SONiC Management Community Test in UHD

Release 2.0

User Guide

KEYSIGHT

202309051211+05:30

Notices

Copyright Notice

Warranty

The material contained in this document is provided “as is,” and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Keysight disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Keysight shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Keysight and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

U.S. Government Rights

The Software is "commercial computer software," as defined by Federal Acquisition Regulation ("FAR") 2.101. Pursuant to FAR 12.212 and 27.405-3 and Department of Defense FAR Supplement ("DFARS") 227.7202, the U.S. government acquires commercial computer software under the same terms by which the software is customarily provided to the public. Accordingly, Keysight provides the Software to U.S. government customers under its standard commercial license, which is embodied in its End User License Agreement (EULA), a copy of which can be found at http://www.keysight.com/find/sweula. The license set forth in the EULA represents the exclusive authority by which the U.S. government may use, modify, distribute, or disclose the Software. The EULA and the license set forth therein, does not require or permit, among other things, that Keysight: (1) Furnish technical information related to commercial computer software or commercial computer software documentation that is not customarily provided to the public; or (2) Relinquish to, or otherwise provide, the government rights in excess of these rights customarily provided to the public to use, modify, reproduce, release, perform, display, or disclose commercial computer software or commercial computer software documentation. No additional government requirements beyond those set forth in the EULA shall apply, except to the extent that those terms, rights, or licenses are explicitly required from all providers of commercial computer software pursuant to the FAR and the DFARS and are set forth specifically in writing elsewhere in the EULA. Keysight shall be under no obligation to update, revise or otherwise modify the Software. With respect to any technical data as defined by FAR 2.101, pursuant to FAR 12.211 and 27.404.2 and DFARS 227.7102, the U.S. government acquires no greater than Limited Rights as defined in FAR 27.401 or DFAR 227.7103-5 (c), as applicable in any technical data. 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Contacting Us

Keysight headquarters

1400 Fountaingrove Parkway
Santa Rosa, CA 95403-1738
www.ixiacom.com/contact/info

Support

Global Support	+1 818 595 2599	support@ixiacom.com
Regional and local support contacts:		
APAC Support	+91 80 4939 6410	support@ixiacom.com
Australia	+61-742434942	support@ixiacom.com
EMEA Support	+40 21 301 5699	support-emea@ixiacom.com
Greater China Region	+400 898 0598	support-china@ixiacom.com
Hong Kong	+852-30084465	support@ixiacom.com
India Office	+91 80 4939 6410	support-india@ixiacom.com
Japan Head Office	+81 3 5326 1980	support-japan@ixiacom.com
Korea Office	+82 2 3461 0095	support-korea@ixiacom.com
Singapore Office	+65-6215-7700	support@ixiacom.com
Taiwan (local toll-free number)	00801856991	support@ixiacom.com

Documentation conventions

The following documentation conventions are used in this guide:

Describing interactions with the UI

You can interact with products by using different input methods: keyboard, mouse, touch, and more. So in most parts of the user documentation, generic verbs have been used that work with any input method. In cases where input-neutral verbs do not work, mouse-specific verbs are used as the first choice, followed by touch-specific verbs as the second choice.

See the following table for examples on how you can interpret the different input methods.

Input-neutral	Mouse	Touch
Select **Modify**.	Click **Modify**.	Tap **Modify**.
Select **Accounts** > **Other accounts** > **Add an account**.	Click **Accounts** > **Other accounts** > **Add an account**.	Tap **Accounts** > **Other accounts** > **Add an account**.
To open the document in Outline view, select **View** > **Outline**.	To open the document in Outline view, click **View** > **Outline**.	To open the document in Outline view, tap **View** > **Outline**.
Select **Protocols**.	Click the **Protocols** tab.	Tap **Protocols**.
-NA-	Double-click the **Client** wizard.	Double-tap the **Client** wizard.
Open the **Packages** context menu.	Right-click **Packages** to open the shortcut menu.	Long tap **Packages** to open the shortcut menu.

