervisor + Virtual Machine	Remote
3	VNF Software Packages	Loadtime	Virtual Machine + Application	Local
4	VNF Software Packages	Loadtime	Virtual Machine + Application	Remote
5a	Hardware and Virtualisation Platform	Runtime	Hypervisor + Virtual Machine	Remote
5b	VNF Software Packages	Runtime	Virtual Machine + Application	Remote

Accordingly, the relevant LoA Levels that relate to the present document are LoA 2, 4, 5a and 5b. Important to note regarding the defined LoAs is that the corresponding attestation scope does not include the hierarchical lower layer implicitly. This means, LoA 4 does not influence the attestation information of LoA 2, although both levels share the Virtual Machine Sub-Scope. Thus, the overall attestation scope for LoA 2 and 5a relates to the Hypervisor and Virtual Machine sub-scopes and for LoA 4 and 5b the overall attestation scope relates to (1) Hypervisor and Virtual Machine sub-scopes and (2) Virtual Machine and Application sub-scopes. In the latter case (i.e. LoA 4 and 5b), two separate but interdependent RA procedures need to be applied to satisfy the requirements defined by LoA.

To conclude, the NFV RA scope depends on multiple distinct systems and components. These systems and components are under control of different organizations with different visibility. This defines natural boundaries between the involved systems and components that are represented by introduced sub-scopes. Moreover, LoA are used to determine the overall RA scope within NFV. Depending on the targeted LoA level, the overall RA scope includes multiple RA procedures that also relate to limited visibility within the system.

Regarding the present document, the targeted overall RA scope considers Hypervisor, Virtual Machine and Application sub-scopes, to satisfy the highest LoA (i.e. 4, 5a and 5b) defined. Consequently, the document discusses all RA relevant systems and components available within NFV and consider them in the design for the RA Architecture appropriately.

4.4 Stakeholders

The stakeholders relevant for RA are derived by the corresponding roles defined in ETSI GS NFV-REL 005 [i.3]. In particular, these roles are: Cloud Service User (CSU), Cloud Service Customer (CSC) and Cloud Service Provider (CSP). The CSP role is further subdivided into NFV Infrastructure (CSP: NFVI) and NFV Management and Orchestration (CSP: MANO) that might be the same or different organizations. The additional CSP roles, i.e. Functional Component (CSP: FC) and Network Provider (CSP: NP) are not considered in the present document. It is assumed that these roles are implicitly provided or not part of the NFV itself.

Accordingly, the stakeholders are identified as representatives of the mentioned roles within RA. Since NFV follows a hierarchical approach based on customer-provider relationships, each stakeholder has a particular interest in the information provided by RA. But, in turn, the information required to provide the RA information is not visible/available for all stakeholders. In addition, the hierarchical model also implies that there is no direct relationship that necessarily extends beyond a certain role boundary. For example, a CSU typically has no business relationship with the CSP and vice versa, so it might not be possible to exchange any RA information directly between them. As a result, two RA Information related roles are introduced that distinguish between an RA Information Provider (RAIP) and Customer/End-user (RAIC). More specifically, the RAIC is interested in the information provided by RIAP, but does not have the capability to acquire them; the information necessary might not be available, for instance, due to limited visibility. Accordingly, the RAIP is responsible to accumulate and provide the relevant RA information instead.

Table 4.4-1

Stakeholder	RAIC…	RAIP…
CSU	of CSC	n/a
CSC	of CSP	for CSU
CSP	n/a	for CSC

Consequently, the different stakeholders can only act as depicted in Table 4.4-1:

- CSU is RAIC of CSC
- CSC is RAIP for CSU
- CSC is RAIC of CSP
- CSP is RAIP for CSC

NOTE 1: A stakeholder with the capability of RAIP might implicitly be a RAIC of itself. For example, the CSP: NFVI could be in the role of the actual RAIP and CSP: MANO in the role of RAIC in this case.

Still, depending on the particular use-case and the exact RA model employed, the RAIP provides the accumulated information only, an already RA-evaluated result or both. Consequently, this means the RAIC either needs to conduct the RA evaluation or rely on the evaluation result provided.

Regarding the exchange of information between stakeholders, there is typically no unbound exchange of information between roles without a direct relationship in a strict hierarchical model. However, a relaxed hierarchical model could be defined that facilitates this exchange of information. Within this relaxed model, all parties provide the necessary information for the RA evaluation process without considering the hierarchical relationships altogether.

Since a completely unconstrained model might not be applicable in certain NFV RA use-cases, but a strict model would impose too many restrictions, a RA Trusted-Party (RATP) stakeholder, which might either be one of the involved parties or an additional independent party, is introduced as an alternative. The RATP is generally trusted by all RA-involved parties and has access to all information relevant to conduct an RA evaluation. Still, this does not involve the accumulation of RA information, because it is not assumed the RATP can freely access all involved systems and components on its own and acquire the information by itself. As a result, the other stakeholders do accumulate the necessary RA information by themselves, but are expected to report them to the RATP. In turn, the RATP acts as a central receiver of all accumulated RA information and conducts the RA evaluation on basis of this information. In this model, the following role-based relationships apply:

- CSP and CSC are RAIP and reports to RATP
- CSP, CSC and CSU are RAIC of RATP (evaluation result)

NOTE 2: In this model, a RAIP only accumulates the RA-information on the relevant systems and components. It does not conduct the RA-evaluation on its own. This means, in this case, a RAIP might not act as a RAIC for itself. Thus, for instance, if CSP: MANO is interested in CSP: NFVI information, it needs to ask the RATP unless it has the capability to do an RA-evaluation on its own. This is not defined or expected within this model and not considered.

4.5 Use Cases

4.5.1 Use Cases Overview

The RA use cases rely most of all on the information that is available to the involved stakeholders. In general there is a distinction between the RA information that is accumulated and the RA information necessary to check these measurement information during evaluation. As described in the previous clauses 4.3 and 4.4, the visibility of measurement information is limited to the stakeholders that control the corresponding system. But, as mentioned, this visibility only related to the actual procedure of the measurement. Hence, this does not affect the reporting of this information and neither limit the possibilities that a stakeholder might offer this information to another stakeholder by providing an interface to acquire this measurement information.

Considering that the access to the accumulated information is provided by an interface that is able to restrict the access to the information, the strict and relaxed model are equal from an architectural point of view. Similarly, the access permissions in the model involving a RATP can be restricted. For this reason, the measurement accumulation and reporting does not limit or affect the RA architecture, besides the definition of the particular interface that provides this information. As a result, the RAIP does not affect the RA Architecture and thus, plays only a minor role during the definition of use-cases. Instead, the RAIC is used to distinguish between the use-cases.

Under the pre-condition that the RA measurement information is available to RAIC, the RA Architecture needs to distinguish between a RAIC that has only limited knowledge or full knowledge during the evaluation procedure. In case this knowledge is limited, the corresponding stakeholder can only evaluate the RA measurement information partly. For instance, if the CSP has only RA evaluation information for the Hypervisor and Virtual Machine attestation scope, it can only determine the reliability of these particular scope, even if additional measurement information would be available to him. Similarly, if the CSC has only access to Virtual Machine and Application scope evaluation information, it cannot determine the reliability of the Hypervisor and Virtual Machine scope, even if the RA measurement information is available. This means, within this model, multiple distinct attestation procedures might be required, depending on the LoA that has to be satisfied.

4.5.2 Transitive Model Use Case

Multiple Independent Logical RA Servers in a single trust domain.

Assumptions:

1) It is assumed that the model satisfies LoA 4 and 5b.

Pre-Conditions:

1) Relevant RA measurement information is available. Access permissions policies are enforced by the Security Controller or available to all RAIC.

2) Role-specific RA evaluation information is available to RAIC, but limited to system and component managed directly by the corresponding RAIC.

Use-case 1 is defined as follows:

There are multiple RAICs, repressed by different stakeholders with limited RA evaluation information. Each RAIC can only evaluate the systems and components it manages and operates. In order to satisfy LoA 4 and 5b requirements, a RAIC might share its RA evaluation result with other RAICs or provide the RA evaluation result to a different system that is eligible to receive this information. In case of RAIC 1 sharing its RA evaluation result with another RAIC 2, RAIC 2 inherently trust the RAIC 1 evaluation result and might use it during its own evaluation procedure. In addition to that, a RAIC might share its evaluation results with a third independent system, eligible to receive this information. In any case, the system receiving the RA evaluation results might combine them and derive the overall reliability state based on the provided evaluation results it received. To satisfy LoA 4 and 5b, the RA evaluation results from (1) Hypervisor and Virtual Machine sub-scopes and (2) all relevant Hypervisor maintained Virtual Machines and Application sub-scopes need to be available.

Table 4.5.2-1

	CSP	CSC	CSU	External System
Provides RA measurement information	To CSP and CSC	To CSC and CSU	-	-
Has RA evaluation information	Only CSP	Only CSC	-	-
Provides RA evaluation results	To CSC	To CSC, CSU and external system (might reuse CSP evaluation result)	-	-
Has access to RA Evaluation Results	CSP only	From CSP and CSC	From CSC only (if eligible)	From CSC (if eligible)
Provided Evaluation results are Loa 4 and 5b compliant?	No	Yes	-	-

4.5.3 Central-Model in a single trust domain

Assumptions:

1) It is assumed that the model satisfies LoA 4 and 5b.

Pre-Conditions:

1) Relevant RA measurement information is available to RATP without any restriction.

2) Role-specific RA evaluation information is available to RATP without any restriction.

Use-case 2 is defined as follows:

There are multiple RAIPs and RAICs that represent different stakeholders within the system. In particular, the RATP is introduced as an additional stakeholder that can act as a RAIP and RAIC. RATP has access to all RA measurement and evaluation information from all other parties. Other stakeholders act in the system as RAIPs, in case they are manage and operate attestable systems. Consequently, they implement an interface to offer the RA measurement information to RATP. In addition to that, all stakeholders can adopt the RAIC role and access RA evaluation result from RATP. The access to the information can be restricted to specific information only or allow full access for one or many stakeholders. RATP has all RA evaluation information available and, hence, can conduct a RA evaluation for all other stakeholders. More specifically, only RATP is meant and considered to conduct the RA evaluation in this model. This means, RATP represents a RAIP that provides RA evaluation results only. RATP provides an interface to offer the RA evaluation results to other relevant stakeholders and implements access restriction to the information as necessary.

