CPS UDC Migration MoP for OpenStack, Release 13.1.0

First Published: 2017-08-18

Last Modified: 2017-08-24

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2017 Cisco Systems, Inc. All rights reserved.

CONTENTS

Method of Procedure for OpenStack

This chapter includes the following topics:

- Apply Orchestration API Patch
- Add New Replica Sets
- Configure Cluster Manager VM
- Initiate UDC VM
- Configure Other VMs
- Verify Deployment

Apply Orchestration API Patch

NOTE: Refer to the patch Release Note to download the Orchestration API patch.

Step 1 In Cluster Manager VM, create a directory `/opt/orchestration_api_server.udc.patch` by running the following command:

```
mkdir /opt/orchestration_api_server.udc.patch
```

Step 2 Move the patch into the `/opt/orchestration_api_server.udc.patch` directory.

Step 3 Create a backup of the existing Orchestration API server directory by running the following command:

```
cp -r /opt/orchestration_api_server /opt/orchestration_api_server.backup
```

Step 4 Create a backup of `orchestration_deploy.tar.gz` by running the following command:

```
cp /opt/cluman/puppet/modules/api_server/files/plugins/orchestration_deploy.tar.gz
/opt/cluman/puppet/modules/api_server/files/plugins/orchestration_deploy.tar.gz.backup
```

Step 5 Replace `orchestration_deploy.tar.gz` with downloaded patch by running the following command:

```
cp /opt/ocrchestration_api_server.udc.patch/ATT_Platform_Patch1_13.0.0.tar.gz
/opt/cluman/puppet/modules/api_server/files/plugins/orchestration_deploy.tar.gz
```

Step 6 Stop the `monit` process of the Orchestration API Server by running the following command:

```
monit stop orchestration-api-server
```

Step 7 Verify if the Orchestration API server `monit` process is stopped by running the following command:

```
monit summary
```

The `orchestration-api-server` process must display its status as `Not monitored`.

Step 8 Run puppet command on cluman to apply the patch as follows:

```
puppet apply --logdest=/var/log/cluman/puppet-custom-run.log --
modulepath=/opt/cluman/puppet/modules --config=/opt/cluman/puppet/puppet.conf
/opt/cluman/puppet/nodes/node_repo.pp
```

Step 9 Start the Orchestration API server by running the following command:

```
monit start orchestration-api-server
```

The `orchestration-api-server` process must display its status as `Running`.

Add New Replica Sets

Perform the following steps to add new Replica Sets:

Step 1 Add a new Session DB replica set using Orchestration API.

For more inofrmation, see the Orchestration API section under /api/system/config/replica-sets in the *CPS Installation Guide for Openstack*.

Step 2 In Cluster Manager VM, modify `/etc/broadhop/mongoConfig.cfg` to add a new replica set for a new Admin DB:

```
[ADMIN-SET<X>]

SETNAME=set<xx>

ARBITER=pcrfclient01:<port number>

ARBITER_DATA_PATH=/var/data/sessions.<x>

MEMBER1=sessionmgr01:<port number>

MEMBER2=sessionmgr02:<port number>

DATA_PATH=/var/data/sessions.<x>

[ADMIN-SET<X>-END]
```

NOTE:

1. Ensure that the port number for this Admin replica set is not currently used.
2. The "`<X>`" in `[ADMIN-SET<X>]` should be a number.

Copy the configuration to all VMs using the following command:

```
SSHUSER_PREFERROOT=true copytoall.sh /etc/broadhop/mongoConfig.cfg
/etc/broadhop/mongoConfig.cfg
```

Step 3 Run `/var/qps/bin/support/mongo/build_set.sh --admin --create --setname set<xx>` where `set<xx>` is the Set name configured in mongoConfig.cfg to create the Admin DB.

For the prompts, enter non-sharded(2) as the option and then yes to continue as described below:

```
Starting Replica-Set Creation

Please select your choice: replica sets sharded (1) or non-sharded (2):
```

Type 2 and Enter

```
WARNING: Continuing will drop mongo database and delete everything under /data/sessions
on all Hosts

CAUTION: This result into loss of data

Are you sure you want to continue (y/yes or n/no)?
```

Type `y` or `yes` and Enter

Step 4 Verify that the Admin DB port is up and in listening mode in both session managers by running the following command:

```
netstat -tupln | grep LISTEN | grep xxxxx
```

Where `xxxxx` is the port number of the new Admin DB.

Step 5 Verify and add appropriate license to the new Admin DB created:

From `root`, run `mongodump --host sessionmgr01 --port yyyyy --db sharding --collection licensedfeats`

Where `yyyyy` is the port number of the existing Admin DB.

Run `mongorestore --host sessionmgr01 --port xxxxx --db sharding --collection licensedfeats /root/dump/sharding/licensedfeats.bson`

Where `xxxxx` is the port number of the new Admin DB.

Configure Cluster Manager VM

Perform the following steps to configure Cluster Manager VM for the new UDC VMs:

For more information about the modifications in the files listed in the following steps, see *Modifications in Platform Files*.

Step 1 Append `/var/qps/install/current/scripts/build/build_fact.sh` configuration as follows:

```
echo "udc_instances=1" >> $FACT_FILE
```

Step 2 Update `/var/qps/install/current/scripts/deployer/support/jconfig.py` to include UDC cluster info similar to QNS.

Step 3 Modify `/var/qps/install/current/scripts/upgrade/reinit.sh` to include UDC regexp and other loops.

Step 4 Add UDC regexp pattern to `/var/qps/install/current/scripts/upgrade/pp_status.sh`

Step 5 Add UDC regexp to `/var/qps/install/current/scripts/bin/control/copytoall.sh`

Step 6 Add UDC regexp pattern to `/var/qps/install/current/scripts/bin/support/hosts.sh`

Step 7 Add UDC regexp pattern to `/var/qps/install/current/scripts/bin/diag/diagnostics.sh`

Step 8 Add UDC regexp to `/var/qps/install/current/scripts/bin/control/runonall.sh`

Step 9 Add UDC regexp to `/var/qps/bin/support/sync_times.sh`

Step 10 Copy `/etc/puppet/modules/qps/manifests/roles/qns.pp` as `/etc/puppet/modules/qps/manifests/roles/udc.pp` by using the following command:

```
cp /etc/puppet/modules/qps/manifests/roles/qns.pp
/etc/puppet/modules/qps/manifests/roles/udc.pp
```

In the `/etc/puppet/modules/qps/manifests/roles/udc.pp` file, change all instances of "`qns`" to "`udc`".

