

Install Guide - Regional Controller Node - Akraino Edge Stack Network Cloud Blueprint Family

PLEASE REFER TO R1 NETWORK CLOUD RELEASE DOCUMENTATION

[NC Family Documentation - Release 1](#)

THIS DOCUMENTATION WILL BE ARCHIVED

This guide instructs how to build and install an Akraino Edge Stack (AES) Regional Controller node.

Contents

- [Overview](#)
 - [Operating System](#)
 - [Regional Controller](#)
 - [Supplementary Components](#)
- [Prerequisites](#)
 - [Build Server](#)
 - [Bare Metal Server](#)
 - [System Check](#)
 - [Build Server](#)
 - [Network Connectivity](#)
- [Installation on Bare Metal server](#)
 - [Repository Cloning](#)
 - [Configuration](#)
 - [Operating System](#)
 - [Regional Controller](#)
- [Installation on an Existing Ubuntu Server or Virtual Machine](#)
 - [Repository Cloning](#)
 - [Regional Controller Installation](#)
- [Akraino Portal Operations](#)
 - [Login](#)

Overview

The 'Build Server' remotely installs the OS and SW packages on a 'Bare Metal' server to create the 'Regional Controller' (i.e. the Bare Metal servers becomes the Regional Controller). Once the RC is build it is used to subsequently deploy either Rover or Unicycle pods. After the Build Server has completed the creation of the Regional Controller node, the Build Server has no further role in any Network Cloud Rover and Unicycle Pod deployment. The Regional Controller Node installation includes the following components:

Operating System

- Redfish Integrated Dell Remote Access Controller (iDRAC) bootstrapping and hardware configuration
- Linux OS (Ubuntu)

Regional Controller

- PostgreSQL DB
- Camunda Workflow and Decision Engine
- Akraino Web Portal
- LDAP configuration

Supplementary Components

Various supporting files are also installed on the Regional Controller, including:

- OpenStack Tempest tests
- YAML builds
- ONAP scripts
- Sample VNFs

NOTE: The Regional Controller can be installed on an existing Ubuntu 16.04 server or virtual machine. See the instructions below labeled: Installation on an Existing Ubuntu Server or Virtual Machine



This installation guide refers to the following by way of example:

- **192.168.2.43 (aknode43):** Build Server (Linux Server with Docker installed)
- **192.168.2.44 (aknode44):** Bare Metal Server on which the Regional Controller will be installed
- **192.168.41.44:** Bare Metal Server iDRAC on which the Regional Controller will be installed

Steps herein presume the use of a root account. All steps are performed from the Build Server.

A clean, out-of-the-box Ubuntu environment is *strongly* recommended before proceeding.

Prerequisites

AES Regional Controller installation is orchestrated from a **Build Server** acting upon a **Bare Metal Server**.

Build Server

- Any server or VM with Ubuntu Release 16.04
- Latest version of the following apt packages:
 - docker (used to run dhcp and web containers)
 - python (used for redfish api calls to bare metal server)
 - python-requests (used for redfish api calls to bare metal server)
 - python-pip (used to install hpe redfish tools)
 - sshpass (used to copy keys to new server)
 - xorriso (used to extract Ubuntu files to web server)
 - make (used to build custom ipxe efi file used during bare metal server boot)
 - gcc (used to build custom ipxe efi file used during bare metal server boot)

Bare Metal Server

- Dell PowerEdge R740 Gen 14 server or HP DL380 Gen10 with no installed OS [Additional types of hardware will be supported in the future release]
- Two interfaces for primary network connectivity bonding
- 802.1q VLAN tagging for primary network interfaces

System Check

Build Server

Ensure **Ubuntu Release 16.04** (specifically) and Docker version is 1.13.1 or newer:

```
# lsb_release -rs
16.04
# docker --version
Docker version 1.13.1, build 092cba3
```

Ensure required packages are installed including **python**, **python-requests**, **python-pip**, **sshpass**, **xorriso**, **make**, and **gcc** are installed. Install any missing packages with apt-get install -y <package name>

```
# apt list python python-requests python-pip sshpass xorriso make gcc
Listing... Done
gcc/xenial,now 4:5.3.1-1ubuntu1 amd64 [installed]
make/xenial,now 4.1-6 amd64 [installed,automatic]
python/xenial-updates,now 2.7.12-1~16.04 amd64 [installed]
python-pip/xenial-updates,xenial-updates,now 8.1.1-2ubuntu0.4 all [installed]
python-requests/xenial-updates,xenial-updates,now 2.9.1-3ubuntu0.1 all [installed]
sshpass/xenial,now 1.05-1 amd64 [installed] xorriso/xenial,now 1.4.2-4ubuntu1 amd64 [installed]
```