The LoA 4 and 5b requirement is implicitly fulfilled by RATP, as long as RATP has access the relevant RA measurement and evaluation information. In order to satisfy LoA 4 and 5b, RATP needs to provide the RA evaluation results from (1) Hypervisor and Virtual Machine sub-scopes and (2) all relevant Hypervisor maintained Virtual Machines and Application sub-scopes need to be available.

Table 4.5.3-1

	CSP	**CSC**	**CSU**	**RATP**	**External System**
Provides RA measurement information	To RATP	To RATP	-	-	
Has RA evaluation information	Only CSP	Only CSC	-	CSP and CSC	
Provides RA evaluation results	-	-	-	For CSP, CSC, CSU, External system (access restrictions might be defined)	
Has access to RA Evaluation Results	From RATP (if eligible)	From RATP (if eligible)	From RATP (if eligible)	From RATP (if eligible)	From RATP (if eligible)
Provided Evaluation results are Loa 4 and 5b compliant?	-	-	-	Yes	-

4.6 Use case scenario examples

4.6.1 General

The RA capabilities of the different stakeholders can be put to use to realize trust establishment in different scenarios. In this clause several use cases are described that aim specific trust establishment between the stakeholders:

1) Measurement of VM during launch.
2) Protected VM launch on a trusted NFVI
3) Measurement of VM during launch and while in use
4) Remote attestation of secret storage
5) Secure VM migration between two trusted NFVIs

Note that these use cases are rather building blocks from which complex and functionally richer use cases can be built.

4.6.2 Measurement of VM during launch

A VM is launched on a NFVI platform and the NFVI is instructed to perform measurements on the VM image or immutable parts thereof as process of the launch and retains these measurement and possibly related logs. The RAIC (e.g. tenant, MANO, etc.) has a reference point to the launched VM through which it can contact the VM RAIP instance. At a later point in time the RAIC might request an attestation and decide if the measurements are:

a) reflecting conditions of proper launch; and

b) if the measurement was performed by a trustworthy NFVI.

4.6.3 Protected VM launch on a trusted NFVI

A protected VM is here understood as a VM image that is delivered encrypted and or integrity protected to the NFVI. Only a NFVI platform that is trusted has access to a given key for decrypting the VM. Typically this is achieved by sealing key material. If unsealing succeeds, the trusted platform can do a key exchange to retrieve the key to decrypt the VM. Trust in the NFVI is established by a remote attestation of its TCB instance. The remote attestation can be done via an entity that serves as a Trusted Third Party from the tenant perspective or by the tenant itself. In either case the verifier has established information of NFVI that allows the verifier to assess whether the NFVI platform has performed a trusted boot.

4.6.4 VM measurement during launch and while in use

A VM is launched on a NFVI platform and the NFVI is instructed to perform measurements on the VM image or immutable parts thereof as process of the launch and retains these measurement and possibly related logs. Furthermore, the VM performs additional measurements to be recorded in a dedicated secure storage space for the VM instance. This snapshot measurements relate typically to static files or configuration data. The RAIC (e.g. tenant, MANO, etc.) has a reference point to the launched VM through which it can contact the VM RAIP instance. At any later point in time the tenant RAIC might request an attestation and decide if the measurements are a) reflecting secure launch of the VM b) reflecting runtime integrity of the VM if the snapshot measurement data is allowed (e.g. has a hash value that appears as one of the whitelisted values).

4.6.5 Remote attestation of secret storage

A VM instance needs to store credentials or sensitive data in a secure place, for example, as part of the migration process. To have better control on the storage service, the VM might use a remote storage service for that purpose. Further, the VM might perform a remote attestation of the remote storage prior to any credentials or data being passed to this storage service provider. The remote attestation might be done via an entity that serves as a Trusted Third Party from the tenant perspective or by the tenant itself. In either case the verifier has established information of the remote storage server that allows the verifier to assess whether the remote storage platform has performed a trusted boot.

4.6.6 Secure VM migration between two trusted NFVIs

In case a NFVI needs to interrupt its service it is beneficial for the service the VM's provide that the VMs is able to be moved to another trustworthy NFVI in a secure manner. Based on a tenant provided policy the source NFVI might attest the target NFVI and establish a key used for encryption and or integrity protection in a manner similar to the launch use case 4.5.3. Only a trustworthy target NFVI might recover the VM. If NFVI support live-migration, even secure live-migration might be realized in this manner.

4.7 Challenges and Limitations

One challenge is obtaining a base set of known good values, e.g.: firmware providers and OEMs have possibilities to customize firmware such that it becomes possible that "identical" machines have subtly different configurations. These might even include the compiler version that generated the firmware binary loaded which in turn affects cryptographic hashes/measurements eventually.

5 NFV Remote Attestation Architecture

5.1 RA High Level Architecture

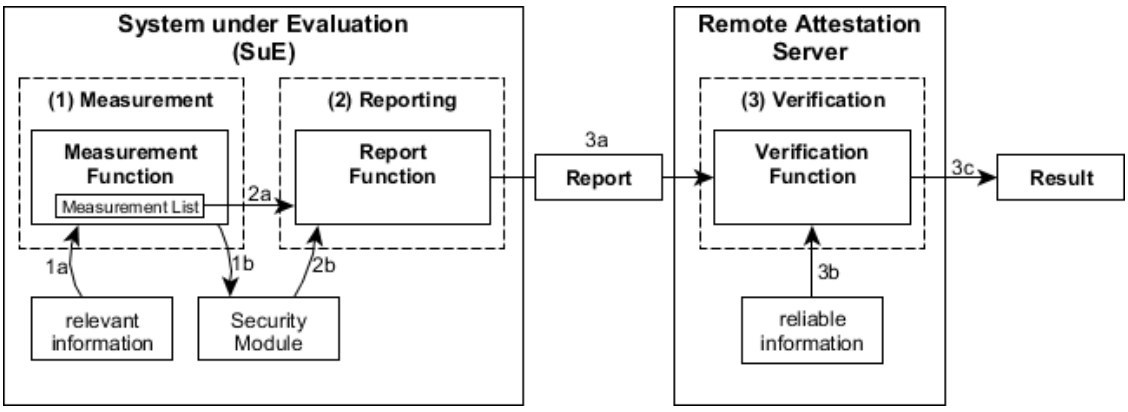

Figure 5.1-1: Remote Attestation High Level Architecture

A high level architecture for remote attestation of non-virtualised environments typically involves two distinct systems and consists of three operations: Measurement, Reporting and Verification.

As illustrated in Figure 5.1-1, there is the System under Evaluation (SuE) which is to be attested and checked whether it is in a trustworthy state or not. For this purpose, a measurement function (1) accumulates relevant information on the SuE (1a), maintains this gathered information in an internal measurement list and protects the maintained measurement list by adding integrity information into a tamper resistant security module (1b).

After relevant information has been accumulated and securely stored within the system, a reporting function (2) uses both, the measurement list (2a) and the security-module-anchored integrity information (2b), to generate a report (3a). The present document is transmitted to the Remote Attestation Server (RAS), the second system, for conducting the actual trustworthiness evaluation of SuE.

The transmission is typically performed on the basis of a remote attestation protocol that involves additional information to assure certain security properties in transit. For instance, integrity, authenticity, confidentiality, freshness and replay-protection properties. For some of these properties it is possible that they are implicitly provided by the security module or they are to be covered explicitly by the protocol itself, depending on the requirements of the use-case.

This evaluation is implemented by a verification function (3) which takes the report (3a) and verifies its contents on the basis of reliable information (3b) which is already present in the RAS. The result (3c) of the verification function is a statement about the trustworthiness of the evaluated SuE. This statement is either that the SuE has been found to be in a trustworthy state or not. The result might then be used by other systems for executing additional procedures like, for instance, enforcing certain policies that trigger remediation procedures in case SuE was found to be untrustworthy. However, specific policy enforcement procedures or other result-related follow-up actions are not in the scope of the present document.

NOTE: Describe the well-known RA Architecture for non-virtualised deployments as a reference architecture the rest of this clause is based upon.

5.2 Architectural Scenarios and Deployment Analysis

5.2.1 Trust at the Service Layer

The trust-state of a provided network service in NFV is determined by the corresponding cross-architectural components that are necessary for its operation. In this context, network service is an abstract definition provided by a software implementation running on a stacked layer of environmental components that are provided by one or multiple VNFs, VMs and hypervisors. Hence, these cross-architectural components involve: (1) the Virtualisation Layer, including the physical and virtualised resources, and (2) the related VNFs that implement the environment and functional capabilities the network service offers.

The virtualisation layer, i.e. NFVI layer, provides a virtualised execution environment for VNFs. For this purpose, the NFVI abstracts available physical resources into virtual resources and provides these virtual resources for the instantiation and execution of VNFs. The building block that offers this kind of virtualisation is commonly known as a hypervisor.

A VNF, in turn, implements the run-time environment and functional capabilities to offer a specific network service. These functional capabilities are typically provided by a software-package. In general, a VNF might implement arbitrary functions that offer an entire network service on its own or that only offers a specific part of a network service. This means, a network service might rely on one single VNF or, in the usual case, on multiple VNFs that work together. In any case, an individual VNF, or more precisely a VNF Instantiation (VNFI) relies on the run-time environment and corresponding software provided by a single building block that is typically referred to as a VM.

In relation to the trust-state of a particular network service the NFV architecture might hence involve different NFVIs, provided by one or multiple VNF Component Instantiations (VNFCIs), whereas each VNFCI might be executed across on or multiple VM's that might also be spread across multiple hypervisors. For instance, as depicted in Figure 5.2.1-1, a network service is provided by two different VNFIs (VNFI 1, VNFI 2) implemented by three different VNFCIs (VNFCI 1, VNFCI 2, VNFCI 3) that are provisioned on three VMs (VM 1-3) on two hypervisors (Hypervisor 1 and Hypervisor 2).

