cnBNG Installation and Configuration

Feature Summary and Revision History

Summary Data

Table 1: Summary Data

Applicable Product(s) or Functional Area	cnBNG
Applicable Platform(s)	SMI
Feature Default Setting	Disabled - Configuration Required
Related Changes in this Release	Not Applicable
Related Documentation	Not Applicable

Revision History

Table 2: Revision History

Revision Details	Release
Introduced support for cnBNG Cluster Deployment Using Inception Server.	2023.04.0

Revision Details	Release
Introduced support for the cnBNG CNF Deployment on AIO BareMetal Server.	2022.02.0
cnBNG CP deployment on bare metal server is supported (with support for IPoE, PPPoE, LAC and LNS call models and High Availability) and fully qualified in this release.	2022.01.0
First introduced.	2021.01.0

Feature Description

This chapter describes cnBNG installation and configuration using the Ultra Cloud Core Subscriber Microservices Infrastructure (SMI) Cluster Manager and the BNG Operations (Ops) Center. The BNG Ops Center is based on the ConfD command line interface (CLI).

To install the SMI Cluster Manager, refer to the "Deploying the SMI Cluster Manager on VMware vCenter" section in the *Ultra Cloud Core Subscriber Microservices Infrastructure - Deployment Guide*.

The SMI Ops Center is the platform to install the cnBNG cluster with the offline or online repository. It is mandatory to install the SMI Ops Center to set up and access the BNG Ops Center.

✎

Note   To access the offline or online repository, contact your Cisco Account Manager or representative to get access to the offline or online repository.
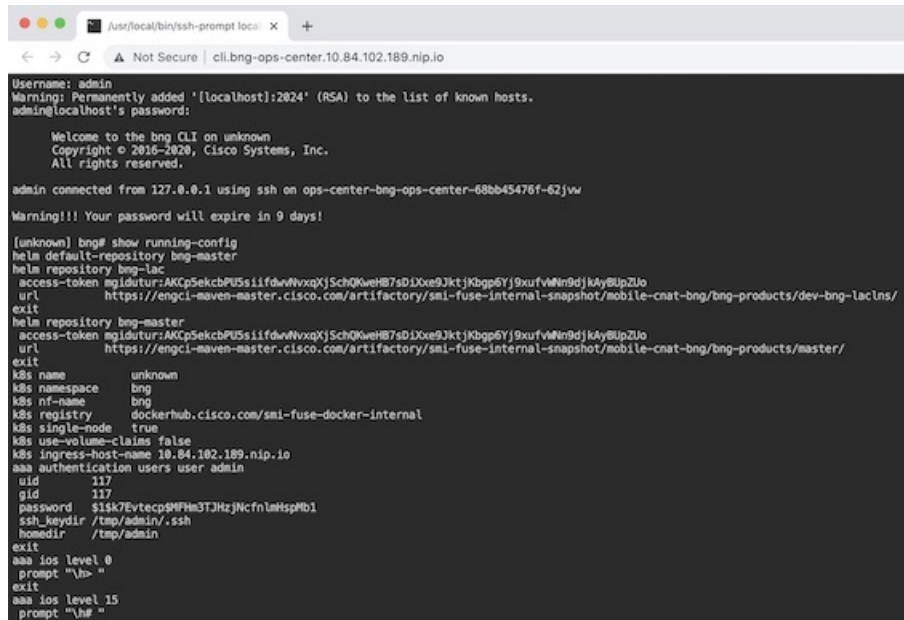
BNG Ops Center

The BNG Ops Center is a system-level infrastructure that provides the following functionality:

- A user interface to trigger a deployment of microservices with the flexibility of providing variable helm chart parameters to control the scale and properties of Kubernetes objects (deployment, pod, services, and so on) associated with the deployment.

- A user interface to push application-specific configuration to one or more microservices through Kubernetes configuration maps.

- A user interface to issue application-specific execution commands (such as show and clear commands). These commands:

  - Invoke some APIs in application-specific pods

  - Display the information returned on the user interface application

The following figure shows a sample of the web-based CLI presented to the user.

The BNG Ops Center allows you to configure features such as licensing, REST endpoint, and CDL.

For information on how to deploy BNG Ops Center on bare metal servers (currently Cisco UCS-C servers) environment, see "Operating the SMI Cluster Manager on Bare Metal" section in the *Ultra Cloud Core Subscriber Microservices Infrastructure — Operations Guide*.

Installing cnBNG and Accessing BNG Ops Center

This section describes how to install cnBNG and access the BNG Ops Center.

The Ultra Cloud Core SMI platform is responsible for setting up and managing the Cloud Native Broadband Network Gateway application.

Prerequisites

Before installing cnBNG on the SMI layer in an offline environment:

- Ensure that the SMI Cluster Manager all-in-one (AIO) is installed. This helps orchestrate the K8s Cluster and load the image.

- Ensure that all SMI K8s cluster nodes are in Ready state.

- Run the SMI synchronization operation for the BNG Ops Center and Cloud Native Common Execution Environment (CN-CEE).

  For CEE installation, refer to the *Ultra Cloud Core Common Execution Environment- Configuration and Administration Guide*.

- Ensure that the local repositories, which host the product offline TAR ball version, is installed.

System Requirements

Feature	Description
Disk Space	2 x 800 GB SSD (RAID 1) or equivalent input/output operations per second (IOPS) and redundancy.

Hardware	• High-performance x86 64-bit chipset
	
	• CPU performance Passmark benchmark of 13K rating per chip and 1,365 rating per thread, or better
	
	• VMware ESXi-compatible
	
	Note The following is recommended:
	
	     • Cisco UCSM5 series blade servers to achieve the best performance.
	
	     • All the host servers should be UCSC-C240-M5SX or UCSC-C220-M5SX.
	
	     • All the UCS systems should have SSD storge type.
	
	     • UCS C240M5 servers for better performance and to avoid infrastructure issues.

Platform	VMware ESXi and VMware vCenter versions 6.5 and 6.7 **Note** SMI Cluster Manger support is qualified on the preceding platforms.
Memory	• At least DDR3-1600 or better than 1600 MT/s • ECC
Deployment Requirement	Hardware oversubscription, network saturation, or CPU oversubscription reduces application performance and productivity. The Cisco Ultra Cloud Core Subscriber Microservices Infrastructure detects and takes action when infrastructure requirements are not met.

Installing cnBNG in an Offline Environment

Using the SMI Cluster Manager, download the offline TAR ball of the cnBNG, the host and its charts, and corresponding images in the local registries. The SMI Cluster Manager supports the deployment of the BNG Ops Center and all the applications and services associated with it. This section describes the procedures involved in installing cnBNG in an offline environment using the SMI Cluster Manager.

To install cnBNG, complete the following steps:

1. Download the TAR ball from the URL.

   software-packages download *URL*

   Example:

```
SMI Cluster Manager# software-packages download
http://<ipv4address>:<port_number>/packages/bng-2021-02-1.tar
```

2. Verify whether the TAR balls are loaded.

software-packages list

Example:

```
BNG Cluster Manager# software-packages list
[ bng-2021-02-1 ]
[ sample ]
```

3. Configure the necessary SMI Ops Center parameters in the cluster to install cnBNG.

config
    cluster *cluster_name*
        ops-centers *app_name instance_name*
            repository *url*
            netconf-ip *ipv4_address*
            netconf-port *port*
            ssh-ip *ipv4_address*
            ssh-port *port*
            ingress-hostname *<ipv4_address>.<customer_specific_domain_name>*
            initial-boot-parameters use-volume-claims *true/false*
            initial-boot-parameters first-boot-password *password*
            initial-boot-parameters auto-deploy *true/false*
            initial-boot-parameters single-node *true/false*
            initial-boot-parameters image-pull-secrets
            exit
    exit

Example:

```
SMI Cluster Manager# config
Entering configuration mode terminal
SMI Cluster Manager(config)# clusters cnbng-smi-cluster-01
SMI Cluster Manager(config-clusters-cnbng-smi-cluster-01)# ops-centers bng bng
SMI Cluster Manager(config-ops-centers-bng/bng)# repository
https://charts.10.10.105.50.nip.io/bng-2021.02.1
SMI Cluster Manager(config-ops-centers-bng/bng)# ingress-hostname 10.10.105.34.nip.io
SMI Cluster Manager(config-ops-centers-bng/bng)# initial-boot-parameters use-volume-claims
 true
SMI Cluster Manager(config-ops-centers-bng/bng)# initial-boot-parameters
first-boot-password test123
SMI Cluster Manager(config-ops-centers-bng/bng)# initial-boot-parameters auto-deploy
false
SMI Cluster Manager(config-ops-centers-bng/bng)# initial-boot-parameters single-node
false
SMI Cluster Manager(config-ops-centers-bng/bng)# exit
SMI Cluster Manager(config-clusters-cnbng-smi-cluster-01)# exit
SMI Cluster Manager(config)#
```

