Install Guide - Akraino Edge Stack Network Cloud Blueprint - Unicycle

PLEASE REFER TO R1 NETWORK CLOUD RELEASE DOCUMENTATION

NC Family Documentation - Release 1

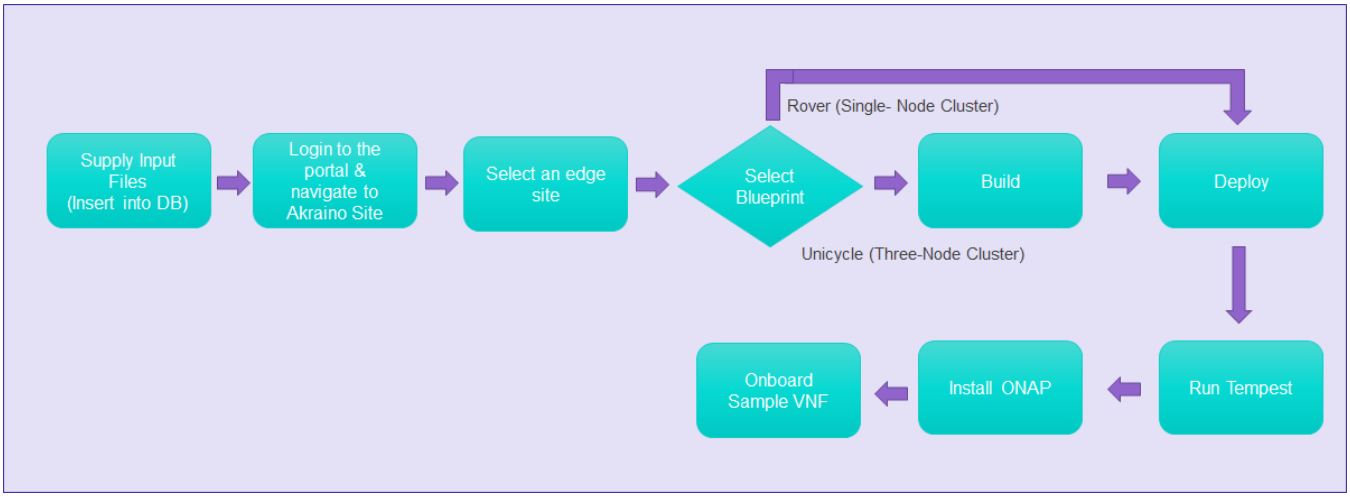
THIS DOCUMENTATION WILL BE ARCHIVED

Contents

- Introduction
- Hardware Requirements for Test
- Akraino Portal Operations
  - Login
  - Deploy a Multi-Node Edge Site
- Appendix
  - Create New Edge Site locations

Introduction

This document describes the steps to create a single and multi-node edge sites.



Unicycle (Three-Node Cluster) - Supported in the future release

Hardware Requirements for Test

Up to 7 servers (3 control plus 1 to 4 workers) x86 Dell R740 servers

Build Server

- Any server or VM with Ubuntu Release 16.04
- Packages: Latest versions of sshpass, xorriso, and python-requests
- Docker 1.13.1 or later

Bare Metal Server

- Dell PowerEdge R740 server with no installed OS [Additional types of hardware will be supported in the future release]
- Two interfaces for primary network connectivity bonding with DPDK enabled NIC
- 802.1q VLAN tagging for primary network interfaces

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:

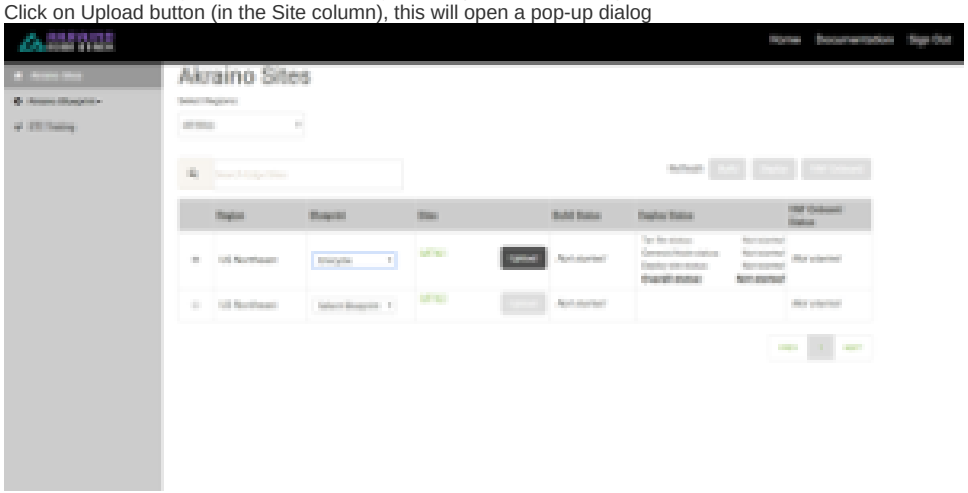
- Username: *akadmin*
- Password: *akraino*

