****

**Title**

**Automated Security Audit For Container Based Cisco Virtualization Platforms**

**Session ID**

**DEVWKS-1428**

**Speakers:**

**Abbas Abidi**

**Asifiqbal Pathan**

**Chocks Ramiah**

# Learning Objectives

Upon completion of this lab, you will be able to:

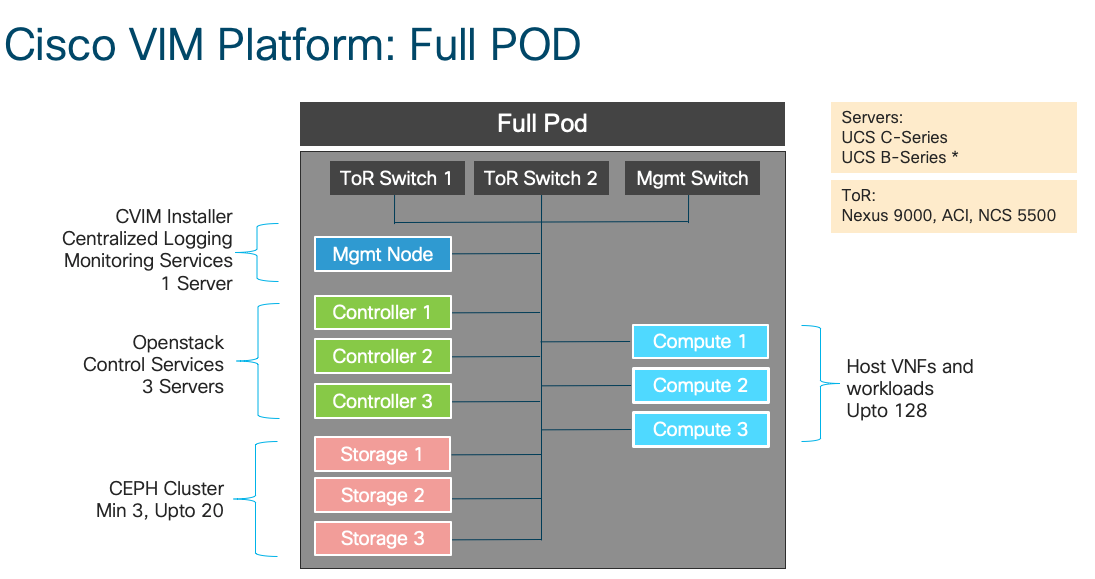
* Understand CVIM Platform
* OpenStack (Queens) running as containers
* CIS Benchmark for Docker containers
* Validating CIS Benchmark with Docker commands

# Scenario

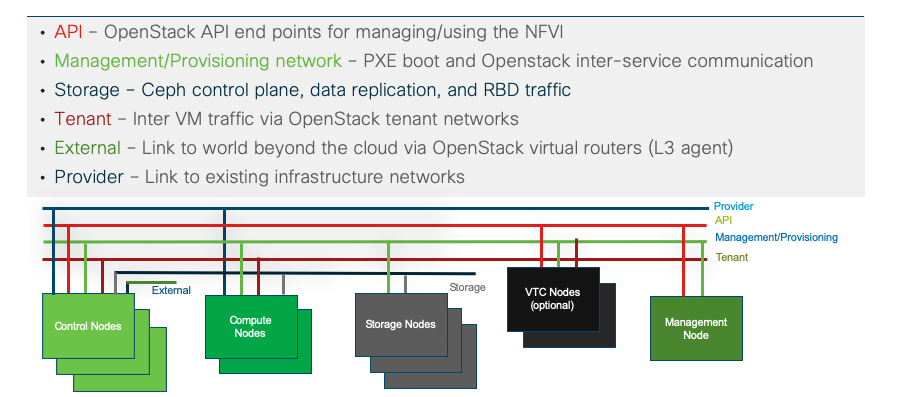
In this lab activity, you will learn how to access Cisco Virtual Infrastructure Manager(CVIM) and validate the OpenStack platform provided in CVIM.

OpenStack Platform

# Network Diagram



Segregated Networks :



# Task 1: Access CVIM Management Node via Jump Server

Use the information given below to access the CVIM management node using jump server.

# Step 1: Jump server information for lab users

Use the IP address assigned to you (CLUSER1 to CLUSER8) in the table above.

Note: Please note the session date and time!

June 12, 2019 04:00PM PDT

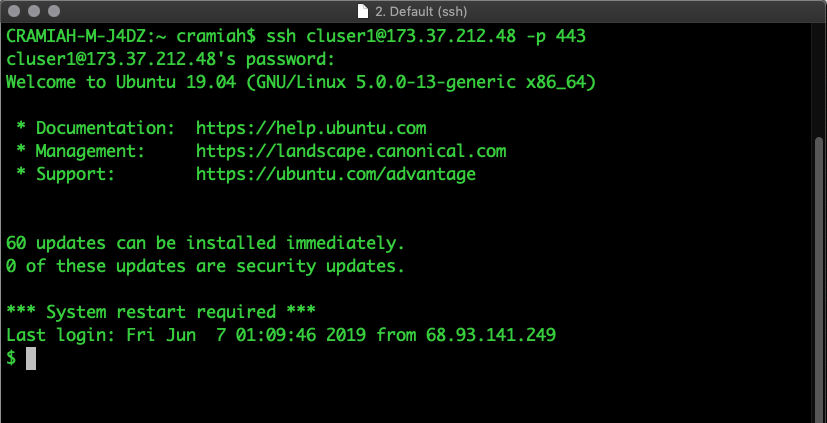
|  |  |  |  |
| --- | --- | --- | --- |
| Jump Server | Username | Password | User (Session 1) |
| 173.37.212.48 | cluser1 | cluser1 | Yasser |
| 173.37.212.48 | cluser2 | cluser2 | Michael |
| 173.37.212.48 | cluser3 | cluser3 | Joshua |
| 173.37.212.48 | cluser4 | cluser4 | Anthony |
| 173.37.212.48 | cluser5 | cluser5 | Dana |
| 173.37.212.48 | cluser6 | cluser6 | Richard |
| 173.37.212.48 | cluser7 | cluser7 | Chris |
| 173.37.212.48 | cluser8 | cluser8 | Yu |