4. Configure the secrets, if your local registry contains secrets.

config
    cluster *cluster_name*
        secrets docker-registry *secret_name*
            docker-server *server_name*
            docker-username *username*
            docker-password *password*

```
                    docker-email email
                    namespace k8s namespace
                    commit
                    exit
                exit
```

Example:

```
SMI Cluster Manager# config
SMI Cluster Manager(config)# clusters test2
SMI Cluster Manager(config-clusters-test2)# secrets docker-registry sec1
SMI Cluster Manager(config-docker-registry-sec1)# docker-server serv1
SMI Cluster Manager(config-docker-registry-sec1)# docker-username user1
SMI Cluster Manager(config-docker-registry-sec1)# docker-password Cisco@123
SMI Cluster Manager(config-docker-registry-sec1)# docker-email reg@cisco.com
SMI Cluster Manager(config-docker-registry-sec1)# bng bng
SMI Cluster Manager(config-docker-registry-sec1)# exit
SMI Cluster Manager(config-clusters-test2)# exit
SMI Cluster Manager(config)#
```

5. Run the cluster synchronization.

clusters *cluster_name* **actions sync run**

Example:

```
SMI Cluster Manager# clusters cnbng-smi-cluster-01 actions sync run
```

Notes:

- **software-packages download** *url*–Specifies the software packages to be downloaded through HTTP/HTTPS.

- **software-packages list**–Specifies the list of available software packages.

- **ops-centers** *app_name instance_name*–Specifies the BNG Ops Center and instance. *app_name* is the application name. *instance_name* is the name of the instance.

- **repository** *url*-Specifies the local registry URL for downloading the charts.

- **netconf-ip** *ipv4_address*–Specifies the BNG Ops Center netconf IPv4 address.

- **netconf-port** *port*–Specifies the BNG Ops Center netconf port number.

- **ssh-ip** *ipv4_address*–Specifies the SSH IPv4 address for the BNG Ops Center.

- **ssh-port** *port*–Specifies the SSH port number for the BNG Ops Center.

- **ingress-hostname** *<ipv4_address>.<customer_specific_domain_name>*–Specifies the ingress hostname to be set to the BNG Ops Center. *<customer_specific_domain_name>* specifies the domain name of the customer.

- **initial-boot-parameters**–Specifies the initial boot parameters for deploying the helm charts.

  - **use-volume-claims** *true/false*–Specifies the usage of persistent volumes. Set this option to True to use persistent volumes. The default value is true.

  - **first-boot-password** *password*–Specifies the first boot password for the product's Ops Center.

  - **auto-deploy** *true/false*–Auto deploys all the services of the product. Set this option to false to deploy only the product's Ops Center.

- **single-node** *true/false*– Specifies the product deployment on a single node. Set this option to false for multi node deployments.

- **image-pull-secrets**–Specifies the docker registry secret name to be used.

- **secrets docker-registry** *secret_name*–Specifies the secret name for your docker registry.

  - **docker-server** *server_name*–Specifies the docker server name.

  - **docker-username** *username*–Specifies the docker registry user name.

  - **docker-password** *password*–Specifies the docker registry password.

  - **docker-email** *email*–Specifies the docker registry email.

  - **namespace** *namespace*–Specifies the docker registry namespace.

Verifying the cnBNG Installation

Verify the status of the cnBNG installation deployment through the cnBNG CLI. To verify, use the following commands:

1. Log in to the cnBNG product CLI.

2. Verify whether the charts are loaded in the specific instance (verify the namespace).

   show helm charts

   Example:

   ```
   bng# show helm charts
   CHART     INSTANCE  STATUS    VERSION  REVISION  RELEASE    NAMESPACE
   ----------------------------------------------------------------------
   infra-charts - DEPLOYED 0.0.6-rel-2021-01-0073-210208130850-fac5207 1 bng-bng-infra-charts
    bng-bng
   oam-pod - DEPLOYED 0.1.2-rel-2021-01-0144-210122165946-fcb74ed 1 bng-bng-oam-pod bng-bng
   bng-dashboard - DEPLOYED 0.0.1-rel-2021-01-0039-210122165311-0d542be 1
   bng-bng-bng-dashboard bng-bng
   etcd-cluster - DEPLOYED 0.7.0-0-7-0060-210203074532-f118407 1 bng-bng-etcd-cluster bng-bng
   ngn-datastore - DEPLOYED 1.3.0-1-3-0782-210125161812-f50a892 1 bng-bng-ngn-datastore
   bng-bng
   ```

3. Verify the status of the system.

   show system status

   Example:

   ```
   bng# show system status
   system status deployed true
   system status percent-ready 100.0
   ```

Notes:

- **show helm charts**–Displays the helm release details.

- **show system status**–Displays the status of the system.

Accessing BNG Ops Center

You can connect to the BNG Ops Center through SSH or the web-based CLI console.

1. SSH:

   ssh admin@ops_center_pod_ip **-p 2024**

2. Web-based console:

   a. Log in to the Kubernetes master node.

   b. Run the following command:

      ```
      kubectl get ingress <namespace>
      ```

      The available ingress connections get listed.

   c. Select the appropriate ingress and access the BNG Ops Center.

   d. Access the following URL from your web browser:

      cli.<namespace>-**ops-center.**<ip_address>.**nip.io**

      By default, the Day 0 configuration is loaded into the cnBNG.

Day 0 Configuration

To view the Day 0 configuration, run the following command.