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

Deploy a Multi-Node Edge Site

From the Portal home page:

1. Select an Edge Site - MTN1 or MTN2 (these are the two default lab sites hosted in middle town NJ) by clicking on radio button provided in the first column of the table.
2. For the selected Edge Site, select the Unicycle Blueprint from the drop-down menu.
3. Click on Upload button (in the Site column), this will open a pop-up dialog



Provide the edge site-specific details such as:

1. Host IP address
2. Host username
3. Host password.

Example: DELL Cluster:

- Host IP address: 192.168.2.40
- Host username: root
- Host password: XXXXXX

Example: HP Cluster

- Host IP address: 192.168.2.30
- Host username: root
- Host password: XXXXXX

4. Click on Browse button, select the input file for Blueprint - Unicycle (Multi-Node Cluster).
The input file is a property file that stores information in `key-value` format. Sample input file used for 'Unicycle' deploy:

Copy and paste the below contents in to a file, and save it as unicycle.yaml. Use this file for uploading as mentioned in step 4.d above. If using Dell Gen10, use the sample YAML input file shown in #1 below. If using HP Gen10, use the sample YAML file in #2 below.

Verify the configuration details as applicable to your environment. For more details refer to Appendix - Edge Site Configuration

Sample YAML Input File #1

```
---
################################################################################
# Copyright (c) 2018 AT&T Intellectual Property. All rights reserved.        #
#                                                                            #
# Licensed under the Apache License, Version 2.0 (the "License"); you may    #
# not use this file except in compliance with the License.                   #
#                                                                            #
# You may obtain a copy of the License at                                    #
#       http://www.apache.org/licenses/LICENSE-2.0                           #
#                                                                            #
# Unless required by applicable law or agreed to in writing, software        #
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT  #
# WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.           #
# See the License for the specific language governing permissions and        #
# limitations under the License.                                             #
################################################################################
site_name: dellgen10
ipmi_admin:
  username: root
  password: calvin
networks:
  bonded: yes
  primary: bond0
  slaves:
    - name: enp94s0f0
    - name: enp94s0f1
  oob:
    vlan: 40
    interface:
    cidr: 192.168.41.0/24
    routes:
      gateway: 192.168.41.1
    ranges:
      reserved:
        start: 192.168.41.2
        end: 192.168.41.12
      static:
        start: 192.168.41.13
        end: 192.168.41.254
  host:
    vlan: 41
    interface: bond0.41
    cidr: 192.168.2.0/24
    routes:
       gateway: 192.168.2.200
    ranges:
      reserved:
        start: 192.168.2.84
        end: 192.168.2.86
      static:
        start: 192.168.2.40
        end: 192.168.2.45
    dns:
      domain: lab.akraino.org
      servers: '192.168.2.85 8.8.8.8 8.8.4.4'
  storage:
    vlan: 42
    interface: bond0.42
    cidr: 172.31.2.0/24
    ranges:
      reserved:
```

```
        start: 172.31.2.1
        end: 172.31.2.10
      static:
        start: 172.31.2.11
        end: 172.31.2.254
  pxe:
    vlan: 43
    interface: eno3
    cidr: 172.30.2.0/24
    gateway: 172.30.2.1
    routes:
      gateway: 172.30.2.40
    ranges:
      reserved:
        start: 172.30.2.2
        end:  172.30.2.10
      static:
        start: 172.30.2.11
        end: 172.30.2.200
      dhcp:
        start: 172.30.2.201
        end: 172.30.2.254
    dns:
      domain: lab.akraino.org
      servers: '192.168.2.85 8.8.8.8 8.8.4.4'
  ksn:
    vlan: 44
    interface: bond0.44
    cidr: 172.29.1.0/24
    local_asnumber: 65531
    ranges:
      static:
        start: 172.29.1.5
        end: 172.29.1.254
    additional_cidrs:
      -  172.29.1.128/29
    ingress_cidr: 172.29.1.129/32
    peers:
    - ip: 172.29.1.1
      scope: global
      asnumber: 65001
    vrrp_ip: 172.29.1.1 # keep peers ip address in case of only peer.
  neutron:
    vlan: 45
    interface: bond0.45
    cidr: 10.0.102.0/24
    ranges:
      reserved:
        start: 10.0.102.1
        end: 10.0.102.10
      static:
        start: 10.0.102.11
        end: 10.0.102.254
dns:
  upstream_servers:
    - 192.168.2.85
