MICRO FOCUS®

OBM Certificate Migration, SHA-1 to SHA-2/SHA-3.

Jun-18-2018

Introduction

This document explains in detail about migrating “OBM” certificates used for “Operations Agent” communication to SHA-2/SHA-3, from SHA-1.

In cryptography, SHA-1 (Secure Hash Algorithm 1) is a cryptographic hash function that produces a 160-bit (20-byte) hash value known as a message digest. The SHA-1 hashing algorithm, which is known to be weak due to advances in cryptographic attacks, hence deprecated by NIST.

SHA-256 (SHA-2) and SHA-512 (SHA-3) are novel hash functions which produce 256 or 512 bit hash values, respectively. Therefore it is recommended to move to SHA-2 signed certificates at a minimum. And this document helps customers to migrate to SHA-2/SHA-3 signed certificates.

There are different ways to do this transition:

1. Re-setup new certificate infrastructure, by completely removing existing certificates on OBM server and OA nodes.

2. Use MigrateAysmKey(.sh|.vbs) tool available as part of Certificate Server to move to SHA-2/SHA-3 signed certificates on OBM server and re-issue certificate on all OA nodes.

   This document – talks more in detail of scenario 2.

IMPORTANT NOTE: Please do not remove any CA Certificates, until the complete process of upgrade is done, to have all the nodes retain the communication for old and new policy deployments and updates.

How to check a certificate’s Hash Algorithm?

- Agent Keystore is available at: %OvDataDir%/datafiles/sec/ks (/var/opt/OV/datafiles/sec/ks) directory.

- The ‘ovrg’ Keystore is available at:
%OvDataDir%/shared/server/datafiles/sec/ks (/var/opt/OV/shared/server/datafiles/sec/ks) directory

- Below command will provide readable description of certificates:

```
#openssl x509 -in <coreid>_cert.pem -text

 #openssl x509 -in CA_<ovrg_coreid>_2048_cert.pem -text
```

Look for the string “***Signature Algorithm***”, to see if the certificates’s hash function.

- ASYMMETRIC_KEY_LENGTH documentation is available here.
- HASH_ALGO documentation is available here.

OBM Keystore Backup

Backup Keystore on DPS Server

The ‘Node’ Keystore on OBM Server (Primary DPS in case of distributed setup) is on filesystem:
%OvDataDir%/datafiles/sec/ks (/var/opt/OV/datafiles/sec/ks) directory.

The ‘ovrg’ Keystore on OBM Server (Primary DPS in case of distributed setup) is on filesystem:
%OvDataDir%/shared/server/datafiles/sec/ks (/var/opt/OV/shared/server/datafiles/sec/ks) directory.

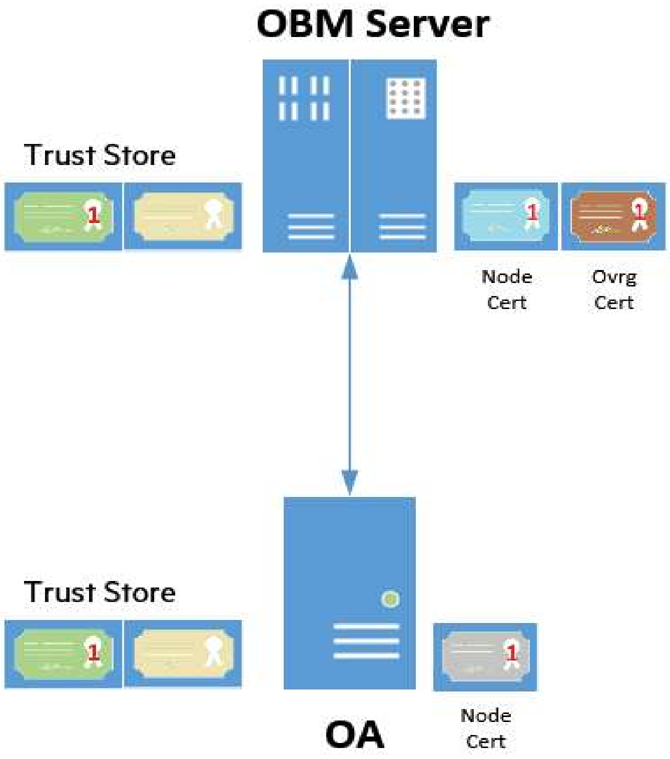
Backup these 2 directories, during the SHA-1 to SHA-2 migration.
Please make sure these backups are ‘safe’ & ‘secure’, during the course of migration, with appropriate permissions.

The back certs can be used in case, the processes needs to be reversed, for unforeseen reasons.

Delete the backups, diligently post successful migration.

OBM Single Server Setup

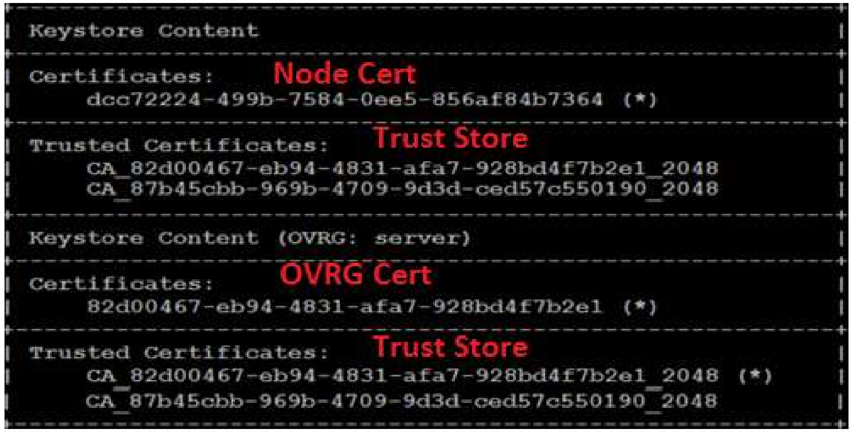
1.0 Single Server - Initial State with SHA-1 Certs



In the existing environment, all the certs are SHA1 based.

