Smart Data Transaction for CPS R7 Test Documentation

Introduction

This document describes the blueprint test environment for the Smart Data Transaction for CPS blueprint. The test results and logs are posted in the Akraino Nexus at the link below:

https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7
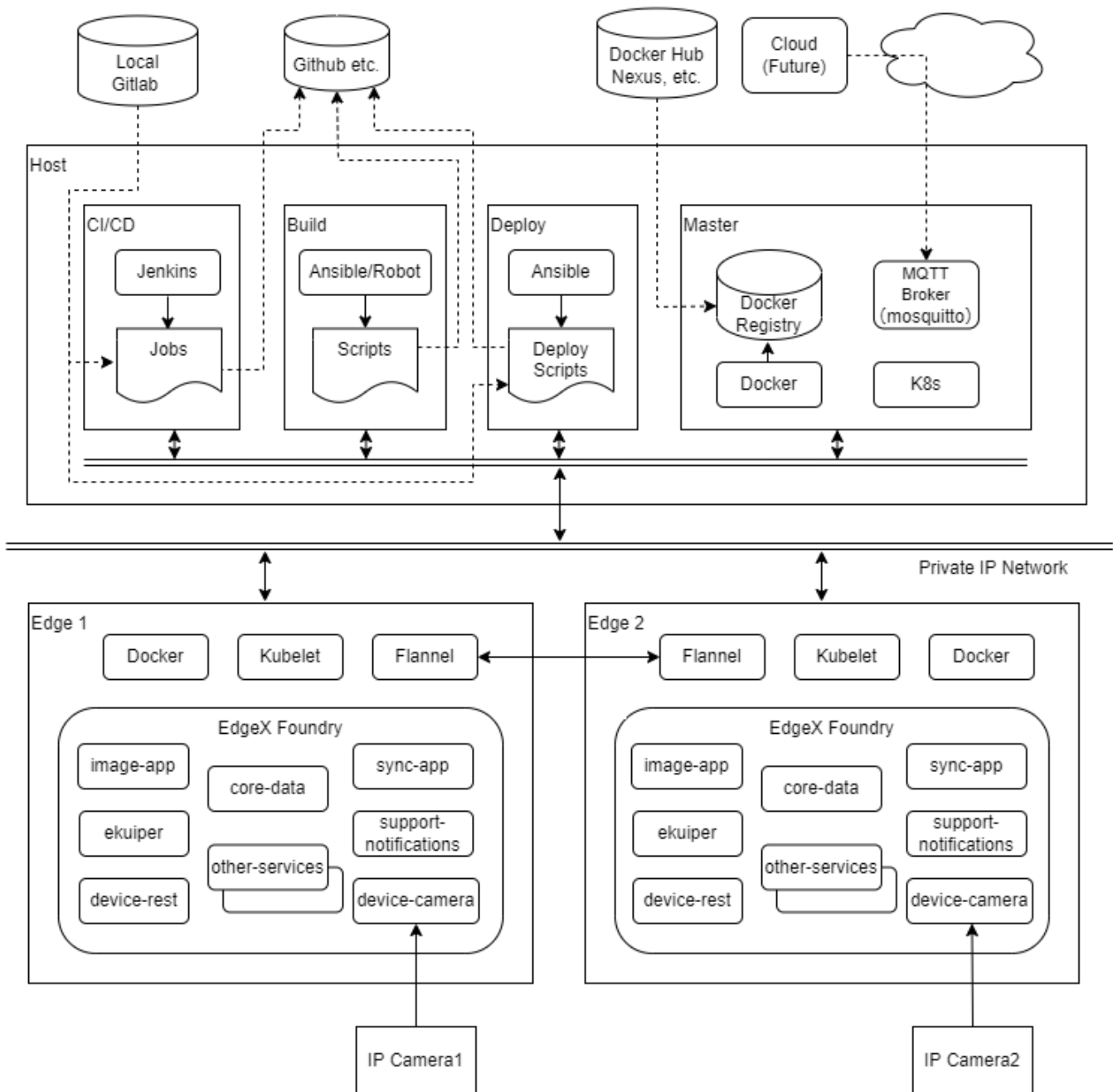
Akarino Test Group Information

N/A

Testing has been carried out at Fujitsu Limited labs without any Akraino Test Working Group resources.

Overall Test Architecture

Tests are carried out on the architecture shown in the diagram below.