    - 8.8.8.8
    - 8.8.8.8
  upstream_servers_joined: '192.168.2.85,8.8.8.8'
  ingress_domain: dellgen10.akraino.org
sriovnets:
- physical: sriovnet1
  interface: enp135s0f0
  vlan_start: 2001
  vlan_end: 3000
  whitelists:
  - "address": "0000:87:02.0"
  - "address": "0000:87:02.1"
  - "address": "0000:87:03.2"
  - "address": "0000:87:03.3"
  - "address": "0000:87:03.4"
```

```
  - "address": "0000:87:03.5"
  - "address": "0000:87:03.6"
  - "address": "0000:87:03.7"
  - "address": "0000:87:04.0"
  - "address": "0000:87:04.1"
  - "address": "0000:87:04.2"
  - "address": "0000:87:04.3"
  - "address": "0000:87:02.2"
  - "address": "0000:87:04.4"
  - "address": "0000:87:04.5"
  - "address": "0000:87:04.6"
  - "address": "0000:87:04.7"
  - "address": "0000:87:05.0"
  - "address": "0000:87:05.1"
  - "address": "0000:87:05.2"
  - "address": "0000:87:05.3"
  - "address": "0000:87:05.4"
  - "address": "0000:87:05.5"
  - "address": "0000:87:02.3"
  - "address": "0000:87:05.6"
  - "address": "0000:87:05.7"
  - "address": "0000:87:02.4"
  - "address": "0000:87:02.5"
  - "address": "0000:87:02.6"
  - "address": "0000:87:02.7"
  - "address": "0000:87:03.0"
  - "address": "0000:87:03.1"
- physical: sriovnet2
  interface: enp135s0f1
  vlan_start: 2001
  vlan_end: 3000
  whitelists:
  - "address": "0000:87:0a.0"
  - "address": "0000:87:0a.1"
  - "address": "0000:87:0b.2"
  - "address": "0000:87:0b.3"
  - "address": "0000:87:0b.4"
  - "address": "0000:87:0b.5"
  - "address": "0000:87:0b.6"
  - "address": "0000:87:0b.7"
  - "address": "0000:87:0c.0"
  - "address": "0000:87:0c.1"
  - "address": "0000:87:0c.2"
  - "address": "0000:87:0c.3"
  - "address": "0000:87:0a.2"
  - "address": "0000:87:0c.4"
  - "address": "0000:87:0c.5"
  - "address": "0000:87:0c.6"
  - "address": "0000:87:0c.7"
  - "address": "0000:87:0d.0"
  - "address": "0000:87:0d.1"
  - "address": "0000:87:0d.2"
  - "address": "0000:87:0d.3"
  - "address": "0000:87:0d.4"
  - "address": "0000:87:0d.5"
  - "address": "0000:87:0a.3"
  - "address": "0000:87:0d.6"
  - "address": "0000:87:0d.7"
  - "address": "0000:87:0a.4"
  - "address": "0000:87:0a.5"
  - "address": "0000:87:0a.6"
  - "address": "0000:87:0a.7"
  - "address": "0000:87:0b.0"
  - "address": "0000:87:0b.1"
storage:
  osds:
    - data: /dev/sda
      journal: /var/lib/ceph/journal/journal-sda
    - data: /dev/sdb
      journal: /var/lib/ceph/journal/journal-sdb
    - data: /dev/sdc
```

```
      journal: /var/lib/ceph/journal/journal-sdc
    - data: /dev/sdd
      journal: /var/lib/ceph/journal/journal-sdd
    - data: /dev/sde
      journal: /var/lib/ceph/journal/journal-sde
    - data: /dev/sdf
      journal: /var/lib/ceph/journal/journal-sdf
  osd_count: 6
  total_osd_count: 18
genesis:
  name: aknode40
  oob: 192.168.41.40
  host: 192.168.2.40
  storage: 172.31.2.40
  pxe: 172.30.2.40
  ksn: 172.29.1.40
  neutron: 10.0.102.40
masters:
  - name : aknode41
    oob: 192.168.41.41
    host: 192.168.2.41
    storage: 172.31.2.41
    pxe: 172.30.2.41
    ksn: 172.29.1.41
    neutron: 10.0.102.41
  - name : aknode42
    oob: 192.168.41.42
    host: 192.168.2.42
    storage: 172.31.2.42
    pxe: 172.30.2.42
    ksn: 172.29.1.42
    neutron: 10.0.102.42
hardware:
  vendor: DELL
  generation: '10'
  hw_version: '3'
  bios_version: '2.8'
disks:
  - name : sdg
    labels:
      bootdrive: 'true'
    partitions:
      - name: root
        size: 20g
        mountpoint: /
      - name: boot
        size: 1g
        mountpoint: /boot
      - name: var
        size: 100g
        mountpoint: /var
  - name : sdh
    partitions:
      - name: ceph
        size: 300g
        mountpoint: /var/lib/ceph/journal
disks_compute:
  - name : sdg
    labels:
      bootdrive: 'true'
    partitions:
      - name: root
        size: 20g
        mountpoint: /
      - name: boot
        size: 1g
        mountpoint: /boot
      - name: var
        size: '>300g'
        mountpoint: /var
  - name : sdh
```

```
    partitions:
      - name: nova
        size: '99%'
        mountpoint: /var/lib/nova
genesis_ssh_public_key: "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC
/n4mNLAj3XKG2fcm+8eVe0NUlNH0g8DA8KJ53rSLKccm8gm4UgLmGOJyBfUloQZMuOpU6a+hexN4ECCliqI7+KUmgJgsvLkJ3OUMNTEVu9tDX5md
XeffsufaqFkAdmbJ/9PMPiPQ3
/UqbbtyEcqoZAwUWf4ggAWSp00SGE1Okg+skPSbDzPVHb4810eXZT1yoIg29HAenJNNrsVxvnMT2kw2OYmLfxgEUh1Ev4c5LnUog4GXBDHQtHAwa
IoTu9s/q8VIvGav62RJVFn3U1D0jkiwDLSIFn8ezORQ4YkSidwdSrtqsqa2TJ0E5w/n5h5IVGO9neY8YlXrgynLd4Y+7 root@pocnjrsv132"
kubernetes:
  api_service_ip: 10.96.0.1
  etcd_service_ip: 10.96.0.2
  pod_cidr: 10.98.0.0/16
  service_cidr: 10.96.0.0/15
regional_server:
  ip: 135.16.101.85
...
```