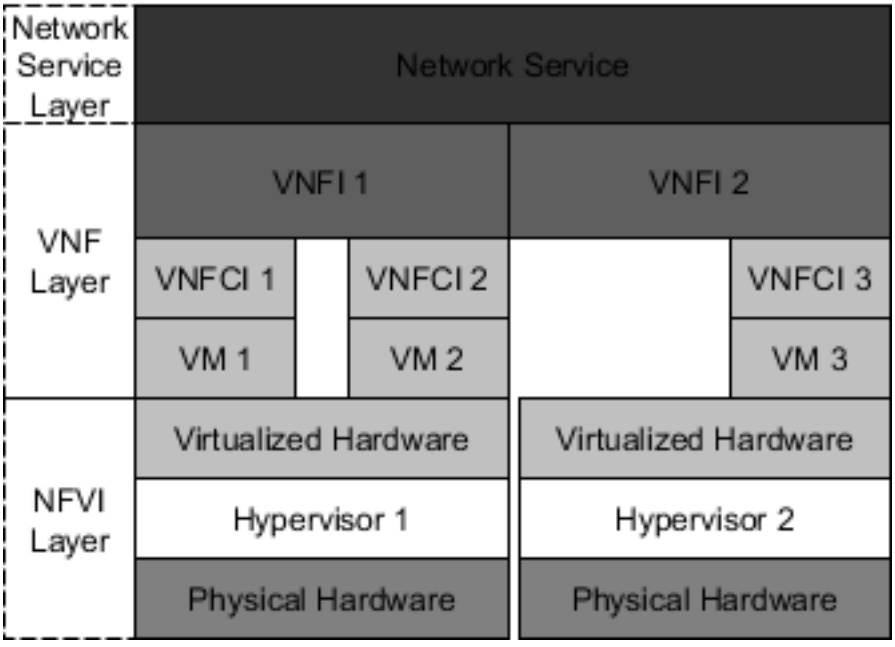

Figure 5.2.1-1: Cross-architectural Composition of NFVI and VNFs for Network Service Provisioning

For the given example this means that determining the trustworthiness of the network service includes all components and systems within its provisioning-chain. The trustworthiness of a network service therefore depends on whether it is provided by systems and components that are trustworthy themselves or not. From a bottom up perspective, this corresponds to the definition of the LoA-levels from the NFVI- to the VNF-layer. However, the defined LoAs do not consider the compositions of multiple related systems and components to derive a statement about the trustworthiness of a network service; assuming that the trustworthiness of the underlying layers is provable, a network service is considered trustworthy if the network service is provided exclusively by a trustworthy and cross-layer hard- and software stack.

For system architectures that depend on virtualisation this means that the trust-state of an upper hierarchical layer depends on the trustworthiness of its corresponding lower layer. Hence, without proving the lower layer's trustworthiness, it is impossible to derive meaningful statements about the trust states of the higher-layers.

In terms of Remote Attestation for NFV the trust-state of the hypervisor-platforms needs to be established first. If a hypervisor-platform is proven to be trustworthy the corresponding NFVI-layer is to be trusted including the physical hardware, the entire software providing the hypervisor capabilities and eventually the virtualised hardware. Next, the provisioned VMs within the VNF-layers are to be attested. Again, this includes the platform of the VM, which relates to the VM's entire software stack and, in addition, the software-package providing the specific VNF. In case that all related VNFIs, their corresponding VNFCIs, VMs and corresponding hypervisors in a service's provisioning-chain are found to be trustworthy, the network service is also trustworthy. The major distinction in this case is that the network service-layer is not explicitly attested during Remote Attestation. Instead, a system which is aware of the provisioning-chain of a network service is able to derive the trust-state of a particular network service on the basis of the Remote Attestation results of the components and systems from the corresponding NFVI- and VNF-layers.

To conclude, the goal of is to establish a chain of trust from the physical hardware up to the network service layer, as depicted in Figure 5.2.1-2.

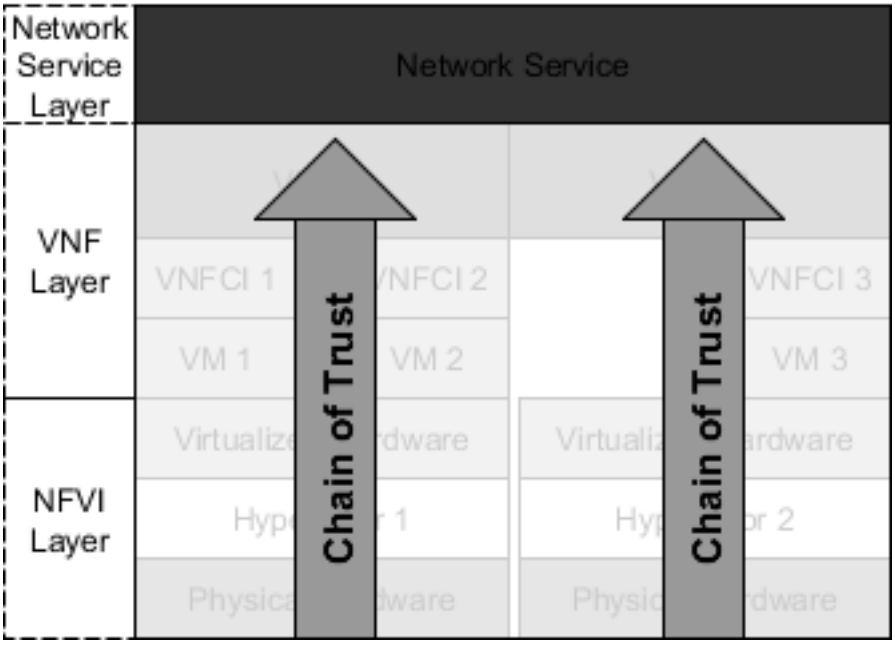

Figure 5.2.1-2: Cross-architectural Chain of Trust Establishment

5.2.2 Considerations for Trust Assurance in NFV

5.2.2.0 Introduction

One major challenge in virtualised architectures and especially in NFV is to prove that a particular VM runs on top of a specific hypervisor. More specifically, it is necessary to assure that a trusted VM/VNFCI is executed on a particular trusted hypervisor, whereas the hypervisor's trust state relies on an attestation that considers the entire corresponding hard and software stack. More precisely, this includes all hardware chips, firmware, OS- and hypervisor-components that are relevant for the hypervisor's trust state determination.

If it is not possible to establish a correlation between VM/VNFCI and hypervisor, an attacker is able to make use of a trusted VM that runs on top of an untrusted hypervisor and it would be impossible to detect any interference made by the malicious hypervisor, e.g. intercepting communication, replacing strong or using weak cryptographic keys, etc. Similarly, trustworthiness in the service-layer might only be established if there is a mechanism to determine that only trusted VNFs, w.r.t trusted VM's, are running on specific trusted hypervisors that are part of the service-provisioning-chain.

5.2.2.1 Security Properties at the Hypervisor Layer

A solution to this challenge is to define mechanisms that allow for establishing these cross-layer-bindings. Within the NFVI layer hardware-based Security Modules (SMs), e.g. TPM's and HSM's, are typically used to form hardware-based Roots of Trust (RoTs). Among other properties and mechanisms, the RoTs of the platform to be attested need to offer the following properties related to Remote Attestation to the NFVI-layer:

1) a unique identity that is usually based on a cryptographic key, w.r.t Root of Trust for Reporting (RTR);

2) a tamper-resistant storage for accumulated measurements, w.r.t. Root of Trust for Storage (RTS);

3) a mechanism to bind the accumulated measurements to the unique identity;

4) protection against disclosure of secrets and cryptographic key material.

In addition, the platform needs to offer a Root of Trust for Measurement (RTM) that leverages the RTS and RTR during measurement and reporting phases of the Remote Attestation procedure. This is typically provided by some a tamper-proof hardware-component initiating the boot process. This component is referred to as the Core Root of Trust for Measurement (CRTM).

Platforms that provide and leverage these RoTs and mechanisms correctly are attestable during Remote Attestation with different levels of confidence in the attestation result. If the RoTs are rooted in hardware, the confidence in the attestation result is very high because the hardware-rooted RoTs are typically well-protected against tampering and disclosure. For instance the cryptographic keys that provide the identity are burned into protected by a discrete chip and their disclosure is protected avoided by using them only inside the SM itself.

At the hypervisor level, the RoTs are typically implemented by one or more discrete hardware-based chips whereas each hypervisor generally contains one or more discrete chips that provide the necessary RoTs and mechanisms. Considering these chips as one unit, each hypervisor implements them once. As a result, the identity of a hypervisor is clearly assignable to the implemented RTR, i.e. the RTR defines the identity of the hypervisor. If the other described properties and mechanisms are fulfilled by the implemented RoTs, this results in a very high confidence in the information measured and therefore also a very high conclusiveness of the attestation. These properties are:

- Tamper-resistance of the discrete chips
- Confidentiality of secrets and cryptographic keys involved
- Establishment of a verifiable and unambiguous chain of trust from boot to operation

5.2.2.2 Security Properties at the VNF Layer

For the VNF-layer, there are usually no discrete hardware-based RoTs available that offer equivalent assurance with regard to their provided security properties. Instead, the mentioned RoTs are typically made available through software implementations and are offered to the VNFs by the hypervisor as a virtual resource. From a functional point of view, the virtual RoTs (vRoTs) are equivalent to their hardware counterparts; however, as expected, they do not provide the same level of trust assurance, unless similar properties are offered by leveraging hardware-backed or hardware-rooted technologies during their instantiation and operation.