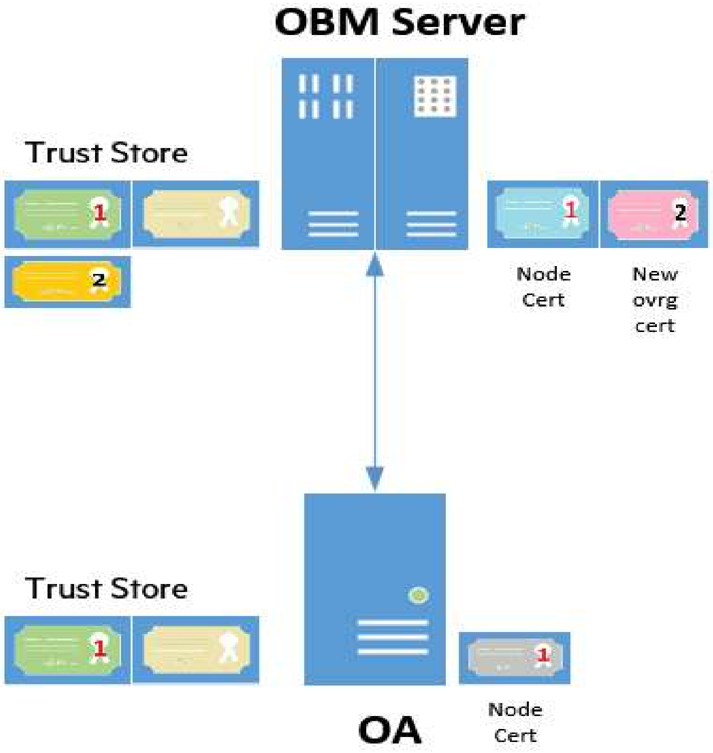
“ovcert -list” command can be run to correlate the certificates in the picture.

Sample Keystore:



```
+---------------------------------------------------------------+
| Keystore Content                                              |
+---------------------------------------------------------------+
| Certificates:      Node Cert                                  |
|     dcc72224-499b-7584-0ee5-856af84b7364 (*)                  |
+---------------------------------------------------------------+
| Trusted Certificates:     Trust Store                         |
|     CA_82d00467-eb94-4831-afa7-928bd4f7b2e1_2048              |
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048              |
+---------------------------------------------------------------+
| Keystore Content (OVRG: server)                               |
+---------------------------------------------------------------+
| Certificates:      OVRG Cert                                  |
|     82d00467-eb94-4831-afa7-928bd4f7b2e1 (*)                  |
+---------------------------------------------------------------+
| Trusted Certificates:     Trust Store                         |
|     CA_82d00467-eb94-4831-afa7-928bd4f7b2e1_2048 (*)          |
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048              |
+---------------------------------------------------------------+
```

1.1 Single Server – Create SHA-2 CA Certificate



- Move to a stronger RSA key size on the OBM server as well as managed nodes by setting ASYMMETRIC_KEY_LENGTH configuration under sec.cm namespace

```
#ovconfchg -ns sec.cm -set ASYMMETRIC_KEY_LENGTH 4096
```

Note: Here were are trying to keep existing certs intact. Hence choosing a different key length to create new CA certs, with retaining coreid.

- Set HASH_ALGO configuration under sec.core namespace to desired and supported hash algorithm on OBM server

```
#ovconfchg -ns sec.core -set HASH_ALGO eSHA256
```

- Run MigrateAsymkey tool with “-createCAcert” option, this creates new CA certificate for 4096 RSA key size, signed using hash algorithm configured.

```
  #/opt/OV/lbin/seccs/install/MigrateAsymKey.sh –createCAcert
  #cscript.exe “%OvInstallDir%”\lbin\seccs\install\MigrateAsymKey.vbs -
createCAcert
```

1.1.1 Single Server – Cert Listing

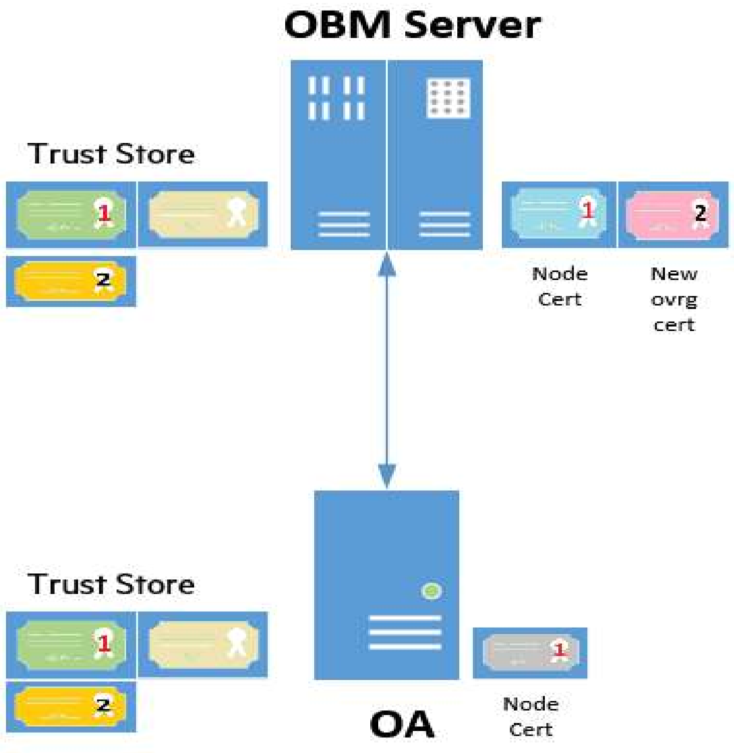
```
#ovcert -list
+----------------------------------------------------------+
| Keystore Content                                         |
+----------------------------------------------------------+
| Certificates:                                            |
|     98fbd3ff-e468-43db-b8f6-dbd801bac59a (*)             |
+----------------------------------------------------------+
| Trusted Certificates:                                    |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_2048         |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048         |        |
+----------------------------------------------------------+

+----------------------------------------------------------+
| Keystore Content (OVRG: server)                          |
+----------------------------------------------------------+
| Certificates:                                            |
|     ae51ac62-783b-75a0-0e10-fe195d13d9cf (*)             |
+----------------------------------------------------------+
| Trusted Certificates:                                    |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_2048 (*)     |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096 (*)     |
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048         |
+----------------------------------------------------------+
```

From the The ‘ovrg’ Keystore

```
#openssl x509 -in CA_<ovrg_coreid>_4096_cert.pem –text
```

“***Signature Algorithm***” will show: sha256WithRSAEncryption

1.2 Single Server – Update Trusted for all Agents



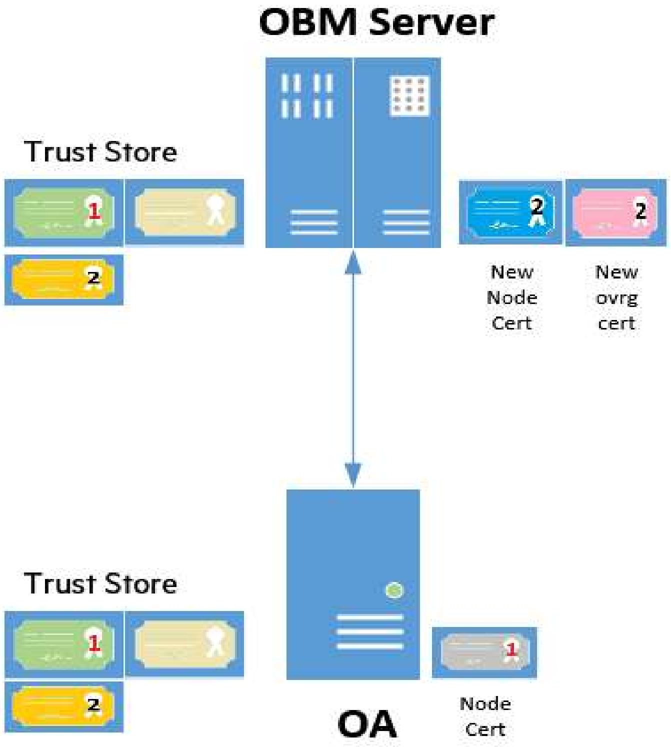
- On OBM Server, make sure “Certificate Server” is running, if its in stopped state, start it.

```
#ovc -status
#ovc -start ovcs
```