Sample YAML Input File #2

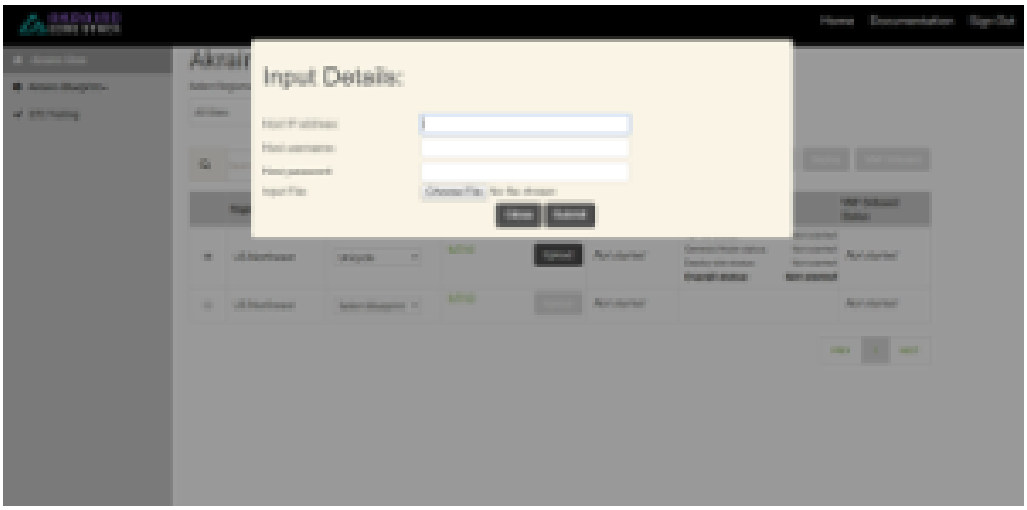
```
---
##############################################################################
# Copyright (c) 2018 AT&T Intellectual Property. All rights reserved.        #
#                                                                            #
# Licensed under the Apache License, Version 2.0 (the "License"); you may    #
# not use this file except in compliance with the License.                   #
#                                                                            #
# You may obtain a copy of the License at                                    #
#       http://www.apache.org/licenses/LICENSE-2.0                           #
#                                                                            #
# Unless required by applicable law or agreed to in writing, software        #
# distributed under the License is distributed on an "AS IS" BASIS, WITHOUT  #
# WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.           #
# See the License for the specific language governing permissions and        #
# limitations under the License.                                             #
##############################################################################
site_name: hpgen10
ipmi_admin:
  username: Administrator
  password: Admin123
networks:
  bonded: yes
  primary: bond0
  slaves:
    - name: ens3f0
    - name: ens3f1
  oob:
    vlan: 40
    interface:
    cidr: 192.168.41.0/24
    routes:
      gateway: 192.168.41.1
    ranges:
      reserved:
        start: 192.168.41.2
        end: 192.168.41.4
      static:
        start: 192.168.41.5
        end: 192.168.41.254
  host:
    vlan: 41
    interface: bond0.41
    cidr: 192.168.2.0/24
    subnet: 192.168.2.0
    netmask: 255.255.255.0
    routes:
       gateway: 192.168.2.200
    ranges:
```

```
      reserved:
        start: 192.168.2.84
        end: 192.168.2.86
      static:
        start: 192.168.2.1
        end: 192.168.2.83
    dns:
      domain: lab.akraino.org
      servers: '192.168.2.85 8.8.8.8 8.8.4.4'
  storage:
    vlan: 42
    interface: bond0.42
    cidr: 172.31.1.0/24
    ranges:
      static:
        start: 172.31.1.2
        end: 172.31.1.254
  pxe:
    vlan:
    interface: eno1
    cidr: 172.30.1.0/24
    gateway: 172.30.1.1
    routes:
       gateway: 172.30.1.30
    ranges:
      reserved:
        start: 172.30.1.1
        end:  172.30.1.10
      static:
        start: 172.30.1.11
        end: 172.30.1.200
      dhcp:
        start: 172.30.1.201
        end: 172.30.1.254
    dns:
      domain: lab.akraino.org
      servers: '192.168.2.85 8.8.8.8 8.8.4.4'
    inf: net4
  ksn:
    vlan: 44
    interface: bond0.44
    cidr: 172.29.1.0/24
    local_asnumber: 65531
    ranges:
      static:
        start: 172.29.1.5
        end: 172.29.1.254
    additional_cidrs:
      -  172.29.1.136/29
    ingress_cidr: 172.29.1.137/32