If there are no safeguarding-mechanisms available that protect the instantiation and operation of VNFs against malicious interference by the hypervisor, a meaningful attestation of a VNF is only possible if the hypervisor itself is in a trustworthy state. Although there are mechanisms available that offer a certain kind of protection in this regard, the operation of virtualised entities in an assumed hostile environment goes beyond the scope of the present document. For this reason, the rest of the present document assumes that the vRoTs themselves are resistant against interference by the hypervisor. This means, the hypervisor is not capable to modify, replace and disable the vRoTs altogether or intercept and tamper communication from the virtualised entity to the vRoTs as long as the hypervisor remains in a trusted state.

Under these assumptions, the vRoTs are still susceptible to different internal and external threats. In particular, the tamper resistance for the RTS and the identity of the VNFs are important in this regard.

5.2.2.3 vRTS Tamper Resistance

The vRTS needs to be tamper resistant to avoid that invalid information is reported during a Remote Attestation process. From the perspective of a VNF, it needs to be impossible to reset or remove any platform integrity measurements from the RTS, once they were taken. For this reason, it is recommended that the vRTS do not implement functions that allow removal of measurements or resetting the vRTS. This behaviour is similar to the implementation inside the discrete hardware chips, which also do not implement these functions.

From the perspective of the hypervisor, the storage is a resource which is available to an internal attacker without explicit usage of a provided API. For instance, the implementation of the vRTS could relate to a file at the hypervisor site. To avoid tampering with this resource, its contents needs to be protected through suitable integrity protection mechanisms, for instance:

- Log and track the internal state of the resource to protect against tampering(E.g. do this internally in the software or publish the current state of the list or use more sophisticated technologies for integrity protection of logs/states such as Blockchains).
- Use encryption paired with Message Authentication Codes (MACs) or only MACs, only if confidentiality is unnecessary(E.g. this might be done securely by leveraging enclave-based technologies or specific hardware. Most importantly, the encryption and MAC keys are not to ever be disclosed).

5.2.2.4 vRTR: VM/VNFCI Identity and Layer Binding

The identity of a VM/VNFCI is a very important property that needs to be fulfilled. If a VM/VNFCI is not identifiable at all, it is not possible to operate or manage it properly, since the MANO would lack to control the VM/VNFCI instantiation, operation and termination. Consequently, it is assumed that each VM/VNFCI has at least some basic property to uniquely identify it within the system and, in addition to that, it is also assumed that a VM/VNFCI is be relatable to a particular hypervisor. This means, MANO maintains a mapping which VM/VNFCI is operated on which hypervisor.

In terms of Remote Attestation, however, this simple mapping is not sufficient. In this case, the identity of a VM/VNFCI is typically defined as a cryptographic key that is unique to a particular VM/VNFCI. This is necessary to implement a sufficient vRTR. Naturally, this cryptographic key needs to be confidential and is not to be made publicly available, since once this key is disclosed it might be used to spoof the identity of the VM/VNFCI.

A second concern is to prove if a specific VM/VNFCI is executed on a particular hypervisor. Since the trust-state of a VM/VNFCI, determined during Remote Attestation, relies on the fact that both the VM/VNFCI and the hypervisor are in a trusted state, it is desired to establish a provable binding between hypervisor and VM/VNFCI. The ideal solutions in terms of provability would be to establish a strong layer binding between the VNF-layer and VNFI-layer directly within the discrete chip that implements the hypervisor's RTR. If such a strong hardware-rooted layer binding is available, it is possible to conduct a deep attestation of the VM/VNFCI which implicitly proves that the VM/VNFCI runs on top of a particular hypervisor with a high level of confidence.

In contrast to the ideal traceability of identity, however, there is the potential problem of scalability under real operating conditions. Under the assumption that many VM/VNFCI are operated at the same time on a single hypervisor, the hardware-based layer-binding is susceptible to impose a huge performance overhead. This is because the proof of layer-binding for each individual Remote Attestation procedure relies on a recent statement from the discrete hardware chip to ascertain whether the layer-binding is established or not. Without this statement the relation between the VM/VNFCI and the hypervisor is not demonstrable without any doubt.

NOTE: This procedure also includes the verification of the hypervisor's identity.

In each case, the layer binding proof involves multiple operations within the chip, including signing the hypervisor-related identity. These signing processes typically take a very long time and are usually synchronous processes that do not support concurrency.

5.3 System and Component Attestation-impact

5.3.1 Procedures Overview

In order to determine the attestation-scope of the involved systems and components, the granularity of the actual attestation needs to be considered and analysed further.

Under the assumption that the necessary security properties are fulfilled for the NFVI-layer's discrete RoTs and the VNF-layer's virtualised RoTs, it is possible to securely collect and report evidences from all parts of the NFV systems that are to be attested. Collecting evidences refers to securely acquiring measurements and securely storing these measurements with the assistance of the corresponding RoTs (RTS/vRTS), whereas the reporting of evidences refers to the utilization of this stored measurements and the generation of a statement under the assistance of the RTR/vRTR.

5.3.2 Evidence Collection on the Hypervisor and Virtual Machine

The collection of evidence is usually carried out by using several individual components that might vary depending on which LoA and which components are ultimately targeted. For load-time integrity this typically relates to leveraging different components that take measurements during the initial boot phase up until the OS kernel takes over control. From this point onward, the OS kernel might implement additional components for measurement acquisition that are concerned with taking and storing measurements of load-time or run-time portions of the system. Important to note is that for the LoA levels 2 (Hardware and Virtualisation Platform load-time) and 5a (Hardware and Virtualisation Platform run-time) measurements are only taken by components on the hypervisor; this refers to the Hypervisor and Virtual Machine sub-scopes defined in clause 4.3. This means that the evidence is to be stored on the hypervisor and anchored within discrete hardware components. In this case, confidence in the evidence might generally be considered very high, as confidence is based on the assumption that the discrete hardware components used are best protected against tampering. In addition, major parts of the VM's software stack, i.e. the early boot stages of VMs (from loading and executing the bootloader firmware to loading and executing the VM image and its kernel), are also under the exclusive control of the hypervisor and therefore do not require the provision of virtualised RoTs and do not require specific measurement components within the VM itself. For this reason, the hypervisor is responsible for: (a) acquiring these corresponding measurements which represent the actual virtual machine sub-scope and (b) securely storing them inside its RTS.

For LoA level 3 (VNF software packages load-time) and 5b (VNF software packages run-time), measurement components within the VM are responsible for measurement acquisition; this refers to the application sub-scope defined in clause 4.3. In this case, the responsible components perform the measurements under the supervision of the VM's kernel and leverage the virtualised RoTs provided by the hypervisor for secure storage. Confidence in the evidences collected depends on the actual implementation of these virtualised RoTs and ultimately an accurate prediction without a detailed analysis of the implementation specifics is not possible. To conclude, evidence collection that leverages the virtualised RoTs, particularly the RTS in this case, is only concerned with the software that is loaded and executed under the control of the VM and not with establishing the VM platform itself. Software, in this case, refers to any kind of software that is loaded or executed after the VM's kernel takes over memory management, process scheduling and the management of IO-operations; this includes also the loading of additional kernel modules during or after the boot process. Noteworthy in this case is that once the VM's kernel takes over control, the hypervisor loses the semantic understanding of the VM's internal state, referred to as the semantic gap issue, and thus is not well suited to take measurements that are related to the VM's internal operations on its own. Technologies such as VM introspection (VMI), which would enable such monitoring functions, are available but are not considered in the present document. In general, VMI is not a desirable technology in NFV, as it conflicts with strong insulation and security guarantees for the tenant. Furthermore, VMI also leads to a significant impairment of performance and still requires overcoming the problem of contextual understanding of the information to be monitored; this problem is called the semantic gap.

5.3.3 Reporting of the Hypervisor and Virtual Machine Current State

Once evidences are collected and securely stored by anchoring them within a discrete or virtualised RTS, these evidences are used to generate a report that represents the current system state. The report itself includes statements that refer to information stored in a measurement list, information that is anchored inside the RTS and a statement that proofs the origin of this information derived from the identity information from a corresponding RTR. The present document is then securely transmitted to a RAS that is concerned with verifying the system state based on trusted information or references that have been generated or which are available otherwise, see clause 5.2. Regardless of the LoA and the actual evidences that have been reported, the overall process for verifying the system state is all the same. After the authenticity of the report, w.r.t. proofing its source of origin on basis of the system's identity, has been verified, the relevant measurements are extracted from the report and verified against the available references. In case that all measurements could be verified successfully, that is for instance that all measurements could be related to a matching counterpart of a reference, the RAS outputs a statement whether the verification has been finished successful or not. This result might be a binary decision determining whether a system is considered trustworthy or not; or it might be a complex report that provides detailed information that needs to be processed further to come to a final conclusion about the trustworthiness of the system. Also, as explained, depending on the specific model adopted, multiple reports and RASs might be part of this verification phase; this affects the contents of the report or the information that is to be extracted, but not the process itself.

Consequently the system state report consists of the following parts:

- S1: A statement about the measurements included inside the report that represent the evidences
- S2: A statement that proves the correctness of the measurements in terms of its integrity
- S3: A statement that proves the source of origin of the collected evidences

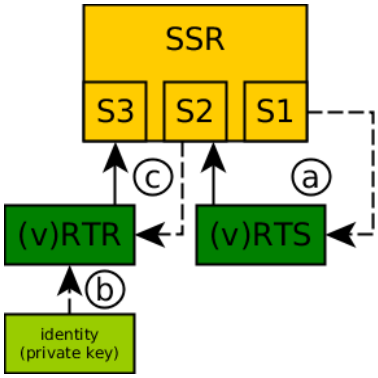

Figure 5.3.3-1: Composition and Generation of an SSR

The information of the report is related to one another and without all three parts a system state is not verifiable beyond any doubt. More specifically, as depicted in Figure 5.3.3-1, an SSR is composed of the above mentioned statements, w.r.t. S1, S2 and S3. The statement about the accumulated measurements (S1) are integrity protected through statement S2 that is generated by the corresponding (v) RTS, see (a) in Figure 5.3.3-1.

