T1611.501 Malicious privileged container-VNF Shared Resource Access

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Who | Current text | Proposed text | Final text |
|  |  |  |  |  |
|  |  |  |  |  |

Description: An adversary may compromise a target VNF to gain unauthorized access to the data in shared resources with other VNFs.

A malicious VNF instantiated in the NFVI may be able to access the resources reserved for another tenant VNF if root or escalated privilege is gained due to misconfiguration of host or container. This exploitation can lead to unauthorized data access in shared resources. Multiple techniques can be used to isolate VNF/VNFc where sharing virtualization resources is a business requirement to ensure a co-resident compromised or malicious VNF/VNFc cannot access shared resources or read data.

Labelling:

* Sub-techniques: none
* Applicable Tactics: Privilege escalation
* Platform(s): CI/CD, OA&M Tools
* Access type required: User/NPE/Administrative access
* Data Sources:
* Theoretical/Proof of concept/Observed:
* Architecture Segment: Impl-OA&M, Impl-Virtualization

Procedure Examples:

|  |  |
| --- | --- |
| **Name** | **Description** |
| S0683 | Peirates can deploy a pod that mounts its node’s root file system, then execute a command to create a reverse shell on the node |
| S0600 | Doki was run through a deployed container |
| G0139 | TeamTNT has deployed different types of containers into victim environments to facilitate execution. |

Mitigations

|  |  |
| --- | --- |
| **Name** | **Description** |
| M1047 | Scan images before deployment, and block those that are not in compliance with security policies. In Kubernetes environments, the admission controller can be used to validate images after a container deployment request is authenticated but before the container is deployed. |
| M1035 | Limit communications with the container service to local Unix sockets or remote access via SSH. Require secure port access to communicate with the APIs over TLS by disabling unauthenticated access to the Docker API, Kubernetes API Server, and container orchestration web applications. |
| M1030 | Deny direct remote access to internal systems through the use of network proxies, gateways, and firewalls. Segment execution environment with node and network. |
| M1018 | Enforce the principle of least privilege by limiting container dashboard access to only the necessary users. |
|  |  |

Pre-Conditions

|  |  |
| --- | --- |
| **Name** | **Description** |
|  |  |

Critical Assets

|  |  |
| --- | --- |
| **Name** | **Description** |
| VNF and VNF Data | Adversary may identify high value 5G network functions targets for its exploits |

Detection

|  |  |
| --- | --- |
| **Name** | **Description** |
| DS0015 | Audit application logs (NFVO, VIM). Configuration management databases (CMDB) and other asset management systems may help with the detection of computer systems or network devices that should not exist on a network. |
| DS0032 | Monitor container creation, container start events |
| DS0009 | Monitor process activity on node, hosts and VNFs. |
| DS0034 | Monitor volume or storage modifications, attachment or read actions. |

Post-Conditions

|  |  |
| --- | --- |
| **Name** | **Description** |
|  |  |

References:

|  |  |
| --- | --- |
| Name | URL |
| Network Functions Virtualisation (NFV) Release 4;  Security;Isolation and trust domain specification  Release 4, section 4.2.1, Access 4/12/2022 | https://docbox.etsi.org/ISG/NFV/Open/Drafts/SEC026\_Isolation\_and\_trust\_domain |

Notes:

<Following needs to be moved to Network Slice attack>

Two users use their own NFVO and VNFM but allocates resources on the same NFVI-PoP(s) (NFV Infrastructure Point of Presence) to build their NSs (Network Slices). Sub attacks:

-This malicious NFVO or VNFM uses for the instantiation of the VNF the resourceGroupId of the legitimate tenant. This VNF sharing the domain of other VNFs of the legitimate tenant may access to the resources of the legitimate VNFs.

- A malicious NFVO or VNFM request reservation of resources (compute, storage, network) using Or-Vi or Vnfm-Vi interface respectively, to jeopardize the NFVI resources available

- A malicious NFVO uses the Or-Vi software image interface on behalf of a legitimate tenant authorized to access the VIM, to add, delete, update or to query information of Software images from the VIM

- A malicious VNFM uses the Vnfm-Vi software image interface on behalf of a legitimate tenant authorized to access the VIM, to query information of Software images from the VIM

- VNF instantiated in the NFVI accesses to the resources reserved for another tenant and not sharable

- A malicious tenant accesses in the WIM to data transiting between another tenant’s two VNF instances instantiated in two different NFVI-PoPs