Step 11 Copy `/etc/puppet/modules/qps/nodes/qns.yaml` as `/etc/puppet/modules/qps/nodes/udc.yaml` by using the following command:

```
cp /etc/puppet/modules/qps/nodes/qns.yaml

/etc/puppet/modules/qps/nodes/udc.yaml
```

Update `/etc/puppet/modules/qps/nodes/udc.yaml` by modifying `qps::roles::qns:` as follows:

```
qps::roles::udc:
```

Step 12 Copy `/var/qps/install/current/puppet/modules/qps/templates/etc/snmp/qns.snmpd.conf` to `/var/qps/install/current/puppet/modules/qps/templates/etc/snmp/udc.snmpd.conf` in the same directory by using the following command:

```
cp /var/qps/install/current/puppet/modules/qps/templates/etc/snmp/qns.snmpd.conf
/var/qps/install/current/puppet/modules/qps/templates/etc/snmp/udc.snmpd.conf
```

Step 13 Copy `/var/qps/install/current/puppet/modules/qps/templates/etc/qns.sysctl.conf` to `/var/qps/install/current/puppet/modules/qps/templates/etc/udc.sysctl.conf` in the same directory by using the following command:

```
cp /var/qps/install/current/puppet/modules/qps/templates/etc/qns.sysctl.conf
/var/qps/install/current/puppet/modules/qps/templates/etc/udc.sysctl.conf
```

Step 14 Delete the following line from `/etc/broadhop/qns.conf`:

```
-Dcom.broadhop.run.clusterId=cluster-1
```

Step 15 Copy `/var/qps/current_config/etc/broadhop/pcrf` recursively (-r option) as `/var/qps/current_config/etc/broadhop/udc` by using the following command:

```
cp -r /var/qps/current_config/etc/broadhop/pcrf /var/qps/current_config/etc/broadhop/udc
```

Step 16 In the `/etc/broadhop/udc/qns.conf` file, remove the following entries:

```
--DjmsFlowControlHost=xxxx

--DjmsFlowControlPort=xxxx
```

Step 17 Add the following line at the top inside the QNS_OPTS string in `/etc/broadhop/udc/qns.conf`:

```
-Dcom.broadhop.run.clusterId=cluster-udc

-Dsession.db.init.1=sessionmgr<xx>

-Dsession.db.init.2=sessionmgr<yy>

-Dsession.db.init.port=zzzzz
```

Where:

- `sessionmgr<xx>` is the primary session manager for UDC's session DB.
- `sessionmgr<yy>` is the secondary session manager for UDC's session DB.
- `zzzzz` is the port number for UDC's session DB.

Step 18 Copy `/var/qps/current_config/etc/broadhop/diameter_endpoint` recursively (-r option) as `/var/qps/current_config/etc/broadhop/udc_diameter_endpoint` by running the following command:

```
cp -r /var/qps/current_config/etc/broadhop/diameter_endpoint
/var/qps/current_config/etc/broadhop/udc_diameter_endpoint
```

Step 19 Add the following line at the top inside the QNS_OPTS string in `/etc/broadhop/udc_diameter_endpoint/qns.conf` :

```
-Dcom.broadhop.run.clusterId=cluster-udc
```

Step 20 Add the following line at the top inside the QNS_OPTS string in `/etc/broadhop/pcrf/qns.conf`:

```
-Dcom.broadhop.run.clusterId=cluster-1

-Dsession.db.init.1=sessionmgr<xx>

-Dsession.db.init.2=sessionmgr<yy>

-Dsession.db.init.port=zzzzz
```

Where:

- `sessionmgr<xx>` is the primary session manager for QNS's session DB.
- `sessionmgr<yy>` is the secondary session manager for QNS's session DB.
- `zzzzz` is the port number for QNS's session DB.

Step 21 Add the following line at the top inside the QNS_OPTS string in `/etc/broadhop/diameter_endpoint/qns.conf`:

```
-Dcom.broadhop.run.clusterId=cluster-1
```

Step 22 Add the following line at the top inside the QNS_OPTS string in `/etc/broadhop/controlcenter/qns.conf`:

```
-Dcom.broadhop.run.clusterId=cluster-1
```

Step 23 Create a `qns.conf` file in `/etc/broadhop/iomanager/` directory with the following details:

```
QNS_OPTS="
-Dcom.broadhop.run.clusterId=cluster-1
"
```

Step 24 Modify `/var/qps/current_config/etc/broadhop/servers` as described below:

a. Replace the lines corresponding to QNS VMs that are migrated to UDC VMs with type as udc. For example, *udc03=udc*.

b. Add an endpoint information for `udc_diameter_endpoint` for the load balancer VMs. For example, `lb<xx>=udc_diameter_endpoint`.

Step 25 Modify `/var/qps/current_config/etc/broadhop/servers.all` as described below:

a. Replace the lines corresponding to QNS VMs that are migrated to UDC VMs with type as udc. For example, `<cluster>-udc03=udc` and so on.

b. Add an enpoint information for `udc_diameter_endpoint` for the load balance VMs. For example, `<cluster>-lb<xx>=udc_diameter_endpoint` and so on.

Step 26 Modify `/var/qps/install/current/scripts/build/build_servers_all.sh` to add UDC regexp.

Step 27 Update feature files as follows:

`/etc/broadhop/pb/features` will have 3 additional features

```
com.broadhop.client.feature.udcclient
com.broadhop.client.feature.udsninterface
com.broadhop.client.feature.udcfe
```

`/etc/broadhop/pcrf/features` will have 1 additional feature

```
com.broadhop.udcclient.service.feature
```

`/etc/broadhop/udc/features` will have 5 additional features

```
com.broadhop.udsninterface.service.feature
com.broadhop.udcfe.service.feature
 com.cisco.api.service.feature
 com.broadhop.ldap.interface.feature
```

```
com.broadhop.ldap.service.feature
```

Step 28 Add `udc=udc` and `lb=udc_diameter_endpoint` to `/var/qps/images/image_map`.

Step 29 Add `udc=udc` and `lb=udc_diameter_endpoint` to `/var/qps/current_config/image_map`.

Step 30 Update `/var/qps/install/current/puppet/modules/qps/templates/collectd_worker/collectd.d/jmxplugin.conf` for UDC stats as described below:

a. Append the following configuration under <Plugin "GenericJMX">:

```
<MBean "qns-udc-events">

    ObjectName
"com.broadhop.udcclient.policy.event.UdcClientStatisticsManager:name=*,type=service"

    InstancePrefix "udc."

    InstanceFrom "name"

    <Value>

        Type "qns_stat"

        Attribute "Success"

        Attribute "Error"

        Attribute "TotalTimeMs"

        Attribute "AverageTime"

    </Value>

</MBean>
```

b. Modify ‘node1’ connection information to include "qns-udc-events" as follows:

```
<Connection>

    InstancePrefix "node1."

    ServiceURL "service:jmx:rmi:///jndi/rmi://localhost:9045/jmxrmi"

    Collect "garbage_collector"

    Collect "java-memory"

    Collect "thread"

    Collect "classes"
```

```
        Collect "qns-counters"
        Collect "qns-spr"
        Collect "qns-actions"
        Collect "qns-messages"
        Collect "qns-andsf-events"
        Collect "qns-entitlement-events"
        Collect "qns-ldap-bind"
        Collect "qns-ldap-search"
        Collect "qns-ldap-connection"
        Collect "qns-ldap-pool"
        Collect "mog-api-stats"
        Collect "scef-api-stats"
        Collect "dra-db-counters"
        Collect "qns-udc-events"
    </Connection>
```

Initiate UDC VM

Perform the following steps on the Cluster Manager VM to initiate UDC VM:

Step 1 Run `/var/qps/install/current/scripts/build_all.sh`

Step 2 Utilize OpenStack HEAT template and add two `udc` VMs.

Step 3 Prepare HEAT template files for UDC VMs.

Refer to `udc-mop-udcvm.hot` and `udc-mop-udcvm.env` in the *Modifications in Platform Files* chapter to see the heat template file and the environment file examples.