Deprecated words

The following words have been replaced with new words, considering the audience profile, our modern approach to voice and style, and our emphasis to use input-neutral terms that support all input methods.

Old usage...	New usage...
shortcut menu, right-click menu	context menu
click, right-click	select
drag and drop	drag

Table of Contents

Contacting Us **3**

Documentation conventions **4**

Chapter 1 SONiC-MGMT Community Test in UHD **1**

Chapter 2 Prerequisites **3**

Chapter 3 How do I... ..... **5**

cEOS Upload 6

Create topology 6

Onboard DUT 6

Create topology for 25G/10G fanout 6

List test cases 7

Execute test 8

Execute custom tests 8

View test logs 10

View test status 10

Generate reports 10

Delete topology 11

Modify layer1 settings 11

Switch mode between UHD and Sonic-ct 11

View Sonic-CT and Sonic-mgmt version 12

Index **13**

CHAPTER 1

SONiC-MGMT Community Test in UHD

Keysight offers you to run the entire SONiC-MGMT testbed in a box with absolutely no change in the scripts. You can run the admin shell operations for SONiC-MGMT Community Test in UHD with the help of ksctl cli. You can view the available ksctl commands by using `ksctl-h`.

For basic operations that can be done using the ksctl cli, see the How do I ... section.

Following are the supported HwSkus:

- Force10-S6000
- Force10-S6100
- Force10-Z9100
- DellEMC-Z9332f-M-O16C64
- DellEMC-Z9332f-O32
- Arista-7050-QX32
- Arista-7050-QX-32S
- Arista-7170-64C
- Arista-7170-32CD-C32
- Mellanox-SN2700-D40C8S8
- Mellanox-SN2700-D48C8
- ACS-MSN2700
- Arista-7060CX-32S-D48C8
- Arista-7260CX3-D108C8
- INGRASYS-S9100-C32
- INGRASYS-S8810-32Q
- INGRASYS-S8900-54XC
- INGRASYS-S8900-64XC
- Accton-AS7712-32X
- Accton-AS7726-32X
- montara
- Celestica-DX010-C32
- Seastone-DX010
- Celestica-E1031-T48S4

- Nokia-7215
- newport

CHAPTER 2
Prerequisites

License requirement for SONiC-MGMT Community Test

Minimum 1 year subscription for SONiC Management Community Test in UHD is needed to run SONiC community test from sonic-mgmt repository. This license must be installed on the SONiC Management Community Test in UHD hardware to make the community available for running tests.

Following is the command to activate the license:

```
-ksctl license activate <activation code> 1
```

NOTE This license is needed only for running SONiC-MGMT Community tests on UHD.

This page intentionally left blank.

CHAPTER 3
How do I...

The 'How do I' section contains a list of topics that will help you to configure SONiC Management Community Test in UHD by using the SONiC Management Community Test in UHD web UI.

Section contents:

cEOS Upload **6**

Create topology **6**

Onboard DUT 6

Create topology for 25G/10G fanout 6

List test cases **7**

Execute test **8**

Execute custom tests 8

View test logs **10**

View test status **10**

Generate reports **10**

Delete topology **11**

Modify layer1 settings **11**

Switch mode between UHD and Sonic-ct **11**

View Sonic-CT and Sonic-mgmt version **12**

cEOS Upload

To create the Virtual Machines (VMs) for topologies (T0, T1, T1-LAG), you must download the cEOS image from Arista website and upload the image to the SONiC Management Community Test in UHD by using the following command:

Command

```
- ksctl ceos upload <cEOS image tar file>
```

Create topology

You can deploy the testbed by using ksctl topo start. After the deployment is complete, you can check the status of the process by using the ksctl topo status command.

Command

```
- ksctl topo start -d <dut ip/mask> -t <topology> -s <hwsku>
- ksctl topo status
```

Onboard DUT

List of supporting HwSkus are mentioned in SONiC-MGMT Community Test in SONiC Management Community Test in UHD.