show running-config

The following is a sample Day 0 configuration:

```
luser@cnbng-smi-cluster-master1:~$ kubectl get svc -n bng-bng | grep
ops-center-bng-bng-ops-center
NAME                                    TYPE        CLUSTER-IP      EXTERNAL-IP    PORT(S)
                                        AGE
ops-center-bng-bng-ops-center          ClusterIP   10.96.151.115   <none>
8008/TCP,8080/TCP,2024/TCP,2022/TCP,7681/TCP   7m37s
luser@cnbng-smi-cluster-master1:~$ ssh admin@10.96.151.115 -p 2024
Warning: Permanently added '[10.96.151.115]:2024' (RSA) to the list of known hosts.
admin@10.96.151.115's password:

     Welcome to the bng CLI on cnbng-smi-cluster/bng
     Copyright © 2016-2020, Cisco Systems, Inc.
     All rights reserved.

admin connected from 192.202.0.1 using ssh on ops-center-bng-bng-ops-center-7bddd4cc48-fmb6l
[cnbng-smi-cluster/bng] bng# show running-config
system mode running
helm default-repository base-repos
helm repository base-repos
url
https://engci-maven-master.cisco.com/artifactory/smi-fuse-internal-snapshot/mobile-cnat-bng/bng-products/master/
username <username>
password <password>
exit
k8s name          cnbng-smi-cluster
k8s namespace     bng-bng
k8s nf-name       bng
k8s registry      dockerhub.cisco.com/smi-fuse-docker-internal
k8s single-node   false
```

```
k8s use-volume-claims true
k8s ingress-host-name 192.0.2.2.nip.io
aaa authentication users user admin
uid        1117
gid        1117
password   $1$EmkQjvc0$o8K5tXmUzN1.drQgCL0A2/
ssh_keydir /tmp/admin/.ssh
homedir    /tmp/admin
exit
aaa ios level 0
prompt "\h> "
exit
aaa ios level 15
prompt "\h# "
exit
aaa ios privilege exec
level 0
  command action
  exit
  command autowizard
  exit
  command enable
  exit
  command exit
  exit
  command help
  exit
  command startup
  exit
exit
level 15
  command configure
  exit
exit
exit
nacm write-default deny
nacm groups group admin
user-name [ admin ]
exit
nacm rule-list admin
group [ admin ]
rule any-access
  action permit
exit
exit
nacm rule-list confd-api-manager
group [ confd-api-manager ]
rule any-access
  action permit
exit
exit
nacm rule-list ops-center-security
group [ * ]
rule change-self-password
  module-name       ops-center-security
  path              /smiuser/change-self-password
  access-operations exec
  action            permit
exit
rule smiuser
  module-name       ops-center-security
  path              /smiuser
  access-operations exec
  action            deny
```

```
            exit
            exit

            deployment
             app-name     BNG
             cluster-name Local
             dc-name      DC
            exit
            k8 bng
             etcd-endpoint      etcd:2379
             datastore-endpoint datastore-ep-session:8882
             tracing
              enable
              enable-trace-percent 30
              append-messages      true
              endpoint             jaeger-collector:9411
             exit
            exit
            k8 label protocol-layer key smi.cisco.com/node-type value protocol
            exit
            k8 label service-layer key smi.cisco.com/node-type value service
            exit
            k8 label cdl-layer key smi.cisco.com/node-type value session
            exit
            k8 label oam-layer key smi.cisco.com/node-type value oam
            exit
            instances instance 1
             system-id  DC
             cluster-id Local
             slice-name 1
            exit
            local-instance instance 1
            system mode shutdown
            helm default-repository base-repos
            helm repository base-repos
             url
            https://engci-maven-master.cisco.com/artifactory/smi-fuse-internal-snapshot/mobile-cnat-bng/bng-products/master/

             username smf-deployer.gen
             password ***
            exit
            k8s name          svi-cn-bng-tb3
            k8s namespace      bng-bng
            k8s nf-name        bng
            k8s registry       dockerhub.cisco.com/smi-fuse-docker-internal
            k8s single-node    false
            k8s use-volume-claims true
            k8s ingress-host-name 10.81.103.86.nip.io
            aaa authentication users user admin
             uid        1117
             gid        1117
             password   $1$vDWeJvJm$v46wiBWqdOj7eWgoPoZZE/
             ssh_keydir /tmp/admin/.ssh
             homedir    /tmp/admin
            exit
            aaa ios level 0
             prompt "\h> "
            exit
            aaa ios level 15
             prompt "\h# "
            exit
            aaa ios privilege exec
             level 0
              command action
              exit
```

```
                command autowizard
                exit
                command enable
                exit
                command exit
                exit
                command help
                exit
                command startup
                exit
               exit
               level 15
                command configure
                exit
               exit
              exit
              nacm write-default deny
              nacm groups group admin
               user-name [ admin ]
              exit
              nacm rule-list admin
               group [ admin ]
               rule any-access
                action permit
               exit
              exit
              nacm rule-list confd-api-manager
               group [ confd-api-manager ]
               rule any-access
                action permit
               exit
              exit
              nacm rule-list ops-center-security
               group [ * ]
               rule change-self-password
                module-name       ops-center-security
                path              /smiuser/change-self-password
                access-operations exec
                action            permit
               exit
               rule smiuser
                module-name       ops-center-security
                path              /smiuser
                access-operations exec
                action            deny
               exit
              exit
```

CP and UP Service Configuration

The CP service requires the basic configuration to process the API calls.

Note For information about the User Plane service configuration, refer to the *Cloud Native BNG User Plane Configuration Guide for Cisco ASR 9000 Series Routers, IOS XR Release 7.3.x*

Configuring the CP

The CP configuration is provided using the Ops Center infrastructure.

The following is a sample CP configuration:

```
ipam
 instance 1
  source local
  address-pool <dummy-pool-1>
   vrf-name default
   ipv4
    split-size
     per-cache 8192
     per-dp    8192
    exit
    address-range 10.0.0.1 10.10.255.255
   exit
   ipv6
    address-ranges
     split-size
      per-cache 8192
      per-dp    8192
     exit
     address-range 2001:DB8::1 2001:DB8::ff00
    exit
    prefix-ranges
     split-size
      per-cache 8192
      per-dp    1024
     exit
     prefix-length 56
     prefix-range 2001:db0:: length 40
    exit
   exit
  exit
  address-pool dummy-pool2
   vrf-name default
   ipv6
    prefix-ranges
     split-size
      per-cache 8192
      per-dp    1024
     exit
     prefix-range 2001:DB8:: length 48
    exit
   exit
  exit
  address-pool slaac-radius
   vrf-name default
   ipv6
    prefix-ranges
     split-size
      per-cache 8192
      per-dp    1024
     exit
     prefix-range 2001:DB8:: length 48
    exit
   exit
  exit
  address-pool static-dummy-pool2
   vrf-name default
   static enable user-plane dummy-asr9k-1
   ipv6
    prefix-ranges
     split-size
      no-split
     exit
```

```
       prefix-range 2001:DB8:: length 48
      exit
     exit
   exit
 exit
exit
cdl node-type session
cdl logging default-log-level error
cdl datastore session
 slice-names [ 1 ]
 endpoint replica 2
 endpoint settings slot-timeout-ms 750
 index replica 2
 index map 1
 slot replica 2
 slot map 2
 slot notification limit 300
exit
cdl kafka replica 1
!
profile dhcp dummy-dhcp-server1
 ipv4
  mode server
  server
   pool-name    dummy-pool-1
   dns-servers [ network-dns ]
   lease days 1
   lease hours 2
   lease minutes 3
  exit
 exit
 ipv6
  mode server
  server
   iana-pool-name dummy-pool-1
   iapd-pool-name dummy-pool-1
   lease days 10
   lease hours 0
   lease minutes 4
  exit
 exit
exit
profile pppoe pppoe-prof
 mtu                      1500
 ctrl-pkt-priority        7
 service-selection-disable true
 max-payload minimum 1492 maximum 1540
 session-limit max 64000 threshold 60000
exit
profile pppoe ppp1
 mtu 1492
exit
profile aaa dummy-aaa-prof
 authentication
  method-order [ dummy-ser-grp ]
 exit
 authorization
  type subscriber method-order [ dummy-ser-grp ]
  username identifier client-mac-address
  password <any-password>
 exit
 accounting
  method-order [ dummy-ser-grp ]
 exit
```

```
            exit
            profile server-group dummy-ser-grp
             radius-group dummy-ser-grp
            exit
            profile attribute-format attr1
             format-order [ client-mac-address ]
            exit
            profile attribute-format attr2
             format-order [ addr ]
            exit
            profile attribute-format attr3
             format-order [ username ]
            exit
            profile subscriber dummy-subs-prof
             dhcp-profile               dummy-dhcp-server1
             pppoe-profile             ppp1
             session-type              ipv4v6
             activate-feature-templates [ svc1 VOICE TV ]
             aaa authorize dummy-aaa-prof
            exit
            profile feature-template VOICE
             qos
              in-policy   VOICE_INGRESS
              out-policy  VOICE_EGRESS
              merge-level 40
             exit
            exit
            profile feature-template APPS
             httpr-policy ACCESS-PBR
            exit
            profile feature-template TV
             qos
              in-policy   TV_INGRESS
              out-policy  TV_EGRESS
              merge-level 50
             exit
            exit
            profile feature-template svc1
             vrf-name default
             ipv4
              mtu                 1492
              disable-unreachables
              verify-unicast-source reachable-via-rx
             exit
             ipv6
              mtu                 1492
              ingress-acl         ipv6-acl-in-1
              egress-acl          ipv6-acl-out-1
              disable-unreachables
              verify-unicast-source reachable-via-rx
             exit
             session-accounting
              enable
              aaa-profile       dummy-aaa-prof
              periodic-interval 1800
             exit
            exit
            !
            profile feature-template HSI_100M_5MQ
             qos
              in-policy   HSI_UPLOAD_RATE_100MB_IN
              out-policy  HSI_DOWNLOAD_RATE_100MB_OUT
              merge-level 30
             exit
```

```
        service-accounting
         enable
         aaa-profile       aaa-prof1
         periodic-interval 120
        exit
       exit
       profile feature-template HSI_100M_30MQ
        qos
         in-policy   HSI_UPLOAD_RATE_100MB_IN
         out-policy  HSI_DOWNLOAD_RATE_100MB_OUT
         merge-level 30
        exit
        service-accounting
         enable
         aaa-profile       dummy-aaa-prof
         periodic-interval 1800
        exit
       exit
       profile radius
        algorithm first-server
        deadtime  2
        detect-dead-server response-timeout 30
        max-retry 3
        timeout   5
        server 10.1.2.3 1812
         type     auth
         secret   <password>
         priority 1
        exit
        server 10.1.2.3 1813
         type     acct
         secret   <password>
         priority 1
        exit
        attribute
         nas-identifier CISCO-BNG
         nas-ip         <209.165.201.1>
         nas-ip user-plane-ip
         instance 1
         exit
        exit
        accounting
         attribute
          nas-ip <209.165.201.1>
         exit
        exit
        server-group dummy-ser-grp
         server auth 10.1.2.3 1812
         exit
         server acct 10.1.2.3 1813
         exit
        exit
       exit
       profile coa
        client 10.1.2.3
         server-key <password>
        exit
       user-plane
        instance 1
         user-plane dummy-asr9k-1
          peer-address ipv4 209.165.201.3
          subscriber-profile dummy-subs-prof
         exit
         user-plane dummy-asr9k-2
```

```
        peer-address ipv4 209.165.201.2
        subscriber-profile dummy-subs-prof
      exit
    exit
  exit
instance instance-id 1
 endpoint sm
  replicas 1
 exit
 endpoint l2tp-tunnel
  replicas 1
 exit
 endpoint nodemgr
  replicas 1
 exit
 endpoint n4-protocol
  replicas 1
  retransmission max-retry 1
 exit
 endpoint dhcp
  replicas 1
 exit
 endpoint pppoe
  replicas 1
 exit
 endpoint radius
  interface coa-nas
   vip-ip <209.165.201.1> vip-port 2000
  exit
 exit
 endpoint udp-proxy
  vip-ip <209.165.201.1>
  interface n4
   sla response 180000
  exit
  interface gtpu
   sla response 180000
  exit
 exit
exit
logging level application error
logging level transaction error
logging level tracing error
logging name infra.affinity_cache.core level application off
logging name infra.application.core level application off
logging name infra.bgipcstream.core level application off
logging name infra.cache_client.core level application off
logging name infra.cdl_update_queue.core level application off
logging name infra.config.core level application off
logging name infra.diagnostic.core level application off
logging name infra.diagnostics.core level application off
logging name infra.dpd.core level application off
logging name infra.ds_client.core level application off
logging name infra.edr.core level application off
logging name infra.heap_dump.core level application off
logging name infra.ipc_action.core level application off
logging name infra.ipcstream.core level application off
logging name infra.memory_cache.core level application off
logging name infra.message_trace.core level application off
logging name infra.resource_monitor.core level application off
logging name infra.resource_monitor_load_factor.core level application off
logging name infra.rest_server.core level application off
logging name infra.session_cache.core level application off
logging name infra.topology.core level application off
```

```
                logging name infra.topology_lease.core level application off
                logging name infra.transaction.core level application off
                logging name infra.virtual_msg_queue.core level application off
                logging name infra.vrf_etcd_update.core level application off
                deployment
                 app-name      BNG
                 cluster-name Local
                 dc-name       DC
                exit
                k8 bng
                 etcd-endpoint      etcd:2379
                 datastore-endpoint datastore-ep-session:8882
                 tracing
                  enable
                  enable-trace-percent 30
                  append-messages      true
                  endpoint             jaeger-collector:9411
                 exit
                exit
                k8 label protocol-layer key smi.cisco.com/proto-type value protocol
                exit
                k8 label service-layer key smi.cisco.com/svc-type value service
                exit
                k8 label cdl-layer key smi.cisco.com/sess-type value cdl-node
                exit
                k8 label oam-layer key smi.cisco.com/node-type value oam
                exit
                resource cache-pod
                 gomaxproc 8
                exit
                instances instance 1
                 system-id  DC
                 cluster-id Local
                 slice-name 1
                exit
                local-instance instance 1
                system mode <shutdown or running>
                commit
                end
```

