REC Installation Guide

Licensing

Radio Edge Cloud is Apache 2.0 licensed. The goal of the project is the packaging and installation of upstream Open Source projects. Each of those upstream projects is separately licensed. For a full list of packages included in REC you can refer to https://logs.akraino.org/production/vex-yul-akraino-jenkins-prod-1/ta-ci-build-amd64/313/work/results/rpmlists/rpmlist (the 313 in this URL is the Akraino REC/TA build number, see https://logs.akraino.org/production/vex-yul-akraino-jenkins-prod-1/ta-ci-build-amd64/ for the latest build.) All of the upstream projects that are packaged into the REC/TA build image are Open Source.

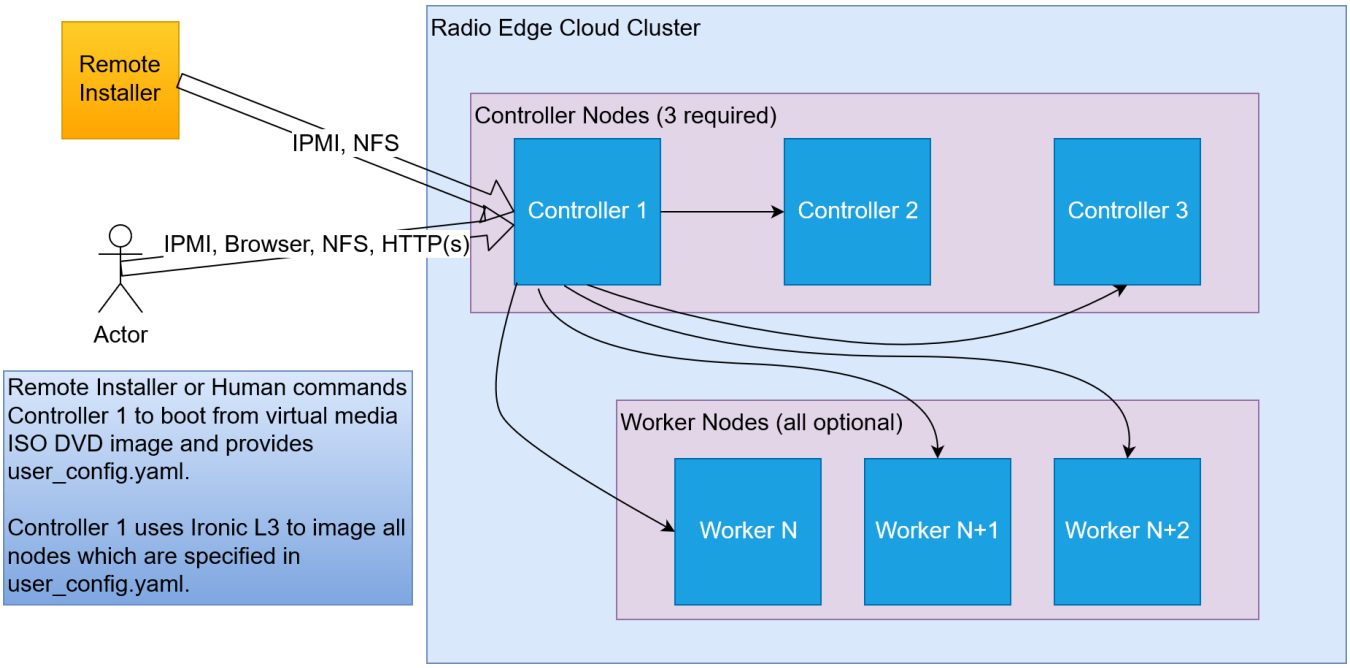
Introduction

This document outlines the steps to deploy Radio Edge Cloud (REC) cluster. It has a minimum of three controller nodes. Optionally it may include worker nodes if desired. REC was designed from the ground up to be a highly available, flexible, and cost-efficient system for the use and support of Cloud RAN and 5G networks. The production deployment of Radio Edge Cloud is intended to be done using the Akraino Regional Controller which has been significantly enhanced during the Akraino Release 1 timeframe, but for evaluation purposes, it is possible to deploy REC without the Regional Controller. Regardless of whether the Regional Controller is used, the installation process is cluster oriented. The Regional Controller or a human being initiates the process on the first controller in the cluster, then that controller automatically installs an image onto every other server in the cluster using IPMI and Ironic (from OpenStack) to perform a zero touch install.

In a Regional Controller based deployment, the Regional Controller API will be used to upload the REC Blueprint YAML (available from the REC repository) which informs the Regional Controller of where to obtain the REC ISO images, the REC workflows (executable code for creating, modifying and deleting REC sites) and the REC remote installer component (a container image which will be instantiated by the create workflow and which will then invoke the REC Deployer (which is located in the ISO DVD disc image file) which conducts the rest of the installation.

The instructions below skip most of this and directly invoke the REC Deployer from the Baseboard Management Controller (BMC), integrated Lights Out (iLO) or integrated Dell Remote Access Controller (iDRAC) of a physical server. The basic workflow of the REC deployer is to copy a base image to the first controller in the cluster and then read the contents of a configuration file (typically called user_config.yaml) to deploy the base OS and all additional software to the rest of the nodes in the cluster.

> ⚠ Although the purpose of the Radio Edge Cloud is to run the RAN Intelligent Controller (RIC), the RIC software (which was written from scratch starting in the spring of 2019) is not quite stable enough yet to incorporate into the REC ISO image. The manual installation procedure described below does not result in the installation of the RIC. In the REC Continuous Deployment system, as of the Akraino Release 1 and 2 timeframe, a Jenkins job deploys a snapshot of the RIC and runs a small set of tests. These steps may be manually executed and this procedure is described in the last section of this WIKI document. Fully automated installation of the RIC as part of the REC will not be complete until sometime in 2020. If you are interested in actually interfacing a REC appliance with eNodeB/gNode B and radio infrastructure you should really join the Radio Edge Cloud Project Meetings on a weekly basis and let the REC team know of your interest. We will be happy to coordinate with you and welcome any testing that you can do.



