**3GPP TSG-SA3 Meeting #101-e *S3-203167***

**e-meeting, 9th – 20th November 2020** Revision of S3-20xxxx

**Source: Nokia, Nokia Shanghai Bell**

**Title: SCAS VNP: Security requirements on the interface between VNF and VNFM**

**Document for: Approval**

**Agenda Item: 5.2**

# 1 Decision/action requested

***SA3 is kindly asked to approve the proposed changes in TR 33.818 v0.8.0.***

# 2 References

[1] 3GPP TR 33.818 v0.8.0 Security Assurance Methodology (SECAM); and Security Assurance Specification (SCAS); for 3GPP virtualized network products

# 3 Rationale

In current TR 33.818 v0.8.0 [1], there are two clauses with the similar titles: “*5.2.5.5.3.7 Security functional requirement related to virtualization*” and “*5.2.5.5.7 Security functional requirements deriving from virtualisation and related test cases*”. In order to avoid confusion, this pCR proposes to merge *5.2.5.5.3.7* into *5.2.5.5.7*, by making all the requirements in *5.2.5.5.3.7* as one of the requirements in *5.2.5.5.7* and add a test purpose and additional execution step in the test case of GVNP lifecycle management security for testing the secure protection on the interface between 3GPP VNF and VNFM.

# 4 Detailed proposal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Start of the Change \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

5.2.5.5.3 Technical baseline for general security functional requirements

5.2.5.5.3.1 Introduction

The technical baseline is a generic set of security requirements to be fulfilled by all virtualized network products.

In particular these requirements counter the security threats identified in clause 5.2.4.2.2 and they basically aim to guarantee the network product confidentiality, integrity and availability.

5.2.5.5.3.2 Protecting data and information

All text from TS 33.117 [4], clause 4.2.3.2 applies to GVNP of type 1.

5.2.5.5.3.3 Protecting availability and integrity

5.2.5.5.3.3.1 System handling during overload situations

All text from TS 33.117 [4], clause 4.2.3.3.1 applies to GVNP of type 1.

5.2.5.5.3.3.2 Boot from intended memory devices only

All text from TS 33.117[4], clause 4.2.3.3.2 applies to GVNP of type 1.

5.2.5.5.3.3.3 System handling during excessive overload situations

All text from TS 33.117 [4], clause 4.2.3.3.3 applies to GVNP of type 1.

5.2.5.5.3.3.4 System robustness against unexpected input

All text from TS 33.117 [4], clause 4.2.3.3.4 applies to GVNP of type 1.

5.2.5.5.3.3.5 Virtualized Network product software package integrity

All text from TS 33.117 [4], clause 4.2.3.3.5 applies to GVNP of type 1.

In addition, VNF package and VNF image integrity shall be validated when on board, and VNF image integrity shall be validated when in instantiated. The detailed security requirements and related test cases are as following.

5.2.5.5.3.3.5.1 VNF package and VNF image integrity

*Requirement Name*: VNF package and VNF image integrity

*Requirement Description*:

1) VNF package and image shall contain integrity validation value (e.g. MAC).

2) VNF package shall be integrity protected during onboarding and its integrity shall be validated by the NFVO.

*Threat Reference*: TR 33.926 [3], Clause 5.3.4.1, "Software Tampering "; TR 33.848, Clause 5.18, “Key Issue 17: Software Catalogue Image Exposure”

*Test case*:

**Test Name:** TC\_VNF PACKAGE AND IMAGE­\_ INTEGRITY

**Purpose:**

1. To test whether the VNF package has been integrity protected or not.

2. To test whether the VNF image has been integrity protected or not.

**Procedure and execution steps:**

**Pre-Condition:**

- The virtualized network product document describes information regarding integrity protection of VNF package and VNF images, including details of how the integrity check is carried out, who makes the digital signatures of VNF package, what evidence is created to prove that the integrity check has been executed and what the result of the check is etc.

- A valid VNF package and a not-valid VNF package (e.g. a tampered image in VNF package) are available.

- A valid VNF image (i.e. a correct HASH value is attached) and a not-valid VNF image (i.e. an incorrect HASH value is attached, e.g. the VNF image can be tampered when the VNF image is sent from the NFVO to the VIM or when the VNF image is stored in the image repository) are available in the image repository of VIM.

- There are NFVO and VIM, or simulated NFVO and VIM.