Configuring the UP

The following is a sample UP configuration:

```
user-plane
 instance 1
  user-plane dummy-asr9k-1
   peer-address ipv4 209.165.201.3
   subscriber-profile dummy-subs-prof
  exit
  user-plane dummy-asr9k-2
   peer-address ipv4 209.165.201.2
   subscriber-profile dummy-subs-prof
  exit
 exit
exit
```

Loading Day1 Configuration

To load the Day 1 configuration for cnBNG, run the following command:

ssh admin@ops_center_pod_ip **-p 2024** < Day1config.cli

Note	The **day1config.cli file** contains the necessary parameters required for the Day 1 configuration.

Alternatively, you can copy the configuration and paste it in the BNG Ops Center CLI to load the Day 1 configuration.

```
config
  <Paste the Day 1 configuration here>
  commit
  exit
```

Day1config.cli

The **day1config.cli file** file contains the Day 1 configuration for cnBNG. For a sample day1 configuration, see Configuring the CP, on page 11.

Mapping Pods with Node Labels

Prerequisites

- Ensure that the node labels are according to the pod deployment layout.

- Ensure that the external VIPs are according to the requirement of NF.

- Enable Istio for pod to pod traffic load balancing.

Node Labels are key and value pairs that are attached to nodes at cluster synchronization. Each node can have a set of key and value labels defined. Each key must be unique for a node. With labels, users can map their NF pods onto nodes in a loosely coupled manner.

Important	• The pod-level labeling configuration is applicable only when the cnBNG CP is deployed on a bare metal server.
	• Ensure to configure the node label on the SMI cluster deployer before mapping the pods. Following is the sample command for master-1 labeling:

```
[cndp-clpnc-cm-cm-primary] SMI Cluster Deployer (config-nodes-master-1)# k8s node-labels
smi.cisco.com/svc-type bng-node
```

To map the pods with node labels, use the following sample configuration:

```
config
   k8 label protocol-layer key label_key value label_value
   k8 label service-layer key label_key value label_value
   k8 label cdl-layer key label_key value label_value
   k8 label oam-layer key label_key value label_value
   end
```

Following is an example configuration of pod to node-label mapping:

```
k8 label protocol-layer key smi.cisco.com/node-type value bng-proto
```

```
exit
k8 label service-layer key vm-type value bng-svc
exit
k8 label cdl-layer key smi.cisco.com/node-type value bng-cdl
exit
k8 label oam-layer key smi.cisco.com/node-type value oam
exit
```

High Availability Support on BareMetal Server

High Availablity on cnBNG CP is validated on BareMetal server deployment. For more informaton about High Availablity, see High Availability and CP Reconciliation.

cnBNG CNF Deployment on AIO BareMetal Server

The cnBNG CNF Deployment on AIO BareMetal Server explains the process of onboarding a cnBNG Cloud Native Function (CNF) on the Cloud Native Deployment Platform (CNDP) on the BareMetal all-in-one (AIO) Kubernetes (K8s) cluster.

In the AIO deployment, all the management VMs are hosted on a different UCS server, however, this depends on the deployment strategy.

The cnBNG CNF is hosted on another UCS server referred as AIO server. During installation, the Cluster Manager (CM) accesses the AIO via the Cisco Integrated Management Controller (IMC) interface and adds the respective image and SMI packages to complete the installation.

✎

Note   The management VMs are the Inception, Cluster Manager, and RADIUS servers.