- Update trusted certificates, using “ovcert -updatetrusted” command.

```
#ovcert -updatetrusted
```

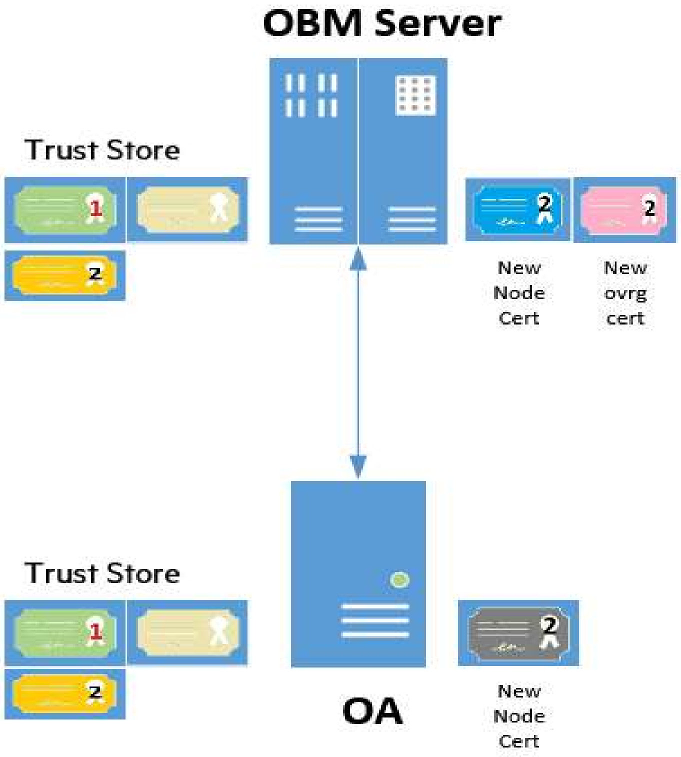
1.3 Single Server – Issue new node cert on OBM



- Create new node certificate for local agent and other keystores using MigrateAsymkey tool with “-createNodecert” option.

```
#/opt/OV/lbin/seccs/install/MigrateAsymKey.sh –createNodecert

#cscript.exe "%OvInstallDir%"\lbin\seccs\install\MigrateAsymKey.vbs
-createNodecert
```

1.4 Single Server – Update Agents with SHA-2 Certs



To have the nodes with only SHA-2 certificates follow below steps

- Remove all existing certificates on the node using “ovcert -remove” command.
- Ensure HASH_ALGO and `ASYMMETRIC_KEY_LENGTH` is the same as the OBM Server

```
#ovconfchg -ns sec.core -set HASH_ALGO eSHA256
#ovconfchg -ns sec.cm -set ASYMMETRIC_KEY_LENGTH 4096
```

- Request for new certificate using “ovcert -certreq” command and grant the same from OBM server.

```
#ovcert -certreq
```

- Grant the certificate request from OBM Server.

Tip: A script can be launched on nodes. Or automation tools can also do it.

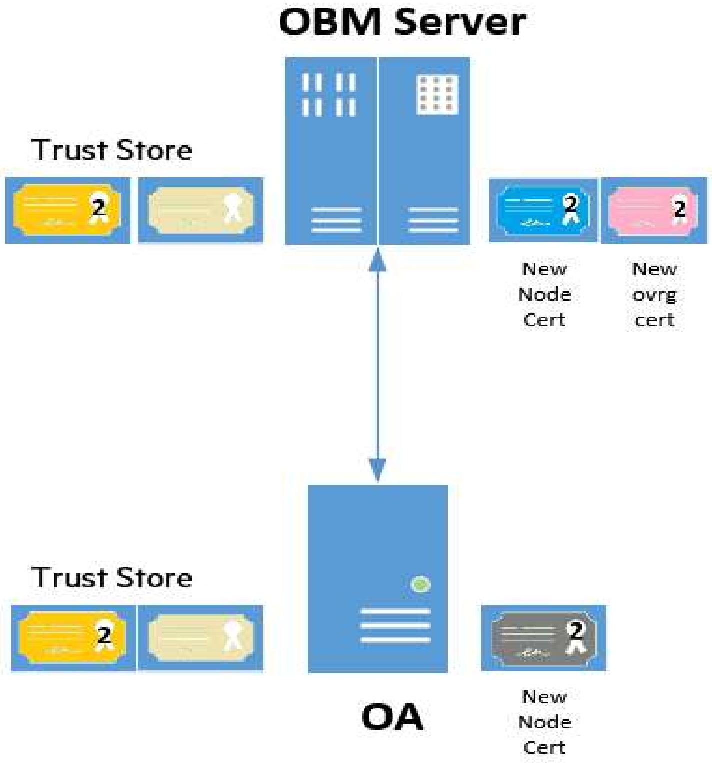
After having new certificates on the Nodes, OBM setup will not be fully operational until all the policies have been redeployed.

Redeployment of policies is required to override the policies with new certificates.

Either GUI or the below CLI can be used, to redeploy policies.

```
#$TOPAZ_HOME/opr/bin/opr-agt.sh(.bat) -username <admin> -password
<admin Password>  -deploy -force -node_list | -nl <nodes comma
separated> or -view_name | -vn <exact view names used for auto
assignment>
```

1.5 Single Server – Remove SHA-1 Certificates



After all the agents are migrated, and communication between OBM and OA is intact remove old CA cert from server trust stores and do update trusted on all agents.

On OBM Server:

- Remove SHA-1 trusted certificates

```
#ovcert -remove <SHA1_CA_Cert>
#ovcert -remove <SHA1_CA_Cert> -ovrg server
```

- Save the certificates permanently in DB.