**Execution Steps**

**Execute the following steps:**

1. Review the documentation provided by the vendor describing how VNF package integrity is verified;

2. During VNF package onboarding, the tester uploads a valid VNF package into a NFVO. The NFVO verifies the integrity of the VNF package by validating the digital signature of the VNF package using the certificate of VNF vendor according to the documentation;

3. During VNF package onboarding, the tester uploads a not-valid VNF package into a NFVO. The NFVO validates the digital signature of the VNF package using the certificate of VNF vendor;

4. During VNF instantiation, the VIM selects a VNF image with a correct integrity protection value from the image repository to instantiate the VNF image.

5. During VNF instantiation, the VIM selects a VNF image with an incorrect integrity protection value from the image repository to instantiate the VNF image.

**Expected Results:**

1. The VNF package is successfully onboarded into the NFVO;

2. The not-valid VNF package is not onboarded;

3. The VNF image with a correct integrity protection value is instantiated by the VIM;

4. The VNF image with an incorrect integrity protection value is not instantiated by the VIM.

**Expected format of evidence:**

Snapshots containing the result of the VNF package on boarding and the VNF image instantiation.

5.2.5.5.3.4 Authentication and authorization

All text from TS 33.117 [4], clause 4.2.3.4 applies to virtualized network products.

5.2.5.5.3.5 Protecting sessions

All text from TS 33.117 [4], clause 4.2.3.5 applies to virtualized network products.

5.2.5.5.3.6 Logging

All text from TS 33.117 [4], clause 4.2.3.6 applies to virtualized network products.

5.2.5.5.4 Operating systems

All text from TS 33.117 [4], clause 4.2.4 is generic and applies to guest operating systems for GVNP of type 1.

5.2.5.5.5 Web servers

All text from TS 33.117 [4], clause 4.2.5 applies to GVNP of type 1.

5.2.5.5.6 Network devices

All text from TS 33.117 [4], clause 4.2.6 applies to GVNP of type 1.

5.2.5.5.7 Security functional requirements deriving from virtualisation and related test cases

5.2.5.5.7.1 Security functional requirements on GVNP lifecycle management

Editor’s Note: GVNP lifecycle management discussed in this clause is different from the product lifecycle management processes in clause 6. The difference between them is to be added.

*Requirement Name*: GVNP lifecycle management security

*Requirement Description*:

1) VNF shall authenticate VNFM when VNFM initiates a communication to VNF.

2) VNF shall be able to establish securely protected connection with the VNFM.

3) VNF shall check whether VNFM has been authorized when VNFM access VNF’s API.

4) VNF shall log VNFM’s management operations for auditing.

*Threat Reference*: Threats on interface between 3GPP VNF and VNFM, in clause 5.2.4.2.2.3

*Test case*:

**Test Name:** TC\_LIFECYCLE MANAGEMENT SECURITY

**Purpose:**

1. To test the VNF authenticates VNFM when VNFM initiates a communication to VNF.

2. To test the VNF establishes secure connection with the VNFM after successful authentication.

3. To test the VNF check whether VNFM has been authorized when VNFM access to VNF’s API.

4. To check whether VNF logs the lifecycle management operations from VNFM.

**Procedure and execution steps:**

**Pre-Condition:**

1. There is a VNFM (or simulated VNFM) in the test environment.

2. The VNF vendor’s document describes how VNF authenticates/authorizes VNFM.

**Execution Steps**

**Execute the following steps:**

1. The tester triggers the establishment of communication between the VNF and the VNFM.

2. The tester captures the communication between the VNF and the VNFM using a tool (e.g. wireshark).

3. The tester checks whether the VNF authenticates the VNFM or not according to the mechenism described in the vendor’s document. For example, the VNF can use HTTPS to communicate with the VNFM, the VNF uses VNFM’s certificate for authentication.

4. The tester checks whether the VNF establishes secure connection with the VNFM after successful authentication. For example, a TLS connection is established after the VNF successfully authenticates the VNFM.

5. The tester using the VNFM to access the VNF’s API and checks whether the VNF authorizes the VNFM or not according to the mechenism described in the vendor’s document. For example, VNF can use OAuth2.0 to authorize the VNFM. The VNF uses VNFM’s token for authorization.

6. The tester checks whether the VNF logs the operations from VNFM or not.

**Expected Results:**

1. Secure communication is established between VNF and VNFM with integrity and confidentiality protection.

2. The VNFM successfully accesses the VNF’s API.

3. The VNF logs the operations from VNFM.

**Expected format of evidence:**

1. Pcap traces contain the authentication and authorization processes.

2. Screenshot contains the logs.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* End of the Change \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*