June 13, 2019 01:00PM PDT

|  |  |  |  |
| --- | --- | --- | --- |
| Jump Server | Username | Password | User (Session 2) |
| 173.37.212.48 | cluser1 | cluser1 | Virginia |
| 173.37.212.48 | cluser2 | cluser2 | James |
| 173.37.212.48 | cluser3 | cluser3 | Nick |
| 173.37.212.48 | cluser4 | cluser4 | Seng Wei |
| 173.37.212.48 | cluser5 | cluser5 | Cindy |
| 173.37.212.48 | cluser6 | cluser6 | Tim |
| 173.37.212.48 | cluser7 | cluser7 | Ed |
| 173.37.212.48 | cluser8 | cluser8 | Jeremy |

Command to use: From your terminal(Mac) or cmd (Windows) do

***“ssh cluser<Number>@173.37.212.48 -p 443”***



# Step 2: Access Management node of CVIM

Use the IP address assigned to you (CLUSER1 to CLUSER8) in the table below.

Note: The following command should be executed from the jump server you logged in the step 1.

Command to use:

From your terminal(Mac) or cmd (Windows) do

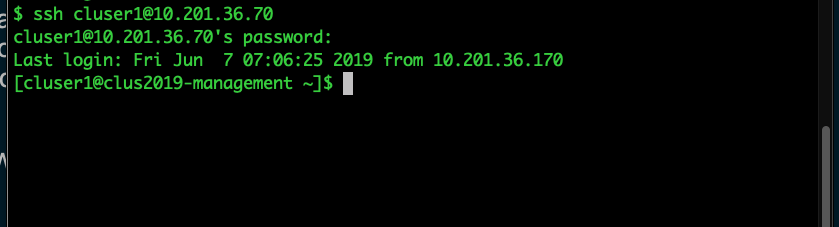
***“ssh cluser<Number>@10.201.36.70”***

June 12, 2019 04:00PM PDT

|  |  |  |  |
| --- | --- | --- | --- |
| Mgmt Node | Username | Password | User (Session 1) |
| 10.201.36.70 | cluser1 | cluser1 | Yasser |
| 10.201.36.70 | cluser2 | cluser2 | Michael |
| 10.201.36.70 | cluser3 | cluser3 | Joshua |
| 10.201.36.70 | cluser4 | cluser4 | Anthony |
| 10.201.36.70 | cluser5 | cluser5 | Dana |
| 10.201.36.70 | cluser6 | cluser6 | Richard |
| 10.201.36.70 | cluser7 | cluser7 | Chris |
| 10.201.36.70 | cluser8 | cluser8 | Yu |

June 13, 2019 01:00PM PDT

|  |  |  |  |
| --- | --- | --- | --- |
| Mgmt Node | Username | Password | User (Session 2) |
| 10.201.36.70 | cluser1 | cluser1 | Virginia |
| 10.201.36.70 | cluser2 | cluser2 | James |
| 10.201.36.70 | cluser3 | cluser3 | Nick |
| 10.201.36.70 | cluser4 | cluser4 | Seng Wei |
| 10.201.36.70 | cluser5 | cluser5 | Cindy |
| 10.201.36.70 | cluser6 | cluser6 | Tim |
| 10.201.36.70 | cluser7 | cluser7 | Ed |
| 173.37.212.48 | cluser8 | cluser8 | Jeremy |



# Step 3: Set the environment for OpenStack API

**Objective:**

Introduce Cisco VIM (virtualization platform) and see if you can get in and perform basic commands in OpenStack.

Execute the command:

**source ~/setlab.sh**

**clus**

**cl\_os**

**Sample:**

**[cluser1@clus2019-management ~]$ source ~/setlab.sh**

**[cluser1@clus2019-management ~]$ clus**

**[cluser1@clus2019-management docker-bench-security]$ cl\_os**

|  |
| --- |
| **Commands with expected output** |
| **COMMAND: 3.1 – Get the servers running in the CVIM pod using openstack API**  openstack server list  **OUTPUT: 3.1**  [cluser1@clus2019-management docker-bench-security]$ openstack server list  +--------------------------------------+-------------+--------+---------------------------------------------+------------------+---------------+  | ID | Name | Status | Networks | Image | Flavor |  +--------------------------------------+-------------+--------+---------------------------------------------+------------------+---------------+  | 0b9a7ab4-3bed-4c92-811f-0a9cd53906a2 | clus-1-vm1 | ACTIVE | net-clus-1=192.167.1.9, 10.201.36.76 | centos-image | centos-flavor |  | c1542989-4e25-4c75-95b0-7a0a4facccad | TestClient1 | ACTIVE | pns-internal-net=192.168.1.6 | RHEL-guest-image | vmtp |  | 6d9b357e-13df-46ae-bbd6-03d5bfa456da | TestServer1 | ACTIVE | pns-internal-net=192.168.1.5, 10.201.34.208 | RHEL-guest-image | vmtp |  +--------------------------------------+-------------+--------+---------------------------------------------+------------------+---------------+ |
| **COMMAND: 3.2: Get the docker images running in the management node of CVIM**  sudo docker ps --format "table {{.ID}}\\t{{.Names}}\\t{{.Image}}\\t\\t{{.Status}}"  **OUTPUT: 3.2**  [cluser1@clus2019-management ~]$ sudo docker ps --format "table {{.ID}}\\t{{.Names}}\\t{{.Image}}\\t\\t{{.Status}}"  CONTAINER ID NAMES IMAGE STATUS  54edf2c0a9db vmtp\_16550 192.168.20.105:5000/cvim-rhel7-osp13/vmtp:16550 Up 10 days  5d6dff207405 vimconfig\_16550 192.168.20.105:5000/cvim-rhel7-osp13/vim-config:16550 Up 10 days  ee766b9cfb53 fluentd\_aggr\_16550 192.168.20.105:5000/cvim-rhel7-osp13/fluentd-aggr:16550 Up 10 days  9105c99fabc1 curator\_16550 192.168.20.105:5000/cvim-rhel7-osp13/curator:16550 Up 10 days  3a1da822ed3d kibana\_16550 192.168.20.105:5000/cvim-rhel7-osp13/kibana:16550 Up 10 days  bb395e4f8a72 elasticsearch\_16550 192.168.20.105:5000/cvim-rhel7-osp13/elasticsearch:16550 Up 10 days  c399a64cc99c tftp\_server\_16550 192.168.20.105:5000/cvim-rhel7-osp13/dockbler-tftp:16550 Up 10 days  b911873bf263 my\_cobbler\_16550 192.168.20.105:5000/cvim-rhel7-osp13/dockbler-app:16550 Up 10 days  0d96613cc5c3 repo\_mirror\_16550 192.168.20.105:5000/cvim-rhel7-osp13/dockbler-web:16550 Up 10 days  13362f5a77a0 container\_registry cloud-docker.cisco.com:8443/redstone/registry-2.6.2:16550 Up 10 days  **COMMAND: 3.3 [Optional] – Get containers inspected in the compute/storage nodes of OpenStack**  **OUTPUT:3.3**  **cat ~/mercury\_server\_info**  **Find the storage nodes from the output and try ssh to one of the management IP address shown below**  **User name: root**  **Password:C1sc0123**      **ssh 192.168.20.148 -l root**  **do simple commands like “dp”**  **dp is an alias for “alias dp='docker ps -a --format "table {{.Names}} {{.Status}}"'** |

# Step 4: CVIM Lab – Best Practices of Containers – Validate CVIM

Execute the commands in your CVIM Management POD.