Step 4 Deploy HEAT stack by running the following command:

```
heat stack-create --environment-file udc-mop-udcvm.env --template-file udc-mop-udcvm.hot
udcvm
```

Step 5 Verify if the udcvm stack is created by running the following command:

```
heat stack-list
```

The following is a sample output:

```
+--------------------------------------+------------+-----------------+----------------------+
| id                                   | stack_name | stack_status    | creation_time        |
+--------------------------------------+------------+-----------------+----------------------+
| f3f6eb03-0b6b-485c-bd1f-9c5535d8fcdb | udc-cluman | CREATE_COMPLETE | 2017-07-20T00:47:17Z |
| d4561791-4e38-45f3-b1d2-4d2270ac4dcb | allvm      | CREATE_COMPLETE | 2017-07-20T21:46:37Z |
| c2505be4-4b3b-4ac6-a0d6-161245067ca5 | udcvm      | CREATE_COMPLETE | 2017-07-21T23:17:54Z |
+--------------------------------------+------------+-----------------+----------------------+
```

Step 6 Verify if the UDC VM is created and the status is active by running the following command:

```
nova list
```

The following is a sample output:

```
+--------------------------------------+--------------------------------+--------+------------+----
---------+-------------------------------------------------------------------------------------------
-----------------------+

| ID                                   | Name                           | Status | Task State |
Power State | Networks
|

+--------------------------------------+--------------------------------+--------+------------+----
---------+-------------------------------------------------------------------------------------------
-----------------------+

| d6df6b9f-a7fd-4b90-b1b0-e0dffeecb6db | Arbiter                        | ACTIVE | -          |
Running     | Replication=192.168.76.50
|

| 59f7127b-b5cd-4cae-aa00-fefbdaab5013 | allvm-lb01-kzxjs3jkh5c2        | ACTIVE | -          |
Running     | Replication=192.168.76.21; InternalB=192.168.78.21; API=192.168.77.21;
Management=172.16.3.120                |

| 081cfa1d-677e-4bc7-b281-1abdf08dfba8 | allvm-lb02-jme2ewxuu7d7        | ACTIVE | -          |
Running     | Replication=192.168.76.22; InternalB=192.168.78.22; API=192.168.77.22;
Management=172.16.3.121                |

| 6520ce6f-7b22-43f6-a913-562023f82ce5 | allvm-pcrfclient01-zzbmbvxo3lju | ACTIVE | -          |
Running     | InternalB=192.168.78.19; Management=172.16.3.118
|
```

```
| 351089d6-5c92-4ef0-8289-d45b33fe3f8c | allvm-pcrfclient02-6on6va3u3suc | ACTIVE | -          |
Running     | InternalB=192.168.78.20; Management=172.16.3.119
|

| 1b6e2096-d580-4575-8f75-bfa90d0d7fa7 | allvm-qns01-qq5oa6f7vkpl        | ACTIVE | -          |
Running     | InternalB=192.168.78.27
|

| 15690dea-0733-4719-8787-06ce04e9f2a3 | allvm-qns02-dqkhxp5uljpv        | ACTIVE | -          |
Running     | InternalB=192.168.78.28
|

| 89f122d2-e37b-48b6-b275-3ede6316163b | allvm-qns03-wkasmt2lvqzv        | ACTIVE | -          |
Running     | InternalB=192.168.78.29
|

| b65bb4cb-5c1d-4e66-ab1b-35064f8976bf | allvm-qns04-bkygbqwckgxk        | ACTIVE | -          |
Running     | InternalB=192.168.78.30
|

| fc37da1b-b5ec-4520-9a6e-4cbfeaf8d63b | allvm-sessionmgr01-oqmggd3kiwgw | ACTIVE | -          |
Running     | InternalB=192.168.78.23; Management=172.16.3.122
|

| e9639fea-fe11-4d83-b605-455d4344812f | allvm-sessionmgr02-osg2tk3fo5b7 | ACTIVE | -          |
Running     | InternalB=192.168.78.24; Management=172.16.3.123
|

| 022a2830-27a8-4931-aff1-d378de32f0b7 | allvm-sessionmgr03-33cprq5lq6ep | ACTIVE | -          |
Running     | InternalB=192.168.78.25; Management=172.16.3.124
|

| 84180c96-17f5-4887-b5ca-398832ab9217 | allvm-sessionmgr04-5kgk5fuw5jdn | ACTIVE | -          |
Running     | InternalB=192.168.78.26; Management=172.16.3.125
|

| 26cf3313-b79d-436a-9b1b-f2e8bc1f4d6c | cluman-MOGA                     | ACTIVE | -          |
Running     | InternalA=192.168.75.5; API=192.168.77.5; Management=172.16.3.110;
Replication=192.168.76.5                     |

| 0a4ebc27-5bd3-4caa-bb1b-7d61fec6ad32 | lb01-MOGA                       | ACTIVE | -          |
Running     | InternalA=192.168.75.8; API=192.168.77.8; Management=172.16.3.113; Rx=155.165.155.155;
Replication=192.168.76.8 |

| 3b07fb3b-13f1-42fa-aa3a-d6502c52a20b | lb02-MOGA                       | ACTIVE | -          |
Running     | InternalA=192.168.75.9; API=192.168.77.9; Management=172.16.3.114; Rx=155.165.155.150;
Replication=192.168.76.9 |

| 1d5741d7-fdd6-4700-9f6c-c57722603e29 | pcrfclient01-MOGA               | ACTIVE | -          |
Running     | InternalA=192.168.75.6; API=192.168.77.6; Management=172.16.3.111;
Replication=192.168.76.6                     |
```

```
| 91031d53-2131-4753-aad7-34adb53f9a97 | pcrfclient02-MOGA               | ACTIVE | -          |
Running     | InternalA=192.168.75.7; API=192.168.77.7; Management=172.16.3.112;
Replication=192.168.76.7                      |

| 06c4b62c-cebb-4e14-b937-105834b6cfce | qns01-MOGA                      | ACTIVE | -          |
Running     | InternalA=192.168.75.14; API=192.168.77.14; Replication=192.168.76.14
|

| 613f2938-93d9-41c6-8d77-1d8fc0ae5169 | qns02-MOGA                      | ACTIVE | -          |
Running     | InternalA=192.168.75.15; API=192.168.77.15; Replication=192.168.76.15
|

| 24185867-4399-44a7-89f5-046c014f2104 | qns03-MOGA                      | ACTIVE | -          |
Running     | InternalA=192.168.75.16; API=192.168.77.16; Replication=192.168.76.16
|

| dcba8533-00aa-4ee3-be1c-86fd9f1bc60b | qns04-MOGA                      | ACTIVE | -          |
Running     | InternalA=192.168.75.17; API=192.168.77.17; Replication=192.168.76.17
|

| 582a6b71-ec03-4c45-9b63-25e6be380788 | sessionmgr01-MOGA               | ACTIVE | -          |
Running     | InternalA=192.168.75.10; API=192.168.77.10; Replication=192.168.76.10
|

| 13d566eb-64a3-411e-a584-3d730cfcb961 | sessionmgr02-MOGA               | ACTIVE | -          |
Running     | InternalA=192.168.75.11; API=192.168.77.11; Replication=192.168.76.11
|

| 72e2ff07-a50e-4d0d-963d-f84eea41f205 | sessionmgr03-MOGA               | ACTIVE | -          |
Running     | InternalA=192.168.75.12; API=192.168.77.12; Replication=192.168.76.12
|

| d6540658-99f9-4828-89cf-d02811b723a9 | sessionmgr04-MOGA               | ACTIVE | -          |
Running     | InternalA=192.168.75.13; API=192.168.77.13; Replication=192.168.76.13
|

| 6ebd98bf-2239-4710-a2e4-f2d5cd36bb0a | udc-cluman-cluman-j7vxkqsnrz4w  | ACTIVE | -          |
Running     | InternalB=192.168.78.18; Management=172.16.3.117
|

| df591a25-7507-48e3-9759-53cb9d5f4e1c | udcvm-udc01-lpkmu7j743qv        | ACTIVE | -          |
Running     | InternalB=192.168.78.91
|

| 80cd7ffe-ac42-4b86-b040-772d6a1c43be | udcvm-udc02-56yezc3pm2x2        | ACTIVE | -          |
Running     | InternalB=192.168.78.92
|

+--------------------------------------+---------------------------------+--------+------------+----
---------+-----------------------------------------------------------------------------------------
-----------------------+
```

Step 7 Create a UDC yaml configuration file.