  #$TOPAZ_HOME/opr/bin/opr-configure-certificates.bat(.sh) -il

On Agents:

- Update trusted certificates, using “ovcert -updatetrusted” command.

```
#ovcert -updatetrusted
```

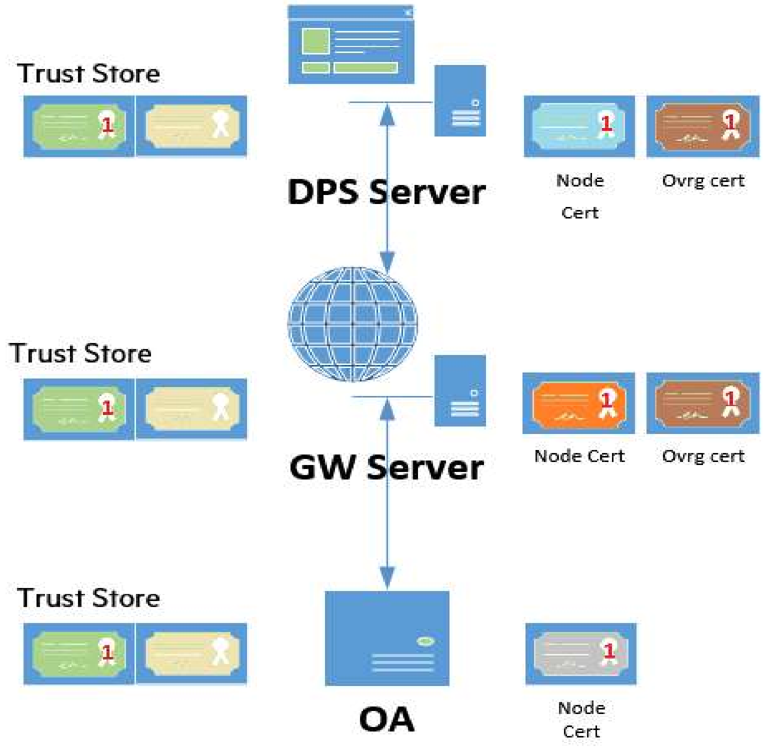
1.6 Single Server – After Successful Migration

```
#ovcert -list (On OBM)
+----------------------------------------------------------+
| Keystore Content                                         |
+----------------------------------------------------------+
| Certificates:                                            |
|     98fbd3ff-e468-43db-b8f6-dbd801bac59a (*)             |
+----------------------------------------------------------+
| Trusted Certificates:                                    |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096         |
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048         |
+----------------------------------------------------------+

+----------------------------------------------------------+
| Keystore Content (OVRG: server)                          |
+----------------------------------------------------------+
| Certificates:                                            |
|     ae51ac62-783b-75a0-0e10-fe195d13d9cf (*)             |
+----------------------------------------------------------+
| Trusted Certificates:                                    |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096 (*)     |
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048         |
+----------------------------------------------------------+
```

```
#ovcert -list (On Agent)
+----------------------------------------------------------+
| Keystore Content                                         |
+----------------------------------------------------------+
| Certificates:                                            |
|     5ba665e4-9509-75a0-162f-a775678d2fcb (*)             |
+----------------------------------------------------------+
| Trusted Certificates:                                    |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096         |
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048         |
+----------------------------------------------------------+

# bbcutil -ping https://<OBM>/com.hp.ov.opc.msgr

<OBM>/com.hp.ov.opc.msgr:
          status=eServiceOK coreID=ae51ac62-783b-75a0-0e10-fe195d13d9cf
          bbcV=12.06.008  appN=OBM appV=10.70.007.001 conn=0 time=160 ms
```

OBM Distributed Setup

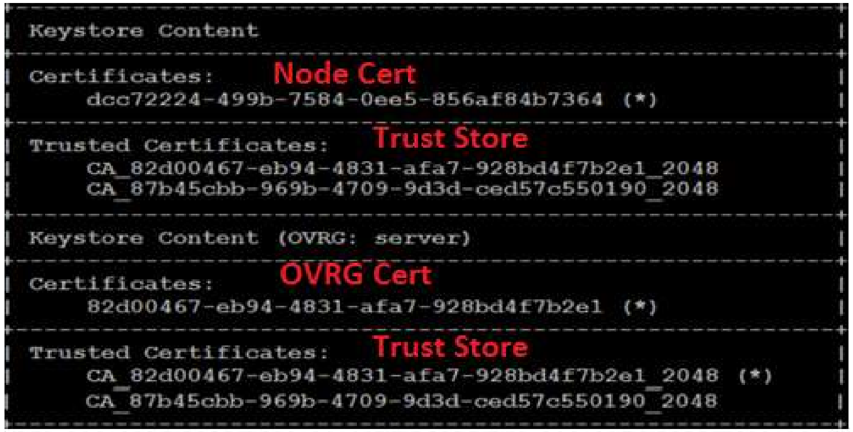
2.0 Distributed Server - Initial State with SHA-1 Certs



In the existing environment, all the certs are SHA1 based.

“ovcert -list” command can be run to correlate the certificates in the picture.

Sample Keystore:



```
+---------------------------------------------------------------+
| Keystore Content                                              |
+---------------------------------------------------------------+
| Certificates:                                                 |
|     dcc72224-499b-7584-0ee5-856af84b7364 (*)                  |
+---------------------------------------------------------------+
| Trusted Certificates:                                         |
|     CA_82d00467-eb94-4831-afa7-928bd4f7b2e1_2048              |
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048              |
+---------------------------------------------------------------+
| Keystore Content (OVRG: server)                               |
+---------------------------------------------------------------+
| Certificates:                                                 |
|     82d00467-eb94-4831-afa7-928bd4f7b2e1 (*)                  |
+---------------------------------------------------------------+
| Trusted Certificates:                                         |
|     CA_82d00467-eb94-4831-afa7-928bd4f7b2e1_2048 (*)          |
|     CA_87b45cbb-969b-4709-9d3d-ced57c550190_2048              |
+---------------------------------------------------------------+
```

