**3GPP TSG-SA3 Meeting #99e *S3-201132***

**e-meeting, 11 -15 May 2020** Revision of S3-20xxxx

**Source: China Mobile**

**Title: Adding security functional requirements deriving virtualisation and related test cases for GVNP of type 1**

**Document for: Approval**

**Agenda Item: 5.6**

# 1 Decision/action requested

It is proposed to add the security functional requirements deriving virtualisation and related test cases for GVNP of type 1 into clause 5.2.5.5.x.

# 2 Rationale

This contribution describes the security functional requirements deriving virtualisation and related test cases for GVNP of type 1 and adds these requirements into clause 5.2.5.5.x.

# 3 Detailed proposal

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Start of the change \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

5.2.5.5.x Security functional requirements deriving from virtualisation and related test cases

5.2.5.5.x.1 Security functional requirements on lifecycle management

*Requirement Name*: lifecycle management security

*Requirement Description*:

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

2) VNF shall check whether VNFM has been authorized when VNFM accesst VNF’s API.

3) VNF shall log VNFM’s management operations for auditting.

*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 check whether VNFM has been authorized when VNFM access to VNF’s API.

3. 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 establishes a communication between a VNF and a 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 mechemism 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 using the VNFM to access the VNF’s API and checks whether the VNF authorizes the VNFM or not according to the mechemism 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.

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

**Expected Results:**

1. Secure communication is established between VNF and VNFM.

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.

5.2.5.5.x.2 Security functional requirements on executive environment provision

*Requirement Name*: secure executive environment provision

*Requirement Description*:

The VNF shall support to compare the owned resource state with the parsed resource state from VNFD (VNF Description) by the VNFM. The VNF can query the parsed resource state by the VNFM from the OAM. The VNF shall send an alarm to the OAM if the two resource states are inconsistent. This comparing process can be trigerred periodically by the VNF, or the administratior can manually trigger the VNF to perform the comparing process.

Editor’s note: The virtualisation layer provides the execution environment for the VNF. The security of the virtualisation layer is a base of theVNF security. The operators should check whether their VNFs are run on the trusted virtualisation layer.

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

*Test case*:

**Test Name:** TC\_SECURE EXECUTIVE ENVIRONMENT PROVISION

**Purpose:**

1. To test whether the VNF compares the owned resource state with the parsed resource state.

2. To test whether the VNF send an alarm to the OAM if the two resource states are inconsistent.

**Procedure and execution steps:**

**Pre-Condition:**

There are a VNF, a virtualisation layer (or simulated virtualisation layer), an OAM, a VNFM, a VIM (or simulated OAM, VNFM, VIM) on the test environment.

**Execution Steps**

**Execute the following steps:**

1. The tester utilizes the virtualisation layer to change the resource state of VNF (e.g. change vCPU size of the VNF).

2. The tester uses the VNF to query the parsed resource state from the OAM.

3. The tester uses the OAM to query the parsed resource state of the VNF from the VNFM and send the received resource state to the VNF.

4. The tester checks whether the VNF sends an alarm to the OAM when the VNF receives the parsed resource state from the OAM and finds that the owned resource state and the parsed resource state are inconsistent.

**Expected Results:**

1. The VNF send an alarm to the OAM when the VNF receives the parsed resource state from the OAM and find that the owned resource state and the parsed resource state are inconsistent.

**Expected format of evidence:**

1. Screenshot contains the alarm on the OAM.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* End of the change \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*