    peers:
    - ip: 172.29.1.1
      scope: global
      asnumber: 65001
    vrrp_ip: 172.29.1.1 # keep peers ip address in case of only peer.
  neutron:
    vlan: 45
    interface: bond0.45
    cidr: 10.0.101.0/24
    ranges:
      static:
        start: 10.0.101.2
        end: 10.0.101.254
dns:
  upstream_servers:
    - 192.168.2.85
    - 8.8.8.8
    - 8.8.8.8
  upstream_servers_joined: '192.168.2.85,8.8.8.8'
  ingress_domain: hpgen10.akraino.org
sriovnets:
```

```
- physical: sriovnet1
  interface: ens6f0
  vlan_start: 2001
  vlan_end: 3000
  whitelists:
  - "address": "0000:af:02.0"
  - "address": "0000:af:02.1"
  - "address": "0000:af:02.2"
  - "address": "0000:af:02.3"
  - "address": "0000:af:02.4"
  - "address": "0000:af:02.5"
  - "address": "0000:af:02.6"
  - "address": "0000:af:02.7"
  - "address": "0000:af:03.0"
  - "address": "0000:af:03.1"
  - "address": "0000:af:03.2"
  - "address": "0000:af:03.3"
  - "address": "0000:af:03.4"
  - "address": "0000:af:03.5"
  - "address": "0000:af:03.6"
  - "address": "0000:af:03.7"
  - "address": "0000:af:04.0"
  - "address": "0000:af:04.1"
  - "address": "0000:af:04.2"
  - "address": "0000:af:04.3"
  - "address": "0000:af:04.4"
  - "address": "0000:af:04.5"
  - "address": "0000:af:04.6"
  - "address": "0000:af:04.7"
  - "address": "0000:af:05.0"
  - "address": "0000:af:05.1"
  - "address": "0000:af:05.2"
  - "address": "0000:af:05.3"
  - "address": "0000:af:05.4"
  - "address": "0000:af:05.5"
  - "address": "0000:af:05.6"
  - "address": "0000:af:05.7"
- physical: sriovnet2
  interface: ens6f1
  vlan_start: 2001
  vlan_end: 3000
  whitelists:
  - "address": "0000:af:0a.0"
  - "address": "0000:af:0a.1"
  - "address": "0000:af:0a.2"
  - "address": "0000:af:0a.3"
  - "address": "0000:af:0a.4"
  - "address": "0000:af:0a.5"
  - "address": "0000:af:0a.6"
  - "address": "0000:af:0a.7"
  - "address": "0000:af:0b.0"
  - "address": "0000:af:0b.1"
  - "address": "0000:af:0b.2"
  - "address": "0000:af:0b.3"
  - "address": "0000:af:0b.4"
  - "address": "0000:af:0b.5"
  - "address": "0000:af:0b.6"
  - "address": "0000:af:0b.7"
  - "address": "0000:af:0c.0"
  - "address": "0000:af:0c.1"
  - "address": "0000:af:0c.2"
  - "address": "0000:af:0c.3"
  - "address": "0000:af:0c.4"
  - "address": "0000:af:0c.5"
  - "address": "0000:af:0c.6"
  - "address": "0000:af:0c.7"
  - "address": "0000:af:0d.0"
  - "address": "0000:af:0d.1"
  - "address": "0000:af:0d.2"
  - "address": "0000:af:0d.3"
  - "address": "0000:af:0d.4"
```

```
  - "address": "0000:af:0d.5"
  - "address": "0000:af:0d.6"
  - "address": "0000:af:0d.7"
storage:
  osds:
    - data: /dev/sdb
      journal: /var/lib/ceph/journal/journal-sdb
    - data: /dev/sdc
      journal: /var/lib/ceph/journal/journal-sdc
    - data: /dev/sdd
      journal: /var/lib/ceph/journal/journal-sdd
    - data: /dev/sde
      journal: /var/lib/ceph/journal/journal-sde
    - data: /dev/sdf