Test Bed

The test bed consists of 4 VMs running on x86 hardware, performing deploy and ci/cd and build and master node roles, two edge nodes on ARM64 (Jetson Nano) hardware, and two sensor nodes on ARM32 (Raspberry Pi) hardware.

Node Type	Count	Hardware	OS
CI/CD	1	Intel i5, 2 cores VM	Ubuntu 20.04
Build	1	Intel i5, 2 cores VM	Ubuntu 20.04
Deploy	1	Intel i5, 2 cores VM	Ubuntu 20.04
Master	1	Intel i5, 2 cores VM	Ubuntu 20.04
Edge	2	Jetson Nano, ARM Cortex-A57, 4 cores	Ubuntu 20.04
Camera	2	H.View HV-500E6A	N/A (pre-installed)

The Build VM is used to run the BluVal test framework components outside the system under test.

Test Framework

BluVal and additional tests are carried out using Robot Framework.

Traffic Generator

N/A

Test API description

Before running the tests below, ensure that the configuration in the chapter `Verifying the Setup` of Smart Data Transaction for CPS R7 Installation Guide has been implemented.

CI/CD Regression Tests: Node Setup

This set of test cases confirms the scripting to change the default runtime of edge nodes.

The Test inputs

The test scripts and data are stored in the source repository's `cicd/tests/sdt_step2/install/` directory.

Test Procedure

The test bed is place in a state where all nodes are prepared with required software. No EdgeX or Kubernetes services are running.

Execute the test scripts:

`robot cicd/tests/sdt_step2/install/`

Expected output

The test scripts will change the default runtime of edge nodes from runc to nvidia.

The robot command should report success for all test cases.

Test Results

Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/lfedge-install/14/

Install Report

<div>
Generated
20221104 10:48:17 UTC+08:00
7 minutes 26 seconds ago
</div>

Summary Information

Status:	All tests passed
Start Time:	20221104 10:47:35.019
End Time:	20221104 10:48:17.727
Elapsed Time:	00:00:42.708
Log File:	log.html

Test Statistics

Total Statistics	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests	1	1	0	0	00:00:41	

Statistics by Tag	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
No Tags						

Statistics by Suite	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
Install	1	1	0	0	00:00:43	
Install . Edge Install	1	1	0	0	00:00:43	

Pass (1/1 test case)

CI/CD Regression Tests: Images Build & Push

These test cases verify that the images for EdgeX microservices can be constructed, and pushed to private registry.

The Test inputs

The test scripts and data are stored in the source repository's `cicd/tests/sdt_step2/build/` directory.

Test Procedure

The test bed is placed in a state where all nodes are prepared with required software and the Docker registry is running.

Execute the test scripts:

```
robot cicd/tests/sdt_step2/build/
```

Expected output

The test scripts will build images of changed services(sync-app/image-app/device-camera), add push the images to private registry.

The robot command should report success for all test cases.

Test Results

Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/lfedge-build/5

Pass (2/2 test cases)

CI/CD Regression Tests: Cluster Setup & Teardown

These test cases verify that the Kubernetes cluster can be initialized, edge nodes added to it and removed, and the cluster torn down.

The Test inputs

The test scripts and data are stored in the source repository's `cicd/tests/sdt_step2/cluster/` directory.

Test Procedure

The test bed is placed in a state where all nodes are prepared with required software and the Docker registry is running. The registry must be populated with the Kubernetes and Flannel images from upstream.

Execute the test scripts:

```
robot cicd/tests/sdt_step2/cluster/
```

Expected output

The test scripts will start the cluster, add all configured edge nodes, remove the edge nodes, and reset the cluster.

The robot command should report success for all test cases.

Test Results

Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/lfedge-cluster/6

Cluster Report

Summary Information

Status:	All tests passed
Start Time:	20221104 10:59:32.393
End Time:	20221104 11:02:08.334
Elapsed Time:	00:02:35.941
Log File:	log.html

Test Statistics

Total Statistics	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests	4	4	0	0	00:02:31	

Statistics by Tag	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
No Tags						

Statistics by Suite	Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
Cluster	4	4	0	0	00:02:36	
Cluster . Init Cluster	1	1	0	0	00:00:42	
Cluster . Join Cluster	1	1	0	0	00:00:48	
Cluster . Delete From Cluster	1	1	0	0	00:00:17	
Cluster . Reset Cluster	1	1	0	0	00:00:49	

Pass (4/4 test cases)

CI/CD Regression Tests: EdgeX Services

These test cases verify that the EdgeX micro-services can be started and that MQTT messages are passed to the master node from the services.

