

# REC Test Document

- [Introduction](#)
- [Overall Test Architecture](#)
  - [Test Bed](#)
  - [Test Framework](#)
  - [Traffic Generator](#)
- [Test API description](#)
  - [Test suites:](#)
  - [Test cases:](#)
- [Test Dashboards](#)
- [Additional Testing](#)
- [Bottlenecks/Errata](#)

## Introduction

REC Blueprint is deployed and validated in Nokia Espoo and AT&T Middletown Radio Edge Cloud Validation Labs. Deployment of REC and test cases have been run automatically with different clusters such as Nokia OpenEdge, Dell and HP Gen 10. REC feature testing is based on Cloud TAF (Test Automation Framework), which tests basic k8s features. Results of deployments and testing are publicly available in LF Akraino nexus servers.

## Overall Test Architecture

### Test Bed

Five server cluster with REC software, as described in [Radio Edge Cloud Validation Lab](#)

### Test Framework

Cloud TAF, robot based test framework for REC <https://gerrit.akraino.org/r/gitweb?p=ta%2Fcloudtaf.git;a=summary> Note that this Gerrit link is authoritative. In the event of discrepancies between this wiki page and Gerrit, Gerrit is correct. There is no mechanism for automatically updating this wiki page when tests are added to the test suite and no commitment on the part of the developers to do double entry of test cases. The testcases are located in \*.robot files in the aptly named "testcases" subdirectory of the cloudtaf repository. The Jenkins servers clone the cloudtaf repo from Gerrit and run whatever tests are contained in it at that time.

### Traffic Generator

N/A

## Test API description

### Test suites:

- post-install (8 test cases)
- kube-namespace (1 test case)
- kube-service (1 test case)
- smoke-tests (6 test cases)
- pm-supports (2 test cases)

### Test cases:

#	Test suite	Test case	Short description
1	post-install	Verify Deployment	Check deployment logs (/srv/deployment/log/bootstrap.log). Looking for "Installation complete, Installation Succeeded" in log.
2		Verify Image Build Number	Show Build number.
3		Verify Docker Version	Check version of the installed docker (needed Docker version 19.03.3)
4		Verify Kubernetes Clusters	Check pods' status in kube-system namespace. All pod needs to be in "Running" state.
5		Verify State of required services	Check the services are running and active.  Checked services: docker.service, kubelet.service

6		Verify Node Functionality	Checks the state of the nodes. All nodes need to be in "Ready" state.
7		Verify Package Manager Status	Try to pull caas/hypercube from common(?) repository. Verify the pull is in success and image is up to date.
8		Verify Helm Caas Infra Status	Check helm status. Caas-infra helm chart execution status needs to be "DEPLOYED"
9	kube-namesp ace	Verify creating and testing Namespaces	Create "test-ns" namespace. Create a pod from nginx image in test-ns namespace. Verify the pod is created. Check the pod is created in test-ns namespace. Delete the created pod and namespace.
10	kube-service	Verify creating and testing services	Create pod from nginx image. Create a service what contains the previously created pod, and targets TCP port 80 to the pod's TCP port 80. Verify that the nginx is reachable on TCP port 80. Delete the created service and pod.
11	smoke-tests	Verify Cluster Config Management	Verify all the nodes names. Verify database: cluster size needs to be 3 (wsrep_cluster_size).
12		Verify Cluster Config Management After Reboot Management VIP Node	Reboot the VIP(?) node. Verify the restart same as 'Verify Cluster Config Management' test case.
13		Verify Database Stop And Start	Stop one of the databases. Check database: cluster size needs to be 2.  Start a database node. Check database: cluster size needs to be 3.
14		Verify Create All And No Roles	Create a user with all-roles.  Create a user with no roles.
15		Verify SudoShells	Verify all the sudo-"node" user. All need to be root user.
16		Verify Remotescript Default	Verify 'remotescript-default' command execution.
17	pm-supports	PM001	Check resource usage of the nodes (kubectl top node).
18		PM002	Check kubectl api metrics (kubectl api-versions   grep metrics). Printout needs to be contain " <a href="https://nexus.akraino.org/content/sites/logs/nokia/job/Cloudtaf_test_suite_Espoo_OE/">custom.metrics.k8s.io/v1beta1</a> " and " <a href="https://nexus.akraino.org/content/sites/logs/nokia/job/Cloudtaf_test_suite_Espoo_OE/">metrics.k8s.io/v1beta1</a> ".

Test execution logs: [https://nexus.akraino.org/content/sites/logs/att/job/Cloudtaf\\_test\\_suite\\_Middletown\\_OE1/](https://nexus.akraino.org/content/sites/logs/att/job/Cloudtaf_test_suite_Middletown_OE1/) or [https://nexus.akraino.org/content/sites/logs/nokia/job/Cloudtaf\\_test\\_suite\\_Espoo\\_OE/](https://nexus.akraino.org/content/sites/logs/nokia/job/Cloudtaf_test_suite_Espoo_OE/)

## Test Dashboards

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

Total Tests	Test Executed	Pass	Fail	In Progress
18	18	18	0	0

## Additional Testing

ORAN RIC applications are installed after each REC deployment using Jenkins and results are pushed to LF nexus [https://nexus.akraino.org/content/sites/logs/att/job/Install\\_RIC\\_on\\_OpenEdge1/](https://nexus.akraino.org/content/sites/logs/att/job/Install_RIC_on_OpenEdge1/)

## Bottlenecks/Errata