*Figure 1: Logical Topology for cnBNG CNF Deployment on AIO BareMetal Server*
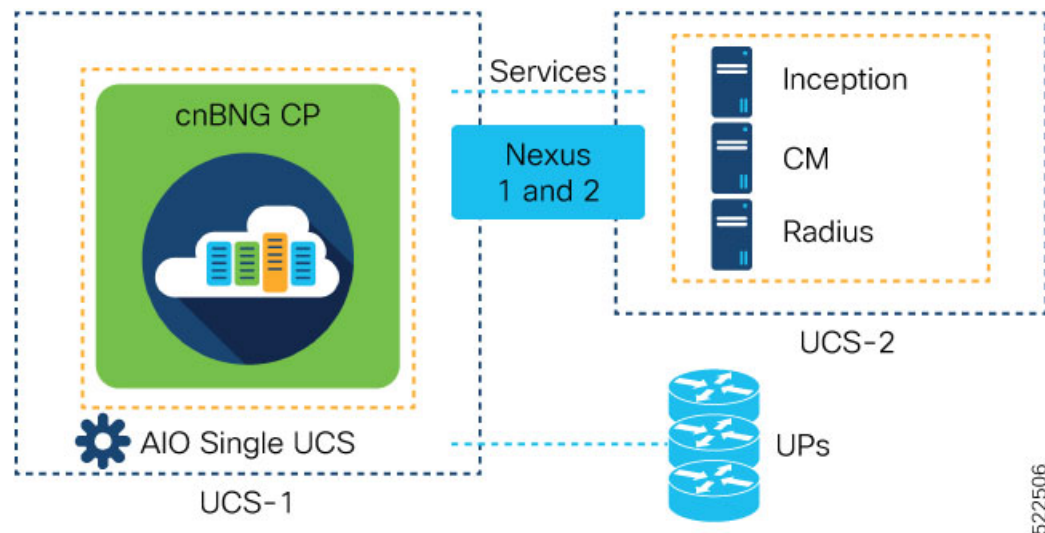
The CNDP is a 'SMI Bare-Metal'. The Cluster Manager uses REST APIs (instead of VIM APIs) of the management cards, which are on the servers, to create a set of Linux servers and then loads the K8s software.
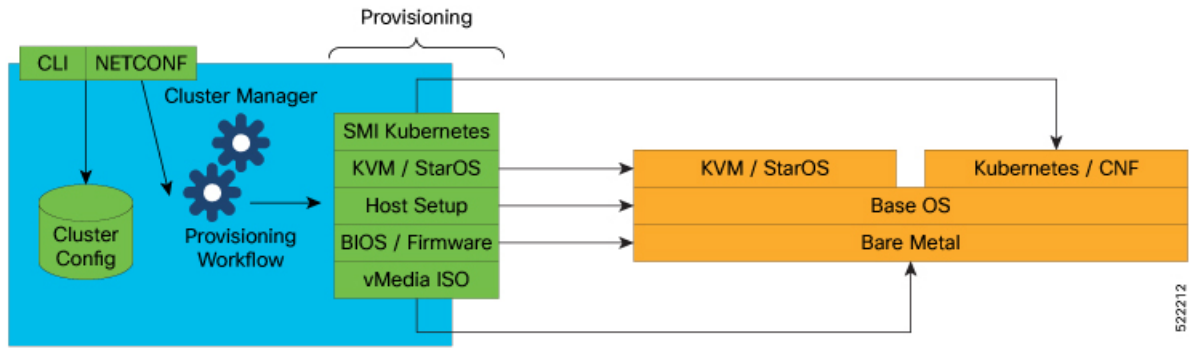
Before installing network functions (NF) on the Cluster Manager, the common containerized software from SMI is installed. For example, monitoring and logging. The SMI NFs include their own common containerized software.

BareMetal CNDP AIO Bring-Up Procedure

The following figure illustrates the step-by-step process that is required to bring up the cnBNG CNF on K8s AIO server.
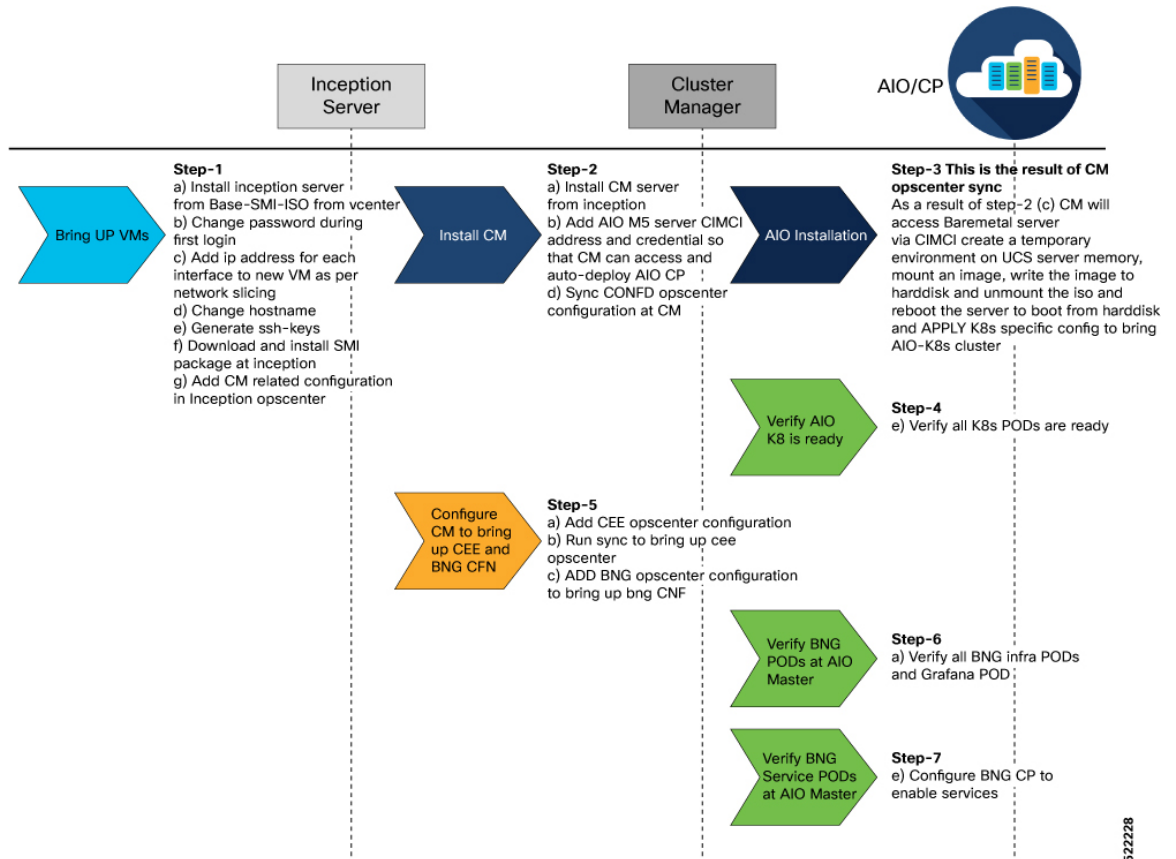
For more information about the Inception, Cluster Manager, and All-in-One server installation, see the "SMI Cluster Manager - Deployment" chapter of the *Ultra Cloud Core Subscriber Microservices Infrastructure - Deployment Guide*.

Limitations and Restrictions

The cnBNG CNF Deployment on AIO BareMetal Server has the following limitations and restrictions:

- Simulated User Planes (UPs) are used in the characterisation activity.
- ASR 9000 routers will be used in the topology based on availability in future releases.
- Actual customer profile must be validated before deployment.

Implementing cnBNG CP Validation with CNDP

Implementing cnBNG CP Validation with CNDP involves the following procedures.

- Prerequisites
- Instantiating and Provisioning Inception Server Instance
- Installing the Cluster Manager Node
- Deploying the All-in-One Cluster
- Integrating RADIUS and UP with the AIO BareMetal Server

Prerequisites

The following sections detail the prerequisites for deploying the cnBNG CNF on the AIO BareMetal Server.

Instantiating and Provisioning Inception Server Instance

The Inception server is used to deploy the CM for CNDP deployment. It is an Ubuntu 18.04 based VM installed with additional packages such as, docker, docker-compose, and its dependencies. The offline tar ball for the CNDP CM is installed on this instance and configured to deploy the CNDP CM nodes.

The following section presents the procedure on how to bring up an Inception server instance on a VM.

Figure 2: CNDP Inception VM
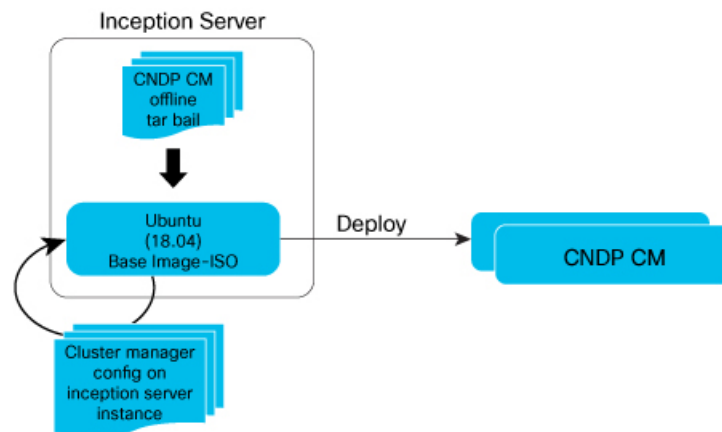
Before beginning the configuration of the Inception server, verify that all the dependent packages such as docker, docker-compose are installed on the VM. This is a prerequisite before loading the tar ball to configure and deploy the CM. If the SMI Base-ISO is used for installing Inception server, the packages are preinstalled.

Installing Cluster Manager Node

The Cluster Manager (CM) handles the installation and upgrade of the Kubernetes (K8s) cluster and associated infrastructure. In this deployment, the Inception server launches two machines that use DRBD to replicate the state to provide High Availability (HA) of the CM. This section covers the bring up procedure of the CM in standalone mode.