Pre-Installation Requirements for REC Cluster

Hardware Requirements:

REC is a fully integrated stack from the hardware up to and including the application, so for best results, it is necessary to use one of the tested hardware configurations. Although REC is intended to run on a variety of different hardware platforms, it includes a hardware detector component that customizes each installation based on the hardware present and will need (possibly minor) changes to run on additional hardware configurations. Preliminary support is present in Akraino Release 1 of REC for HP DL380 generation 9 and 10, Dell R740xd and Nokia Open Edge servers, but the primary focus of Release 1 testing is the Nokia Open Edge servers, so some issues may be encountered with other server types.

- Minimum of 3 nodes
- Total Physical Compute Cores: 60 (120 vCPUs)
- Total Physical Compute Memory: 192GB minimum per node
- Total SSD-based OS Storage: 2.8 TB (6 x 480GB SSDs)
- Total Application-based Raw Storage: 5.7 TB (6 x 960GB SSD0
- Networking Per Server: Apps - 2 x 25GbE (per Server) and DCIM - 2 x 10GbE + 1 1Gbt (shared)

The specific recommended configuration as of the Release 1 timeframe is the Open Edge configuration documented in the Radio Edge Cloud Validation Lab

BIOS Requirements:

- BIOS set to Legacy (Not UEFI, although UEFI support is partially implemented and should be available in 2020)
- CPU Configuration/Turbo Mode Disabled
- Virtualization Enabled
- IPMI Enabled
- Boot Order set with Hard Disk listed as first in the list.

As of Release 1 and 2, Radio Edge Cloud does not yet include automatic configuration for a pre-boot environment. The following versions were manually loaded on the Open Edge servers in the Radio Edge Cloud Validation Lab using the incomplete but functional script available here. In the future, automatic configuration of the pre-boot environment is expected to be a function of the Regional Controller under the direction of the REC pod create workflow script.

- BIOS1: 3B06
- BMC1: 3.13.00
- BMC2: 3.08.00
- CPLD: 0x01

Network Requirements:

The REC cluster requires the following segmented (VLAN), routed networks accessible by all nodes in the cluster:

- External Operations, Administration and Management (OAM) Network
- Out Of Band (OOB) (iLO/iDRAC) network(s)

- Storage/Ceph network(s)
- Internal network for Kubernetes connectivity
- NTP and DNS accessibility

The REC installer will configure NTP and DNS using the parameters entered in the user_config.yaml. However, the network must be configured for the REC cluster to be able to access the NTP and DNS servers prior to the install.

About user_config.yaml

The user_config.yaml file contains details for your REC cluster such as required network CIDRs, usernames, passwords, DNS and NTP server ip addresses, etc. The REC configuration is flexible, but there are dependencies: e.g., using DPDK requires a networking profile with ovs-dpdk type, a performance profile with CPU pinning & hugepages and performance profile links on the compute node(s).

The following link points to the latest user_config template with descriptions and examples for every available parameter:

user_config.yaml template

Note: the version number listed in the user_config.yaml needs to follow closely the version from the template. There is a strict version checking during deployment for the first two part of the version number. The following rules apply to the yaml's version parameter:

```
### Version numbering:
###    X.0.0
###        - Major structural changes compared to the previous version.
###        - Requires all users to update their user configuration to
###          the new template
###    a.X.0
###        - Significant changes in the template within current structure
###          (e.g. new mandatory attributes)
###        - Requires all users to update their user configuration according
###          to the new template (e.g. add new mandatory attributes)
###    a.b.X
###        - Minor changes in template (e.g. new optional attributes or
###          changes in possible values, value ranges or default values)
###        - Backwards compatible
```

Example user_config.yaml

user_config.yaml

```
---
version: 2.0.5
name: rec-sample

description: REC Deployment on Nokia OpenEdge Server

time:
    ntp_servers: [216.239.35.4, 216.239.35.5]
    zone: America/New_York

users:
    admin_user_name: cloudadmin
    admin_user_password: "$9$bl0ck=959000$V07qrQ4tKMbDTWTj$wl9cTTqThWTEWm33THH29SZeIGU66K2FHffF$1wIvh9CACKJ
/HvZFGdbedw79ag2.2AqtDRoTTTCWK8Eq0kQn/"
    initial_user_name: myadmin
    initial_user_password: FY625czv5R
    admin_password: ycjPSE4mA

networking:
    dns: [ 8.8.8.8, 8.8.4.4 ]
    mtu: 9000

    infra_external:
        network_domains:
            rack-1:
                cidr: 192.168.10.0/24
                vlan: 141
```