The Test inputs

The test scripts and data are stored in the source repository's `cicd/tests/sdt_step2/edgex/` directory.

Test Procedure

The test bed is placed in a state where the cluster is initialized and all edge nodes have joined. The Docker registry and mosquitto MQTT broker must be running on the master node. The registry must be populated with all upstream images and custom images. Either the `device-camera` service should be enabled, or `device-virtual` should be enabled to provide readings.

Execute the test scripts:

```
robot cicd/tests/sdt_step2/edgex/
```

Expected output

The test scripts will start the EdgeX micro-services on all edge nodes, confirm that MQTT messages are being delivered from the edge nodes, and stop the EdgeX micro-services.

The robot command should report success for all test cases.

Test Results

Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/edgex-install/7/

Edgex Report

Summary Information

Status:	All tests passed
Start Time:	20221104 11:48:34.716
End Time:	20221104 11:59:00.926
Elapsed Time:	00:10:26.210
Log File:	log.html

Test Statistics

Total Statistics		Total ⇕	Pass ⇕	Fail ⇕	Skip ⇕	Elapsed ⇕	Pass / Fail / Skip
All Tests		8	8	0	0	00:10:21	

Statistics by Tag		Total ⇕	Pass ⇕	Fail ⇕	Skip ⇕	Elapsed ⇕	Pass / Fail / Skip
No Tags							

Statistics by Suite		Total ⇕	Pass ⇕	Fail ⇕	Skip ⇕	Elapsed ⇕	Pass / Fail / Skip
Edgex		8	8	0	0	00:10:26	
Edgex . Start Edgex		4	4	0	0	00:06:23	
Edgex . Check Mosquitto		1	1	0	0	00:02:02	
Edgex . Check Overlay Network		1	1	0	0	00:00:02	
Edgex . Stop Edgex		2	2	0	0	00:01:59	

Pass (8/8 test cases)

CI/CD Regression Tests: Camera Device Service

These test cases verify that the `device-camera` service can get image from IP Camera, the `sync-app` service can share the image to other edge node, the `image-app` service can analyze the image, and the support-notification can receive the crowded notification.

The Test inputs

The test steps and data are contained in the scripts in the source repository `cicd/tests/sdt_step2/camera/` directory.

Test Procedure

The test bed is initialized to the point of having all EdgeX services running, with `device-camera` and `image-app` enabled.

Execute the test scripts:

```
robot cicd/tests/sdt_step2/camera/
```

Expected output

The test cases will check if MQTT messages and the core-data service containing the data of image acquisition, image sharing and image analysis, and check whether the support-notification service having the notification data of crowded after setting the crowded rule.

The Robot Framework should report success for all test cases

Test Results

Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/camera/10

Camera Report

Summary Information

Status:	All tests passed
Start Time:	20221104 13:09:12.886
End Time:	20221104 13:15:23.870
Elapsed Time:	00:06:10.984
Log File:	log.html

Test Statistics

Total Statistics		Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
All Tests		9	9	0	0	00:06:07	

Statistics by Tag		Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
No Tags							

Statistics by Suite		Total	Pass	Fail	Skip	Elapsed	Pass / Fail / Skip
Camera		9	9	0	0	00:06:11	
Camera . Set Crowded Rule		1	1	0	0	00:00:02	
Camera . Camera Running		7	7	0	0	00:06:07	
Camera . Check Crowded Notification		1	1	0	0	00:00:02	

Pass (9/9 test cases)

Feature Project Tests

N/A

BluVal Tests

BluVal tests for Lynis, Vuls, and Kube-Hunter were executed on the test bed.