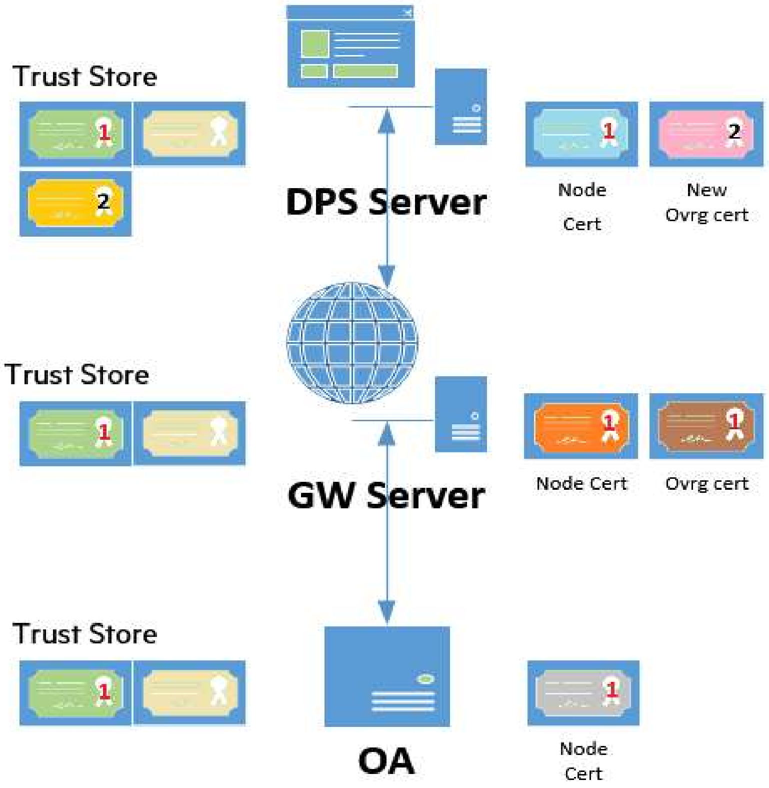
Node Cert

Trust Store

OVRG Cert

Trust Store

2.1 Distributed Server – Create SHA-2 CA Certificate



- On “Primary DPS”, Move to a stronger RSA key size on the OBM server as well as managed nodes by setting ASYMMETRIC_KEY_LENGTH configuration under sec.cm namespace

```
#ovconfchg -ns sec.cm -set ASYMMETRIC_KEY_LENGTH 4096
```

```
Note: Here were are keeping existing certs intact, so that Agent
communication is not broken.
Hence choosing a different key length to create new CA certs with
the same server coreid, later old certificates are discarded.
```

- Set HASH_ALGO configuration under sec.core namespace to desired and supported hash algorithm on OBM server

```
#ovconfchg -ns sec.core -set HASH_ALGO eSHA256
```

- Run MigrateAsymkey tool with “-createCAcert” option, this creates new CA certificate for 4096 RSA key size, signed using hash algorithm configured.

```
#/opt/OV/lbin/seccs/install/MigrateAsymKey.sh –createCAcert
#cscript.exe “%OvInstallDir%”\lbin\seccs\install\MigrateAsymKey.vbs -
createCAcert
```

2.1.1 Distributed Server – Cert Listing

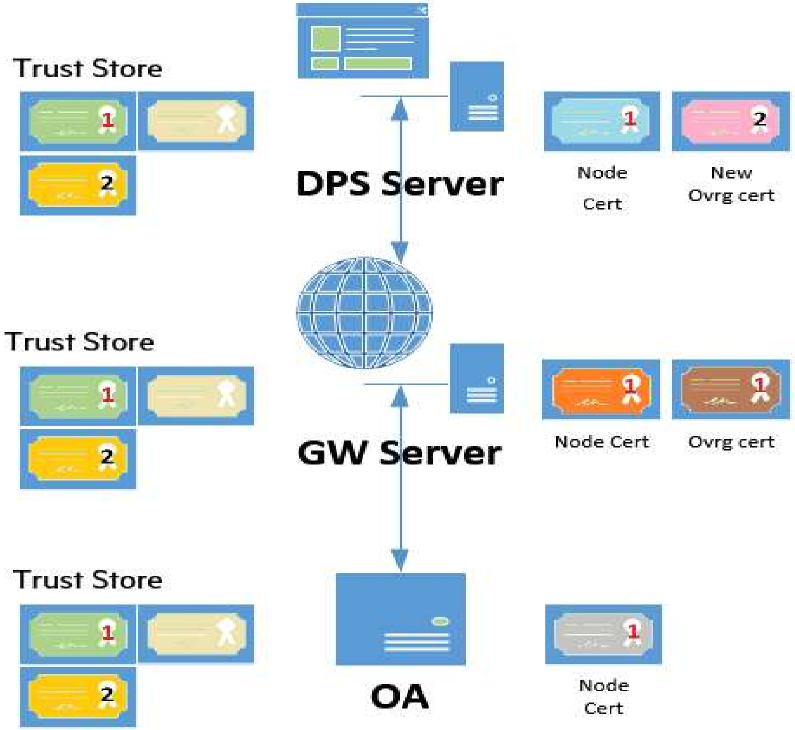
```
#ovcert -list
+----------------------------------------------------------+
| Keystore Content                                         |
+----------------------------------------------------------+
| Certificates:                                            |
|     98fbd3ff-e468-43db-b8f6-dbd801bac59a (*)             |
+----------------------------------------------------------+
| Trusted Certificates:                                    |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_2048         |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096         |
+----------------------------------------------------------+

+----------------------------------------------------------+
| Keystore Content (OVRG: server)                          |
+----------------------------------------------------------+
| Certificates:                                            |
|     ae51ac62-783b-75a0-0e10-fe195d13d9cf (*)             |
+----------------------------------------------------------+
| Trusted Certificates:                                    |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_2048 (*)     |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096 (*)     |
+----------------------------------------------------------+
```

From the The ‘ovrg’ Keystore

```
#openssl x509 -in CA_<ovrg_coreid>_4096_cert.pem –text
```

“*Signature Algorithm*” will show: sha256WithRSAEncryption

2.2 Distributed Server – Update Trusted on all GW + Agents



- On Primary DPS Server, make sure “Certificate Server is running. If its in stopped state, start it.

```
#ovc -status
#ovc -start ovcs
```

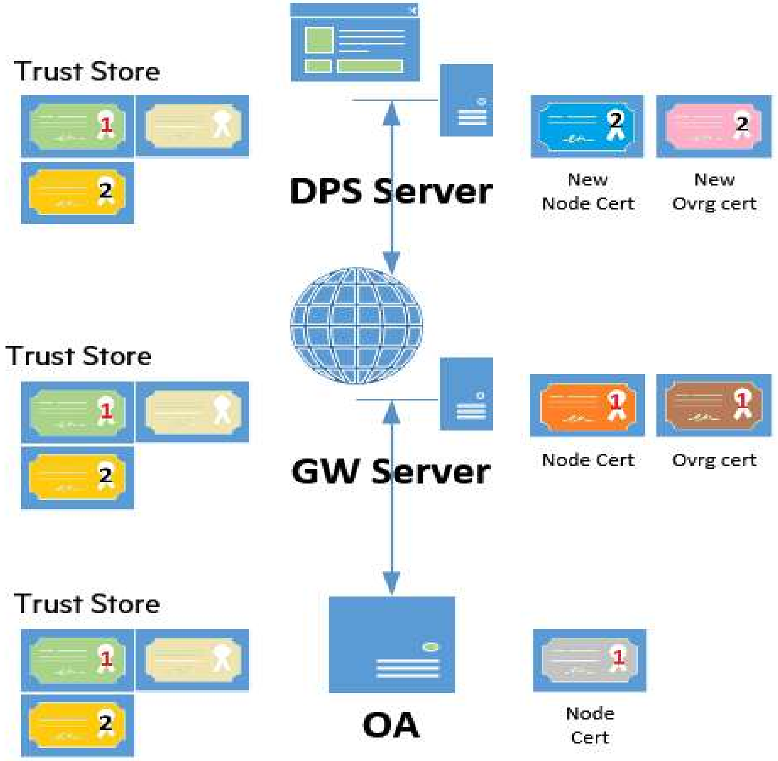
- On all GW Server, update trusted certs.

```
#ovcert -updatetrusted
```