Report a B

```
                gateway: 192.168.10.1
                ip_range_start: 192.168.10.210
                ip_range_end: 192.168.10.213

    infra_storage_cluster:
        network_domains:
            rack-1:
                cidr: 192.169.10.0/24
                ip_range_start: 192.169.10.211
                ip_range_end: 192.169.10.213
                vlan: 142

    infra_internal:
        network_domains:
            rack-1:
                cidr: 192.167.10.0/24
                ip_range_start: 192.167.10.211
                ip_range_end: 192.167.10.250
                vlan: 144

    provider_networks:
        providerInternal:
            vlan_ranges: "2002:2003"
        providerExternal:
            vlan_ranges: "2004:2005"
        providerSriov:
            vlan_ranges: "2006:2008"

caas:
    docker_size_quota: 2G
    helm_operation_timeout: 900
    docker0_cidr: 172.17.0.1/16
    instantiation_timeout: 60
    encrypted_ca:
["U2FsdGVkX1+iaWyYk3W01IFpfVdughR5aDKo2NpcBw2UStYnepHlr5IJD3euo1lS\n7agR5K2My8zYdWFTYYqZncVfYZt7Tc8zB2yzATEIHEV8
PQuZRHqPdR+/OrwqjwA6\ni/R+4Ec3Kko6eS0VWvwDfdhhK/nwczNNVFOtWXCwz/w7AnI9egiXnlHOq2P
/tsO6\np3e9J6ly5ZLp9WbDk2ddAXChnJyC6PlF7ou
/UpFOvTEXRgWrWZV6SUAgdxg5Evam\ndmmwqjRubAcxSo7Y8djHtspsB2HqYs90BCBtINHrEj5WnRDNMR/kWryw1+S7zL1G\nwrpDykBRbq
/5jRQjqO/Ct98yNDdGSWZ+kqMDfLriH4pQoOzMcicT4KRplQNX2q9O\nT
/7CXKmmB3uBxM7a9k2LS22Ljszyd2vxth4jA+SLNOB5IT8FmfDY3PvNnvKaDGQ4\nuWPASyjpPjms3LwsKeu+T8RcKcJJPoZMNZGLm
/5jVqm3RXbMvtI0oEaHWsVaSuwX\nnMgGQHNHop+LK+5a0InYn4ZJo9sbvrHp9Vz4Vo+AzqTVXwA4NEHfqMvpphG+aRCb\ncPJggJqnF6s5CAPDR
vwXzqjjVQy2P1/AhJugW7HZw3dtux4xe3RZ+AMS2YW+fSi1\nIxAGlsLL28KJMc5ACxX5cuSB
/nO19afpf6zyOPIk0ZVh8+bxmB4YBRzGLTSnFNr3\ndauT9
/gCU85ThE93rIfPW6PRyp9juEBLjgTpqDQPn5APoJIIW1ZQWr6tvSlT04Hc\nw0HZ7EcAC7EmmaQYTyL6iifHiZHop9g2clXA0MU9USQggMOKxFr
xEyF4iWdsCCXP\nfTA3bgzvlvqfk9p2Cu9DOmRHGLby2YSj+oghsFDCfhfM1v2Ip2YGPdJM6y7kNX19\nkBpV4Rfcw0NCg2hhXbHZ7LtejlQ1ht8
HnmY5/AnJ/HRdnPb+fcdgS9ZFcGsAH2ze\nSe7hb+MNp80JsuX4A+jOjBacjwL+KbX5RDJp//5dEmqJDkbfMctL1KukBaDrbpci\np
/TeVmLhwlQogeVuF
/Y5vCokq6M5+f28jFJ+R+P2oBY3fAvBhmd+ZmGbUWXxmMF+\nV3mpFkYqXWS+mtVh8Fs0nhrCkqRLTmBj5UNhsMcZ4vGfiu+dPMQi62wa6GoGVju
s\nIj/Upal9RYwthSykUKcWu0KEB929/e4Sz0Y6s3Pzy1+xdmKDPtaBUH9UT3LjMVvY\nordeL0UjKYqWcvpb7Vfma3UD0tz6n/CyHNDVhA
/FioadEy6iJvL316Kf3to69cN+\nvKWav/IeazxdhBSbatPKN3qwESkzr3el2yrdZL4qehflRMp0rFuzZfRB69UFPbgq\nkTQlJHb0OaJTt6er
/XfjtMZoctW7xtYf58CqMJ06QxK5kLKc5Yib73cVyzhmmIz4\nEtUs10QCA5AihHgVES8ZrgZKWDhR+pmFPG3eVitJoUeDNEe9vVEEX8TiWu+H1O
HG\n8UyCKFyyPCj5OwVbwGSgQg=="]
    encrypted_ca_key:
["U2FsdGVkX1+WlNST+WkysFUHYAPfViWe01tCCQsXPsWsUskB4oNNC78bXdEv33+3\ncDlubc9F0ZiHxkng70LKCFV5KQneHfg6c3lPaM4zwaJ3
4UCf80riIoYVozxqnK/S\nTAs0i0rJmzRz4hkTre4xV0I2ZucW3gquP4
/s1yUK3IJF84SDfEi26uPsBOrUpU9Q\nIBxY2rldK+yZUZUFehQb82dvin0CSiXDY63cYLJMYEwWBfJEeY+RGMuZuuGp3qgy\nyVfByZ5
/kwF9qa6+ToYw2zXiokGFfBqiAFnXU7Q6Wcu2qndMQoiy3jFU2DjEQi6N\nVgZHzrPUUUrmQGALyA5blVvNHVQyq4rmMmsTEI02xclz8m7Yzd
/HEFo/C5z5x+My\n2SOIBIRCy6bTSpzU7iixl5U6r5/XfrfQoJ+OwRq1
/P2QmJ2swqzcLOUpDlquDeuP\nd46ceWMO8nlimRps4cX5nQRI1SLaypH1rRiQpnIP7q+jrHEco6wStc458rzX1WxW\nhPMjnnlVhH4sJNqh5c5
/1BvzSBdnx0qIBcFA6fR8XfL
//DmRFsAfRaxVVWadpusc\nXfh4LNNqR9HmoNH6yfBpd66yBYsjFbWip0WKMwdhNBqN1a94OFvRS4+iUfskjC2w\n4w4YjPluRBxI5t9eT4wX8D3
28ikgP4ZQrPdUZoDpLThhRZ62pTOknOeVj+C7799O\nEbopqGg+6BIXZHakmzB6I
/fyjthoLBbxpyqNvKlGGamMNI3d7wq1vwTHch5QLO+w\n5fuRqoIRUtGscSQXp8EOb4kiaxhXXJLkVJw7auOdqxqxQbIf+dt2ViwdyFNjdHz8\ng
PFcAom0GO+T7xHMF1H6xqUXkB4QzTK934pMVoIwu5MezBlz8bxj5+EeF7Ptkdnj\nq4rwihGY7aEhPrXVoq19tsbMYwDGZQvbTKtWDOxrD6ruTDT
wZxVZcEOAX5KCF0Oq\nqRcrCBcLNERm4FSAgUK90v71TNQoMpVea3/01Ec8GbHJfozvrmAVqBpbF0ajlM1/\nZvGrnmVrJEk
/PelCEu+Ni9zrn7DxGZqJ7lbcDU7Nq/18KNvOQah4Ryh9aDKVSD4r\nvgZKzIHPRgKoHTxTZ2uP1LBgK2Ux1RjhlAcZFAmWYxg
/qluxnHKCimZ04rIjI0if\nN0wSI7uh8TsyidZv+iKpG+JqW5oe7R8xLlU3ceFllkghAGVRn
/UyirGXYPzxXbfB\naphYFBuj6FbtdisM7euX2A9F2OUM2reditR
/z6q1Ety1xX9aNudQJ1YcL6yr7pGI\nIX3NANlp2Ra9Fr95ne9aEnwdMmGsQ5DjxHczEc
/GzSfOsErvA+EVWKmU8PiLl461s/OV25m0thG5+03yXKRsymX371\nXAg+hHqe2x5PRjw
/38GgB1XNAuDu0R\n3EyV01Umm6IrYDQWpngjGGmiimOdpLFHkQbxDNiRr8QX5eshAbVl
```