The Test inputs

Bluval User Guide

Steps To Implement Security Scan Requirements

https://vuls.io/docs/en/tutorial-docker.html

Test Procedure

1. Copy the folder ~/.kube from Kubernetes master node to Build VM
2. Create SSH Key on Build VM to access Kubernetes master node

Vuls

We use Ubuntu 20.04, and behind a proxy, so we run Vuls test as follows:

1. Create directory

```
$ mkdir ~/vuls
$ cd ~/vuls
$ mkdir go-cve-dictionary-log goval-dictionary-log gost-log
```

2. Fetch NVD

```
$ docker run --rm -it \
    -v $PWD:/go-cve-dictionary \
    -v $PWD/go-cve-dictionary-log:/var/log/go-cve-dictionary \
    vuls/go-cve-dictionary fetch nvd --http-proxy $http_proxy
```

3. Fetch OVAL

```
$ docker run --rm -it \
    -v $PWD:/goval-dictionary \
    -v $PWD/goval-dictionary-log:/var/log/goval-dictionary \
    vuls/goval-dictionary fetch ubuntu 14 16 18 19 20 --http-proxy $http_proxy
```

4. Fetch gost

```
$ docker run --rm -it \
    -e http_proxy=$http_proxy \
    -e https_proxy=$https_proxy \
    -v $PWD:/gost \
    -v $PWD/gost-log:/var/log/gost \
    vuls/gost fetch ubuntu --http-proxy $http_proxy
```

5. Create config.toml

```
[servers]

[servers.master]
host = "192.168.51.22"
port = "22"
user = "test-user"
keyPath = "/root/.ssh/id_rsa" # path to ssh private key in docker
```

6. Start vuls container to run tests

```
$ docker run --rm -it \
    -v ~/.ssh:/root/.ssh:ro \
    -v $PWD:/vuls \
    -v $PWD/vuls-log:/var/log/vuls \
    -v /etc/localtime:/etc/localtime:ro \
    -v /etc/timezone:/etc/timezone:ro \
    vuls/vuls scan \
    -config=./config.toml \
    --http-proxy $http_proxy
```

7. Get the report

```
$ docker run --rm -it \
    -v ~/.ssh:/root/.ssh:ro \
    -v $PWD:/vuls \
    -v $PWD/vuls-log:/var/log/vuls \
    -v /etc/localtime:/etc/localtime:ro \
    vuls/vuls report \
    -format-list \
    -config=./config.toml \
    --http-proxy $http_proxy
```

Lynis/Kube-Hunter

1. Create ~/validation/bluval/bluval-sdtfc.yaml to customize the Test

```
blueprint:
    name: sdtfc
    layers:
        - k8s
        - os

    k8s: &k8s
        -
            name: kube-hunter
            what: kube-hunter
            optional: "False"

    os: &os
        -
            name: lynis
            what: lynis
            optional: "False"
```

2. Update ~/validation/bluval/volumes.yaml file

```
volumes:
    # location of the ssh key to access the cluster
    ssh_key_dir:
        local: '/home/ubuntu/.ssh'
        target: '/root/.ssh'
    # location of the k8s access files (config file, certificates, keys)
    kube_config_dir:
        local: '/home/ubuntu/kube'
        target: '/root/.kube/'
    # location of the customized variables.yaml
    custom_variables_file:
        local: '/home/ubuntu/validation/tests/variables.yaml'
        target: '/opt/akraino/validation/tests/variables.yaml'
    # location of the bluval-<blueprint>.yaml file
    blueprint_dir:
        local: '/home/ubuntu/validation/bluval'
        target: '/opt/akraino/validation/bluval'
    # location on where to store the results on the local jumpserver
    results_dir:
        local: '/home/ubuntu/results'
        target: '/opt/akraino/results'
    # location on where to store openrc file
    openrc:
        local: ''
        target: '/root/openrc'

# parameters that will be passed to the container at each layer
layers:
    # volumes mounted at all layers; volumes specific for a different layer are below
    common:
        - custom_variables_file
        - blueprint_dir
        - results_dir
    hardware:
        - ssh_key_dir
    os:
        - ssh_key_dir
    networking:
        - ssh_key_dir
    docker:
        - ssh_key_dir
    k8s:
        - ssh_key_dir
        - kube_config_dir
    k8s_networking:
        - ssh_key_dir
        - kube_config_dir
    openstack:
        - openrc
    sds:
    sdn:
    vim:
```

3. Update ~/validation/tests/variables.yaml file

```
### Input variables cluster's master host
host: <IP Address>              # cluster's master host address
username: <username>            # login name to connect to cluster
password: <password>           # login password to connect to cluster
ssh_keyfile: /root/.ssh/id_rsa          # Identity file for authentication
```

4. Run Blucon

```
$ bash validation/bluval/blucon.sh sdtfc
```

Expected output

BluVal tests should report success for all test cases.