For High Availability CM deployment, see the "SMI Cluster Manager - Deployment" chapter of the *Ultra Cloud Core Subscriber Microservices Infrastructure - Deployment Guide*.

Configuring the Cluster Manager - Single Instance

Configure the single instance of the CM:

1. Login to the Ops Center CLI of the Inception server. Use the following steps to install and configure the CM.

   Use the IP address of the Inception server to login.

   ```
    ssh admin@<ip_address> -p 2022   (or)
   https://cli.smi-deployer.<ip_address>.nip.io
   ```

2. Configure the Inception server cluster deployer to install the CM.

   Note Ensure the following before configuring the CM.

   - All the passwords must be typed manually because copying and pasting the encrypted passwords from the configuration throws an error during validation

   - Private key and public key must be generated in the Inception VM and copied. Both the keys must match the keys present in the Inception VM. Use the **ssh-keygen** command in the Inception VM and follow the prompts to generate the keys.

   - Private key and public key are multiline commands. For instance, after typing **node-defaults ssh-connection- private-key**, press Enter to paste the keys.

   For the Inception configuration, see the "SMI Cluster Manager - Deployment" chapter of the *Ultra Cloud Core Subscriber Microservices Infrastructure - Deployment Guide*.

   Note Modify IPs, password, keys , username , cluster-name, and CNF name based on the specified configurations.

Installing the All-in-One Cluster

This section provides the detailed steps to deploy the CNDP AIO (K8s) cluster from the CM node. It also specifies the CLI that is used on the CM to configure and perform a cluster synchronization operation.

Configuring the AIO Cluster

This section describes the procedure to configure and deploy the CNDP AIO cluster from the CM using the CLI method.

1. Login to the CM Ops Center and load the SMI cluster, cnBNG, CEE, and Ops Center node configurations. Multiline configuration for private key must be pasted separately.

```
ssh admin@<ip_address> -p 2022
```

2. Update the Sha256 value, which was generated for the software from the previous step, in the configuration for the respective software under the sha256 section.

> **Note**
> - All the passwords must be typed manually because copying and pasting the encrypted passwords from the configuration throws an error during validation
>
> - Private key and public key must be generated in the CM and copied in. Both the keys must match the keys present in the CM. Use the **ssh-keygen** command in the Inception VM and follow the prompts to generate the keys.

3. Before running cluster synchronization, enable detail logging using the following configuration.

```
clusters <cluster_name>
 configuration restrict-logging false exit
```

4. From the SMI cluster configuration, configure the Software CNF repository to use the cnBNG image, CEE, and include the sha256 checksum as generated previously and provide the path of the image.

For more information, see the "SMI Cluster Manager - Deployment" chapter of the *Ultra Cloud Core Subscriber Microservices Infrastructure - Deployment Guide*.

```
[inception] SMI Cluster Deployer# show running-config
software cnf <cnf_software_version>c
 url      <repo_url>
 user     <user_name>
 password <password>
 sha256   <SHA256_hash_key>
exit
```

Example:

```
Cluster Manager# config
Cluster Manager(config)# software cnf <example=cm-2020-02-0-i06>
Cluster Manager(config)# url <repo_url>
Cluster Manager(config)#user <username>
Cluster Manager(config)#password "<password>"
Cluster Manager(config)#sha256 <sha256_key>
Cluster Manager(config)#exit
```

In this deployment model, a single AIO node is deployed.

From a CM configuration perspective, the AIO node definition, corresponding Ops Center CEE and cnBNG instances are defined as part of a single AIO cluster.

The following configuration snippet shows the sample configuration for a cluster from the cluster manager

```
config
  software cnf <cnf_software_version>
  url <repo_url>
  user <user_name>
```

```
        password <password>
        sha256 <SHA256_hash_key>
        exit
   environments bare-metal
    ucs-server
   exit
   clusters <cluster_name> #For example, cndp-testbed
    environment bare-metal
    addons ingress bind-ip-address <IPv4address>
    addons cpu-partitioner enabled
    configuration allow-insecure-registry true
    node-defaults ssh-username <username>
    node-defaults ssh-connection-private-key
     "-----BEGIN OPENSSH PRIVATE KEY-----\n
   <SSH_private_key>
     -----END OPENSSH PRIVATE KEY-----\n"
    node-defaults initial-boot default-user <username>
    node-defaults initial-boot default-user-ssh-public-key
     "<SSH_Public_Key>"
    node-defaults initial-boot default-user-password #For example, Csco123#
    node-defaults os proxy https-proxy <proxy_server_url>
    node-defaults os proxy no-proxy <proxy_server_url/IPv4address>
    node-defaults os ntp enabled
    node-defaults os ntp servers <ntp_server>
    exit
   node-defaults initial-boot netplan ethernets <interface_name> #For example, eno1
    dhcp4 false
    dhcp6 false
    gateway4 <IPv4address>
    nameservers search <nameserver>
    nameservers addresses <IPv4addresses>
   exit
   node-defaults initial-boot netplan ethernets eno2 # same like eno1 other interfaces to
   be configured
    dhcp4 false                                       # without any ip address
    dhcp6 false
   exit
   node-defaults initial-boot netplan ethernets eno5
    dhcp4 false
    dhcp6 false
   exit
   node-defaults initial-boot netplan ethernets eno6
    dhcp4 false
    dhcp6 false
   exit
   node-defaults initial-boot netplan ethernets enp216s0f0
    dhcp4 false
    dhcp6 false
   exit
   node-defaults initial-boot netplan ethernets enp216s0f1
    dhcp4 false
    dhcp6 false
   exit
   node-defaults initial-boot netplan ethernets enp94s0f0
    dhcp4 false
    dhcp6 false
   exit
   node-defaults initial-boot netplan ethernets enp94s0f1
    dhcp4 false
    dhcp6 false
   exit
   node-defaults initial-boot netplan vlans <vlan_name> #For example, vlan309
    dhcp4 false
    dhcp6 false
```

```
 id    <vlan_id> #For example, 309
 link  eno6
exit
node-defaults initial-boot netplan vlans <vlan_name> #For example, vlan310
 dhcp4 false
 dhcp6 false
 id    <vlan_id> #For example, 310
 link  eno6
exit
node-defaults initial-boot netplan vlans <vlan_name> #For example, vlan311
 dhcp4 false
 dhcp6 false
 id    <vlan_id> #For example, 311
 link  enp94s0f0
exit
node-defaults ucs-server cimc user admin
node-defaults ucs-server cimc storage-adaptor create-virtual-drive true
node-defaults ucs-server cimc remote-management sol enabled
node-defaults ucs-server cimc remote-management sol baud-rate 115200
node-defaults ucs-server cimc remote-management sol comport com0
node-defaults ucs-server cimc remote-management sol ssh-port 2400
node-defaults ucs-server cimc networking ntp enabled
node-defaults ucs-server cimc networking ntp servers <example: ntp.server1.com>
exit
node-defaults ucs-server cimc networking ntp servers <example: ntp.server2.com>
exit
node-defaults os ntp enabled
node-defaults os ntp servers <example: ntp.server1.com>
exit
node-defaults os ntp servers <example: ntp.server1.com>
exit

 nodes <aio>  #For example it can be master or aio
 k8s node-type master
 k8s ssh-ip <IPv4address>
 k8s node-ip <IPv4address>
 k8s node-labels disktype ssd
 exit
 k8s node-labels smi.cisco.com/node-type oam
 exit
 ucs-server cimc user admin
 ucs-server cimc password <IPv4address> #this CIMCI address of the AIO UCS SERVER
 ucs-server cimc ip-address 10.81.103.117
 initial-boot netplan ethernets eno1
 addresses [ <IPv4address-mgmt>/24 ]
 gateway4    <gateway-address>
 exit
 initial-boot netplan vlans vlan309
 addresses [ <IPv4address-k8s>/24 ]
 exit
 initial-boot netplan vlans vlan310
 addresses [ <IPv4address-SMI>/24 ]
 exit
 initial-boot netplan vlans vlan311
 addresses [ <IPv4address-services>/24 ]
 exit
exit
```

Each CNF provides a ConfD based Ops Center CLI to configure and manage the CNF pods. There is a separate Ops Center required for each CNF deployed on the AIO node.