/DBhgvr2jnq9ed3dIKlHbrkw3sjBuwINZjw\naduL3U+WTUvUCY/VtlxJZdU1kVLwSnkDh+8HK
/eZ7AuHWjQjD9JzArCo5CCMMFJL\noY0IKxzhhP+4BmaMabwcuooxMjWR3fu3T0sgcTEZtG61wcSUDW0gw6c5QAxmq7It\nqzP2b1eNPp05oMJ6A
LIe+8MQMM94HigbSiLB3
/rFS8KkhZcdJliBc+Ig6TBFx9QW\nS0Jh4WgJn0B5laiI7DRp0E9bUUnLLEFTdA9P9T1DcIwngPuv6IYNQdzYluaX6cvy\nNhCH+XdbaFkA9KOsp
69uZWqzweoejAo24Cj71J9H4yMzBDWi7/fL4YQqjS6zC9JY\ny3zhk8VGi9SYtMB1bPdmxBlCyLElZ6qf
/cyjsWN89oTTITCYbSuIrB4piJH35t17\nd7eFZ7QXMampJzCQyAcKsxTDVdeKhHjVxsnSWuvmlR31Hmrxw3yQQH2pbGLcHBWJ\ngz+
/xpgxh5x0dGzqOKqgfGOtBOSpzHFMuuoXToYbcAIwMVRcTPnVR7B1kOm2OiLG\nhuOxX29DypSM9HjsmoeffJaUoZ2wvBK4QZNpe5Jb80An/aO+8
/oKmtaZgJqectsM\nfrVSLZtdPnH62lPy1i5CnoFI6JkX7oficJw8YQqswRp2z5HL9cSEAiR3MOr/Yco+\njJu5IidT3u5+hUlIdZtEtA==”]

```
    tenant_networks: [ providerExternal ]

storage:
    backends:
        lvm:
            enabled: false

        ceph:
            osd_pool_default_size: 2
            enabled: true

network_profiles:
    controller_network:
        linux_bonding_options: "mode=lacp"
        ovs_bonding_options: "mode=lacp"

        bonding_interfaces:
            bond0: [enp94s0f0,enp94s0f1]
            bond1: [enp135s0f0,enp135s0f1]

        interface_net_mapping:
            bond0: [infra_internal, infra_external, infra_storage_cluster]

        provider_network_interfaces:
            bond1:
                type: caas
                provider_networks: [ providerInternal, providerExternal ]

    compute_network:
        linux_bonding_options: "mode=lacp"
        ovs_bonding_options: "mode=lacp"

        bonding_interfaces:
            bond0: [ens94s0f0,ens94s0f1]
            bond1: [enp135s0f0,enp135s0f1]

        interface_net_mapping:
            bond0: [ infra_internal ]

        provider_network_interfaces:
            bond1:
                type: caas
                provider_networks: [ providerInternal, providerExternal ]

performance_profiles:
    caas_cpu_profile:
        caas_cpu_pools:
            exclusive_pool_percentage: 34
            shared_pool_percentage: 66
        tuning: standard

storage_profiles:
    caas_worker_docker_profile:
        lvm_instance_storage_partitions: ["1"]
        backend: bare_lvm
        lv_name: docker

    ceph_backend_profile:
        backend: ceph
        nr_of_ceph_osd_disks: 2
        ceph_pg_openstack_caas_share_ratio: "0:1"

hosts:
```

```
    controller-1:
        service_profiles:     [ caas_master, storage ]
        network_profiles:     [ controller_network ]
        storage_profiles:     [ ceph_backend_profile ]
        performance_profiles: [ caas_cpu_profile ]
        network_domain: rack-1
        hwmgmt:
            address:  192.166.10.211
            user:     root
            password: c5zgUQ6f

    controller-2:
        service_profiles:     [ caas_master, storage ]
        network_profiles:     [ controller_network ]
        storage_profiles:     [ ceph_backend_profile ]
        performance_profiles: [ caas_cpu_profile ]
        network_domain: rack-1
        hwmgmt:
            address:  192.166.10.212
            user:     root
            password: c5zgUQ6f

    controller-3:
        service_profiles:     [ caas_master, storage ]
        network_profiles:     [ controller_network ]
        storage_profiles:     [ ceph_backend_profile ]
        performance_profiles: [ caas_cpu_profile ]
        network_domain: rack-1
        hwmgmt:
            address:  192.166.10.213
            user:     root
            password: c5zgUQ6f

host_os:
    grub2_password: grub.pbkdf2.sha512.10000.
CC6F56BFCFB90C49E6E16DC7234BF4DE4159982B6D121DC8EC6BF0918C7A50E8604CA40689A8B26EA01BF2A76D33F7E6C614E6289ABBAA69
44ECB2B6DEB2F3CF.
4B929016A827C36142CC126EB47E86F5F98E92C8C2C924AD0C98436E4699DF7536894F69BB904FDB5E609B9A5D67E28A7D79E8521C0B0AE6
C031589FA0452A21
...
```

YAML Requirements

- The YAML files need to edited/created using Linux editors or in Windows Notepad++
- YAML files do not support TABS. You must space over to the location for the text.