|  |
| --- |
| **Commands with Expected Output** |
| **COMMAND: 4.1 – Confirm the experimental features are disabled (set to Falise)**  sudo docker version --format '{{ .Server.Experimental }}'  **OUTPUT: 4.1**  false |
| **COMMAND:4.2 - Get the docker service path to get permissions**  sudo systemctl show -p FragmentPath docker.service  **OUTPUT: 4.2**  FragmentPath=/usr/lib/systemd/system/docker.service |
| **COMMAND: 4.3 – Use the docker service path and get permissions**  sudo stat -c "%a %n" /usr/lib/systemd/system/docker.service  **OUTPUT: 4.3**  644 /usr/lib/systemd/system/docker.service |
| **COMMAND: 4.4: Get the docker owner and group**  stat -c %U:%G /etc/docker  **OUTPUT: 4.4**  root:root |
| **COMMAND: 4.5: Get the docker binary mode permissions**  stat -c "%a %n" /etc/docker/  **OUTPUT: 4.5**  755 /etc/docker/ |
| **COMMAND: 4.6 – Get the list of Docker images**  sudo docker images  **OUTPUT: 4.6**  [cluser1@clus2019-management ~]$ sudo docker images  REPOSITORY TAG IMAGE ID CREATED SIZE  dockbler-keys latest 2fd49f643c91 7 months ago 788 MB  192.168.20.105:5000/cvim-rhel7-osp13/elasticsearch 16550 752dafd8e7a3 7 months ago 895 MB  192.168.20.105:5000/cvim-rhel7-osp13/vmtp 16550 5c64b3f8dbd2 7 months ago 1.93 GB  192.168.20.105:5000/cvim-rhel7-osp13/vim-config 16550 0c37f97a1467 7 months ago 1.44 GB  192.168.20.105:5000/cvim-rhel7-osp13/curator 16550 a2ff0481287a 7 months ago 812 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-web 16550 bf5efff2dfae 7 months ago 788 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-optional-rpms 16550 a61246a8435f 7 months ago 6.21 GB  192.168.20.105:5000/cvim-rhel7-osp13/fluentd-aggr 16550 2f9125458b59 7 months ago 918 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-openstack-13-devtools-rpms 16550 8878156f48b0 7 months ago 836 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-app 16550 bd73674074ee 7 months ago 789 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-mercury-cloudpulse-rpms 16550 3db4bb0862db 7 months ago 836 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-mercury-buildnode-rpms 16550 4ec0c31ab262 7 months ago 840 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-rpms 16550 b3c432de09a9 7 months ago 5.91 GB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-openstack-13-optools-rpms 16550 e74ec179f723 7 months ago 843 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-extras-rpms 16550 e65cd0472bc3 7 months ago 1.09 GB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-rh-common-rpms 16550 97fceb0447b3 7 months ago 1.33 GB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-mercury-thirdparty-hw-binary-utilities-rpms 16550 b3e1848f8e51 7 months ago 840 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-openstack-13-rpms 16550 eac86e66fe18 7 months ago 4.03 GB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-rhceph-3-osd-rpms 16550 8381a16303cd 7 months ago 916 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-7-server-openstack-13-tools-rpms 16550 b81e428a5ed7 7 months ago 853 MB  192.168.20.105:5000/cvim-rhel7-osp13/kibana 16550 f4b91eead188 7 months ago 1.17 GB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-cisco-rhel-server-7-openstack-13-hotfix-rpms 16550 f2807b2cab58 7 months ago 841 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-ha-for-rhel-7-server-rpms 16550 40e762218861 7 months ago 1.17 GB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-mercury-common-rpms 16550 5cc8701ba677 7 months ago 843 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-tftp 16550 cec4be197d0d 7 months ago 788 MB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-rhel-boot 16550 1441d6414a1a 7 months ago 1.47 GB  192.168.20.105:5000/cvim-rhel7-osp13/dockbler-repofiles 16550 7d93cb352fbd 7 months ago 788 MB  cloud-docker.cisco.com:8443/redstone/registry-2.6.2 16550 a07e3f32a779 19 months ago 33.3 MB |