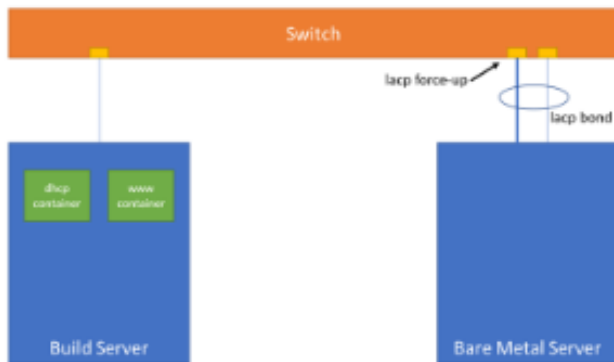
Network Connectivity

The Build Server *must* have connectivity to the Bare Metal Server iDRAC interface on ports 80 (http) and 443 (https).

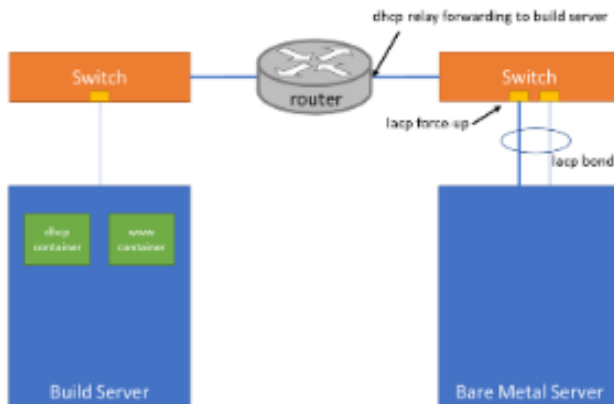
- The Bare Metal Server iDRAC/iLO interface and bonded production interfaces *must* be reachable from the Build Server.
- The Build Server and Bare Metal Server primary networks must have one of the following characteristics:
 - The networks must be located on the same L2 network, *or*
 - DHCP requests must be forwarded from the Bare Metal Server primary network interface to the Build Server (e.g., via a DHCP relay /helper).
- The switch configuration for the network bond of the Bare Metal Server primary interfaces must be set to bring up the bond on the first interface prior to lacp completing negotiation.
 - For Junos OS this option is typically called **force-up** and should be set on the first interface in the bond.
 - For Arista, this option is typically called **lacp fallback**.
 - Please refer to your network switch documentation to determine the correct configuration.

Here are logical views of two supported topologies as examples:

Sharing same layer 2 network



Separate layer 2 networks



Specific steps to achieve this connectivity are beyond the scope of this guide. However, some verification can be performed.

First, verify that at least port 443 is open on the Bare Metal Server iDRAC/iLO interface:

```
# nmap -sS 192.168.41.44
Starting Nmap 7.01 ( https://nmap.org ) at 2018-07-10 13:55 UTC Nmap scan report for 192.168.41.44 Host is up
(0.00085s latency). Not shown: 996 closed ports PORT STATE SERVICE 22/tcp open ssh 80/tcp open http 443/tcp
open https 5900/tcp open vnc Nmap done: 1 IP address (1 host up) scanned in 1.77 seconds
```

Next, use nmap to check for a "clean slate" Bare Metal Server. The results will show the host as being down (due to no OS).

```
# nmap -sS 192.168.2.42

Starting Nmap 7.01 ( https://nmap.org ) at 2018-07-10 13:55 UTC
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.63 seconds
```

Verification of the Build Server and Bare Metal Server primary networks is beyond the scope of this guide.