Note: You have a better chance at creating a working YAML by editing an existing file or using the template rather than starting from scratch.

Installing REC

Obtaining the ISO Image

Recent builds can be obtained from the Akraino Nexus server. Choose either "latest" or a specific build number from the old release images directory for builds prior to the AMD/ARM split or the AMD64 builds or the ARM64 builds and download the file install.iso.

Akraino Release	REC or TA ISO Build	Build Date	Notes
1	Build 9. This build has been removed from Nexus (probably due to age)	2019-05-30	Build number 9 is known to NOT work on Dell servers or any of the ARM options listed below. If attempting to install on Dell servers, it is suggested to use builds from no earlier than June 10th

Report a B

2	Build 237. This build has been removed from Nexus (probably due to age)	2019-11-18	It is possible that there may still be some issues on Dell servers. Most testing has been done on Open Edge. Some builds between June 10th and November 18th have been successfully used on Dell servers, but because of a current lack of Remote Installer support for Dell (or indeed anything other than Open Edge), the manual testing is not as frequent as the automated testing of REC on Open Edge. If you are interested in testing or deploying on platforms other than Open Edge, please join the Radio Edge Cloud Project Meetings.
3 - AMD64	Build 237. This build has been removed from Nexus (probably due to age)	2020-05-29	This is a minor update to Akraino Release 2 of AMD64 based Radio Edge Cloud
3 - ARM64	Arm build 134. This build has been removed from Nexus (probably due to age)	2020-04-13	This is the first ARM based release of Radio Edge Cloud
4 - AMD64	AMD64 build 239	2020-11-03	The ARM build is unchanged since Release 3

Options for booting the ISO on your target hardware include NFS, HTTP, or USB memory stick. You must place the ISO in a suitable location (e.g., NFS server, HTTP(S) server or USB memory stick before starting the boot process. The file bootcd.iso, which is also in the same directory, is used only when deploying via the Akraino Regional Controller using the Telco Appliance Remote Installer. You can ignore bootcd.iso when following the manual procedure below.

Preparing for Boot from ISO Image

Nokia OpenEdge Servers

Using the BMC, configure a userid and password on each blade and ensure that the VMedia Access checkbox is checked.

The expected physical configuration as described in Radio Edge Cloud Validation Lab is that each server in the cluster has two SSD 480GB SATA 1dwpd M.2 2280 on a riser card inside the server and two SSD 960GB SATA 3dwpd 2.5 inch on the front panel. There is no RAID configuration used. The reference implementation in the Radio Edge Cloud Validation Lab uses one M.2 drive as the physical volume for LVM and both 2.5 inch SSDs as Ceph volumes.

HP Servers

Dell Servers

Provision the disk configuration of the server via iDRAC such that the desired disks will be visible to the OS in the desired order. The installation will use /dev/sda as the root disk and /dev/sdb and /dev/sdc as the Ceph volumes.

Ampere Servers

Darrin Vallis

Download and print hardware configuration guide for REC test installation

> Each server requires 2 SSDs

Report a B

Each server requires 3 NIC ports and 1 BMC connection

"Dumb" switch or vLAN is connected to two NIC ports

1 NIC port and BMC are connected to router via switch

REC ISO will recognize Ampere based Lenovo HR330 1U, HR350 2U or openEDGE sleds with "Hawk" motherboard

Designate 1 server as Node 1. It runs all code to complete the installation

Boot each server with a monitor attached to the VGA port. Note the BMC IP address.

Boot note 1 into Operating System. Note Linux names for all Ethernet ports on hardware guide.

Download and edit user_config.yaml

cidr is range of IPs

infra_external has network access

infra_internal is VLAN or dumb switch without network access

infra_storage are IPs on internal network used for storage

Interface_net_mapping must be set with NIC port names previously obtained for Node 1

hwmgmt is IP addresses of all BMCs. Node 1 is the master

Marvell Servers

@ Carl Yang <carlyang@marvell.com>

Booting from the ISO Image

Nokia OpenEdge Servers

Login to the controller-1 BMC ip using a web browser (https://xxx.xxx.xxx.xxx).

Go to **Settings/Media Redirection/General Settings**.

Select the **Remote Media Support**.

Select the **Mount CD/DVD**.

Type the NFS server IP address.

Type the NFS share path.

Select the nfs in **Share Type for CD/DVD**.

Click Save.

Click OK to restart the **VMedia Service**.

Go to **Settings/Media Redirection/Remote Images**.

Select the image for the first CD/DVD device from the drop-down list.

Click the play button to map the image with the server's CD /DVD devices. The Redirection Status changes to Started when the image redirection succeeds.

Go to **Control & Maintain /Remote Control** to open the **Remote Console**.

Reset the server.

Press F11 to boot menu and select boot from CD/DVD device.

HP Servers

Login to iLo for Controller 1 for the installation

Go to Remote Console & Media

Scroll to **HTML 5 Console**

- URL

  http://XXX.XXX.XXX.XX:XXXX/REC_RC1/install.iso -> Virtual Media URL

- NFS

  < IP to connect for NFS file system>/<file path>/install.iso

Check "Boot on Next Reset" -> Insert Media

Reset System

Dell Servers

Go to **Configuration/Virtual Media**

Scroll down to Remote File Share and enter the url for ISO into the **Image File Path** field.

- URL:

  http://XXX.XXX.XXX.XX:XXXX/REC_RC1/install.iso

- NFS

  < IP to connect for NFS file system>/<file path>/install.iso>

Select **Connect**.