| **COMMAND**: **4.7 – Get the list of docker images filtered by configured ones**  sudo docker images --quiet | xargs sudo docker inspect --format '{{ .Id }}:Image={{ .Config.Image }}'  **OUTPUT: 4.7**  [cluser1@clus2019-management docker-bench-security]$ sudo docker images --quiet | xargs sudo docker inspect --format '{{ .Id }}:Image={{ .Config.Image }}'  sha256:2fd49f643c9123de8305a42c53f06df3a7a3e7f6e9aa476d237521c2ea55ebe6 : Image=sha256:cbc568dfccd7a2934701a1fb3b8106360ab2c3da88c3f51decc2510332dad2cf  sha256:752dafd8e7a31ae0b7ce24d693fd0048e993d2b8d721d959019d2cf5614fb576 : Image=sha256:b56b7bace290f3a6d4997be6aed004cb210b77a89fb93068a67ec8c5a797a6dc  sha256:5c64b3f8dbd287d0fe58dcd8c59d4f0ab68719d9ac10e5228b0311b251e57c7f : Image=sha256:040e52257161f170fbc012b6fec032d13633305e31594d15e418307fe1614feb  sha256:0c37f97a14679eba807c6c7c0a0a6a816fac8de8188b359a419af922b5ae5d82 : Image=sha256:d09c31c592969b39098d78994e9c1adfdd66ad311107e268c4a61c4e23a13b91  sha256:a2ff0481287a9a10406c31d01ae2a4a2df7d997a815b7211767ed28aadfbe92b : Image=sha256:393214748a588f4566125038624f4545202f4d1d49514d82a2d631f8590e9fbb  sha256:bf5efff2dfaeacd5adfc18ab4980c9255065dbf52cd2dc578d689c5793ec38e9 : Image=sha256:13a1ffb6bebe940bc809003a0d500131f9ae00890719b8d4f446d7ed6150f76a  sha256:a61246a8435fe9540d4274ed801fcf5bb442ccefddc7df19bca80aea6c46f454 : Image=sha256:4bc4b92592389a18a1640d45af249ae83094686ae85e3d72b6570dba8ea6494d  sha256:2f9125458b5963bd90065bc90906358765387453cda25682f683393f04e875ad : Image=sha256:0c64c88cd74aa63c34596cb3f862b97db3898f2dc2a3f18bac04e11d0ba655a4  sha256:8878156f48b08bc027cebcaec7ebbf1d22f44af72dbb8dd40503d737ec54066b : Image=sha256:4ee38232698f0b0af2b09070ba46f706ee1b45eaadb5236739db740f45e3fe6e  sha256:bd73674074eeb710e8a1c2d4cf7a0babf1a7f71430d888e78545a1ebeeee70af : Image=sha256:745f26bf7c3874e31ccddbedea7b893688f08ed6c93194c30d2c4b95fd17b4f4  sha256:3db4bb0862db85a02d2ffa4d2ffbc836feed2774a86f8a31b466c1413d7dce52 : Image=sha256:a22462e6ecd0f59bc2503c0c8a875c1c103c4eb8458c24540e655eee050449c1  sha256:4ec0c31ab262c49b31bc388d8fd7829f4c885194536d06a698f81364d15327e3 : Image=sha256:774fcb0167927b628c9b47da8dfc646f061c9ca1b1adc9d5b389c44decce1b46  sha256:b3c432de09a984dcb6b2e173b680850fa8eaa802d40be6992b842c6633de2570 : Image=sha256:50d795e917a9d1560f1c6f12fa74a4d09a952427bed7f8dd7b33c317c7846ae2  sha256:e74ec179f723c3ac2b93825b7db3a1cc8c82f5b572d40edd9c0ed28503b87f1f : Image=sha256:e01c68bcadaaf96d7acc40c689988d7fe7ea7fcd58015fa8ced251778034de28  sha256:e65cd0472bc3a5264c7b83d08e9e8adfe4f6e8f3564ed0af6f5bb57a79681dee : Image=sha256:75630a04be6011161d690cb74cb6a514a1fb350e60d18217b5ab2a0c578e104a  sha256:97fceb0447b3394974f950789efa36666baf37fc1c219472c31038cc645f3aa2 : Image=sha256:d59178bad6eb564b25eb43421cc397b529816f7f2b8482298b3e50ea9bd59612  sha256:b3e1848f8e51fc32cd716e5947d54d57fd9c0b562eadbee04d26a75a7b1a4943 : Image=sha256:9ff7835ac8bf91f2d743c16a9a5560df43e43c04af17381cf0fd0ea3f074e389  sha256:eac86e66fe18b6cc972183718ef5742da65f64b3a547847c6b29d99046da328d : Image=sha256:ff76078b52aee8512d0ff7fe68d32c3c82989c3125451c2b22c3cdfdefb26b3b  sha256:8381a16303cd2fe953ae71a8369c2830aa8193baf9098f01ee576d9d7264dc40 : Image=sha256:9b7b6c36abd0776f2599a2ac262479c2d2cbb8972f4db2210ce9b88640376e6d  sha256:b81e428a5ed7b958e8ef9278d8b242a5fc82f0c580933cf5e197191150134ae5 : Image=sha256:5e13983be3b07b5634b68121c25bbf37e55ed5a1a28496583ff155b0b0068528  sha256:f4b91eead1884f50c3cd559caafaa9b7bc59e512de4b36e840cf8d5ec8fb25c6 : Image=sha256:d7808c6af3deba12249df66e2d06af64e29f7e72615b96ad12248099aec15fbe  sha256:f2807b2cab5803b10fc942ac13830d7cf1091196c0cdf4e65bbc6b24e4fa60aa : Image=sha256:26a1bac086356a3bfd97f2456280fbc8e2deaf2cfd9a36503a511e9944fc2dc8  sha256:40e7622188615bbb7505e121c8230d3517cfab1b08ad5fcbc07d28f7636f0532 : Image=sha256:8a8755ec1916228fccaa7e863a03be9d3709796b4a9f505dfa3d7212f06b878c  sha256:5cc8701ba6778b4b592064c3e2c4376150e5a68bb746535f7dfc281ad5d75edb : Image=sha256:db7fa000c7a9629e0d60752200f8230a7de0c21d2cd95f8be1fbd83920a9f03f  sha256:cec4be197d0dc85d09c8d44d11c6fd565d080b47eeb327e0a052bade637c8d8f : Image=sha256:401a47181f211142dff14bb22bb43321ed98b5ab0b2dbc189ba649a6651adbf2  sha256:1441d6414a1ab6c83b9607f66434d2ee098d2af9986cdc0085efcba79db6e370 : Image=sha256:15157746395237b289d7e9590f325fbecf94f78215f362d5b678fe3b57cf1b6f  sha256:7d93cb352fbd911d3bc4b2c1ca20b503f1053733f65faa5f828c73ad750bbe72 : Image=sha256:5a275c90303af899908488d825124e3e272702c8573d400eee8f2318c002a424  sha256:a07e3f32a779aa924fd47f6797d4d5c93061c50c0eb97d464f08365a3a30200b : Image=sha256:a827c74403d090fedc4c3e00f2e3759fe3761fb3fa05b439ec533d5aebd3cc39 |
| **COMMAND: 4.8 – Get the images count based on the configured images**  sudo docker images --quiet | xargs sudo docker inspect --format '{{ .Id }}:Image={{ .Config.Image }}' | wc -l  **OUTPUT: 4.8**  28 |
| **COMMAND: 4.8 – Get the image count**  sudo docker images | wc -l  **OUTPUT: 4.8**  29 |
| **COMMAND: 4.9 - Get the images count except dockbler image used internally by CVIM**  sudo docker images | grep -v dockbler-keys | wc -l  **OUTPUT: 4.9**  28 |
| **COMMAND: 4.10 – Get the Docker containers count**  sudo docker info --format '{{ .Containers }}'  **OUTPUT:4.10**  28 |
| **COMMAND: 4.11- Get the count of docker images in stopped state**  sudo docker info | grep "Stopped"  **OUTPUT: 4.11**  WARNING: You're not using the default seccomp profile  Stopped: 18 |
| **COMMAND: 4.12 = Get the count of docker images in running state**  sudo docker info | grep "Running"  **OUTPUT: 4.12**  WARNING: You're not using the default seccomp profile  Running: 10 |
| **COMMAND: 4.13 – Get the Docker information showing all the attributes**  sudo docker info | more  **OUTPUT: 4.13**  WARNING: You're not using the default seccomp profile  **Containers: 28**  **Running: 10**  **Paused: 0**  **Stopped: 18**  Images: 30  Server Version: 1.13.1  Storage Driver: overlay2  Backing Filesystem: xfs  Supports d\_type: true  Native Overlay Diff: true  Logging Driver: journald  Cgroup Driver: cgroupfs  Plugins:  Volume: local  Network: bridge host macvlan null overlay  Authorization: rhel-push-plugin  Swarm: inactive  Runtimes: docker-runc runc  Default Runtime: docker-runc  Init Binary: /usr/libexec/docker/docker-init-current  containerd version: (expected: aa8187dbd3b7ad67d8e5e3a15115d3eef43a7ed1)  runc version: 5eda6f6fd0c2884c2c8e78a6e7119e8d0ecedb77 (expected: 9df8b306d01f59d3a8029be411de015b7304dd8f)  init version: fec3683b971d9c3ef73f284f176672c44b448662 (expected: 949e6facb77383876aeff8a6944dde66b3089574)  **Security Options:**  **seccomp**  **Profile: /etc/docker/seccomp.json**  **selinux**  **Kernel Version: 3.10.0-862.11.6.el7.x86\_64**  Operating System: OpenStack  OSType: linux  Architecture: x86\_64  Number of Docker Hooks: 2  CPUs: 48  Total Memory: 251.7 GiB  Name: clus2019-management  ID: MVMZ:3TCQ:X5PR:MQ3D:AQHN:NLL7:7NSB:PFQE:7ZMU:AY2M:4NZH:PIR7  Docker Root Dir: /var/lib/docker  Debug Mode (client): false  Debug Mode (server): false  Registry: https://cloud-docker.cisco.com:8443/v1/  **Experimental: false**  Insecure Registries:  192.168.20.105:5000  127.0.0.0/8  Live Restore Enabled: false  Registries: cloud-docker.cisco.com:8443 (secure), registry.access.redhat.com (secure) |
| **COMMAND: 4.14 – Get the Docker version including API version**  sudo docker version  **OUTPUT: 4.14**  Client:  Version: 1.13.1  API version: 1.26  Package version: docker-1.13.1-68.gitdded712.el7.x86\_64  Go version: go1.9.2  Git commit: dded712/1.13.1  Built: Tue Jun 12 18:30:09 2018  OS/Arch: linux/amd64  Server:  Version: 1.13.1  API version: 1.26 (minimum version 1.12)  Package version: docker-1.13.1-68.gitdded712.el7.x86\_64  Go version: go1.9.2  Git commit: dded712/1.13.1  Built: Tue Jun 12 18:30:09 2018  OS/Arch: linux/amd64  Experimental: false |