- On all Agents Update trusted certificates, using “ovcert -updatetrusted” command.

```
#ovcert -updatetrusted
```

2.3 Distributed Server – Issue new node cert on DPS



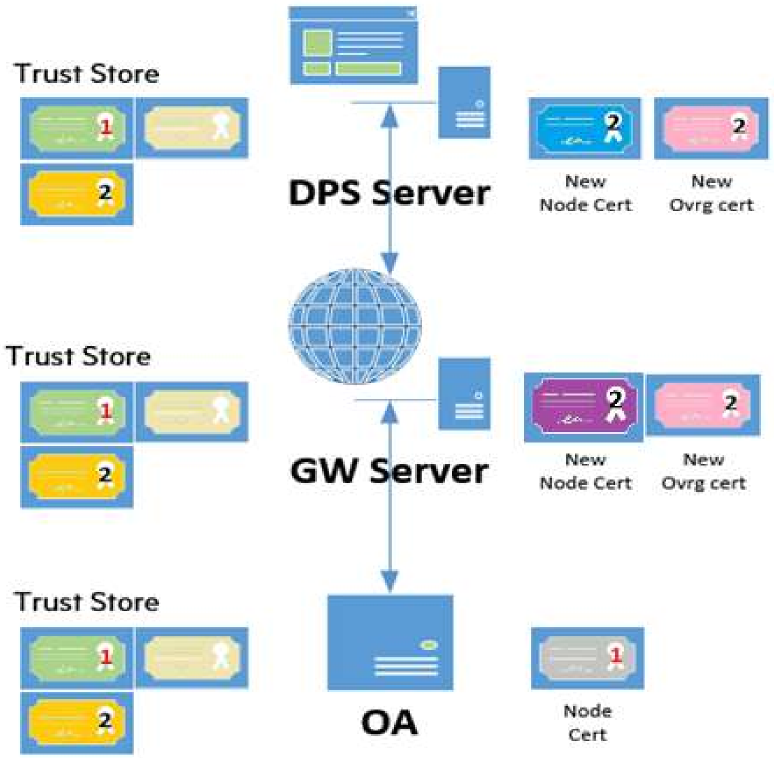
- Create new node certificate for local agent and other keystores using MigrateAsymkey tool with "-createNodecert" option.

```
#/opt/OV/lbin/seccs/install/MigrateAsymKey.sh –createNodecert

#cscript.exe "%OvInstallDir%"\lbin\seccs\install\MigrateAsymKey.vbs
–createNodecert
```

- Update the DB with new certs, so that GW Servers pick the new Certs, during next step.

```
#$TOPAZ_HOME/opr/bin/opr-configure-certificates.bat(.sh) -il
```

2.4 Distributed Server – ReConfigure GW Servers



- Configure the Algorithm and Key length as per new requirements.

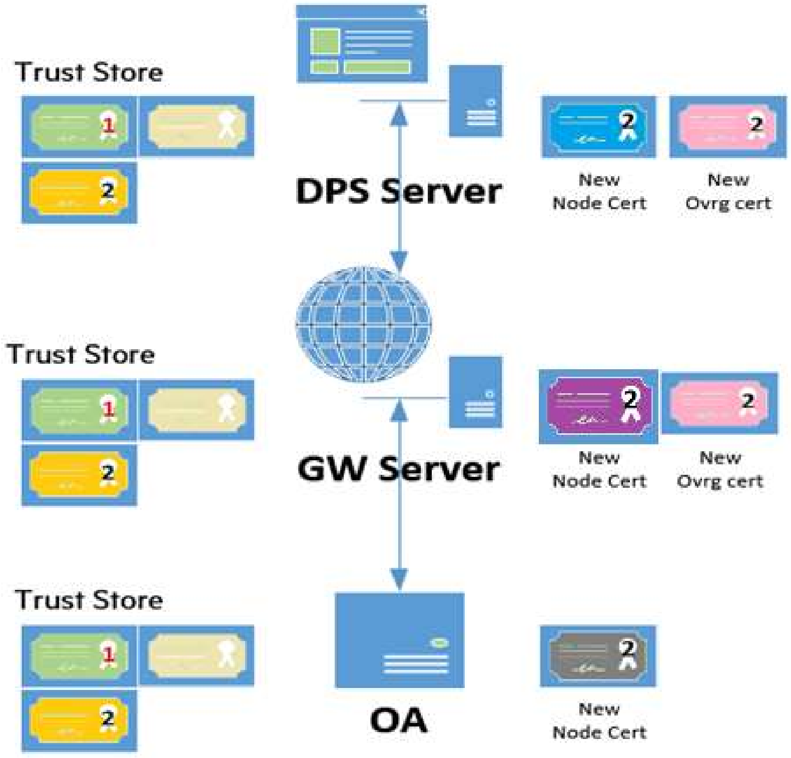
```
#ovconfchg -ns sec.cm -set ASYMMETRIC_KEY_LENGTH 4096
#ovconfchg -ns sec.core -set HASH_ALGO eSHA256
```

- Remove the node certificate:

```
#ovcert -remove <coreid>
```

- Run the Configuration Wizard on GW Servers.

2.5 Distributed Server – Update Agents with SHA-2 Certs



To have the nodes with only SHA-2 certificates follow below steps

- Remove all existing certificates on the node using “ovcert -remove” command.
- Ensure HASH_ALGO and `ASYMMETRIC_KEY_LENGTH` is the same as the OBM Server

```
#ovconfchg -ns sec.core -set HASH_ALGO eSHA256
#ovconfchg -ns sec.cm -set ASYMMETRIC_KEY_LENGTH 4096
```

- Request for new certificate using “ovcert -certreq” command and grant the same from OBM server.

```
#ovcert -certreq
```

- Grant the certificate request from OBM Server.

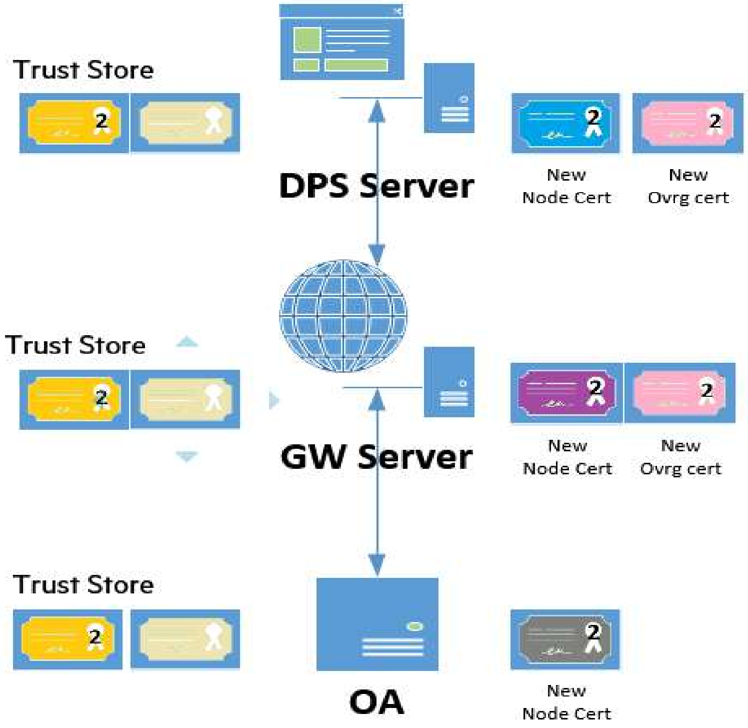
Tip: A script can be launched on nodes. Or automation tools can also do it.