Open **Virtual Console**, and go to **Boot**

Set **Boot Action** to **Virtual CD /DVS/ISO**

Then **Power/Reset System**

Be sure to read the note below on Dell servers

Ampere Servers

Darrin Vallis

Download install.aarch64.iso from the latest Telco Appliance build. A complete list of ARM aarch64 builds is available here.

Mount install.aarch64.iso as NFS share on Linux file system in same network as REC cluster

Download and unzip ampere_virtual_media_v2.zip

Edit mount_media.sh and dismount_media.sh. Set IPMI_LOCATION to IP address of Node 1 BMC, NFS_IP to IP of NFS server, ISO_LOCATION to NFS path for install.aarch64.iso

Run mount_media.sh. This will connect install.aarch64.iso as a CDROM on Node 1

Boot Node 1 into BIOS. Force boot from CD by selecting "Save & Exit" tab on BIOS, Boot Override CDDROM

REC Telco Appliance will begin installation

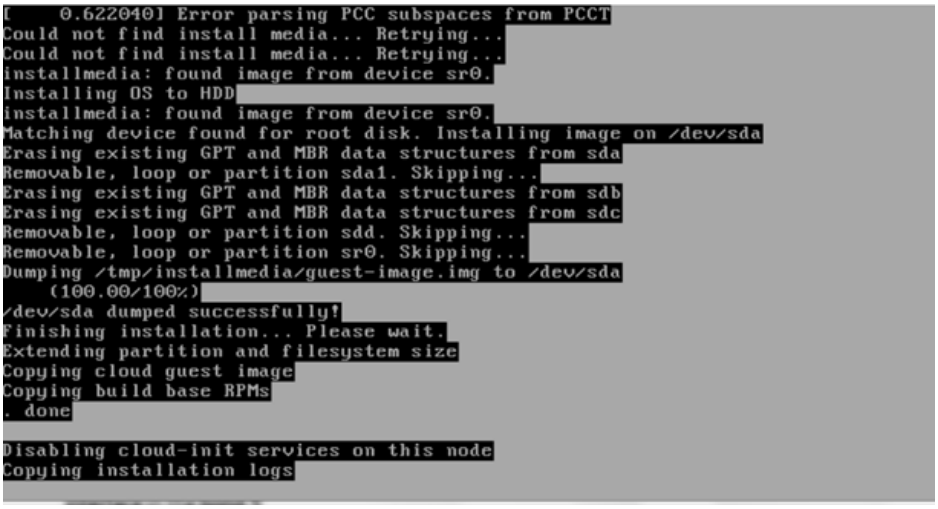
See instructions below

Marvell Servers

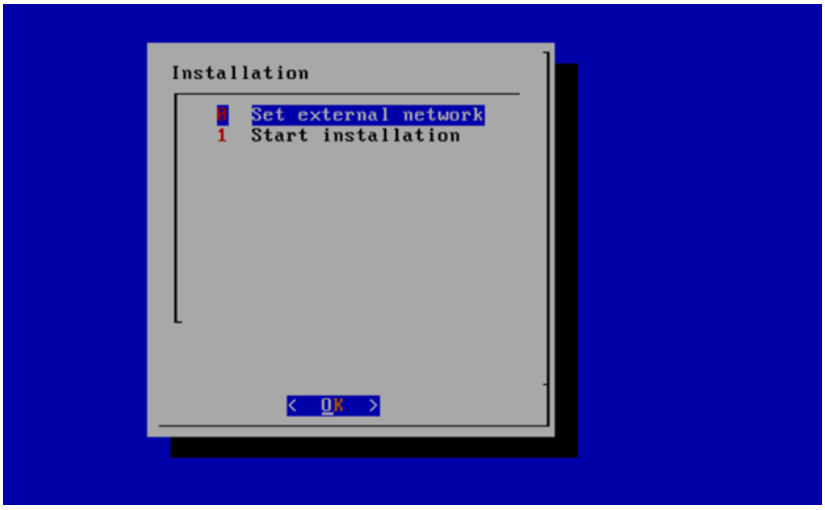
@ Carl Yang <carlyang@marvell.com>

After rebooting, the installation will bring up the Akraino Edge Stack screen.

The first step is to clean all the drives discovered before installing the ISO image.

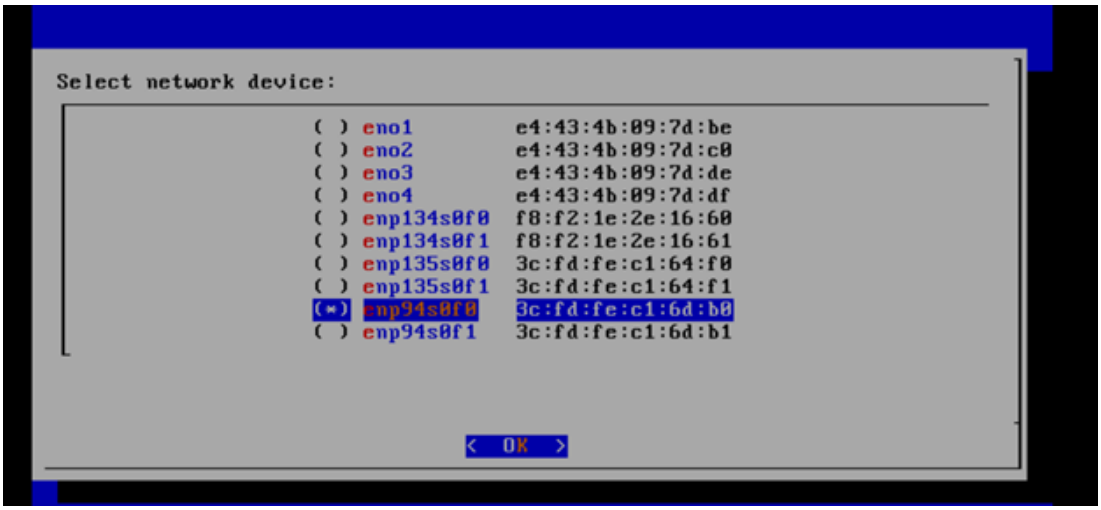


Select, **0 Set external network** at the Installation window, press **OK.**



Arrow down to and press the spacebar to select the **network interface** to be used for the **external network**.