Refer to `udc-mop-udcvm-add-config.yaml` in the *Modifications in Platform Files* chapter to see the yaml configuration file example.

Step 8 Apply the configuration for UDC VMs using the Orchestration API by running the following command:

```
curl -i -X PATCH http:// *installer IP*/api/system/config/hosts -H "Content-Type:
application/yaml" --data-binary @udc-mop-udcvm-add-config.yaml
```

Step 9 After UDC VM is initiated, copy `/etc/snmp/scripts` recursively (-r) from any of the QNS VMs to UDC VM by running the following command:

```
scp -r /etc/snmp/scripts root@<UDC VM IP>:/etc/snmp/scripts
```

Configure Other VMs

Perform the following steps to configure all other VMs with Cluster Manager VM changes:

Step 1 Reinitialize the VMs from the Cluster Manager by running the following script:

```
/var/qps/install/current/scripts/upgrade/reinit.sh
```

NOTE: Perform these steps when `reinit.sh` is run.

Step 2 Verify if `/etc/broadhop/iomanager/qns.conf` changes in *Step 23* is reflected in both the LB VMs.

If the changes are not reflected after running the `reinit.sh` script, repeat *Step 23* on both the LB VMs.

Step 3 In each pcrfclient VM, modify /etc/broadhop/qns.conf to add the following two lines at the top inside the QNS_OPTS string:

```
-Dcom.broadhop.run.systemId=system-1

-Dcom.broadhop.run.clusterId=cluster-1
```

Step 4 Navigate back to Cluster Manager VM.

Step 5 To migrate existing sessions to UDC, see *CPS UDC MoP for Session Migration*.

Verify Deployment

After the system configuration and services are started on the UDC VMs, verify the deployment with peripheral scripts described in the following sections.

Verify UDC VM

1. Verify if all peripheral script output includes details on UDC VMs and no UDC related errors are seen by running the following commands:

a. hosts.sh

Sample Output

```
lb01

lb02

qns01

qns02

udc03

udc04

pcrfclient01

pcrfclient02
```

b. diagnostics.sh

Sample Output

```
CPS Diagnostics HA Multi-Node Environment

---------------------------

Running diagnostics.sh is only supported from the Cluster Manager and
pcrfclient nodes.
```

c. about.sh

Sample Output

```
Cisco Policy Suite - Copyright (c) 2015. All rights reserved.


CPS Multi-Node Environment


CPS Installer Version - 13.0.0

CPS Core Versions

---------------------------

         lb01: qns-1            (iomanager): 13.0.1.r104788

         lb01: qns-2    (diameter_endpoint): 13.0.1.r104788

         lb01: qns-3    (diameter_endpoint): 13.0.1.r104788
```

```
          lb01: qns-4    (diameter_endpoint): 13.0.1.r104788
          lb01: qns-5  (udc_diameter_endpoint): 13.0.1.r104788
          lb02: qns-1            (iomanager): 13.0.1.r104788
          lb02: qns-2    (diameter_endpoint): 13.0.1.r104788
          lb02: qns-3    (diameter_endpoint): 13.0.1.r104788
          lb02: qns-4    (diameter_endpoint): 13.0.1.r104788
         lb02: qns-5  (udc_diameter_endpoint): 13.0.1.r104788
        qns01: qns-1                 (pcrf): 13.0.1.r104788
        qns02: qns-1                 (pcrf): 13.0.1.r104788
        udc03: qns-1                  (udc): 13.0.1.r104788
        udc04: qns-1                  (udc): 13.0.1.r105254
 pcrfclient01: qns-1        (controlcenter): 13.0.1.r105254
 pcrfclient01: qns-2                  (pb): 13.0.1.r104788
 pcrfclient02: qns-1        (controlcenter): 13.0.1.r105254
 pcrfclient02: qns-2                  (pb): 13.0.1.r105254

CPS Patch History
---------------------------
No patches have been applied

CPS Feature Versions
---------------------------
 Features installed on all (combined)
   com.broadhop.balance.service.feature             13.0.1.r105254
   com.broadhop.balance.spr.feature                 13.0.1.r105254
   com.broadhop.controlcenter.feature               13.0.1.r105254
   com.broadhop.custrefdata.service.feature         13.0.1.r104892
```

```
com.broadhop.diameter2.local.feature              13.0.1.r105296
com.broadhop.externaldatacache.memcache.feature   13.0.1.r105254
com.broadhop.faultmanagement.service.feature      13.0.1.r104255
com.broadhop.infrastructure.feature               13.0.1.r105254
com.broadhop.iomanager.feature                    13.0.1.r105254
com.broadhop.ldap.interface.feature               13.0.1.r105030
   com.broadhop.ldap.service.feature                 13.0.1.r105030
   com.broadhop.policy.feature                       13.0.1.r105254
   com.broadhop.server.runtime.product               13.0.1.r105254
   com.broadhop.snmp.feature                         13.0.1.r105254
   com.broadhop.spr.dao.mongo.feature                13.0.1.r104255
   com.broadhop.spr.feature                          13.0.1.r104255
   com.broadhop.udcclient.service.feature            13.0.1.r104255
   com.broadhop.udcfe.service.feature                13.0.1.r104255
   com.broadhop.udsninterface.service.feature        13.0.1.r104255
   com.broadhop.ui.controlcenter.feature             13.0.1.r104255
   com.broadhop.unifiedapi.interface.feature         13.0.1.r104255
   com.broadhop.unifiedapi.ws.service.feature        13.0.1.r104255
   com.broadhop.vouchers.service.feature             13.0.1.r105254
   com.broadhop.ws.service.feature                   13.0.1.r104255
   com.cisco.api.service.feature                     13.0.1.r104255


Deprecated CPS URLs
---------------------------
               Control Center: https://172.19.65.106:443
               Policy Builder: https://172.19.65.106:7443/pb
                      Grafana: https://172.19.65.106:9443/grafana
```

```
              Unified API WSDL:
https://172.19.65.106:8443/ua/wsdl/UnifiedApi.wsdl

               Unified API XSD:
https://172.19.65.106:8443/ua/wsdl/UnifiedApi.xsd

                  Unified API: https://172.19.65.106:8443/ua/soap

                 CRD REST API: https://172.19.65.106:8443/custrefdata

                HAProxy Status: http://172.19.65.106:5540/haproxy?stats

 Service Orchestration API Doc: https://172.19.65.106:7443/doc/index.html

             Import/Export API:
https://172.19.65.106:7443/doc/import.html

                  CPS CENTRAL: https://172.19.65.106:7443/central/




CPS URLs

---------------------------

                Control Center: https://172.19.65.106:443

                Policy Builder: https://172.19.65.106:443/pb

                       Grafana: https://172.19.65.106:443/grafana

              Unified API WSDL:
https://172.19.65.106:443/ua/wsdl/UnifiedApi.wsdl

               Unified API XSD:
https://172.19.65.106:443/ua/wsdl/UnifiedApi.xsd

                  Unified API: https://172.19.65.106:443/ua/soap

                 CRD REST API: https://172.19.65.106:443/custrefdata

                HAProxy Status: http://172.19.65.106:5540/haproxy?stats

 Service Orchestration API Doc: https://172.19.65.106:443/doc/index.html

             Import/Export API: https://172.19.65.106:443/doc/import.html

                  CPS CENTRAL: https://172.19.65.106:443/central/
```