In order to onboard a DUT, whose HwSku is not present in the list of supported HwSkus, do the following:

1. Get the onie_build_platform info of the DUT from /host/machine.conf file from the DUT.
2. Get the HwSku info of the DUT using #show version.
3. Navigate to /usr/share/sonic/device/<onie_build_platform>/<HwSku>/ location on the DUT.
4. Copy the port_config.ini file into /home/admin/ path in the UHD setup.
5. After the port_config.ini file is in place, the topology can be created using ksctl topo start with -c flag as shown in the following example:

Example

```
ksctl topo start -d <dut ip/mask> -t <topology> -s <hwsku> -c <port_config.ini file>
```

Create topology for 25G/10G fanout

The default speed mode of the topology is in 100G mode.To create a topology with 25G /10G speed modes, do the following steps:

1. Change the speed mode in the UHD by using the command `ksctl layer1 speedmode 25G/10G`. The first eight front panel ports of the UHD is considered for the test.
2. Start the topology by using the command `ksctl topo start -d <dutip/mask> -s <hwsku> -t < t1 / t0 / t1-lag / ptf32 >`.
3. Change the speed mode of the DUT ports from 100G to 25G/10G and ensure the links are up.
4. Run a test with `ksctl test run <testcase>`.

List test cases

You can run the test scripts from sonic-mgmt/tests location by using the `ksctl test run` command. You can run the specific testcases with `‘::’` followed by the testcase name as shown in the example.

You can also view the list of available scripts from sonic-mgmt by using the `ksctl test list` command.

You can list all the test files (python files) and the test cases (the pytest definitions present inside the test files). It is an equivalent of –collect-only flag of pytest.

The following command lists all the test files:

```
- ksctl test list
```

The following command lists all the test cases:

```
- ksctl test list --testcases '*'
```

The following command lists a specific test case, for example bgp

```
- ksctl test list --testcases bgp
```

A flag named filter returns test files/cases from tests and/or custom_tests directory.

The following command lists the test files from both the directories tests and custom_tests:

```
- ksctl test list --filter all
```

The following command lists all test cases from both directories:

```
- ksctl test run --filter all --testcases '*'
```

The following command lists test files from custom_tests directory:

```
- ksctl test list --filter custom
```

The following command lists the bgp test cases from custom_tests directory:

```
- ksctl test list --filter custom --testcases 'bgp'
```

Examples

```
-ksctl test run acl/test_acl.py
-ksctl test run acl/test_acl.py::TestBasicAcl
```

Execute test

You can run the test scripts from sonic-mgmt/tests location by using the `ksctl test run` command. You can run the s pecific testcases with `'::'` followed by the testcase name as shown in the example.

You can also view the list of available scripts from sonic-mgmt by using the `ksctl test list` command.

Command

```
- ksctl test run <testscript>
```

Examples

```
-ksctl test run acl/test_acl.py
-ksctl test run acl/test_acl.py::TestBasicAcl
```

Selective Test Execution using Expression flag

The `ksctl test run` command uses an expression that enables you to run specific test cases with a particular pattern. It works like the -k flag of pytest.

Flag Schema

```
ksctl test run <testname> --expression 'expression-pattern as string'
```

The following command runs all the tests having **bgp** in the test case name:

```
ksctl test run '*' --expression 'bgp'
```

The following command runs all bgp tests having **neighbor** in its terminology:

```
ksctl test run bgp --expression "neighbor"
```

Selective Test execution using Topology flag

The `ksctl test run` command uses the **topology** flag that allows you to run test cases having the specified topology in their pytest marker. This mirrors the topology flag functionality present in sonic-mgmt.

Flag Schema

```
ksctl test run –topology <topology type>
```

The following command runs tests having t0 topology in their pytest marker:

```
ksctl test run --topology t0
```

Execute custom tests

You can create your own custom tests and smoothly run them within the sonic-mgmt framework. Additionally, you have the option to download tests from any branch or commit of sonic-mgmt and execute them.