In addition, the report's statements need to be designed such that the proof of origin (S3) does not only prove the actual origin of the information but also the authenticity of the information used to prove the integrity of the measurements (S2). For example, assuming that the identity of a system relates to a private key stored inside a RTR, creating a digital signature over S2 provides both at the same time: (I) the proof about the source of origin and (II) the proof of authenticity of the information encapsulated in S2. This means that the verification of S3 implicitly proves both properties at the same time. First, the source of origin is proven by assuring that the correct private key has been used by the (v) RTR, see (b) in Figure 5.3.3-1. Second, the authenticity of S2 is proven by verifying that the corresponding integrity statement S2 served as the input of the (v) RTR during the creation of S3, see (c) in Figure 5.3.3-1.

Once the digital signature has been verified successfully, the actual measurement from S1 is typically considered as authentic and thus is usable as a reliable source during the actual system state verification on the basis of references.

Specific for NFV is that for LoA 3 and 5b two reports are required to determine the system state including the application sub-scope from the VM, which refers to the state of the VNFCI executing a process that represents the, or part of the, VNFI. One report therefore represents the hypervisor's state and another report the VNFCI's state. Whether or not the verification of those reports are carried out by distinct or the same RAS does not affect the reports structure or the processes. Nevertheless, it is recommended that the verification process of S3 for the VNFCI report also enables to prove that the VNFCI was executed on a particular hypervisor. Regardless of whether this is implicit proof established through strong layer binding or explicit proof through inspection of other available information, this step is necessary for making a convincing verification feasible.

5.4 RA Deployment Alternatives

5.4.1 Location of RA and relations to MANO

The RA service might be a service that the tenant has access to when he/she wants, for example when populating a VNF with sensitive data or application keys. However, the RA service could also be a service that the tenant is not using directly him/herself but delegates to RA service operation to the MANO domain. In this clause two approaches to place an RA service are depicted.

Assuming a model for the RA functions that the actions to perform the verification are as described in Figure 5.4.1-1, the verification involves trusted chain of verification keys that might be used to verify the attested data and the presence of set of rules at the verifier by which the attested information is interpreted. The verification result is here assumed to be a simple response declaring the attested information to be compliant to the rules or to be none-compliant. Additional information could be considered as output of the verification but for the sake of clarity exposition is simplified by not consider such generalizations.

Since some RoTs are part of the HW, the trust chain starts in the hardware that the owner of the NFVI has selected. Thus, the NFVI needs to provide a service that is usable by the RAs in order to securely fetch information that is needed to perform a verification. This might be certificates or public key material or secure referrals to places to other services, databases, or ledgers where the required information might be found.

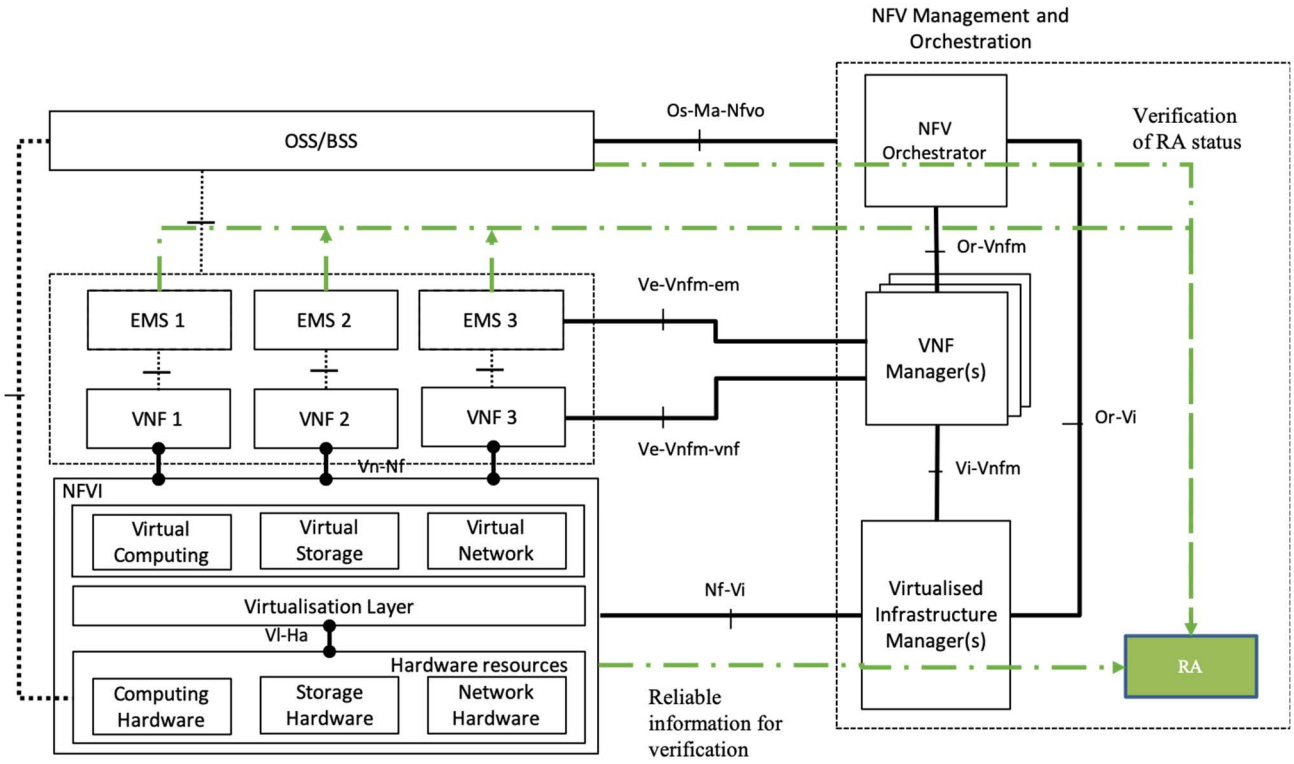

Figure 5.4.1-1: RA in MANO. OSS/BSS is able to (via intermediate interfaces) reach RA

5.4.2 RA in MANO space

In the case the RA is in MANO, it is likely that one only provides attestation of components in the NFVI. In this case the tenant in the OSS/BSS layer (or EMS) would be able via MANO, to fetch the attested information and logs of the RA in the MANO for itself. Such information could be used as supportive evidence for compliance when the tenant system is evaluated.

When RA service is in MANO domain as depicted in Figure 5.4.1-1, some standard interfaces between MANO and the OSS/BSS system are necessary and have to be defined. These interfaces could potentially be exposed over the Os-Ma-Nfvo reference point or it could be directly exposed by the RA service from within MANO.

5.4.3 RA in tenant space

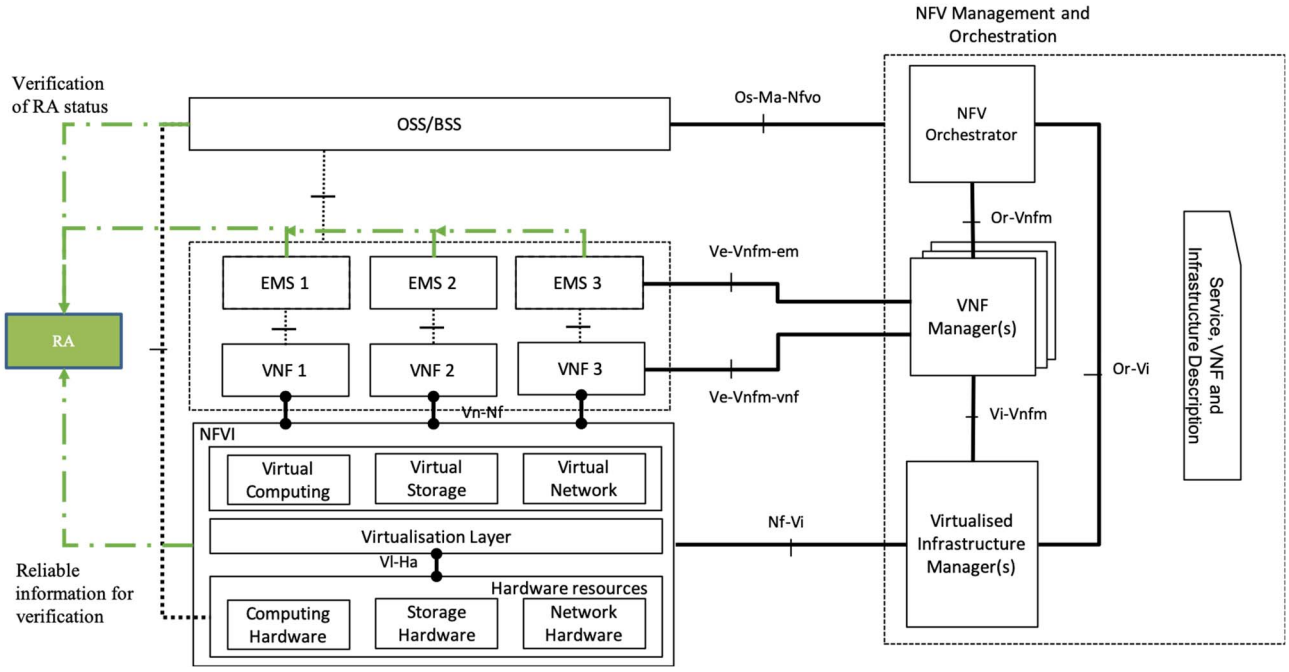

Figure 5.4.3-1: Example of RA in tenant space supporting the SEMS

In case the tenant wants to use attestations between an EM and a VNF or between the OSS/BSS and a VNF the verifying RA at each specific level needs to be provisioned with the information to perform the verification. In this approach to attestation, the tenant also realizes the RA function(s). The verification keys (certificates) that relate to the reporting engine in the NFVI need to be provided by the owner of the NFVI. This might be done out of band, that is the OSS/BSS gets this data via non-standardized, yet secured, procedures, or the OSS/BSS might request this information via MANO services which act as proxy to an Infrastructure Attestation Information Service (IAIS). In this setup the relation of the attest verifiers to the NFVI IAIS is similar to that of a certificate verifier and a certificate validation service for a PKI. This approach provides for better automatization but also in this setup some keys material related to the NFVI needs to be provisioned to the OSS/BSS by secure out-of-band procedures.