d. statusall.sh

Sample Output

```
The Monit daemon 5.17.1 uptime: 1d 0h 36m
Process 'whisper'
  status                            Running
  monitoring status                 Monitored
  uptime                            3d 16h 5m
 Process 'snmpd'
  status                            Running
  monitoring status                 Monitored
  uptime                            1d 0h 36m
Process 'qns-1'
  status                            Running
  monitoring status                 Waiting
  uptime                            16h 15m
  status                            Accessible
  monitoring status                 Monitored
Process 'consul'
  status                            Running
  monitoring status                 Monitored
  uptime                            3d 16h 5m
Program 'cpu_load_monitor'
  status                            Status ok
  monitoring status                 Monitored
Program 'cpu_load_trap'
  status                            Status ok
  monitoring status                 Waiting
```

```
Program 'gen_low_mem_trap'
  status                            Status ok
  monitoring status                 Waiting
Process 'collectd'
  status                            Running
  monitoring status                 Monitored
  uptime                            3d 16h 5m
Process 'auditrpms.sh'
  status                            Running
  monitoring status                 Monitored
  uptime                            3d 16h 5m
System 'udc01-UDC-SiteA'
  status                            Running
  monitoring status                 Monitored
The Monit daemon 5.17.1 uptime: 1d 0h 35m
Process 'whisper'
  status                            Running
  monitoring status                 Monitored
  uptime                            3d 16h 4m
Process 'snmpd'
  status                            Running
  monitoring status                 Monitored
  uptime                            1d 0h 35m
Process 'qns-1'
  status                            Running
  monitoring status                 Waiting
  uptime                            16h 15m
```

```
    status                          Accessible
    monitoring status               Monitored
  Process 'consul'
    status                          Running
    monitoring status               Monitored
    uptime                          3d 16h 4m
  Program 'cpu_load_monitor'
    status                          Status ok
    monitoring status               Monitored
  Program 'cpu_load_trap'
    status                          Status ok
    monitoring status               Waiting
  Program 'gen_low_mem_trap'
    status                          Status ok
    monitoring status               Waiting
  Process 'collectd'
    status                          Running
    monitoring status               Monitored
    uptime                          3d 16h 4m
  Process 'auditrpms.sh'
    status                          Running
    monitoring status               Monitored
    uptime                          3d 16h 5m
  System 'udc02-UDC-SiteA'
    monitoring status               Monitored
```

e. list_installed_features.sh

Sample Output

```
Features installed on udc03:9045
 com.broadhop.balance.service.feature=13.0.1.r105254
 com.broadhop.balance.spr.feature=13.0.1.r105254
 com.broadhop.custrefdata.service.feature=13.0.1.r104892
 com.broadhop.diameter2.local.feature=13.0.1.r105455
 com.broadhop.externaldatacache.memcache.feature=13.0.1.r105479
 com.broadhop.infrastructure.feature=13.0.1.r105479
 com.broadhop.ldap.interface.feature=13.0.1.r105421
 com.broadhop.ldap.service.feature=13.0.1.r105421
 com.broadhop.policy.feature=13.0.1.r105479
 com.broadhop.server.runtime.product=13.0.1.r105479
 com.broadhop.snmp.feature=13.0.1.r105479
 com.broadhop.spr.dao.mongo.feature=13.0.1.r104255
 com.broadhop.spr.feature=13.0.1.r104255
 com.broadhop.udcfe.service.feature=13.0.1.r105434
 com.broadhop.udsninterface.service.feature=13.0.1.r105411
 com.broadhop.ui.controlcenter.feature=13.0.1.r104255
 com.broadhop.unifiedapi.interface.feature=13.0.1.r104255
 com.broadhop.unifiedapi.ws.service.feature=13.0.1.r104255
 com.broadhop.vouchers.service.feature=13.0.1.r105254
 com.broadhop.ws.service.feature=13.0.1.r104255
 com.cisco.api.service.feature=13.0.1.r104255
Features installed on udc04:9045
 com.broadhop.balance.service.feature=13.0.1.r105254
 com.broadhop.balance.spr.feature=13.0.1.r105254
 com.broadhop.custrefdata.service.feature=13.0.1.r104892
 com.broadhop.diameter2.local.feature=13.0.1.r105455
```

```
com.broadhop.externaldatacache.memcache.feature=13.0.1.r105479

com.broadhop.infrastructure.feature=13.0.1.r105479

com.broadhop.ldap.interface.feature=13.0.1.r105421

com.broadhop.ldap.service.feature=13.0.1.r105421

com.broadhop.policy.feature=13.0.1.r105479

com.broadhop.server.runtime.product=13.0.1.r105479

com.broadhop.snmp.feature=13.0.1.r105479

com.broadhop.spr.dao.mongo.feature=13.0.1.r104255

com.broadhop.spr.feature=13.0.1.r104255
```

2. From UDC VM, run `ssh` for all other VMs from UDC nodes to verify login is successful without entering the password.

Verify Cluster Manager VM

1. Verify if all peripheral script output includes details on UDC VMs and no UDC related errors are seen by running the following commands:

   a. `hosts.sh`

   b. `diagnostics.sh`

   Sample Output

```
CPS Diagnostics HA Multi-Node Environment

---------------------------

Ping check for all VMs...

Hosts that are not 'pingable' are added to the IGNORED_HOSTS variable...[PASS]

Checking basic ports for all VMs...[PASS]

Checking qns passwordless logins for all VMs...[PASS]

Checking disk space for all VMs...[PASS]

Checking swap space for all VMs...[PASS]

Checking for clock skew for all VMs...[PASS]

Checking CPS diagnostics...
```

```
  Retrieving diagnostics from qns01:9045...[PASS]

  Retrieving diagnostics from qns02:9045...[PASS]

  Retrieving diagnostics from udc03:9045...[PASS]

  Retrieving diagnostics from udc04:9045...[PASS]

  Retrieving diagnostics from lb01:9045...[PASS]

  Retrieving diagnostics from lb01:9046...[PASS]

  Retrieving diagnostics from lb01:9047...[PASS]

  Retrieving diagnostics from lb01:9048...[PASS]

  Retrieving diagnostics from lb02:9045...[PASS]

  Retrieving diagnostics from lb02:9046...[PASS]

  Retrieving diagnostics from lb02:9047...[PASS]

  Retrieving diagnostics from lb02:9048...[PASS]

  Retrieving diagnostics from pcrfclient01:9045...[PASS]

  Retrieving diagnostics from pcrfclient02:9045...[PASS]

Checking redis server instances status on lb01...[PASS]

Checking redis server instances status on lb02...[PASS]

Checking svn sync status between pcrfclient01 & pcrfclient02...[PASS]

Checking HAProxy statistics and ports...[PASS]

Checking TACACS Server Reachability...[PASS]

Checking CPU and memory allocation for all VMs...[PASS]

Checking Virtual IPs...[PASS]

Checking replica sets...

|-------------------------------------------------------------------------------------
-----------------------------------------|

| Mongo:3.2.10                         MONGODB REPLICA-SETS STATUS INFORMATION
Date : 2017-07-17 20:25:59 |

|-------------------------------------------------------------------------------------
-----------------------------------------|
```

```
|  SET NAME  -  PORT  : IP ADDRESS       - REPLICA STATE -           HOST NAME
-  HEALTH  -   LAST SYNC   - PRIORITY |

|-------------------------------------------------------------------------------
-----------------------------------------|

| ADMIN:set05
|

|  Member-1  -  27721 :                  -  ARBITER      -  pcrfclient01
- ON-LINE  -   --------    -    0     |

|  Member-2  -  27721 :                  -  PRIMARY      -  sessionmgr01
- ON-LINE  -   --------    -    3     |

|  Member-3  -  27721 :                  -  SECONDARY    -  sessionmgr02
- ON-LINE  -    1 sec      -    2     |

|-------------------------------------------------------------------------------
-----------------------------------------|

| ADMIN:set06
|

|  Member-1  -  27731 :                  -  ARBITER      -  pcrfclient01
- ON-LINE  -   --------    -    0     |

|  Member-2  -  27731 :                  -  PRIMARY      -  sessionmgr01
- ON-LINE  -   --------    -    3     |

|  Member-3  -  27731 :                  -  SECONDARY    -  sessionmgr02
- ON-LINE  -    1 sec      -    2     |

|-------------------------------------------------------------------------------
-----------------------------------------|

| BALANCE:set02
|

|  Member-1  -  27718 :                  -  ARBITER      -  pcrfclient01
- ON-LINE  -   --------    -    0     |

|  Member-2  -  27718 :                  -  PRIMARY      -  sessionmgr01
- ON-LINE  -   --------    -    3     |

|  Member-3  -  27718 :                  -  SECONDARY    -  sessionmgr02
- ON-LINE  -    0 sec      -    2     |

|-------------------------------------------------------------------------------
-----------------------------------------|
```

```
| REPORTING:set03
|

|  Member-1  -  27719 :                -  ARBITER      -  pcrfclient01
- ON-LINE  -   --------    -    0     |

|  Member-2  -  27719 :                -  PRIMARY      -  sessionmgr01
- ON-LINE  -   --------    -    3     |

|  Member-3  -  27719 :                -  SECONDARY    -  sessionmgr02
- ON-LINE  -    0 sec      -    2     |

|-------------------------------------------------------------------------------
-----------------------------------------|

| SESSION:set01
|

|  Member-1  -  27717 :                -  ARBITER      -  pcrfclient01
- ON-LINE  -   --------    -    0     |

|  Member-2  -  27717 :                -  PRIMARY      -  sessionmgr01
- ON-LINE  -   --------    -    3     |

|  Member-3  -  27717 :                -  SECONDARY    -  sessionmgr02
- ON-LINE  -    0 sec      -    2     |

|-------------------------------------------------------------------------------
-----------------------------------------|

| SESSION:set07
|

|  Member-1  -  27727 :                -  ARBITER      -  pcrfclient01
- ON-LINE  -   --------    -    0     |

|  Member-2  -  27727 :                -  PRIMARY      -  sessionmgr01
- ON-LINE  -   --------    -    1     |

|  Member-3  -  27727 :                -  SECONDARY    -  sessionmgr02
- ON-LINE  -    0 sec      -    1     |

|-------------------------------------------------------------------------------
-----------------------------------------|

| SESSION:set07_udc
|

|  Member-1  -  27737 :                -  ARBITER      -  pcrfclient01
- ON-LINE  -   --------    -    0     |
```

```
|  Member-2  -  27737 :                  -  PRIMARY      -  sessionmgr01
- ON-LINE  -   --------    -    1     |

|  Member-3  -  27737 :                  -  SECONDARY    -  sessionmgr02
- ON-LINE  -    0 sec      -    1     |

|-------------------------------------------------------------------------------
-----------------------------------------|

| SPR:set04
|

|  Member-1  -  27720 :                  -  ARBITER      -  pcrfclient01
- ON-LINE  -   --------    -    0     |

|  Member-2  -  27720 :                  -  PRIMARY      -  sessionmgr01
- ON-LINE  -   --------    -    3     |

|  Member-3  -  27720 :                  -  SECONDARY    -  sessionmgr02
- ON-LINE  -    0 sec      -    2     |

|-------------------------------------------------------------------------------
-----------------------------------------|
```

c. `about.sh`

   For more information about `about.sh` sample output, see *Verify UDC VM* `about.sh` sample output section.

d. `statusall.sh`

   For more information about `statusall.sh` sample output, see *Verify UDC VM* `statusall.sh` sample output section.

e. `list_installed_features.sh`

   For more information about `list_installed_features.sh` sample output, see *Verify UDC VM* `list_installed_features.sh` sample output section.

f. `runonone.sh udc03 /etc/init.d/qns status`

2. From Cluster Manager VM, run `ssh udc<xx>` for all UDC nodes to verify login is successful without entering the password.

Verify PCRFClient VMs

1. Verify if all peripheral script output includes details on UDC VMs and no UDC related errors are seen by running the following commands:

   a. `hosts.sh`

      For more information about `hosts.sh` sample output, see *Verify UDC VM* `hosts.sh` sample output section.

b. `diagnostics.sh`

   For more information about `diagnostics.sh` sample output, see *Verify Cluster Manager VM* `diagnostics.sh` sample output section.

c. `about.sh`

   For more information about `about.sh` sample output, see *Verify UDC VM* `about.sh` sample output section.

d. `statusall.sh`

   For more information about `statusall.sh` sample output, see *Verify UDC VM* `statusall.sh` sample output section.

e. `list_installed_features.sh`

   For more information about `list_installed_features.sh` sample output, see *Verify UDC VM* `list_installed_features.sh` sample output section.

2. From PCRFClient VM, run `ssh udc<xx>` for all UDC nodes to verify login is successful without entering the password.

Verify LB VMs

1. Verify if all peripheral script output includes details on UDC VMs and no UDC related errors are seen by running the following commands:

   a. `hosts.sh`

      For more information about `hosts.sh` sample output, see *Verify UDC VM* `hosts.sh` sample output section.

   b. `about.sh`

      For more information about `about.sh` sample output, see *Verify UDC VM* `about.sh` sample output section.

   c. `statusall.sh`

      For more information about `statusall.sh` sample output, see *Verify UDC VM* `statusall.sh` sample output section.

   d. `list_installed_features.sh`

      For more information about `list_installed_features.sh` sample output, see *Verify UDC VM* `list_installed_features.sh` sample output section.

2. From LB VM, run `ssh udc<xx>` for all UDC nodes to verify login is successful without entering the password.

Modifications in Platform Files

The platform files have some changes and modifications in them. This section describes the diff output of each of the files and the changes in the files.

The following table describes various changes in each of the platform scripts/files:

Diff Output Indicator	Type of Change in File
< hello	Deprecated lines to be deleted.
> goodbye	Lines to be added.
39,40c39,40	The letter in the middle explains how the file was changed. Given below are the different letters used: • C – Changed • A – added • D – deleted The numbers before the letter denotes the line numbers in the original file. The numbers after the letter denotes the line numbers in the updated file.
Test	Lines in bold indicate that they need to be added or modified.