If using **bonded nics**, select the **first interface** in the **bond**.

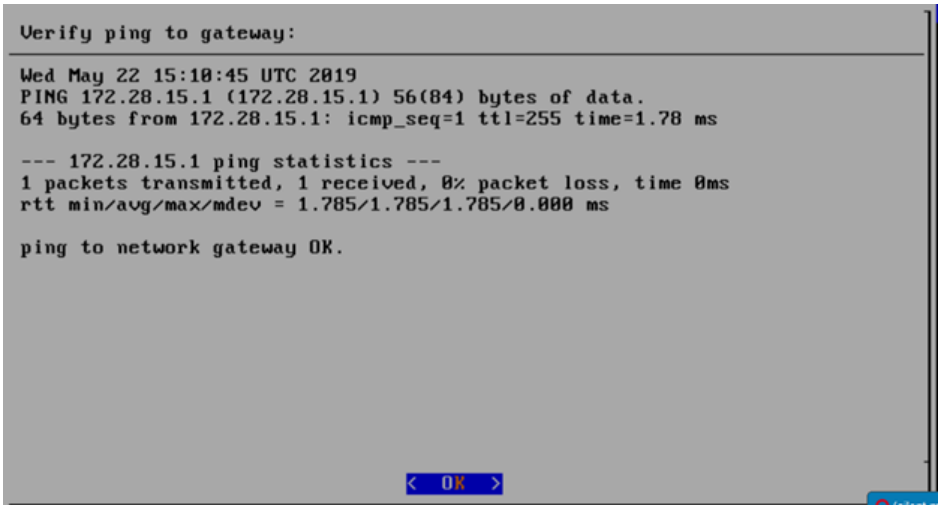


Enter the **external ip address with CIDR** for **controller-1: 172.28.15.211/24**

Enter the **gateway** ip address for the external ip address just entered: **172.28.15.1**

Enter the **VLAN** number: **141**

The installation will check the link and connectivity of the IP addresses entered.



If the connectivity test passed, then **Installation** window will return.

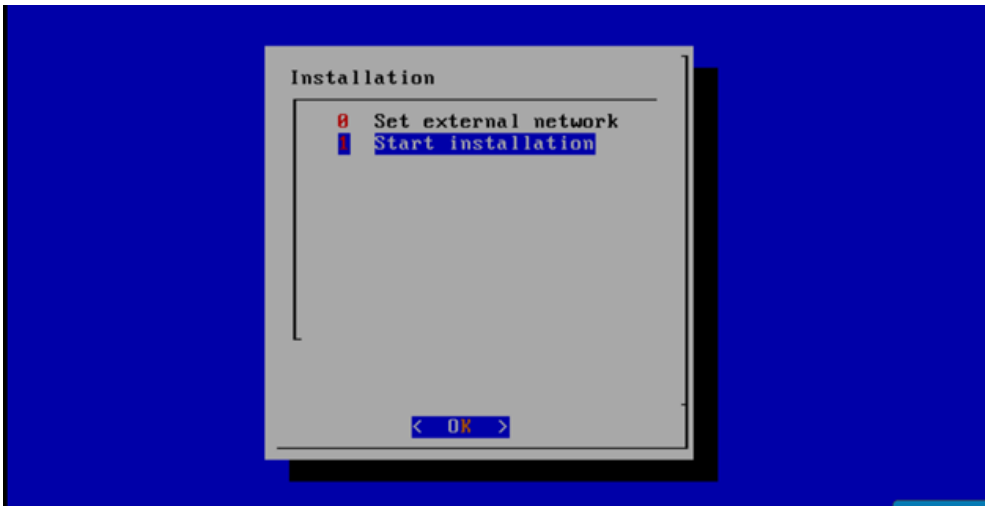
Uploading user_config.yaml

Go to your RC or jump server and scp (or sftp) your user_config.yaml to controller-1's /etc/userconfig directory.

initial credentials: root/root.

scp user_config.yaml root@<controller-1 ip address>/etc/userconfig/

Select, **1 Start installation** and **OK**.



After selecting **Start Installation**, the installation should start automatically, and the content of /srv/deployment/log/bootstrap.log should be displayed on the remote console.

Monitoring Deployment Progress/Status

You can monitor the REC deployment by checking the remote console screen or by tailing the logs on controller-1 node's /srv/deployment/log/ directory.

There are two log files:

bootstrap.log: deployment status log

cm.log: ansible execution log

tail -f /srv/deployment/log/cm.log

tail -f /srv/deployment/log/bootstrap.log

Note: When the deployment to all the nodes has completed, “controller-1” will reboot automatically.

Special Attention Required on Dell

A Note on deploying on DELL severs:

Currently, a manual step is required when doing an installation on Dell servers. After the networking has been set up and the deployment has started, the following message will be shown on the console screen on controller-2 and controller-3:

```
Strike F1 to retry boot, F2 for system setup, F10 for lifecycle controller, F11
for boot manager.
Note that in F2/F10/F11 cases a system reboot will be initiated

Booting from Hard drive C:

Booting from IBA 40G Slot 1900 v1066

Intel(R) Boot Agent 40G v1.0.66
Copyright (C) 1997-2016, Intel Corporation

PXE-E61: Media test failure, check cable
PXE-M0F: Exiting Intel Boot Agent.

No boot device available.
Current boot mode is set to BIOS.
Please ensure compatible bootable media is available.
Use the system setup program to change the boot mode as needed.
```

At this point, both controller-2 and controller-3 should be set to boot from virtual CD/DVD/ISO.