After having new certificates on the Nodes, OBM setup will not be fully operational until all the policies have been redeployed.

Redeployment of policies is required to override the policies with new certificates.
Either GUI or the below CLI can be used, to redeploy policies.

```
#$TOPAZ_HOME/opr/bin/opr-agt.sh(.bat) -username <admin> -password
<admin Password>  -deploy -force -node_list | -nl <nodes comma
separated> or -view_name | -vn <exact view names used for auto
assignment>
```

2.6 Distributed Server – Remove SHA-1 Certificates



After all the agents are migrated, and communication between OBM and OA is intact remove old CA cert from server trust stores and do update trusted on all agents.

On Primary DPS Server:

- Remove SHA-1 trusted certificates

```
#ovcert -remove <SHA1_CA_Cert>
#ovcert -remove <SHA1_CA_Cert> -ovrg server
```

- Save the certificates permanently in DB.

```
#$TOPAZ_HOME/opr/bin/opr-configure-certificates.bat(.sh) -il
```

On GW:

- Update trusted certificates, using “ovcert -updatetrusted” command.

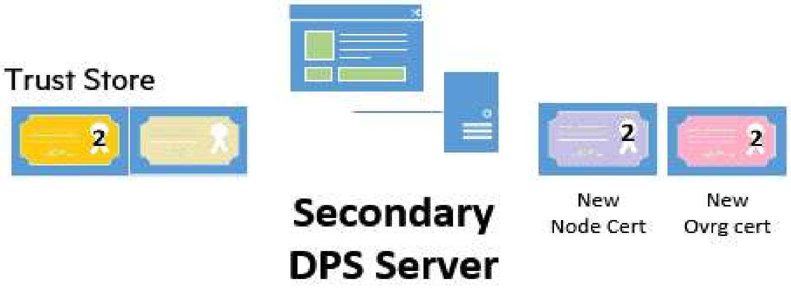
```
#ovcert -updatetrusted
```

On Agents:

- Update trusted certificates, using “ovcert -updatetrusted” command.

```
#ovcert -updatetrusted
```

2.7 Distributed Server – Update Secondary DPS



If there is a Secondary DPS in the environment:

- Configure the Algorithm and Key length as per new requirements.

```
#ovconfchg -ns sec.cm -set ASYMMETRIC_KEY_LENGTH 4096
#ovconfchg -ns sec.core -set HASH_ALGO eSHA256
```

- Remove the node certificate:

```
#ovcert -remove <coreid>
```

- Run the Configuration Wizard on GW Servers.

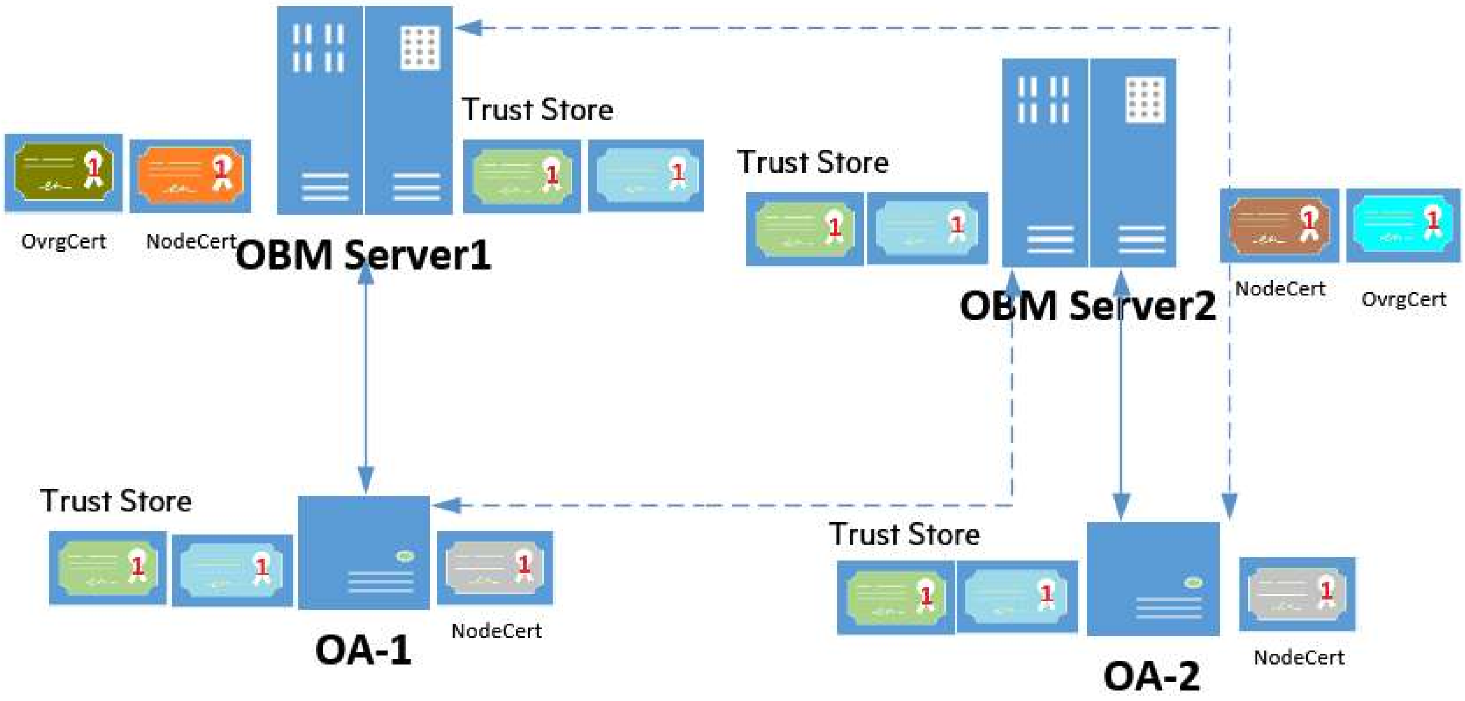
2.8 Distributed Server – After Successful Migration

```
#ovcert –list (On OBM)
+---------------------------------------------------------+
| Keystore Content                                        |
+---------------------------------------------------------+
| Certificates:                                           |
|     98fbd3ff-e468-43db-b8f6-dbd801bac59a (*)            |
+---------------------------------------------------------+
| Trusted Certificates:                                   |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096        |
+---------------------------------------------------------+

+---------------------------------------------------------+
| Keystore Content (OVRG: server)                         |
+---------------------------------------------------------+
| Certificates:                                           |
|     ae51ac62-783b-75a0-0e10-fe195d13d9cf (*)            |
+---------------------------------------------------------+
| Trusted Certificates:                                   |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096 (*)    |
+---------------------------------------------------------+
```

```
#ovcert –list (On Agent)
+---------------------------------------------------------+
| Keystore Content                                        |
+---------------------------------------------------------+
| Certificates:                                           |
|     5ba665e4-9509-75a0-162f-a775678d2fcb (*)            |
+---------------------------------------------------------+
| Trusted Certificates:                                   |
|     CA_ae51ac62-783b-75a0-0e10-fe195d13d9cf_4096        |
+---------------------------------------------------------+

# bbcutil -ping https://<GW>/com.hp.ov.opc.msgr

<GW>/com.hp.ov.opc.msgr:
          status=eServiceOK coreID=ae51ac62-783b-75a0-0e10-fe195d13d9cf
          bbcV=12.06.008  appN=OBM appV=10.70.007.001 conn=0 time=160 ms
```