Note that in this picture the RA service is drawn as a separate entity in the tenant domain. However, this is a logical entity that might be placed within the EMS or OSS/BSS or even within the security manager entity as depicted in [i.2].

5.5 Remote Attestation Protocol Recommendations

A remote attestation protocol is concerned with the secure exchange of the collected evidences between the systems that are to be evaluated and the RAS.

The SSR that was described in clause 5.3.3 is in itself already coherent and contains all information necessary for its evaluation. Furthermore, it is not possible to manipulate the SSR's encapsulated data and it is not possible to forge the originator of the data without detection from RAS. The major goal of the remote attestation protocol it therefore to assure properties that are not addressed implicitly by the SSR.

Freshness and replay protection are the primary targets that the remote attestation protocol needs to address. This is typically solved by introducing a nonce-value that is unique and only valid in the context of a single protocol session that has been established between one SuE and the RAS.

A remote attestation protocol session is initiated by RAS that generates and sends the nonce value to the SuE. The SuE now uses this nonce value during the SSR generation process and hence generates an SSR that is contextually bound to this nonce. Contextually bound means that the function that generates the SSR's statement S2 uses both, the statement S1 and the nonce as an input, as depicted in Figure 5.5-1. Using S1 and the nonce renders the output S2 of the function unique and bound to that particular nonce. This means that even if the actual state of the SuE does not change between two consecutive attestation processes, S2 adopts two different values for each individual attestation procedure since the nonce is different.

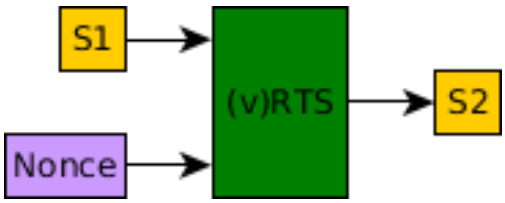

Figure 5.5-1: Generation of Statement S2 with Nonce

By introducing the nonce value, the remote attestation protocol offers a solution for providing the required freshness and replay-protection properties.

The properties are provided by the fact that an SSR is only valid within a single session that is under the full control of RAS. Consequentially, it is not possible for a SuE to generate a valid SSR response before knowing the nonce value and, at the same time, it is not possible for the SuE, or any other party, to re-use a system state other than the current one.

The freshness property does not represent an actual time frame for the validity of the SSR. How long a single session might last needs to be defined otherwise, for instance by defining a timeout until a response needs to be received by RAS. This means that the freshness property is related to a system state in the scope of a single session.

It is recommended that the remote attestation protocol is performed via a secure channel because the state information transferred might reveal details about the underlying SuE that is not to be known to anyone other than RAS. However, the confidentiality of SSRs is not required for conducting a reliable remote attestation. Similarly, the authenticity of the SuE and the SSR is also assured due to the known and verifiable identities of the RTRs involved.

Figure 5.5-2 depicts the SSR generation process using a nonce created by the RAS during the Remote Attestation Protocol procedures.

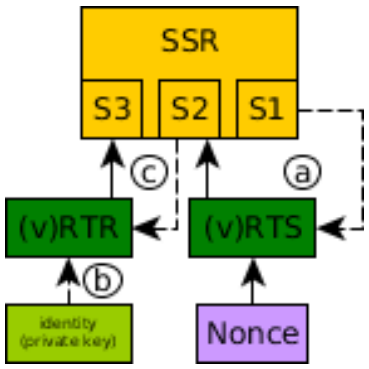

Figure 5.5-2: SSR Generation with RAS provided Nonce

5.6 Remote Attestation Architecture Instantiations

5.6.1 Transitive Model Architecture Instantiation

The assumption for the transitive model is that the information for verifying individual SSRs from a system within a specific domain is only possible in corresponding related security domain. Accordingly, as depicted in Figure 5.6.1-1, a Hypervisor from within the CSP Domain might only be verified by a RAS in the CSP Security Domain. VMs, on the other hand, belong to one particular CSC Domain and hence might only be verified by the specific RAS from the corresponding CSC Security Domain.

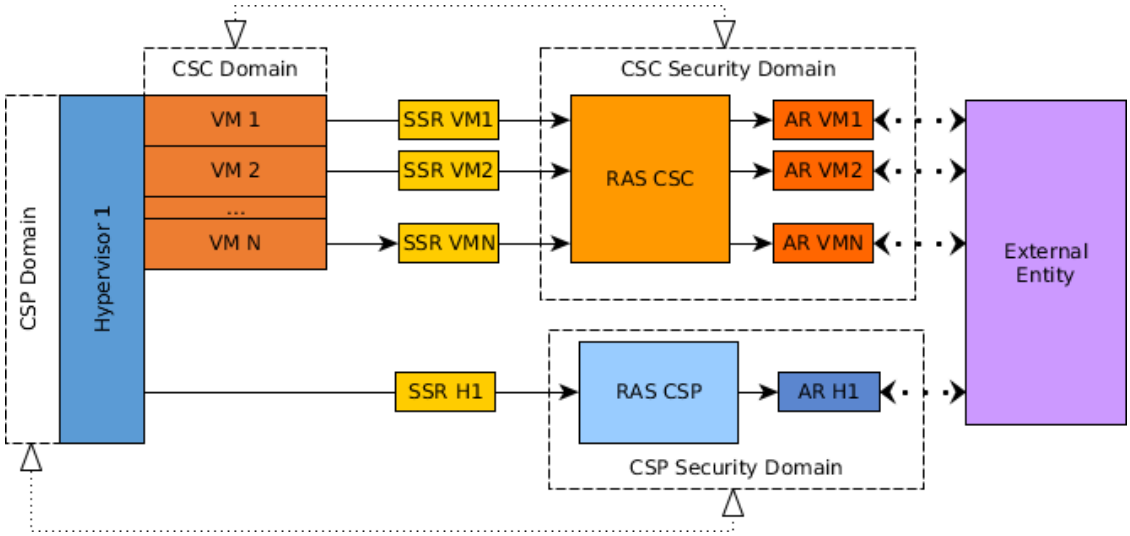

Figure 5.6.1-1: Instantiated Remote Attestation Architecture for Transitive Model

Each individual RAS then verifies the SSRs it receives and generates an Attestation Result (AR). The individual ARs might be shared with other entities, for instance an external entity or with RASs from other Security Domains. Whether sharing is possible is to be defined by a policy that enforces an ACL defining specific access policies.

The overall assumption for the transitive model is that the AR that have been determined within a security domain are considered as trusted in terms of correctness; there needs to be no doubt about the actual results of an AR. Assuming that the ARs are trusted, the external entity might use them for all kinds of operations. For instance, assuming the external entity is responsible for maintaining state information about VNFIs, it collects all necessary ARs for all available/visible systems and updates the state information for the VNFIs accordingly.

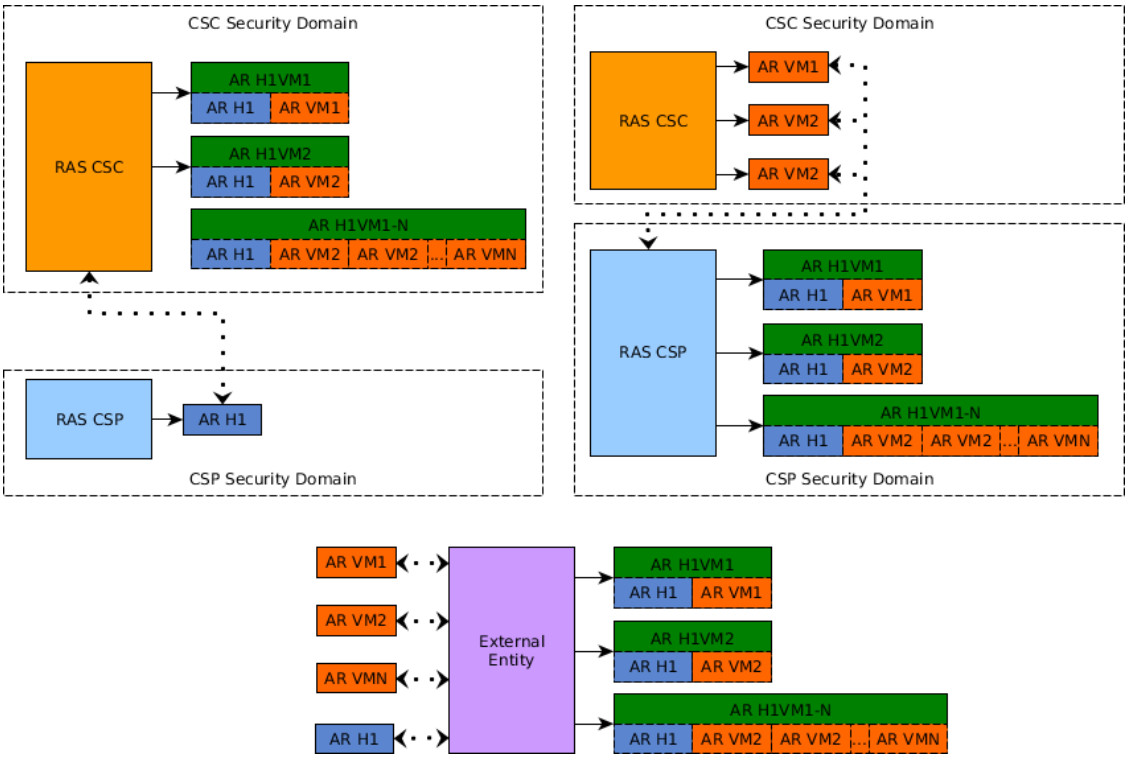

Figure 5.6.1-2: Composition of Attestation Results from different Entities in the Transitive Model

Alternatively, any system that has access to the ARs from both the CSP and CSC might establish a composition of individual ARs, this is depicted in Figure 5.6.1-2 this composite AR then describes the state of two or more related systems. Under the assumption the necessary information on how VNFIs are composed in terms of interconnected systems, it is also possible to generate a composite AR that corresponds to all systems that are related to one or multiple VNFIs.