To add custom tests, a directory is mounted in the root directory of the ksctl shell, also known as custom_tests. It can be accessed using the following command:

```
cd /custom_tests
```

Using sonic-mgmt branch in custom_tests

The customized testing process allows you to download and execute tests from any specific branch or commit from sonic-mgmt. To initiate tests from a particular branch, like 202205, you need to do the following:

- Clone the desired branch/commit onto their local system or UHD. For instance, the user can clone sonic-mgmt branch 202205 to the directory /home/admin/github/202205.
- Copy the contents of the test folder to the /custom_tests directory. In a shell environment, this can be done using the following command:

```
cp -R /home/admin/github/202205/sonic-mgmt/tests/* /custom_tests/
```

```
1ae6d4436c53:~$
1ae6d4436c53:~$ cp -R /home/admin/github/202205/sonic-mgmt/tests/* /custom_tests/.
1ae6d4436c53:~$
```

- Validate the contents of custom_tests folder and check if all the files and folders have been copied successfully.

NOTE

When you clone directly from the sonic-mgmt branch, the test files will carry their imports from the tests folder of sonic-mgmt. Note that bundled tests might not include certain packages or libraries, especially those that the community has deprecated. If you are working with older cloned versions, they might still be expecting these packages.

To address this, you might need to adjust the imports to reference the custom_tests directory. This directory contains all the cloned data for the specific branch.

In specific situations, you may need to modify file references and imports. For instance, with recent updates, the **CustomSkipIf** plugin has been removed. Thus, when running a build from an earlier branch (such as 202205), you may need to point to CustomSkipIf within the custom_test directory. This adjustment can be carried out by adding the reference to custom_tests directory:

```
pytest_plugins = ('tests.common.plugins.ptfadapter',
                  'tests.common.plugins.ansible_fixtures',
                  'tests.common.plugins.dut_monitor',
                  'tests.common.plugins.loganalyzer',
                  'tests.common.plugins.pdu_controller',
                  'tests.common.plugins.sanity_check',
                  'tests.common.plugins.custom_markers',
                  'tests.common.plugins.custom_skipif.CustomSkipIf',
                  'tests.common.plugins.test_completeness',
                  'tests.common.plugins.log_section_start',
                  'tests.common.plugins.custom_fixtures',
                  'tests.common.dualtor',
                  'tests.decap',
                  'tests.common.plugins.allure_server',
                  'tests.common.plugins.conditional_mark')
```

You can use any one of the following commands to run tests from custom_tests folder:

```
ksctl test run <test_name> --custom-test
ksctl test run /custom_tests/<testname>
```

View test logs

You can view the realtime log of the test execution by using the `ksctl test log` command.

Command

```
- ksctl test log <run_id> or
- ksctl test log –follow ( for the running test )
```

View test status

To check the status of the current run or to get the run ID of the previous runs, use the following command:

Command

```
- ksctl test status
```

Generate reports

You can generate the report of the run history in json, csv and yaml formats by using the following command:

```
- ksctl test report <run_id> -o csv/yaml/json\
```

Generate allure report

With sonic-ct, you can create and upload allure reports from their custom tests. The `ksctl test report` command now includes the introduction of two flags, described as follows:

The following command generates allure report for a specific test run:

```
ksctl test report –o allure <test run id>
```

The following command generates allure reports for all the test runs:

```
ksctl test report –o allure
```

This command generates all the allure reports in zip format and can be transferred and uploaded to a remote allure server.