NOTE: These changes are applicable to 13.0.0 platform scripts/files.

jconfig.py

File Name

/var/qps/install/current/scripts/deployer/support/jconfig.py

Diff Output

```
127a128
> **    cluster.UDC = 0**
146a148,149
> **                    elif vmType == "udc":**
> **                        cluster.UDC += 1**
694a698,699
```

```
>         line = "udc_instances=%s\n"%clusterInfo.UDC

>         f.write(line)
```

reinit.sh

File Name

/var/qps/install/current/scripts/upgrade/reinit.sh

Diff Output

```
39,40c39,40

<     fetch_list_of_host_required_reboot
"`/var/qps/install/current/scripts/bin/support/hosts-all.sh '^(sessionmgr[0-9]+|portal[0-
9]+|pcrfclient[0-9]+|qns[0-9]+|lb[0-9]+)$' | tr '\n' ' '`"

<     for host in `/var/qps/install/current/scripts/bin/support/hosts-all.sh
'^(sessionmgr[0-9]+|portal[0-9]+|pcrfclient[0-9]+|qns[0-9]+|lb[0-9]+)$'`

---

>     fetch_list_of_host_required_reboot
"`/var/qps/install/current/scripts/bin/support/hosts-all.sh '^(sessionmgr[0-9]+|portal[0-
9]+|pcrfclient[0-9]+|udc[0-9]+|qns[0-9]+|lb[0-9]+)$' | tr '\n' ' '`"

>     for host in `/var/qps/install/current/scripts/bin/support/hosts-all.sh
'^(sessionmgr[0-9]+|portal[0-9]+|pcrfclient[0-9]+|udc[0-9]+|qns[0-9]+|lb[0-9]+)$'`

99a100,113

>

>   UDC_VMS=`/bin/echo $REBOOT_NEED_HOST_LIST | tr ' ' '\n' | grep '^udc[0-9]'`

>   i=0

>   for udc in $UDC_VMS

>   do

>       rem=$(( $i % 2 ))

>       if [[ $rem -eq 0 ]]; then

>           REBOOT_SET1="$REBOOT_SET1 $udc"

>       else

>           REBOOT_SET2="$REBOOT_SET2 $udc"
```

```
>        fi
>        i=`expr $i + 1`
>   done
>
```

pp_status.sh

File Name

/var/qps/install/current/scripts/upgrade/pp_status.sh

Diff Output

```
41c41
<            hosts=`/var/qps/install/current/scripts/bin/support/hosts-all.sh
'^(sessionmgr[0-9]+|portal[0-9]+|pcrfclient[0-9]+|qns[0-9]+|lb[0-9]+)$'`
---
>            hosts=`/var/qps/install/current/scripts/bin/support/hosts-all.sh
'^(sessionmgr[0-9]+|portal[0-9]+|pcrfclient[0-9]+|udc[0-9]+|qns[0-9]+|lb[0-9]+)$'`
```

copytoall.sh

File Name

/var/qps/install/current/scripts/bin/control/copytoall.sh

Diff Output

```
48c48
< for HOST in $($BIN_SUPPORT/hosts-all.sh '(^pcrfclient[0-9][0-9]$|^lb[0-9][0-9]$|^lbs[0-
9][0-9]$|^qns[0-9][0-9]$|^qnss[0-9][0-9]$|^sessionmgr[0-9][0-9]$|^portal[0-9][0-
9]$|^portallb[0-9][0-9]$|*arbiter)' 'diam|esx'); do
---
> for HOST in $($BIN_SUPPORT/hosts-all.sh '(^pcrfclient[0-9][0-9]$|^lb[0-9][0-9]$|^lbs[0-
9][0-9]$|^udc[0-9][0-9]$|^qns[0-9][0-9]$|^qnss[0-9][0-9]$|^sessionmgr[0-9][0-
9]$|^portal[0-9][0-9]$|^portallb[0-9][0-9]$|*arbiter)' 'diam|esx'); do
```

hosts.sh

File Name

/var/qps/install/current/scripts/bin/support/hosts.sh

Diff Output

```
21c21
< "$BIN_HOME/support/hosts-all.sh" '^(pcrfclient[0-9]+|qns[0-9]+|lb[0-9]+)$' 'diam|esx'
---
> "$BIN_HOME/support/hosts-all.sh" '^(pcrfclient[0-9]+|udc[0-9]+|qns[0-9]+|lb[0-9]+)$'
'diam|esx'
```

diagnostics.sh

File Name

/var/qps/install/current/scripts/bin/diag/diagnostics.sh

Diff Output

```
145c145
<     if [[ $rolename =~ ^(sessionmgr|portal|lb|qns)[0-9]+$ ]]
---
>     if [[ $rolename =~ ^(sessionmgr|portal|lb|udc|qns)[0-9]+$ ]]
151c151
<     QNS_VMS=$($BIN_SUPPORT/hosts.sh | grep qns)
---
>     QNS_VMS=$($BIN_SUPPORT/hosts.sh | egrep 'qns|udc')
338,339c338,343
<             check_port $QNS 8090 "Control Center"
<             #check_port $QNS 9443 "Grafana"
---
>             ## not udc
```

```
>              if [[ ! $QNS =~ ^udc ]]
>              then
>                  check_port $QNS 8090 "Control Center"
>                  #check_port $QNS 9443 "Grafana"
>              fi
```

runonall.sh

File Name

/var/qps/install/current/scripts/bin/control/runonall.sh

Diff Output

```
16c16
< for HOST in $($BIN_SUPPORT/hosts-all.sh '^(sessionmgr[0-9]+|portal[0-9]+|pcrfclient[0-9]+|qns[0-9]+|lb[0-9]+)$'); do
---
> for HOST in $($BIN_SUPPORT/hosts-all.sh '^(sessionmgr[0-9]+|portal[0-9]+|pcrfclient[0-9]+|udc[0-9]+|qns[0-9]+|lb[0-9]+)$'); do
```

sync_times.sh

File Name

/var/qps/bin/support/sync_times.sh

Diff Output

```
53c53
<             output=$($BIN_SUPPORT/hosts-all.sh '(^pcrfclient[0-9][0-9]$|^lb[0-9][0-9]$|^lbs[0-9][0-9]$|^qns[0-9][0-9]$|^qnss[0-9][0-9]$|^sessionmgr[0-9][0-9]$|^portal[0-9][0-9]$|^portallb[0-9][0-9]$|*arbiter)')
---
>             output=$($BIN_SUPPORT/hosts-all.sh '(^pcrfclient[0-9][0-9]$|^lb[0-9][0-9]$|^lbs[0-9][0-9]$|^udc[0-9][0-9]$|^qns[0-9][0-9]$|^qnss[0-9][0-9]$|^sessionmgr[0-9][0-9]$|^portal[0-9][0-9]$|^portallb[0-9][0-9]$|*arbiter)')
```

udc.pp

File Name

/etc/puppet/modules/qps/manifests/roles/udc.pp

This is a new file based on /etc/puppet/modules/qps/manifests/roles/qns.pp in which all instances of the string “qns” is replaced by “udc”.

Diff Output

```
1c1

< # == Class: qps::role:qns

---

> # == Class: qps::role:udc

11c11

< class qps::roles::qns inherits qps::roles::qps_base {

---

> class qps::roles::udc inherits qps::roles::qps_base {

25c25

<       vmType => "qns",

---

>       vmType => "udc",

28c28

<       vmType => "qns",

---

>       vmType => "udc",

33c33

<   Class['qps::collectd_worker'] -> Class['qps']

---

>   Class['qps::collectd_worker'] -> Class['udc']
```

udc.yaml

File Name

/etc/puppet/modules/qps/nodes/udc.yaml

This is a new file.