The following is the Ops Center configuration for the AIO node, which has the Ops Center configuration for CEE and CNF.

```
ops-centers bng bng
  repository       <url> or offline-tarball
  username         <username>
  password         <password>
  ingress-hostname <ip-address>.nip.io
  initial-boot-parameters use-volume-claims false
  initial-boot-parameters first-boot-password <password>
  initial-boot-parameters auto-deploy false
  initial-boot-parameters single-node true
 exit
 ops-centers cee cee
  repository-local        cee-release-build
  sync-default-repository true
  netconf-ip              <ip-address>
  netconf-port            2024
  ssh-ip                   <ip-address>
  ssh-port                2022
  ingress-hostname        <ip-address>.nip.io
  initial-boot-parameters use-volume-claims true
  initial-boot-parameters first-boot-password <password>
  initial-boot-parameters auto-deploy true
  initial-boot-parameters single-node true
 exit
exit
```

> ✎
>
> **Note** To bring the network function NF at the AIO K8 cluster, always use the "initial-boot-parameters single-node true" option.

5. Run the cluster synchronization to deploy the cluster, cnBNG, and CEE Ops Centers

```
clusters cndp-cm actions sync run debug true
```

The cluster synchronization operation takes approximately 45 minutes to complete.

6. Monitor the cluster synchronization operation using the following command.

```
monitor sync-logs cndp-cm
```

After cluster synchronization is completed, a message is shown indicating a successful cluster synchronization.

Integrating RADIUS and UP with the AIO BareMetal Server

The RADIUS and UP are part of the services network and therefore should be part of the same network. If they are not in the same VLAN, then the necessary routing should be available to have reachability between them.

The AIO services interface is also part of the services VLAN, which has routable reachability between AIO UDP proxy interface, RADIUS, and the User Plane function (UPF).

Figure 3: Logical Network Connectivity
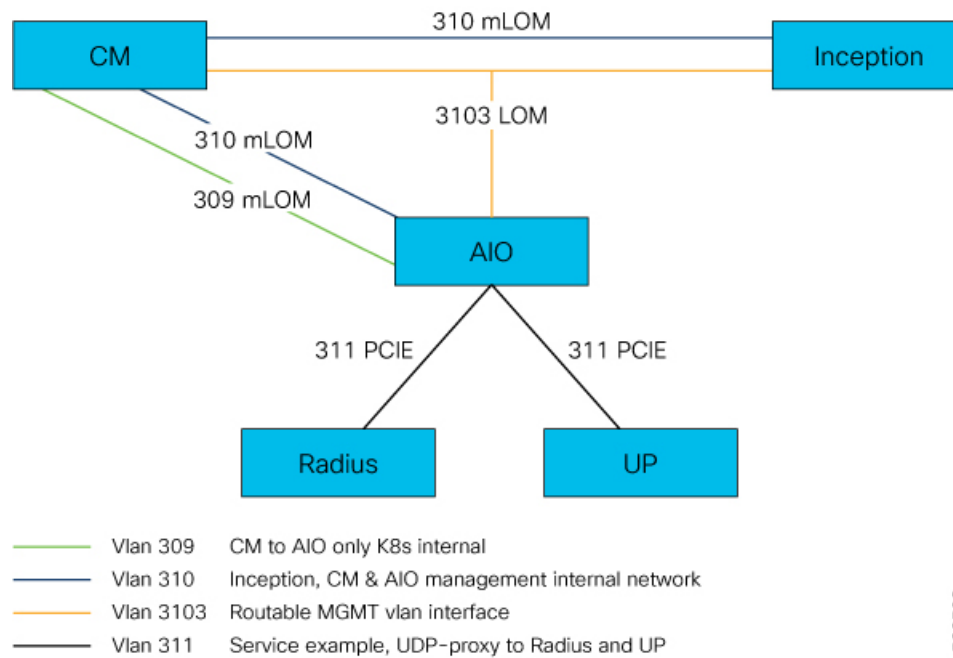
cnBNG Cluster Deployment Using Inception Server

You can now deploy the cnBNG cluster using the Inception server alone. You do not require the SMI Cluster Manager, which was previously used along with the Inception server to deploy the cnBNG cluster. This enhancement can help you save on hardware resources (servers).

Installing Inception Server on Baremetal

The procedure to install the the Inception server on baremetal is as follows:

1. Clear a Boot drive.

2. Create a virtual drive from unused physical drives.

3. Install Base ISO image.

4. Configure User and Network Parameters.

5. Install Inception Server.

6. Deploy SMI Cluster.

7. Add images to Inception Server.

8. Create SSH keys.

9. Add SMI Cluster Deployer configuration.

Clear a Boot Drive

You must clean up the server storage before installing the Base ISO image. To clear the boot drive configurations on the Cisco Integrated Management Controller (CIMC) server, perform the following steps:

1. Log in to the CIMC Web UI using admin privileges.

2. In the **Navigation** pane, click the **Storage** menu.

3. On the **Storage** menu, click the appropriate LSI MegaRAID or Host Bus Adapter (HBA) controller.

   In the **RAID Controller** area, the **Controller Info** tab is displayed by default.

4. In the **Actions** area, click **Clear Boot Drive**.

Note The **Clear Boot Drive** option is enabled only if the server was used previously. If the server is new, this option is disabled.

5. Click **OK** to confirm.

Create Virtual Drive from Unused Physical Drives

1. Log in to the CIMC Web UI using admin privileges.

2. On the **Storage** menu, click the appropriate LSI MegaRAID or HBA controller.

3. In the **Actions** area, click **Create Virtual Drive from Unused Physical Drives**.

   The **Create Virtual Drive from Unused Physical Drives** dialog box appears.

4. In the **Create Virtual Drive from Unused Physical Drives** dialog box, select **1** from the **RAID Level** drop-down list.

5. In the **Create Drive Groups** area, choose one or more physical drives to include in the group.

   Use the **>>** button to add the drives to the **Drive Groups** table. Use the **<<** button to remove physical drives from the drive group.

Note
   - The size of the smallest physical drive in the drive group defines the maximum size used for all the physical drives. To ensure maximum use of space for all physical drives, it is recommended that the size of all the drives in the drive group are similar.

   - CIMC manages only RAID controllers and not Host Bus Adapters (HBAs) attached to the server.

   - You must have multiple drive groups available to create virtual drives (VDs) for certain RAID levels. While creating drives for these RAID levels, the create drive option is available only if the required number of drives are selected.

6. In the **Virtual Drive Properties** area, update the following properties:

Name	Description
Virtual Drive Name field	The name of the new virtual drive you want to create.

Name	Description
Read Policy drop-down list	The read-ahead cache mode.
Cache Policy drop-down list	The cache policy used for buffering reads.
Strip Size drop-down list	The size of each strip, in MB.
Write Policy drop-down list	This can be one of the following • **Write Through**— Data is written through the cache and to the physical drives. Performance is improved, because subsequent reads of that data can be satisfied from the cache. • **Write Back**— Data is stored in the cache, and is only written to the physical drives when space in the cache is needed. Virtual drives requesting this policy fall back to **Write Through** caching when the BBU cannot guarantee the safety of the cache in the event of a power failure. • **Write Back Bad BBU**—With this policy, write caching remains **Write Back** even if the battery backup unit is defective or discharged.
Disk Cache Policy drop-down list	This can be one of the following • **Unchanged**— The disk cache policy is unchanged. • **Enabled**— Allows IO caching on the disk. • **Disabled**— Disallows disk caching.
Access Policy drop-down list	This can be one of the following • **Read Write**— Enables host to perform read-write on the VD. • **Read Only**— Host can only read from the VD. • **Blocked**— Host can neither read nor write to the VD.
Size field	The size of the virtual drive (VD) you want to create. Enter a value and select one of the following units: • MB • GB • TB

7. Click **Create Virtual Drive**.

8. After the virtual drive is created, click the **Virtual Drive Info** tab.

   In the **Virtual Drives** area, choose the drive from which the controller must boot, and then click **Set as Boot Drive**.

9. Click **OK** to confirm.

Install Base ISO image

Perform the following steps to install the Base ISO image.

1. Log in to the CIMC Web UI using admin privileges.

2. Click the **Virtual KVM** tab.

3. In the **Actions** area, click **Launch KVM Console**.

4. Click the **Virtual Media** menu, and click **Activate Virtual Devices**. A virtual media session is activated, and that allows you to attach a drive or image file from your local computer or network.

5. Click the **Virtual Media** menu again, and select **Map CD/DVD**. You can map a CD or a DVD image from your local machine and map the drive to the image.