To do this:

- Log on to the iDrac web interface
- Select "Launch Virtual Console"
- In the Virtual Console:
  - Select "Boot | Virtual CD/DVD/ISO" and confirm
  - Select "Power | Reset System (Warm Boot)" and confirm

Again, this needs to be done for both controller-2 and controller-3. After this, the installation should continue normally.

As a reference, during this time, viewing the file /srv/deployment/log/cm.log on controller-1 will show the following:

FAILED - RETRYING: Verify node provisioning state. Waiting for 60mins max. (278 retries left).

FAILED - RETRYING: Verify node provisioning state. Waiting for 60mins max. (277 retries left).

FAILED - RETRYING: Verify node provisioning state. Waiting for 60mins max. (276 retries left).

This will continue until the above manual step is completed or a timeout happens. After the manual step, the following messages will appear:

ok: [controller-2 -> localhost]

ok: [controller-3 -> localhost]

Verifying Deployment

A post-installation verification is required to ensure that all nodes and services were properly deployed.

You need to establish an ssh connection to the controller's VIP address and login with administrative rights.

tail /srv/deployment/log/bootstrap.log

You should see: *Installation complete, Installation Succeeded*.

Go to REC Test Document and follow the steps outlined there to ensure that all nodes and services were properly deployed.

Deployment Failures

Sometimes failures happen, usually due to misconfigurations or incorrect addresses entered.

Report a B

To re-launch a failed deployment

There are two options for redeploying. (Execute as root)

1. /opt/cmframework/scripts/bootstrap.sh /etc/userconfig/user_config.yaml --install &
2. openvt -s -w /opt/start-menu/start_menu.sh &

Note: In some cases modifications to the user_config.yaml may be necessary to resolve a failure.

If re-deployment is not possible, then the deployment will need to be started from booting to the REC.iso,

RIC R0 Installation onto REC

REC Release R1 includes the ability to run the R0 version of the RIC.
This R0 is limited functionality of the RIC plaform, but demonstrates the basic RIC platform components of appmgr, rtmgmr, redis, e2term, etmgr.

Additionally it is possible to load robot test suites to verify functionality, but not all test cases will work in this version of RIC.
As more functionality becomes available, more test cases will work and more tests will be added.

Onboarding RIC R0 is a manual step at present, in future it will be included in the REC build process.
RIC R0 includes scripts to bring up the RIC onto a generic kubernetes platform.
To bring it up on the REC, follow the steps below.

Step 1:

Login to the controller 1 as cloudadmin and clone the scripts used to bring up the RIC on a REC cluster.

```
git clone https://gerrit.akraino.org/r/rec.git
```

Step 2:

Copy the scripts to your home directory

```
cp rec/workflows/ric_automation.sh rec/workflows/robot_test_ric.sh rec/workflows/nanobot.sh ./
```

Step 3:

Run the ric_automation.sh script.

```
bash ric_automation.sh
```

Once the script completes, verify the output below indicating successful deployment of the RIC helm charts.

```
+ helm install localric/ric --namespace ricplatform --name ric-full --set appmgr.appmgr.service.appmgr.extport=30099 --set e2mgr.e2mgr.service.http.
extport=30199
```

```
LAST DEPLOYED: Fri Jun 28 17:42:53 2019
NAMESPACE: ricplatform
STATUS: DEPLOYED

RESOURCES:
==> v1/ConfigMap
NAME DATA AGE
ric-full-appmgr-appconfig 1 0s
ric-full-appmgr-appenv 1 0s
ric-full-e2mgr-router-configmap 1 0s
ric-full-e2term-router-configmap 1 0s

==> v1/Deployment
NAME READY UP-TO-DATE AVAILABLE AGE
ric-full-appmgr 0/1 1 0 0s
ric-full-dbaas 0/1 1 0 0s
ric-full-e2mgr 0/1 1 0 0s
ric-full-e2term 0/1 1 0 0s
ric-full-rtmgr 0/1 1 0 0s

==> v1/Pod(related)
NAME READY STATUS RESTARTS AGE
ric-full-appmgr-74b4f68459-rhwf6 0/1 ContainerCreating 0 0s
ric-full-dbaas-877f5788d-rpg87 0/1 ContainerCreating 0 0s
ric-full-e2mgr-f6956b9f8-kxc6q 0/1 ContainerCreating 0 0s
ric-full-e2term-f6556544c-pzxgv 0/1 ContainerCreating 0 0s
ric-full-rtmgr-95f7cb5cc-bfhdx 0/1 ContainerCreating 0 0s

==> v1/Service
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
ric-full-appmgr ClusterIP 10.254.254.49 <none> 8080/TCP 0s
ric-full-dbaas ClusterIP 10.254.188.227 <none> 6379/TCP 0s
ric-full-e2mgr ClusterIP 10.254.93.207 <none> 3800/TCP,3801/TCP 0s
ric-full-e2term ClusterIP 10.254.208.204 <none> 38000/TCP 0s
ric-full-rtmgr ClusterIP 10.254.70.83 <none> 5656/TCP 0s
```

Step 4:
Verify the RIC pods are all coming up and running.

kubectl get pods -n ricplatform

```
NAME READY STATUS RESTARTS AGE
ric-full-appmgr-74b4f68459-rhwf6 1/1 Running 0 2m5s
ric-full-dbaas-877f5788d-rpg87 1/1 Running 0 2m5s
ric-full-e2mgr-f6956b9f8-kxc6q 1/1 Running 0 2m5s
ric-full-e2term-f6556544c-pzxgv 1/1 Running 0 2m5s
ric-full-rtmgr-95f7cb5cc-bfhdx 1/1 Running 0 2m5s
```

Note: It may take a little time so repeat the command at some intervals. If they do not come up as Running, you should use standard kubernetes command such as kubectl describe, kubectl logs to troubleshoot and resolve the issue.

Step5:
Now that you have running RIC R0, on the REC, you can install robot test scripts to do more verification of the RIC components. These at present only provide limited testing but more tests will be added later as more functionality is developed in later RIC releases.