OBM MoM Setup

3.0 MoM – Initial Setup



3.1 MoM – Update OBM-1 Environments

- Create new SHA2 signed CA certificate on OBM-1 (Refer 1.1 section).
- Update trusted certificates on all OBM-1 Agents (Refer 1.2 section)
- Export trusted certificates from OBM-1

  ```
  #overt -exporttrusted -file OBM-1_trusted.cert
  ```

  Transfer obm_trusted.cert to OBM-2 Server, import trusted certificates on OBM2.

  ```
  #ovcert -importtrusted -file OBM-1_trusted.cert
  ```

- Update trusted on all OBM-2 Agents

  ```
  #ovcert -updatetrusted
  ```

- Issue new sever node certificate on OBM-1 (Refer 1.3 section)
- Migrate all OBM-1 Agents to new certificates and redeploy all policies afterwards (per node) (Refer 1.4 section)
- After all the agents are migrated remove old CA cert from server trust stores and do update trusted on all agents. (Refer 1.5 section)

3.2 MoM – Update OBM-2 Environment

- Create new SHA2 signed CA certificate on OBM-2 (Refer 1.1 section).

- Update trusted certificates on all OBM-2 Agents (Refer 1.2 section)

- Export trusted certificates from OBM-2

  ```
  #overt -exporttrusted -file OBM-2_trusted.cert
  ```

  Transfer obm_trusted.cert to OBM-1 Server, import trusted certificates on OBM2.

  ```
  #ovcert -importtrusted -file OBM-2_trusted.cert
  ```

- Update trusted on all OBM-2 Agents

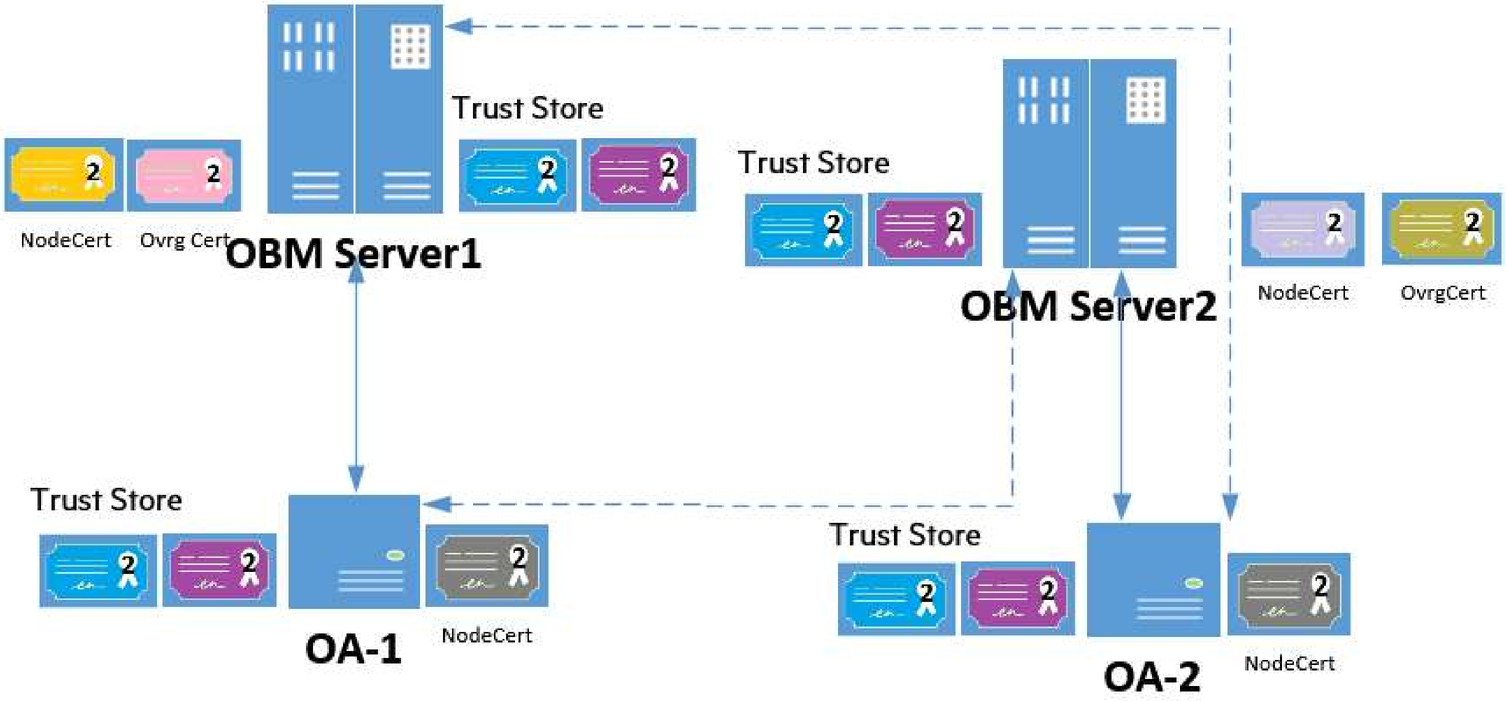
  ```
  #ovcert -updatetrusted
  ```

- Issue new sever node certificate on OBM-1 (Refer 1.3 section)

- Migrate all OBM-2 Agents to new certificates and redeploy all policies afterwards (per node) (Refer 1.4 section)

- After all the agents are migrated remove old CA cert from server trust stores and do update trusted on all agents. (Refer 1.5 section)

3.3 MoM – Final Setup


Trust Store
NodeCert
Ovrg Cert
OBM Server1
Trust Store
OBM Server2
NodeCert
OvrgCert
Trust Store
OA-1
NodeCert
Trust Store
OA-2
NodeCert