Installation on Bare Metal server

Repository Cloning

Repositories are located under /opt/akraino. On the Build Server:

```
mkdir /opt/akraino
```

Clone the Redfish Bootstrapping Script repository for use as part of Akraino tools. On the Build Server:

```
## Download the latest redfish artifacts from LF Nexus

mkdir -p /opt/akraino/redfish
NEXUS_URL=https://nexus.akraino.org
curl -L "$NEXUS_URL/service/local/artifact/maven/redirect?r=snapshots&g=org.akraino.redfish&a=redfish&v=0.0.2-SNAPSHOT&e=tgz" | tar -xozv -C /opt/akraino/redfish
```

Clone the Akraino Regional Controller repository. On the Build Server::

```
## Download the latest Regional_controller artifacts from LF Nexus ##

mkdir -p /opt/akraino/region
NEXUS_URL=https://nexus.akraino.org
curl -L "$NEXUS_URL/service/local/artifact/maven/redirect?r=snapshots&g=org.akraino.regional_controller&a=regional_controller&v=0.0.2-SNAPSHOT&e=tgz" | tar -xozv -C /opt/akraino/region
```

Configuration

Copy the Bare Metal Server configuration template into /opt/akraino/server-config/AKRAIN0_NODE_RC, where AKRAIN0_NODE_RC is the Bare Metal Server name followed by rc. On the Build Server:

```
mkdir -p /opt/akraino/server-config
cp /opt/akraino/redfish/serverrc.template /opt/akraino/server-config/aknode44rc
vi /opt/akraino/server-config/aknode44rc
```

A sample configuration file for the Regional Controller follows. Ensure the following:

- SRV_NAME is the Bare Metal Server name
- SRV_00B_IP is the Bare Metal Server iDRAC or iLO IP
- SRV_IP is the Bare Metal Server IP
- Passwords are chosen for SRV_00B_PWD and SRV_PWD
- All remaining SRV_ prefixed options are adjusted as appropriate for the Bare Metal server and network

```

# host name for server
SRV_NAME=aknode44

# out of band interface information for server (idrac/ilo/etc)
SRV_OOB_IP=192.168.41.44
SRV_OOB_USR=root
SRV_OOB_PWD=ROOT_PASSWORD

# mac address of server to be used during the build - not required for Dell servers
# SRV_MAC=3c:fd:fe:b8:10:60

# the boot device is the device name on which the OS will be loaded
SRV_BOOT_DEVICE=sdg

# ipxe script to use - based on the os version and kernel to install
# valid options are script-hwe-16.04.5-amd64.ipxe or script-16.04.5-amd64.ipxe
SRV_BLD_SCRIPT=script-hwe-16.04.5-amd64.ipxe

# template xml file to set bios and raid configuration settings
SRV_BIOS_TEMPLATE=dell_r740_g14_uefi_base.xml.template
SRV_BOOT_TEMPLATE=dell_r740_g14_uefi_httpboot.xml.template
SRV_HTTP_BOOT_DEV=NIC.Slot.7-1-1

# VLAN to use during build and for final network configuration
SRV_VLAN=41

# basic network information for dhcp config and final server network settings
SRV_MTU=9000
SRV_IP=192.168.2.44      #Note: This address is the same as the 'TARGET_SERVER_IP' defined below and in the
akrainorc file#
SRV_SUBNET=192.168.2.0
SRV_NETMASK=255.255.255.0
SRV_GATEWAY=192.168.2.200
SRV_DNS=192.168.2.85
SRV_DOMAIN=lab.akraino.org
SRV_DNSSEARCH=lab.akraino.org
SRV_NTP=ntp.ubuntu.org

# root password for server being built
SRV_PWD=SERVER_PASSWORD

# network bond information
SRV_BOND=bond0
SRV_SLAVE1=enp135s0f0
SRV_SLAVE2=enp135s0f1

```

Operating System

Begin the Regional Controller OS installation from the Build Server. You must be root on the Build Server to fully install:

```
/opt/akraino/redfish/install_server_os.sh --rc /opt/akraino/server-config/aknode44rc --skip-confirm
```

This will take time. This is an excellent time to enjoy a favorite beverage. 😊



In Case of Errors

The Operating System installation may produce errors. As of this writing, the following errors may be safely ignored:

```
- FAIL, detailed job message is: [{u'Message': u'Staged component configuration completed with errors.',
u'MessageId': u'SYS033', u'MessageArgs': [], u'MessageArgs@odata.count': 0}]

- FAIL: detailed error message: {"error":{"@Message.ExtendedInfo":[{"Message":"Unable to create a
configuration job because an existing configuration job is already in progress.", "MessageArgs":[], "Message
Args@odata.count":0, "MessageId":"iDRAC.1.6.RAC052", "RelatedProperties":[], "RelatedProperties@odata.count":
0, "Resolution":"Retry the operation after the existing configuration job is complete, or cancel the
existing configuration job and retry the operation.", "Severity":"Warning"}], "code":"Base.1.0.
GeneralError", "message":"A general error has occurred. See ExtendedInfo for more information"}}
```