Test Results

Vuls results (manual) Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/sdt-vuls/2/

Lynis results (manual) Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/sdt-lynis/2/

Kube-Hunter results Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/sdt-bluval/1/

Vuls

Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/sdt-vuls/2/

There are 4 CVEs with a CVSS score >= 9.0. These are exceptions requested here:

Release 7: Akraino CVE and KHV Vulnerability Exception Request

CVE-ID	CVSS	NVD	Fix/Notes
CVE-2022-3643	10.0	https://nvd.nist.gov/vuln/detail/CVE-2022-3643	Fix not yet available Ubuntu CVE record
CVE-2016-1585	9.8	https://nvd.nist.gov/vuln/detail/CVE-2016-1585	No fix available Ubuntu CVE record
CVE-2022-0318	9.8	https://nvd.nist.gov/vuln/detail/CVE-2022-0318	Fix not yet available Ubuntu CVE record
CVE-2022-3649	9.8	https://nvd.nist.gov/vuln/detail/CVE-2022-3649	Fix not yet available Ubuntu CVE record

Lynis

Nexus URL (manual run, with fixes): https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/lynis/3/

The results compare with the Lynis Incubation: PASS/FAIL Criteria, v1.0 as follows.

The Lynis Program Update test MUST pass with no errors.

```
2022-09-14 16:19:49 Test: Checking for program update...
2022-09-14 16:19:49 Result: Update check failed. No network connection?
2022-09-14 16:19:49 Info: to perform an automatic update check, outbound DNS connections should be allowed (TXT
record).
2022-09-14 16:19:49 Suggestion: This release is more than 4 months old. Check the website or GitHub to see if
there is an update available. [test:LYNIS] [details:-] [solution:-]
```

The test environment is a proxied private network inside the Fujitsu corporate network which does not allow direct DNS lookups using tools such as dig. Therefore the update check cannot be performed automatically.

The latest version of Lynis, 3.0.8 at time of execution, was downloaded and run directly on the SUT. See the link below:

Steps To Implement Security Scan Requirements#InstallandExecute

The following list of tests MUST complete as passing

No.	Test	Result	Notes
1	Test: Checking PASS_MAX_DAYS option in /etc/login.defs	2022-12-16 18:45:05 Test: Checking PASS_MAX_DAYS option in /etc/login.defs 2022-12-16 18:45:05 Result: max password age is 180 days 2022-12-16 18:45:05 Hardening: assigned maximum number of hardening points for this item (3). Currently having 21 points (out of 35)	Required configuration
2	Performing test ID AUTH-9328 (Default umask values)	2022-12-16 18:45:05 Performing test ID AUTH-9328 (Default umask values) ... 2022-12-16 18:45:05 Test: Checking /etc/login.defs 2022-12-16 18:45:05 Result: file /etc/login.defs exists 2022-12-16 18:45:05 Test: Checking umask value in /etc/login.defs 2022-12-16 18:45:05 Result: umask is 027, which is fine 2022-12-16 18:45:05 Hardening: assigned maximum number of hardening points for this item (2). Currently having 35 points (out of 49)	Required configuration
3	Performing test ID SSH-7440 (Check OpenSSH option: AllowUsers and AllowGroups)	2022-12-16 18:45:14 Performing test ID SSH-7440 (Check OpenSSH option: AllowUsers and AllowGroups) 2022-12-16 18:45:14 Result: AllowUsers set, with value sdt-admin 2022-12-16 18:45:14 Result: AllowGroups is not set 2022-12-16 18:45:14 Result: SSH is limited to a specific set of users, which is good 2022-12-16 18:45:14 Hardening: assigned maximum number of hardening points for this item (2). Currently having 164 points (out of 231)	Required configuration