File Content

```
parameters:

  qnsUserName: param

classes:

  qps::roles::udc:
```

servers

File Name

/var/qps/current_config/etc/broadhop/servers

Diff Output

```
5a6

> lb01=udc_diameter_endpoint

9a11

> lb02=udc_diameter_endpoint

12,13c14,15

< qns03=pcrf

< qns04=pcrf

---

> udc03=udc

> udc04=udc
```

servers.all

File Name

/var/qps/current_config/etc/broadhop/servers.all

Diff Output

```
4c4

< <cluster-name>-lb01=diameter_endpoint

---

> <cluster-name>-lb01=udc_diameter_endpoint

8c8

< <cluster-name>-lb02=diameter_endpoint

---

> <cluster-name>-lb02=udc_diameter_endpoint

11,12c11,12

< <cluster-name>-qns03=pcrf

< <cluster-name>-qns04=pcrf

---

> <cluster-name>-udc03=udc

> <cluster-name>-udc04=udc
```

build_servers_all.sh

File Name

/var/qps/install/current/scripts/build/build_servers_all.sh

Diff Output

```
21c21

< CPS_HOSTS=$($BIN_SUPPORT/hosts-all.sh '(^pcrfclient[0-9][0-9]$|^lb[0-9][0-9]$|^lbs[0-
9][0-9]$|^qns[0-9][0-9]$|^qnss[0-9][0-9]$)')

---
```

```
> CPS_HOSTS=$($BIN_SUPPORT/hosts-all.sh '(^pcrfclient[0-9][0-9]$|^lb[0-9][0-9]$|^lbs[0-
9][0-9]$|^udc[0-9][0-9]$|^qns[0-9][0-9]$|^qnss[0-9][0-9]$)')
```

/images/image-map

File Name

/var/qps/images/image-map

Diff Output

```
4a5
> lb=udc_diameter_endpoint
5a7
> udc=udc
```

/env_config/image-map

File Name

/var/qps/env_config/image-map

Diff Output

```
5a6
> lb=udc_diameter_endpoint
6a7
> udc=udc
```

udc-mop-udcvm.env

File Name

udc-mop-udcvm.env

Example

```
parameters:
  base_vm_image_name: base_image13s1
  cps_az_1: az-1
```

```
  cps_az_4: az-4


  internal_net_name: internalb
  internal_net_cidr: 192.168.78.0/24


  cluman_internal_ip: 192.168.78.18




  udc_flavor_name: qns
  udc01_internal_ip: 192.168.78.91
  udc02_internal_ip: 192.168.78.92
```

udc-mop-udcvm.hot

File Name

udc-mop-udcvm.hot

Example

```
heat_template_version: 2014-10-16


description: A minimal CPS deployment for big bang deployment


parameters:
#=========================
# Global Parameters
#=========================
  base_vm_image_name:
    type: string
    label: base vm image name
```

```
    description: name of the base vm as imported into glance
  cps_install_type:
    type: string
    label: cps installation type (mobile|wifi|mog|pats|arbiter|dra)
    description: cps installation type (mobile|wifi|mog|pats|arbiter|dra)
    default: mobile
  cps_az_4:
    type: string
    label: first availability zone
    description: az for "first half" of cluster
    default: nova
  cps_az_1:
    type: string
    label: second availability zone
    description: az for "second half" of cluster
    default: nova


#=========================
# Network Parameters
#=========================
  internal_net_name:
    type: string
    label: internal network name
    description: name of the internal network
  internal_net_cidr:
    type: string
    label: cps internal cidr
```

```
    description: cidr of internal subnet



#=========================
# Instance Parameters
#=========================
  udc_flavor_name:
    type: string
    label: udc flavor name
    description: flavor udc vm will use
    default: udc
  udc01_internal_ip:
    type: string
    label: internal ip of udc01 vm
    description: internal ip of udc01 vm
  udc02_internal_ip:
    type: string
    label: internal ip of udc02 vm
    description: internal ip of udc02 vm
  cluman_internal_ip:
    type: string
    label: internal ip of cluster manager
    description: internal ip of cluster manager


resources:
#=========================
# Instances
```

```
#=========================


  udc01:
    type: OS::Nova::Server
    properties:
      availability_zone: { get_param: cps_az_1 }
      config_drive: "True"
      image: { get_param: base_vm_image_name }
      flavor: { get_param: udc_flavor_name }
      networks:
        - port: { get_resource: udc01_internal_port }
      user_data_format: RAW
      user_data: { get_resource: udc01_config }
  udc01_internal_port:
    type: OS::Neutron::Port
    properties:
      network: { get_param: internal_net_name }
      fixed_ips: [{ ip_address: { get_param: udc01_internal_ip }}]
  udc01_config:
    type: OS::Heat::CloudConfig
    properties:
      cloud_config:
        write_files:
          - path: /var/lib/cloud/instance/payload/launch-params
          - path: /etc/broadhop.profile
            content: "NODE_TYPE=udc01\n"
          - path: /etc/sysconfig/network-scripts/ifcfg-eth0
```

```
              content:
                str_replace:
                  template: |
                    DEVICE=eth0
                    BOOTPROTO=none
                    NM_CONTROLLED=no
                    IPADDR=$ip
                  params:
                    $ip: { get_param: udc01_internal_ip }
          runcmd:
            - str_replace:
                template: echo $ip installer >> /etc/hosts
                params:
                  $ip: { get_param: cluman_internal_ip }
            - str_replace:
                template: ipcalc -m $cidr >> /etc/sysconfig/network-scripts/ifcfg-eth0
                params:
                  $cidr: { get_param: internal_net_cidr }
            - ifdown eth0 && ifup eth0
            - echo HOSTNAME=udc01 >> /etc/sysconfig/network
            - hostname udc01
      udc02:
        type: OS::Nova::Server
        properties:
          availability_zone: { get_param: cps_az_4 }
          config_drive: "True"
          image: { get_param: base_vm_image_name }
```

```
    flavor: { get_param: udc_flavor_name }
    networks:
      - port: { get_resource: udc02_internal_port }
    user_data_format: RAW
    user_data: { get_resource: udc02_config }
udc02_internal_port:
  type: OS::Neutron::Port
  properties:
    network: { get_param: internal_net_name }
    fixed_ips: [{ ip_address: { get_param: udc02_internal_ip }}]
udc02_config:
  type: OS::Heat::CloudConfig
  properties:
    cloud_config:
      write_files:
        - path: /var/lib/cloud/instance/payload/launch-params
        - path: /etc/broadhop.profile
          content: "NODE_TYPE=udc02\n"
        - path: /etc/sysconfig/network-scripts/ifcfg-eth0
          content:
            str_replace:
              template: |
                DEVICE=eth0
                BOOTPROTO=none
                NM_CONTROLLED=no
                IPADDR=$ip
              params:
```

```
                $ip: { get_param: udc02_internal_ip }
        runcmd:
          - str_replace:
              template: echo $ip installer >> /etc/hosts
              params:
                $ip: { get_param: cluman_internal_ip }
          - str_replace:
              template: ipcalc -m $cidr >> /etc/sysconfig/network-scripts/ifcfg-eth0
              params:
                $cidr: { get_param: internal_net_cidr }
          - ifdown eth0 && ifup eth0
          - echo HOSTNAME=udc02 >> /etc/sysconfig/network
          - hostname udc02
```

udc-mop-udcvm-add-config.yaml

File Name

udc-mop-udcvm-add-config.yaml

Example

```
---
- op: add
  name: "udc01"
  alias: "udc01"
  interfaces:
    - network: "Internal"
      ipAddress: "192.168.78.91"
- op: add
  name: "udc02"
```

```
alias: "udc02"
interfaces:
  - network: "Internal"
    ipAddress: "192.168.78.92"
```