A successful installation will start and end as follows:

```
/opt/akraino/tools/install_server_os.sh --rc /opt/akraino/server-config/aknode44rc --skip-confirm
Beginning /opt/akraino/tools/install_server_os.sh as user [root] in pwd [/opt/akraino/server-config] with home [
/root] Tools are ready in [/opt/akraino] WARNING: Preparing to build server [aknode44] using oob ip
[192.168.41.44]. Beginning in 10 seconds ..... Beginning bare metal install of os at Mon Jul 2 18:57:32
UTC 2018 ... Processing triggers for libc-bin (2.23-0ubuntu10) ... SUCCESS: Completed bare metal install of
regional server [aknode44] at Mon Jul 2 20:09:35 UTC 2018 SUCCESS: Try connecting with 'ssh root@192.168.2.42'
as user root Elapsed time was 9 minutes and 22 seconds
```

Note that any time estimates (e.g., *"This step could take up to 15 minutes"*) and elapsed times are likely inaccurate. The total install time is longer, on the order of hours. Enjoy that beverage. 😊

Regional Controller

On the Build Server, update the Akraino run command (rc) file in /opt/akraino/region:

```
vim /opt/akraino/region/akrainorc
```

Set TARGET_SERVER_IP to the Bare Metal Server IP. This is the IP address of the Regional Controller. All other values may be left as-is.

```
export TARGET_SERVER_IP=192.168.2.44
```

On the Build Server, begin the Regional Controller's software installation:

```
/opt/akraino/region/install_akraino_portal.sh
```

This will take time. This is an excellent time to enjoy another favorite beverage. 😊



In Case of Errors

The Regional Controller installation is not idempotent at this time. If errors are encountered during this phase, it is recommended that the errors be triaged and resolved, followed by an Operating System re-installation. This will ensure a "clean slate" Bare Metal Server before trying again.

A successful installation will end as follows. Note that any time estimates (e.g., *"This step could take up to 15 minutes"*) and elapsed times are likely inaccurate.

```
...

Setting up tempest content/repositories
Setting up ONAP content/repositories
Setting up sample vnf content/repositories
Setting up airshipinabottle content/repositories
Setting up redfish tools content/repositories
SUCCESS: Portal can be accessed at http://192.168.2.44:8080/AECPortalMgmt/
SUCCESS: Portal install completed
```

The Regional Controller Node installation is now complete.

Please note: It will be necessary to generate rsa keys on the RC which must be copied and inserted into the 'genesis_ssh_public_key' attribute in site input yaml file used when subsequently deploying each Unicycle pod at any edge site controlled by the newly built RC.

Installation on an Existing Ubuntu Server or Virtual Machine

Repository Cloning

Manually install Ubuntu 16.04 on a physical server or virtual machine in your environment that will be used as the Regional Controller. On the server or virtual machine that you just installed, clone the Akraino Regional Controller repository:

```
## Download the latest Regional_controller artifacts from LF Nexus ##

mkdir -p /opt/akraino/region
NEXUS_URL=https://nexus.akraino.org
curl -L "$NEXUS_URL/service/local/artifact/maven/redirect?r=snapshots&g=org.akraino.
regional_controller&a=regional_controller&v=0.0.2-SNAPSHOT&e=tgz" | tar -xozv -C /opt/akraino/region
```

Regional Controller Installation

Change to the /opt/akraino/region directory and run the start_regional_controller.sh script:

```
cd /opt/akraino/region/
./start_akraino_portal.sh
```

The install should take 15 to 45 minutes depending on the speed of the internet connection. A successful installation will end as follows.

```
...

Setting up tempest content/repositories
Setting up ONAP content/repositories
Setting up sample vnf content/repositories
Setting up airshipinabottle content/repositories
Setting up redfish tools content/repositories
SUCCESS: Portal can be accessed at http://192.168.2.44:8080/AECPortalMgmt/
SUCCESS: Portal install completed
```

The Regional Controller installation is now complete.

Akraino Portal Operations

Login

Visit the portal URL http://REGIONAL_NODE_IP:8080/AECPortalMgmt/ where REGIONAL_NODE_IP is the Portal IP.

Use the following credentials:

- Username: *akadmin*
- Password: *akraino*

Upon successful login, the Akraino Portal home page will appear.



Next Steps

Edge Site Installation, please follow the [Install Guide Edge Site - Rover](#) or [Install Guide Edge Site - Unicycle](#) documentation.