      journal: /var/lib/ceph/journal/journal-sdf
    - data: /dev/sdg
      journal: /var/lib/ceph/journal/journal-sdg
    - data: /dev/sdh
      journal: /var/lib/ceph/journal/journal-sdh
    - data: /dev/sdi
      journal: /var/lib/ceph/journal/journal-sdi
  osd_count: 8
  total_osd_count: 24
genesis:
  name: aknode30
  oob: 192.168.41.130
  host: 192.168.2.30
  storage: 172.31.1.30
  pxe: 172.30.1.30
  ksn: 172.29.1.30
  neutron: 10.0.101.30
  root_password: akraino,d
  oem: HPE
  mac_address: 3c:fd:fe:aa:90:b0
  bios_template: hpe_dl380_g10_uefi_base.json.template
  boot_template: hpe_dl380_g10_uefi_httpboot.json.template
  http_boot_device: NIC.Slot.3-1-1
masters:
  - name : aknode31
    oob: 192.168.41.131
    host: 192.168.2.31
    storage: 172.31.1.31
    pxe: 172.30.1.31
    ksn: 172.29.1.31
    neutron: 10.0.101.31
    oob_user: Administrator
    oob_password: Admin123
  - name : aknode32
    oob: 192.168.41.132
    host: 192.168.2.32
    storage: 172.31.1.32
    pxe: 172.30.1.32
    ksn: 172.29.1.32
    neutron: 10.0.101.32
    oob_user: Administrator
    oob_password: Admin123
workers:
  - name : aknode33
    oob: 192.168.41.133
    host: 192.168.2.33
    storage: 172.31.1.33
    pxe: 172.30.1.33
    ksn: 172.29.1.33
    neutron: 10.0.101.33
    oob_user: Administrator
    oob_password: Admin123
#  - name : aknode34
#    oob: 192.168.41.134
#    host: 192.168.2.34
#    storage: 172.31.1.34
#    pxe: 172.30.1.34
```

```
#    ksn: 172.29.1.34
#    neutron: 10.0.101.34
hardware:
  vendor: HP
  generation: '10'
  hw_version: '3'
  bios_version: '2.8'
disks:
  - name : sdj
    labels:
      bootdrive: 'true'
    partitions:
      - name: root
        size: 20g
        mountpoint: /
      - name: boot
        size: 1g
        mountpoint: /boot
      - name: var
        size: '>300g'
        mountpoint: /var
  - name : sdk
    partitions:
      - name: cephj
        size: 300g
        mountpoint: /var/lib/ceph/journal
disks_compute:
  - name : sdj
    labels:
      bootdrive: 'true'
    partitions:
      - name: root
        size: 20g
        mountpoint: /
      - name: boot
        size: 1g
        mountpoint: /boot
      - name: var
        size: '>300g'
        mountpoint: /var
  - name : sdk
    partitions:
      - name: nova
        size: '99%'
        mountpoint: /var/lib/nova
genesis_ssh_public_key: "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC
/n4mNLAj3XKG2fcm+8eVe0NUlNH0g8DA8KJ53rSLKccm8gm4UgLmGOJyBfUloQZMuOpU6a+hexN4ECCliqI7+KUmgJgsvLkJ3OUMNTEVu9tDX5md
XeffsufaqFkAdmbJ/9PMPiPQ3
/UqbbtyEcqoZAwUWf4ggAWSp00SGE1Okg+skPSbDzPVHb4810eXZT1yoIg29HAenJNNrsVxvnMT2kw2OYmLfxgEUh1Ev4c5LnUog4GXBDHQtHAwa
IoTu9s/q8VIvGav62RJVFn3U1D0jkiwDLSIFn8ezORQ4YkSidwdSrtqsqa2TJ0E5w/n5h5IVGO9neY8YlXrgynLd4Y+7 root@pocnjrsv132"
kubernetes:
  api_service_ip: 10.96.0.1
  etcd_service_ip: 10.96.0.2
  pod_cidr: 10.99.0.0/16
  service_cidr: 10.96.0.0/14
regional_server:
  ip: 135.16.101.85
...
```