6. Browse, and select the Base ISO image from the image folder.

7. From the tool bar, click the **Host Power** link. Select **Power Cycle** from the drop-down list. The chosen server is powered on with the mapped Base ISO image.

8. From the tool bar, click **Launch KVM**. The **KVM Console** opens in a separate window.

9. Once you see the login prompt and other options, press the **F6** function key.

10. On the boot menu, select the CIMC based vKVM-Mapped option to which the Base ISO image is mapped.

    The server boots with the required ISO image.

11. Using CIMC, you can also configure the order in which the server attempts to boot from the available boot device types.

    a. Click the **Compute** menu.

    b. In the **BIOS** tab, click the **Configure Boot Order** tab. **Configure Boot Order** dialog box appears.

    c. In the **Configure Boot Order** dialog box, you can select the order of boot, and click **Save Changes**.

12. Power cycle the server to trigger the Base ISO installation. You can view the installation status in the KVM console.

Configure User and Network Parameters

Before installing the Inception server you must configure the user and network parameters.

To configure the network and static IP addressing:

1. Login with the default *cloud-init* credentials.

✎

| Note | You must change the password immediately after logging in. |

2. Update the network configuration in `/etc/netplan/50-cloud-init.yaml` file.

   The following is a sample network configuration:

```
network:
   ethernets:
     eno1:
       addresses:
       - 10.0.0.20/8
       dhcp4: false
       gateway4: 10.197.209.193
       nameservers:
           addresses:
       - 10.104.128.236
       - 10.163.128.140
       - 10.37.142.73
version: 2
```

3. Run the following command to apply the configuration:

   sudo netplan apply

4. To preserve hostname or make the hostname persistent after reboot, you must edit the cloud configuration file:

   ```
   sudo vi /etc/cloud/cloud.cfg
   ```

   Set **preserve_hostname: true**

5. Modify the hostname.

   ```
   sudo hostnamectl set-hostname hostname
   sudo vi /etc/hosts
   127.0.0.1 hostname
   ```

Install Inception Server

Perform the following steps to install the Inception server:

1. Copy the SMI offline package to the images folder.

   Example:

   ```
   cloud-user@cnbng4-cxbgl:/data/software/images$ pwd /data/software/images
   cloud-user@cnbng4-cxbgl:/data/software/images$
   ```

2. Untar the the downloaded file.

   Example:

   ```
   cloud-user@cnbng4-cxbgl:/data/software/images$ tar -xvf
   cluster-deployer-2023.01.1.i18.tar
   ```

3. Navigate to the `deployer-inception` folder which has the required charts and docker files.

   Example:

   ```
   cd /data/deployer-inception/
   ```

4. Run the following command to install the Inception server.

   Example:

   ```
   sudo ./deploy --external-ip external_ipaddress --first-boot-password
   first_boot_password
   ```

Note The *external_ipaddress* is the management IP address of the inception server.

Deploy SMI Cluster

Perform the following steps to deploy the SMI cluster:

1. Log in to the cluster using the following:

   ```
   ssh admin@<ip_address> -p 2022
   ```

2. Add the cluster level configurations for one or more K8s clusters.

Add Images to Inception Server

1. Fetch the offline tarball for SMI, cnBNG, & CEE and save it to the `/data/software/images` folder. You can fetch the tarball either from the artifactory or copy it securely through the **scp** or **winscp** command.

2. Untar the offline tarball, and copy the tar file to the respective path.

   Example:

   ```
   root@cnbng-inception:/data/downloads# tar -xvzf bng.2020.04.m0.i37.SSA.tgz ./
   ./bng.2020.04.m0.i37.tar.SSA.README
   ./CNBNG_DEV_KEY-CCO_DEV.cer
   ./trca.cer
   ./Innerspace_DEV.cer ./cisco_x509_verify_dev.py ./bng.2020.04.m0.i37.tar.signature.SSA
   ./bng.2020.04.m0.i37.tar
   ```

3. Generate **sha256** checksum for the images and verify them with artifactory checksum.

   Example:

   ```
   cloud-user@cnbng4-cxbgl:/data/software/images$ sudo sha256sum bng.2020.04.m0.i37.tar
   2e4fe956daf4afa13909d6fa89be5e727b9e4c03619436ecd04805045b780c0b bng.2020.04.m0.i37.tar
   cloud-user@cnbng4-cxbgl:/data/software/images$ sudo sha256sum cee-2023.01.1.i18.tar
   320e61f56976a2c107fa489a2a12d16301671f28212ec5b7d902b608d2e6ab80 cee-2023.01.1.i18.tar
   cloud-user@cnbng4-cxbgl:/data/software/images$ sudo sha256sum cluster-deployer-
   2023.01.1.i18.tar
   929dd80a840483f65a9f4afa314144f0f544e3bee23703214c03c831465ae707 cluster-deployer-
   2023.01.1.i18.tar
   ```

4. Add the images to Inception deployer cluster configuration. The inception deployer uses cnBNG & CEE images from the provided file path to bring up cnBNG control plane & CEE ops-center.

   Example:

   ```
   software cnf cnbng
   url file:///data/software/images/bng.2020.04.m0.i37.tar
   sha256 2e4fe956daf4afa13909d6fa89be5e727b9e4c03619436ecd04805045b780c0b
   exit
   software cnf cee-2023.01.1.i18
   url file:///data/software/images/cee-2023.01.1.i18.tar
   sha256 320e61f56976a2c107fa489a2a12d16301671f28212ec5b7d902b608d2e6ab80
   exit
   software host-profile bios-ht-25
   url file:///data/software/images/ht.tgz
   sha256 aa7e240f2b785a8c8d6b7cd6f79fe162584dc01b7e9d32a068be7f6e5055f664
   exit
   environments bare-metal
   ```

```
                    ucs-server
                    exit
                    !
```

Generate SSH Keys

Generate SSH public and private keys.

Example:

```
cloud-user@inception-28:~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:YQ3OfuEinrurnkkFlLW/vBDu5Zkti8zMt8dfIxpgYIE ubuntu@nfs-server-cn-svi
The key's randomart image is:
+---[RSA 2048]----+
| .+o . |
| .E = o |
| . + = o |
| o = o . |
| + S o |
| + * = |
| . = +.. . o |
| . O *o=oo o . |
| .=.@+B=o.. |
+----[SHA256]-----+
cloud-user@inception-28:~/.ssh$ ls -ltr ~/.ssh/*id*
-rw------- 1 cloud-user cloud-user 577 Jan 19 2023 /home/cloud-user/.ssh/id_rsa.pub
-rw------- 1 cloud-user cloud-user 2484 Jan 19 2023 /home/cloud-user/.ssh/id_rsa
```

Add SMI Cluster Deployer Configuration

Perform the following steps to add the SMI Cluster Deployer configurations:

1. Log in to the SMI cluster deployer.

    Example:

    ```
    cloud-user@incep.on-28:~$ ssh admin@127.0.0.1 -p 2022
    admin@127.0.0.1's password:
    Welcome to the Cisco SMI Cluster Deployer on inception-28
    Copyright Â© 2016-2020, Cisco Systems, Inc.
    All rights reserved.
    admin connected from 10.0.1.1 using ssh on c0993db6451a
    ```

2. Add SSH-keys to the Inception Manager.

    Example:

    ```
    clusters <cluster_name>
    environment bare-metal
    node-defaults initial-boot default-user-ssh-public-key
    "<SSH_Public_key>"
    node-defaults ssh-connection-private-key
    "-----BEGIN OPENSSH PRIVATE KEY-----
    <SSH_Private_Key>
    -----END OPENSSH PRIVATE KEY-----\n"
    CNTRL+D
    ```

3. Add CNF to clusters.

   You can either download CNF images from online repositories, web servers, or local servers, or copy the tarball images to the Inception server folder and configure the folder path.

4. You can verify the configuration using the following command:

```
SMI Cluster Deployer# show running-config
software cnf cee-2023.01.1.i18
url file:///data/software/images/cee-2023.01.1.i18.tar
sha256 320e61f56976a2c107fa489a2a12d16301671f28212ec5b7d902b608d2e6ab80
exit
software cnf cnbng
url file:///data/software/images/bng-dev-private.tar
sha256 2e4fe956daf4afa13909d6fa89be5e727b9e4c03619436ecd04805045b780c0b
exit
software host-profile bios-ht-25
url file:///data/software/images/ht.tgz
sha256 aa7e240f2b785a8c8d6b7cd6f79fe162584dc01b7e9d32a068be7f6e5055f664
exit
```