# Step 5: Docker CIS Benchmark Lab

For this lab, please use the OpenStack management Node & VM provided.

|  |
| --- |
| Commands with expected output |
| **COMMAND: 5.1 Change the directory to execute the benchmark (CIS)**  clus  OR  **COMMAND: 5.2 - Change the directory to execute the benchmark (CIS)**  cd /home/cluser1/clus19-audit/cis\_benchmark/docker-bench-security  **OUTPUT: 5.1 and 5.2**  [cluser1@clus2019-management docker-bench-security]$  OR  [cluser1@clus2019-management docker-bench-security]$ |
| **COMMAND: 5.3 Execute the benchmark (CIS) in the Management Node of CVIM**  sudo sh docker-bench-security.sh  **OUTPUT: 5.3**  [cluser1@clus2019-management docker-bench-security]$ sudo sh docker-bench-security.sh  # ------------------------------------------------------------------------------  # Docker Bench for Security v1.3.4  #  # Docker, Inc. (c) 2015-  #  # Checks for dozens of common best-practices around deploying Docker containers in production.  # Inspired by the CIS Docker Community Edition Benchmark v1.1.0.  # ------------------------------------------------------------------------------  Initializing Sat Jun 8 02:00:09 UTC 2019  ***\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Section 1 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\****  [INFO] 1 - Host Configuration  [WARN] 1.1 - Ensure a separate partition for containers has been created  [NOTE] 1.2 - Ensure the container host has been Hardened  [INFO] 1.3 - Ensure Docker is up to date  [INFO] \* Using 1.13.1, verify is it up to date as deemed necessary  <snip>  ***\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Section 2 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\****  [INFO] 2 - Docker daemon configuration  [WARN] 2.1 - Ensure network traffic is restricted between containers on the default bridge  **[PASS] 2.2 - Ensure the logging level is set to 'info'**  **[PASS] 2.3 - Ensure Docker is allowed to make changes to iptables**  [WARN] 2.4 - Ensure insecure registries are not used  **[PASS] 2.5 - Ensure aufs storage driver is not used**  <snip>  **[PASS] 2.17 - Ensure experimental features are avoided in production**  <snip>  ***\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Section 3 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\****  [INFO] 3 - Docker daemon configuration files  [**PASS] 3.1 - Ensure that docker.service file ownership is set to root:root**  **[PASS] 3.2 - Ensure that docker.service file permissions are set to 644 or more restrictive**  [INFO] 3.3 - Ensure that docker.socket file ownership is set to root:root  [INFO] \* File not found  <snip>  ***\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Section 4 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\****  [INFO] 4 - Container Images and Build File  [WARN] 4.1 - Ensure a user for the container has been created  [WARN] \* Running as root: container\_registry  [NOTE] 4.2 - Ensure that containers use trusted base images  [NOTE] 4.3 - Ensure unnecessary packages are not installed in the container  [NOTE] 4.4 - Ensure images are scanned and rebuilt to include security patches  <snip>  ***\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Section 5 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\****  [INFO] 5 - Container Runtime  [WARN] 5.1 - Ensure AppArmor Profile is Enabled  [WARN] \* No AppArmorProfile Found: vmtp\_16550  **<snip>**  **[PASS] 5.3 - Ensure Linux Kernel Capabilities are restricted within containers**  [WARN] 5.4 - Ensure privileged containers are not used  [WARN] \* Container running in Privileged mode: vmtp\_16550  [**PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers**  **[PASS] 5.6 - Ensure ssh is not run within containers**  **[PASS] 5.7 - Ensure privileged ports are not mapped within containers**  [NOTE] 5.8 - Ensure only needed ports are open on the container  [WARN] \* MaximumRetryCount is not set to 5: container\_registry  [**PASS] 5.15 - Ensure the host's process namespace is not shared**  **[PASS] 5.16 - Ensure the host's IPC namespace is not shared**  **[PASS] 5.17 - Ensure host devices are not directly exposed to containers**  [INFO] 5.18 - Ensure the default ulimit is overwritten at runtime, only if needed  [INFO] \* Container no default ulimit override: vmtp\_16550  [**PASS] 5.19 - Ensure mount propagation mode is not set to shared**  **[PASS] 5.20 - Ensure the host's UTS namespace is not shared**  **[PASS] 5.21 - Ensure the default seccomp profile is not Disabled**  **[NOTE] 5.22 - Ensure docker exec commands are not used with privileged option**  **[NOTE] 5.23 - Ensure docker exec commands are not used with user option**  **[PASS] 5.24 - Ensure cgroup usage is confirmed**  **[WARN] 5.25 - Ensure the container is restricted from acquiring additional privileges**  [WARN] \* Privileges not restricted: vmtp\_16550  [WARN] \* Privileges not restricted: vimconfig\_16550  [WARN] \* PIDs limit not set: container\_registry  **[PASS] 5.29 - Ensure Docker's default bridge docker0 is not used**  **[PASS] 5.30 - Ensure the host's user namespaces is not shared**  **[PASS] 5.31 - Ensure the Docker socket is not mounted inside any containers**  **[INFO] 6 - Docker Security Operations**  [INFO] 6.1 - Avoid image sprawl  [INFO] \* There are currently: 28 images  [INFO] 6.2 - Avoid container sprawl  [INFO] \* There are currently a total of 28 containers, with 10 of them currently running  **[INFO] 7 - Docker Swarm Configuration**  **[PASS] 7.1 - Ensure swarm mode is not Enabled, if not needed**  **[PASS] 7.2 - Ensure the minimum number of manager nodes have been created in a swarm (Swarm mode not enabled)**  **[PASS] 7.3 - Ensure swarm services are binded to a specific host interface (Swarm mode not enabled)**  **[PASS] 7.4 - Ensure data exchanged between containers are encrypted on different nodes on the overlay network**  **[PASS] 7.5 - Ensure Docker's secret management commands are used for managing secrets in a Swarm cluster (Swarm mode not enabled)**  [**PASS] 7.6 - Ensure swarm manager is run in auto-lock mode (Swarm mode not enabled)**  **[PASS] 7.7 - Ensure swarm manager auto-lock key is rotated periodically (Swarm mode not enabled)**  **[PASS] 7.8 - Ensure node certificates are rotated as appropriate (Swarm mode not enabled)**  **[PASS] 7.9 - Ensure CA certificates are rotated as appropriate (Swarm mode not enabled)**  **[PASS] 7.10 - Ensure management plane traffic has been separated from data plane traffic (Swarm mode not enabled)**  **[INFO] Checks: 105**  **[INFO] Score: 13** |