The composite AR is trusted ultimately in case it has been generated within a particular security domain. If it was created by an external entity, the processor of the composite AR needs to decide for itself whether to trust it or not. In case the processor does not trust the composite AR, the processor needs to implement checks that verifies the encapsulated individual ARs by itself.

5.6.2 Transitive Model Architecture Instantiation using PDLT

In a transitive model, attestation servers are hosted by separate service providers. Each attestation server is involved in the remote attestation of VNFs owned by the corresponding service provider. The VNFs running in a NFVI platform might be owned by different service providers and attested by several attestation servers. An attestation server needs assurance or proof that the platform of the CSP (CSP-NFVI) has been attested and that all VNFs running on the platform have been attested as well, but is not able to attest the VNFs owned by another service provider itself.

There is a need for each attestation server to establish a trust relationship with the others, to obtain this proof of attestation as described in clause 5.6.3. A permissioned distributed ledger technology (or permissioned blockchain) might be used to establish this trust relationship with each other at runtime.

The goal of the permissioned distributed ledger is to provide each attestation server with some distributed ledger entries such as:

- The identity of all certification authorities authorizing service providers into the distributed ledger. The only authority able to insert such certification authority in the ledger is the distributed ledger authority (BCA).
- The identity of the various service providers that have been authorized into the ecosystem. Only certification authorities described in the first bullet are able to insert service providers into the distributed ledger.
- The identity and the related metadata of the attestation servers that the service providers want to provision into the ecosystem. Only service providers are able to insert the attestation servers within the distributed ledger.

The metadata related to attestation server includes information on the way to communicate with the said attestation server. For each attestation server, a digital certificate traces back the identity of the service provider. With these information attestation servers are able to establish a trust relationship with each other and exchange at runtime continuously and securely some information such as ephemeral (short-lived) key material. The key material could rotate quickly over time to reduce the value of the cryptographic material valid only during a short time, and to increase the level of security. A Hardware Security Module attached to the attestation server ensure that the permissioned distributed ledger and the generation of keys are securely implemented. The HSM ensures a high quality on the random generator and on keys.

This key material could be used for proof of attestation using symmetric keys as defined in clause 5.6.3.

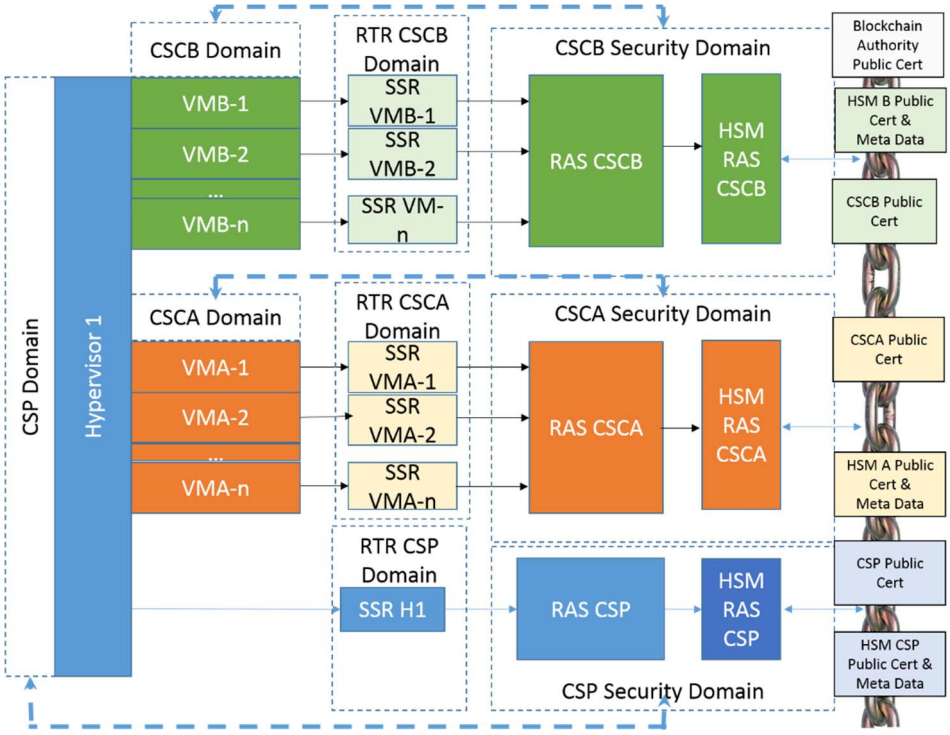

Figure 5.6.2-1: Instantiated Remote Attestation Architecture for Transitive Model using DLT

The RTR CSCA and RTR CSCB are roots of trust that are to be implemented in a secure environment such as a Hardware-Mediated Execution Enclave (HMEE) and are used to translate the local attestation of a VNF instance to a remote attestation using a key material controlled by the CSCA for the RTR CSCA or using a key material controlled by the CBCB for the RTR CSCB.

If the VNFI is instantiated in a HMEE, the local attestation is done using the key material of the CPU, and verifiable by the RTR running on the same CPU in an HMEE.

Apart from attestation, the RTR HMEE also offers data sealing services to VNFI running in a HMEE, unique cryptographic keys provided per enclave, such that any data sealed by the enclave might later be unsealed by this enclave only. This is a way to securely share the NFVI storage. There could be different data sealing keys on a per-enclave, per-platform and per-CSC basis.

A secure channel is established between the RTR HMEE of the CSC and the corresponding HSM of the RAS. Key material is then securely provided by the HSM directly to the RTR running in a HMEE.

This data sealing service is used for the proof of attestation using symmetric keys as described in clause 5.6.3.

5.6.3 Proof of attestation using symmetric keys

The goal of this proof of attestation using symmetric keys, is a trust establishment between codes running within different service provider ecosystem. Without exchanging golden measurement, the service provider obtains the proof that the code of the other service provider has been attested.

In most of the case, remote attestation is done using digital certificates. The benefit of this type of attestation is that only one public certificate needs to be published to verify the authenticity of attestation certificates. As such, crypto key and crypto certificate management throughout the ecosystem is minimized.

However, attestation using digital certificates is computationally very expensive and slow.

A proof of attestation using symmetric keys has the benefit to be very fast compared to attestation using digital certificates and to be more quantum resistant.

The proof of attestation is based on the fact that a cryptographic sequence (see Figure 5.6.3-2) is able to successfully complete between two endpoints. This success offers an indirect proof that both endpoints have already been verified as trustworthy by their respective attestation platforms.

After an initial remote attestation of the RTR CSCA by the HSM of the remote attestation server RAS CSCA, a secure channel is established between the HSM and the RTR CSCA. The HSM then provides to the RTR CSCA an array of master symmetric keys, with a global UUID represented by a large 128 bit GUID and a Time-to-Live (TTL) metadata for each key as shown in the Figure 5.6.3-1. These master keys never leave the HMEE of the RTR.

After instantiation of a VNF and successful local attestation of this VNFI, the RTR CSCA derives from the master symmetric a derived key and releases it to the VNF instance, along with the GUID of the master key from which it was derived from and the large nonce used for the derivation, as shown in Figure 5.6.3-1. Possession of this derived key by the VNFI is the proof that it has been successfully attested by Root of Trust in relation with a trusted attestation server. The VNFI further uses these materials to proof to another VNFI or attestation server that it has been attested as described in the flow of Figure 5.6.3-2.

At this end of the procedure the two VNFIs have exchanged a session key.

To ensure a high reliability of this proof of attestation, it is recommended that the VNFI contains a component in a HMEE, which is used for a secure communication with the RTR through a secure channel, for the storage of the derived key and the process described in the flow of Figure 5.6.3-2.

This could be used also to obtain a proof of attestation of the user device application communicating with a VNFI. In this case the derived key is provisioned in a secure and tamper resistant part of the device application (e.g. TEE, SE) when the device application has been attested by a remote attestation server. In the flow of Figure 5.6.3-2, the VNFI1 is for this case the User device application.