Upload allure report

The following command uploads the allure report:

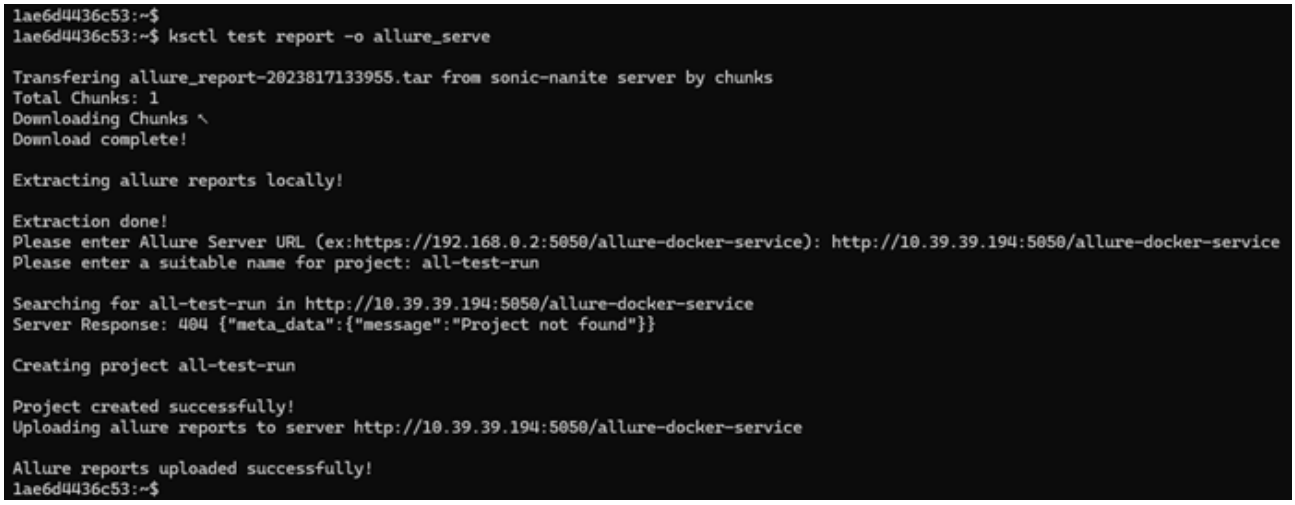
```
ksctl test report –o allure_serve <test run id>
```

The following command generates allure reports for all the test runs:

```
ksctl test report –o allure_serve <test run id>
```

The allure service initiates a fresh CLI thread, prompting the user to input the allure server's URL and the corresponding project name. The URL format should follow this structure:

https://192.168.0.2:5050/allure-docker-service, while the project name must adhere to the specific allure format requirements.



The following command is used to clean the allure reports in the internal system:

```
ksctl test cleanup
```

Delete topology

To delete the deployed testbed, use the following command:

Command

```
- ksctl topo stop
```

Modify layer1 settings

You can modify the layer1 settings, such as AN and FEC by using the `ksctl layer1` commands.

Command

```
ksctl layer1 an-set <port#> <0->auto,1->enable,2>disable>
ksctl layer1 fec-set <port#> <fec (NONE, FC, RS)>
ksctl layer1 speedmode < 100G /25G /10G >
```

Switch mode between UHD and Sonic-ct

You can switch the mode of the box between UHD and sonic-ct by using the following commands:

- If the current mode is in UHD, that is kcos cli, switch to sonic-ct mode by using the following command:

```
ksctl system switch sonic-ct
```

- If the current mode is in sonic-ct, that is ksctl cli, switch to UHD mode by using the following command:

```
ksctl system switch uhd
```

After you switch the mode, reboot your system by using the kcos system reboot command for the change to take effect.

View Sonic-CT and Sonic-mgmt version

To get the image version for Sonic-ct and the git version of Sonic-Mgmt repo use the following command:

```
- ksctl version
```

Index

C

customer assistance 3

D

deleting topology 11

documentation conventions 4

E

executing custom tests 8

executing test 8

G

generating reports 10

H

hardware prerequisites 3

hardware requirements 3

how do I 5

how to 5

K

keyboard interactions 4

L

listing test cases 7

M

modifying layer1 settings 11

mouse interactions 4

O

onboarding a DUT 6

P

prerequisites 3

product support 3

R

requirements 3

S

software prerequisites 3

software requirements 3

supported browsers 3

switching mode 11

T

technical support 3

touch interactions 4

U

uploading cEOS 6

V

viewing test logs 10

viewing test status 10

viewing version 12

www.keysight.com