5. Click on Submit. This will upload the input file and the site details into the portal.

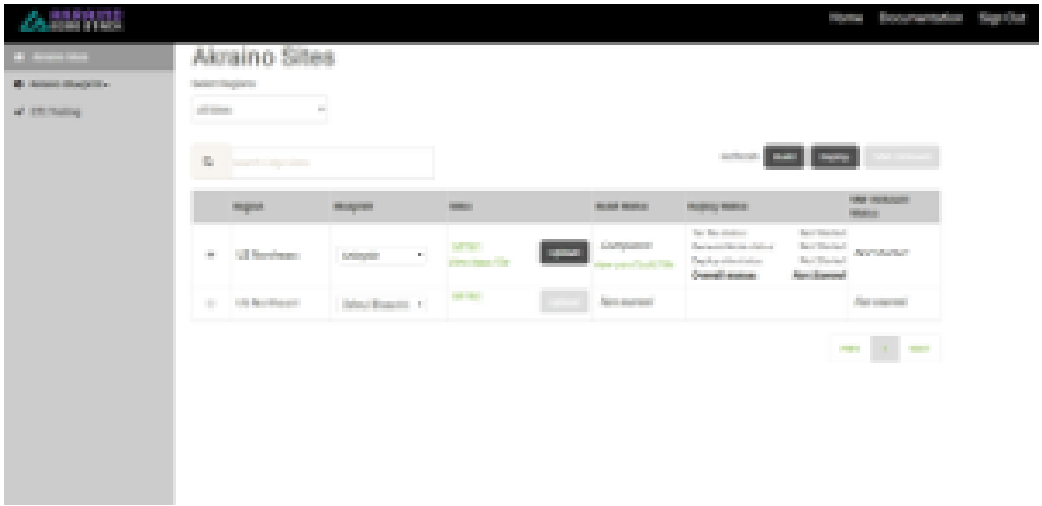
6. User will see the file uploaded successfully message in the sites column then **Build** button is enabled.

   a. Click on **Build** to begin the build process.

   b. User can click on Refresh (link) to update the status of the build on the portal.

   c. The build status changes from ‘Not started’ to ‘In progress’ to ‘Completed’.

   d. The build process will generate all the required yaml files with site details. User can view the generated yaml files by clicking ‘view yaml build file’ provided in Build status column.