| **Compare OpenStack with VM running Docker**  Switch to the VM with IP 10.201.36.76  **COMMANDS:5.4 – Log into the VM running in CVIM management node**  ssh <username>@10.201.36.76  cd sec\_audit/cisc\_benchmark/docker-bench-security  **COMMAND:5.5**  **Execute the benchmark (CIS)**  sudo sh docker-bench-security.sh  **OUTPUT: 5.5**  [cluser1@clus-1-vm1 docker-bench-security]$ sudo sh docker-bench-security.sh  We trust you have received the usual lecture from the local System  Administrator. It usually boils down to these three things:  #1) Respect the privacy of others.  #2) Think before you type.  #3) With great power comes great responsibility.  [sudo] password for cluser1:  # ------------------------------------------------------------------------------  # Docker Bench for Security v1.3.4  #  # Docker, Inc. (c) 2015-  #  # Checks for dozens of common best-practices around deploying Docker containers in production.  # Inspired by the CIS Docker Community Edition Benchmark v1.1.0.  # ------------------------------------------------------------------------------  Initializing Sat Jun 8 14:24:51 UTC 2019  [INFO] 1 - Host Configuration  [WARN] 1.1 - Ensure a separate partition for containers has been created  [NOTE] 1.2 - Ensure the container host has been Hardened  <snip>  **[INFO] 2 - Docker daemon configuration**  [WARN] 2.1 - Ensure network traffic is restricted between containers on the default bridge  **[PASS] 2.2 - Ensure the logging level is set to 'info'**  **[PASS] 2.3 - Ensure Docker is allowed to make changes to iptables**  **[PASS] 2.4 - Ensure insecure registries are not used**  **[PASS] 2.5 - Ensure aufs storage driver is not used**  [INFO] 2.6 - Ensure TLS authentication for Docker daemon is configured  [INFO] \* Docker daemon not listening on TCP  [INFO] 2.7 - Ensure the default ulimit is configured appropriately  [INFO] \* Default ulimit doesn't appear to be set  [WARN] 2.8 - Enable user namespace support  **[PASS] 2.9 - Ensure the default cgroup usage has been confirmed**  **[PASS] 2.10 - Ensure base device size is not changed until needed**  [WARN] 2.11 - Ensure that authorization for Docker client commands is enabled  **[PASS] 2.12 - Ensure centralized and remote logging is configured**  [WARN] 2.13 - Ensure operations on legacy registry (v1) are Disabled  [WARN] 2.14 - Ensure live restore is Enabled  [WARN] 2.15 - Ensure Userland Proxy is Disabled  [INFO] 2.16 - Ensure daemon-wide custom seccomp profile is applied, if needed  **[PASS] 2.17 - Ensure experimental features are avoided in production**  [WARN] 2.18 - Ensure containers are restricted from acquiring new privileges  [INFO] 3 - Docker daemon configuration files  **[PASS] 3.1 - Ensure that docker.service file ownership is set to root:root**  **[PASS] 3.2 - Ensure that docker.service file permissions are set to 644 or more restrictive**  [INFO] 3.3 - Ensure that docker.socket file ownership is set to root:root  [INFO] \* File not found  [INFO] 3.4 - Ensure that docker.socket file permissions are set to 644 or more restrictive  [INFO] \* File not found  **[PASS] 3.5 - Ensure that /etc/docker directory ownership is set to root:root**  **[PASS] 3.6 - Ensure that /etc/docker directory permissions are set to 755 or more restrictive**  **[PASS] 3.7 - Ensure that registry certificate file ownership is set to root:root**  **[PASS] 3.8 - Ensure that registry certificate file permissions are set to 444 or more restrictive**  [INFO] 3.9 - Ensure that TLS CA certificate file ownership is set to root:root  [INFO] \* No TLS CA certificate found  [INFO] 3.10 - Ensure that TLS CA certificate file permissions are set to 444 or more restrictive  [INFO] \* No TLS Key found  **[PASS] 3.15 - Ensure that Docker socket file ownership is set to root:docker**  **[PASS] 3.16 - Ensure that Docker socket file permissions are set to 660 or more restrictive**  **[PASS] 3.17 - Ensure that daemon.json file ownership is set to root:root**  **[PASS] 3.18 - Ensure that daemon.json file permissions are set to 644 or more restrictive**  [INFO] 3.19 - Ensure that /etc/default/docker file ownership is set to root:root  [INFO] \* File not found  [INFO] 3.20 - Ensure that /etc/default/docker file permissions are set to 644 or more restrictive  [INFO] \* File not found  [INFO] 4 - Container Images and Build File  [WARN] 4.1 - Ensure a user for the container has been created  **[PASS] 4.7 - Ensure update instructions are not use alone in the Dockerfile**  [NOTE] 4.8 - Ensure setuid and setgid permissions are removed in the images  <snip>  **[INFO] 5 - Container Runtime**  [WARN] 5.1 - Ensure AppArmor Profile is Enabled  [WARN] \* No AppArmorProfile Found: infallible\_albattani  [WARN] \* No AppArmorProfile Found: romantic\_noether  [WARN] 5.2 - Ensure SELinux security options are set, if applicable  [WARN] \* No SecurityOptions Found: infallible\_albattani  [WARN] \* No SecurityOptions Found: romantic\_noether  **[PASS] 5.3 - Ensure Linux Kernel Capabilities are restricted within containers**  **[PASS] 5.4 - Ensure privileged containers are not used**  **[PASS] 5.5 - Ensure sensitive host system directories are not mounted on containers**  **[PASS] 5.6 - Ensure ssh is not run within containers**  **[PASS] 5.7 - Ensure privileged ports are not mapped within containers**  [NOTE] 5.8 - Ensure only needed ports are open on the container  [PASS] 5.9 - Ensure the host's network namespace is not shared  [WARN] \* Container running with root FS mounted R/W: romantic\_noether  **[PASS] 5.13 - Ensure incoming container traffic is binded to a specific host interface**  [WARN] 5.14 - Ensure 'on-failure' container restart policy is set to '5'  [WARN] \* MaximumRetryCount is not set to 5: infallible\_albattani  [WARN] \* MaximumRetryCount is not set to 5: romantic\_noether  **[PASS] 5.15 - Ensure the host's process namespace is not shared**  **[PASS] 5.16 - Ensure the host's IPC namespace is not shared**  [PASS] 5.17 - Ensure host devices are not directly exposed to containers  <snip>  [INFO] 6 - Docker Security Operations  [INFO] 6.1 - Avoid image sprawl  [INFO] \* There are currently: 4 images  [INFO] 6.2 - Avoid container sprawl  [INFO] \* There are currently a total of 21 containers, with 2 of them currently running  [INFO] 7 - Docker Swarm Configuration  [**PASS] 7.1 - Ensure swarm mode is not Enabled, if not needed**  **[PASS] 7.2 - Ensure the minimum number of manager nodes have been created in a swarm (Swarm mode not enabled)**  **[PASS] 7.3 - Ensure swarm services are binded to a specific host interface (Swarm mode not enabled)**  **[PASS] 7.4 - Ensure data exchanged between containers are encrypted on different nodes on the overlay network**  **[PASS] 7.5 - Ensure Docker's secret management commands are used for managing secrets in a Swarm cluster (Swarm mode not enabled)**  **[PASS] 7.6 - Ensure swarm manager is run in auto-lock mode (Swarm mode not enabled)**  **[PASS] 7.7 - Ensure swarm manager auto-lock key is rotated periodically (Swarm mode not enabled)**  **[PASS] 7.8 - Ensure node certificates are rotated as appropriate (Swarm mode not enabled)**  **[PASS] 7.9 - Ensure CA certificates are rotated as appropriate (Swarm mode not enabled)**  **[PASS] 7.10 - Ensure management plane traffic has been separated from data plane traffic (Swarm mode not enabled)**  [INFO] Checks: 105  [INFO] Score: 17 |