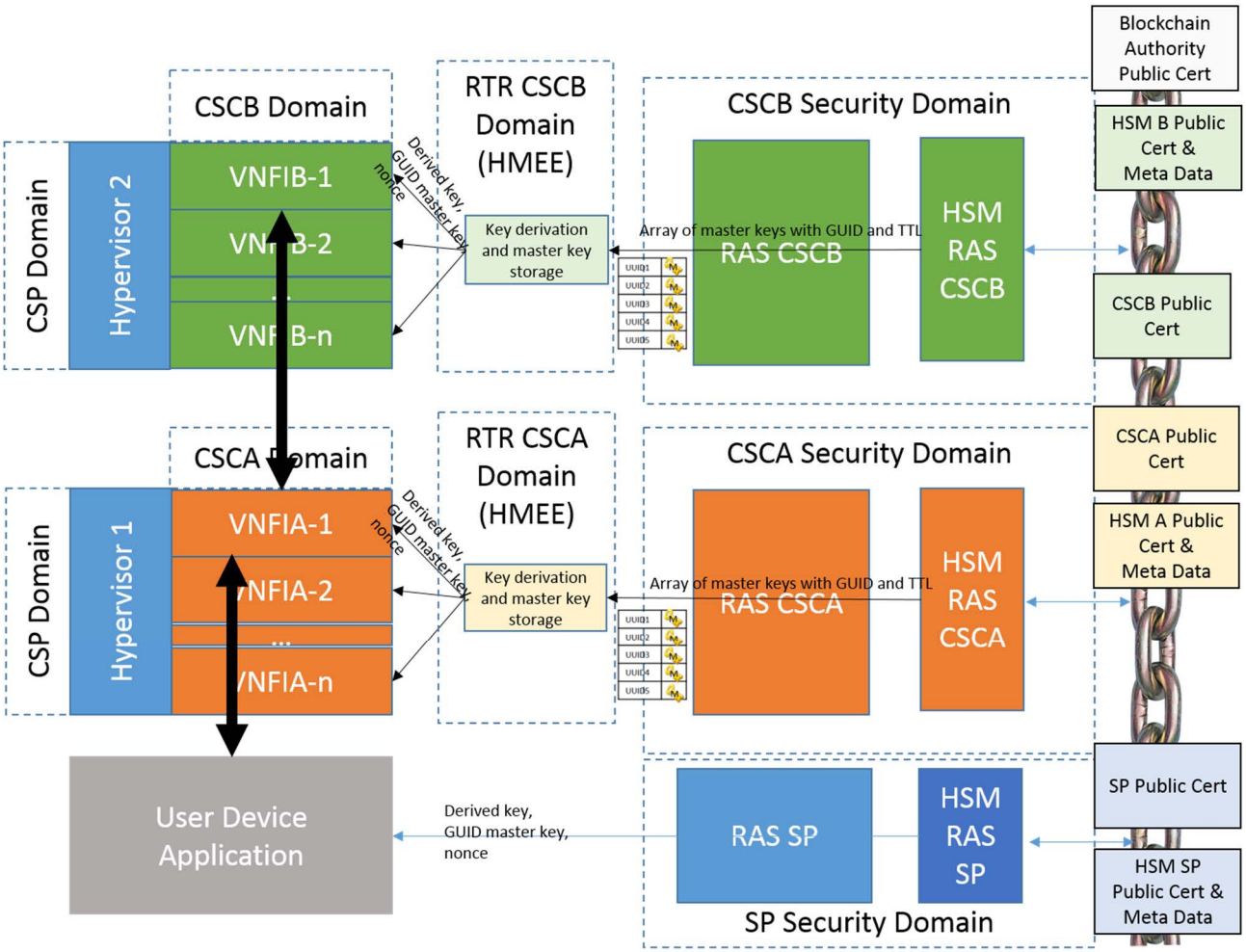

Figure 5.6.3-1: Architecture diagram for proof of attestation using symmetric keys

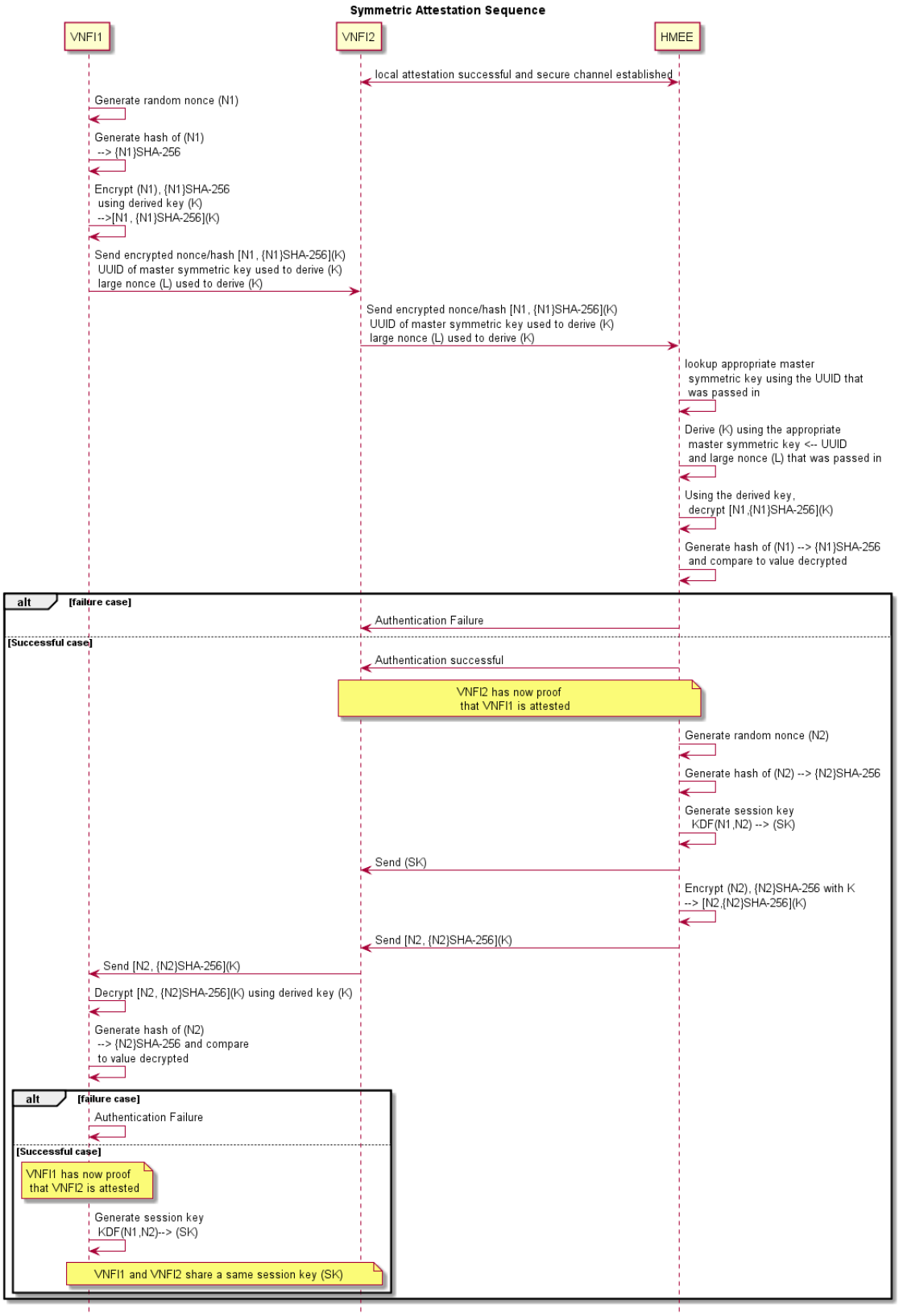

Figure 5.6.3-2: Proof of attestation using symmetric keys sequence diagram

5.6.4 Centralized Model Architecture Instantiation

In the centralized model, depicted in Figure 5.6.4-1, the SSRs of all systems (Hypervisors and VMs) and Domains (CSP-, CSC-Domain) are collected and verified by RAS in the RATP Security Domain. The generated ARs from RATP's RAS are ultimately trusted in terms of correctness by all external entities that process the ARs and take decisions. Whether a specific external entity has access to the ARs might be defined in additional ACL policies and needs to be enforced by the system which is responsible for doing that.

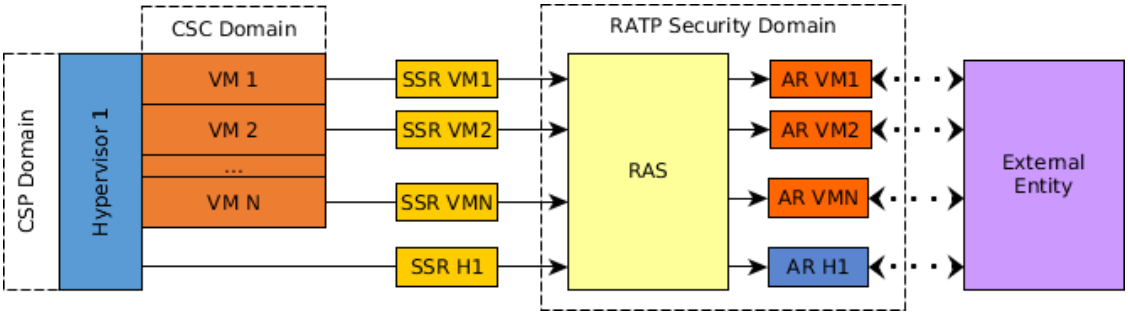

Figure 5.6.4-1: Instantiated Remote Attestation Architecture for Centralized Model

Alternatively, a composite AR might be generated by RATP's RAS that makes a statement about a set of related systems, as depicted in Figure 5.6.4-2. This typically corresponds to one or more specific hypervisors and VMs under their control. Similarly to the transitive model it is possible that RATP's RAS generates composite ARs for one or multiple VNFIs, assuming the information on how these systems are interconnected is available.

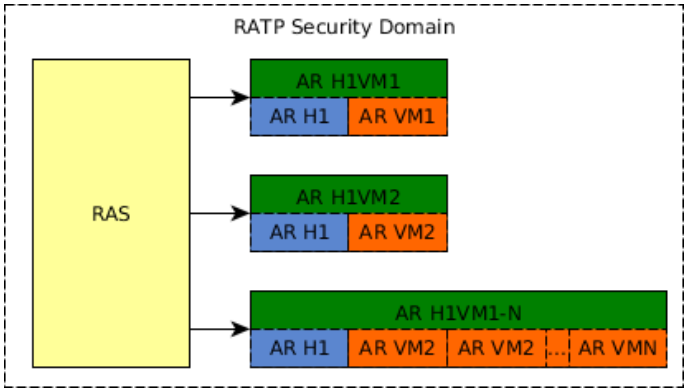

Figure 5.6.4-2: Composition of Attestation Results from RATP's RAS in the Centralized Model

Annex A:
Change History

Date	Version	Information about changes
2017-10-04	V0.0.2	ToC and Scope added
2018-08-15	V0.0.8	– Architectural Use Cases (NFVSEC(18)000037) – Use Cases Scenarios (NFVSEC(18)000023r1) – High Level Architecture (NFVSEC(18)000036) – Trust Assurance Considerations(NFVSEC(18)000056r1) – Trust at the Service Layer (NFVSEC(18)000055r2) – System and Component Attestation-impact (NFVSEC(18)000134) – Remote Attestation Architecture: Instantiations (NFVSEC(18)000135) – Update_to_Motivation_and_Problem_Description (NFVSEC(18)000143r1) – Update to Challenges and Limitations (NFVSEC(18)000145)
2019-05-22	V0.1.1	– Remote Attestation Protocol Recommendations (NFVSEC(19)000065r1) – SEC018 Distributed model using DLT and symmetric-based attestation (NFVSEC(19)000017r1) – Remote attestation service placement in NFV Ericsson Contribution (NFVSEC(19)000064) – Review Contribution (NFVSEC(19)000067r2)
2019-08-21	V0.1.2	– Editorial changes to prepare for edit help process

History

Document history		
V1.1.1	November 2019	Publication