7. User will see the ‘Completed’ status in build status column then **Deploy** button is enabled.

   a. Click on **Deploy** to begin the deploy process.

   b. User can click on Refresh (link) to update the status of the build on the portal.

Note: In portal when the overall status of the Deploy is success, login to each node and check deploy site logs under /*var/log*/deploy_site_yyyymmddhhmm.log file by using the command tail -f /var/log/deploy_site_yyyymmddhhmm.log file

Check the deployment process logs under “tail –f /var/log/scriptexecutor.log” or “/var/log /yaml_builds/” on regional_controller node.

- Once the deploy status got “completed” on the portal, then

This is to check the status of deploy_site.

Following is the snippet from root@aknode44:/var/log# vi scriptexecutor.log

2018-10-02 17:28:58.464 DEBUG 12751 --- [SimpleAsyncTaskExecutor-2] a.b.s.i. RemoteScriptExecutionServiceImpl : + deploy_site

2018-10-02 17:28:58.464 DEBUG 12751 --- [SimpleAsyncTaskExecutor-2] a.b.s.i. RemoteScriptExecutionServiceImpl : + sudo docker run -e OS_AUTH_URL=http://keystone-api.ucp.svc. cluster.local:80/v3 -e OS_PASSWORD=86db58e20de93ef55477 -e OS_PROJECT_DOMAIN_NAME=default -e OS_PROJECT_NAME=service -e OS_USERNAME=shipyard -e OS_USER_DOMAIN_NAME=default -e OS_IDENTITY_API_VERSION=3 --rm --net=host quay.io/airshipit/shipyard: 165c845e3e7459d2a4892ed4ca910b00675e7561 create action deploy_site

2018-10-02 17:29:02.273 DEBUG 12751 --- [SimpleAsyncTaskExecutor-2] a.b.s.i. RemoteScriptExecutionServiceImpl : Name Action Lifecycle Execution Time Step Succ/Fail/Oth

2018-10-02 17:29:02.274 DEBUG 12751 --- [SimpleAsyncTaskExecutor-2] a.b.s.i. RemoteScriptExecutionServiceImpl : deploy_site action/01CRTX8CTJ8VHMSNVC2NHGWKCY None 2018-10-02T17:29:53 0/0/0

2018-10-02 17:29:02.546 DEBUG 12751 --- [SimpleAsyncTaskExecutor-2] a.b.s.i. RemoteScriptExecutionServiceImpl : Script exit code :0

Based on the above snippet you can frame a command like following(**just concatenate highlighted partes and add describe in the middle**) and run it on aknode40 to see the status deploy_site,

root@aknode40:~# docker run -e OS_AUTH_URL=http://keystone-api.ucp.svc.cluster.local:80/v3 -e OS_PASSWORD=86db58e20de93ef55477 -e OS_PROJECT_DOMAIN_NAME=default -e OS_PROJECT_NAME=service -e OS_USERNAME=shipyard -e OS_USER_DOMAIN_NAME=default -e OS_IDENTITY_API_VERSION=3 --rm --net=host quay.io/airshipit/shipyard:165c845e3e7459d2a4892ed4ca910b00675e7561 describe action /01CRTX8CTJ8VHMSNVC2NHGWKCY

Appendix

Create New Edge Site locations

> The Akraino seed code comes with default two sites: MTN1, MTN2 representing two lab sites in Middletown, NJ. This step of connecting to the database and creating edge_site records are only required if the user wishes to deploy on other sites.

To deploy a Unicycle (Multi-Node Cluster) Edge Node, perform the following steps:

- Check if the Akraino (Docker Containers) packages are stood up.
- Connect to PostgreSQL database providing the host IP (name).

```
jdbc:postgresql://<IP-address-of-DB-host>:6432/postgres
user name = admin
password = abc123
```

> use ‘pgAdmin |||’ Postgres client tool or connect to Postgres DB using SQL interface

- Execute the following SQL insert, bearing in mind these value substitutions:
  - `edge_site_id`: Any unique increment value. This is usually 1 but does not have to be.
  - `edge_site_name`: Human-readable Edge Node name.
  - `region_id`: Edge Node region number. Use `select * from akraino.Region;` to determine the appropriate value. Observe the region number associations returned from the query: Use 1 for *US East*, 2 for *US West*, and so on.

```
> insert into akraino.edge_site(edge_site_id, edge_site_name, crt_login_id, crt_dt, upd_login_id, upd_dt, region_id)
  values( 1, 'Atlanta', user, now(), user, now(),1);
```