4	Test: checking for file /etc/network/if-up.d/ntpdate	2022-12-16 18:45:16 Test: checking for file /etc/network/if-up.d/ntpdate 2022-12-16 18:45:16 Result: file /etc/network/if-up.d/ntpdate does not exist 2022-12-16 18:45:16 Result: Found a time syncing daemon/client. 2022-12-16 18:45:16 Hardening: assigned maximum number of hardening points for this item (3). Currently having 173 points (out of 246)	
5	Performing test ID KRNL-6000 (Check sysctl key pairs in scan profile) : Following sub-tests required	N/A	
5a	sysctl key fs.suid_dumpable contains equal expected and current value (0)	2022-12-16 18:45:27 Result: sysctl key fs.suid_dumpable contains equal expected and current value (0)	Required configuration
5b	sysctl key kernel.dmesg_restrict contains equal expected and current value (1)	2022-12-16 18:45:27 Result: sysctl key kernel.dmesg_restrict contains equal expected and current value (1)	Required configuration
5c	sysctl key net.ipv4.conf.default. accept_source_route contains equal expected and current value (0)	2022-12-16 18:45:27 Result: sysctl key net.ipv4.conf.default. accept_source_route contains equal expected and current value (0)	Required configuration
6	Test: Check if one or more compilers can be found on the system	2022-12-16 18:45:28 Performing test ID HRDN-7220 (Check if one or more compilers are installed) 2022-12-16 18:45:28 Test: Check if one or more compilers can be found on the system 2022-12-16 18:45:28 Result: no compilers found 2022-12-16 18:45:28 Hardening: assigned maximum number of hardening points for this item (3). Currently having 212 points (out of 312)	Required removal of build-essential package and apt autoremove, and /bin/as

Kube-Hunter

Nexus URL: https://nexus.akraino.org/content/sites/logs/fujitsu/job/sdt/r7/sdt-bluval/1/

There are no reported vulnerabilities. Note, this release includes fixes for vulnerabilities found in release 6. See the release 6 test document for details on those vulnerabilities and the fixes.

Kube-Hunter Log

Generated
20220914 16:19:45 UTC+09:00
22 days 18 hours ago

REPORT

Test Statistics

Total Statistics	Total ⇕	Pass ⇕	Fail ⇕	Elapsed ⇕	Pass / Fail
Critical Tests	2	2	0	00:00:01	
All Tests	3	2	1	00:00:16	

Statistics by Tag	Total ⇕	Pass ⇕	Fail ⇕	Elapsed ⇕	Pass / Fail
non-critical (non-critical)	1	0	1	00:00:15	

Statistics by Suite	Total ⇕	Pass ⇕	Fail ⇕	Elapsed ⇕	Pass / Fail
Kube-Hunter	3	2	1	00:00:16	
Kube-Hunter . Kube-Hunter	3	2	1	00:00:16	

Test Execution Log

SUITE Kube-Hunter	00:00:16.188
Full Name: Kube-Hunter	
Source: /opt/akraino/validation/tests/k8s/kube-hunter	
Start / End / Elapsed: 20220914 16:19:29.736 / 20220914 16:19:45.924 / 00:00:16.188	
Status: 2 critical test, 2 passed, 0 failed 3 test total, 2 passed, 1 failed	

SUITE Kube-Hunter	00:00:16.163
Full Name: Kube-Hunter.Kube-Hunter	
Documentation: Hunt for security weaknesses in Kubernetes cluster	
Source: /opt/akraino/validation/tests/k8s/kube-hunter/kube-hunter.robot	
Start / End / Elapsed: 20220914 16:19:29.759 / 20220914 16:19:45.922 / 00:00:16.163	
Status: 2 critical test, 2 passed, 0 failed 3 test total, 2 passed, 1 failed	

TEST Cluster Remote Scanning	00:00:00.827
TEST Node Remote Scanning	00:00:00.149
TEST Inside-a-Pod Scanning (non-critical)	00:00:15.070

Note that the results still show one test failure. The "Inside-a-Pod Scanning" test case reports failure, apparently because the log ends with "Kube Hunter couldn't find any clusters" instead of "No vulnerabilities were found." This also occurred during release 6 testing. Because vulnerabilities were detected and reported in release 6 by this test case, and those vulnerabilities are no longer reported, we believe this is a false negative, and may be caused by this issue: https://github.com/aquasecurity/kube-hunter/issues/358

Test Dashboards

Single pane view of how the test score looks like for the Blue print.

Total Tests	Test Executed	Pass	Fail	In Progress
29	29	27	2	0

*Vuls is counted as one test case.

*One Kube-Hunter failure is counted as a pass. See above.

Vuls and Lynis test cases are failing, an exception request is filed for Vuls-detected vulnerabilities that cannot be fixed. The Lynis results have been confirmed to pass the Incubation criteria.

Additional Testing

None at this time.

Bottlenecks/Errata

None at this time.