Boomi Cloud™ API Management - Local Edition

Migration and Upgrade Guide

Version 5.6.2 | November 2024

Contents

Migrating to Boomi Cloud™ API Management - Local Edition 5.x

The Boomi Cloud™ API Management - Local Edition(Local Edition) architecture provides migration path for 5.x projects with a few exceptions. This document explains how to migrate to Local Edition 5.x.

Migration Prerequisites

Download a new version of Local Edition

Download the version of Local Edition to which you want to migrate.

Create Local Edition Images

The Local Edition has the following containers:

- tml-cache
- tml-sql
- tml-tm
- tml-log
- tml-cm
- tml-nosql
- tml-reporting

For more information, see the "Boomi Cloud™ API Management - Local Edition Components" topic in the *Boomi Cloud™ API Management - Local Edition Installation and Configuration Guide*.

Container image is a template, comprising of the application along with the required binaries and libraries, needed to build a running container or a pod (the running instance of that image). The new Local Edition cluster may upgrade one or more container types.

The container images can be built on any machine and not necessarily on the machines hosting the Local Edition Cluster.

To create the images for all containers from the Local Edition-installer Jenkins job, see the "Boomi Cloud™ API Management - Local Edition Installer" topic in the *Boomi Cloud™ API Management - Local Edition Installation and Configuration Guide*.

Migrating MySQL Data from Version 5.4.1 Untethered Mode to Version 5.5.0 Untethered Mode

You can migrate MySQL data from the cluster created in an older version of Local Edition.

The following section gives the procedure to migrate MySQL data for the following versions and modes of Local Edition.

- 5.3.1 to 5.5.0 (untethered)
- 5.4.1 to 5.5.0 (untethered)

Procedure

1. Dump MySQL Data from the cluster created in the 5.4.1 version.
    a. Login to any tml-sql pod running in the 5.4.1 cluster.

    ```
    kubectl exec -it mysql-set-0-0 -- /bin/bash
    ```

    b. Run the following command to dump MySQL data. Update the MySQL password for the cluster.

    ```
    mysqldump --no-create-info --complete-insert --single-transaction -u root -
    p'changeme' \
      --ignore-table=masherysolar.area_config --ignore-table=masherysolar.areas \
      --ignore-table=masherysolar.package_key_audit_log --setgtid-purged=OFF
    masherysolar \
      > /tmp/mash541_data.sql
    ```

> **Note:** In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.
>
> - masherysolar.member_activity_log
> - masherysolar.member_audit_log
> - masherysolar.member_role_audit_log
> - masherysolar.method_override_audit_log
> - masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
kubectl cp mysql-set-0-0:/tmp/mash531_data.sql mash541_data.sql
```

2. Copy the MySQL data dumped from the 5.4.1 cluster to the 5.5.0 cluster.

```
kubectl cp mash541_data.sql mysql-set-0-0:/tmp
```

3. Clear the MySQL data in the 5.5.0 cluster.

a. In the 5.5.0 cluster, login to the tml-sql pod where the MySQL data file was copied.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

b. Run the following command to clear data in MySQL.

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar \
| sed -r 's/\b(area_config|areas|package_key_audit_log)\b//g' \
| sed '/^$/d' | while read table; do mysql -u root -p'changeme' \
-e "truncate table $table" masherysolar; done
```

4. Import MySQL data dumped from the 5.4.1 cluster to the 5.5.0 cluster.

   In the tml-sql pod where the MySQL data file is copied, run the following command to import the MySQL data dumped from the 5.4.1 cluster to the 5.5.0 cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash541_data.sql
```

5. Enable revert function.

   In the tml-sql pod run the following commands to enable revert function.

   ```
   cd /opt/mashery/containeragent/resources/sql/scripts
   ./sql-upgrade.sh 5.4.1
   ```

6. Refresh cache of the 5.5.0 cluster.

   Login to each of the tml-cache pods of the 5.5.0 cluster and run the following command to refresh the cache.

   ```
   /opt/javaproxy/proxy/cacheloader --service --mapi --devclass --packager \
   --httpsclientsecurity --env production --verbose
   ```

Migrating Local Edition Cluster from 4.4.x to 5.5.1

The Local Edition cluster can be migrated from versions 4.4.x to 5.5.1.

Before you begin

- Local Edition 5.5.1 cluster must be set up and running in tethered mode.
- Apache Cassandra cluster in 5.5.1 must be set up to enable connection from 4.4.x cluster.
- Local Edition 5.5.1 cluster is up and running but in stand by mode.

Perform the following steps in sequence:

Procedure

1. Installing updated traffic manager on 4.4.x runing in OVA
2. Installing updated traffic manager on 4.4.x runing in Docker
3. Migrating tokens from 4.4.x to 5.5.1

Installing Updated Traffic Manager on 4.4.x Running in OVA

The following section gives the procedure to install updated traffic manager on 4.4.x running in OVA:

Procedure

1. Save or backup the existing mashery-proxy-config.json.
2. Download TIB_mashery_local_4.4.3.HF001_ova.tar.gz from the Boomi Support portal and extract it to /opt/mashery/install.

   It contains mashery-local-4.4.3.HF001-GA.x86_64.rpm and mashery-proxy-config.json.tmpl.

3. Update mashery-proxy-config.json.tmpl with values from the existing configuration.
4. Update the Cassandra client configuration section com.mashery.cassandra.client.cassandra-client with the properties from 5.5.1 cluster.

   This is to enable 4.4.x installation to access the Cassandra database for token access and migration.

   It is recommended that the client connection to Cassandra is a secure connection.
5. If you are using SSL, update the SSL configuration section factory:com.mashery.proxy.server.ssl, otherwise delete the section.
6. Update the following sections accordingly:
   - com.mashery.service.onprem.oauth.authenticator.oauth-service-authenticator
   - com.mashery.service.onprem.api-server
7. Update traffic manager with:

   ```
   yum install -y mashery-local-4.4.3.HF001-GA.x86_64.rpm
   ```
8. Replace /etc/mashery-proxy-config.json with the updated mashery-proxy-config.json.tmpl
9. Restart traffic manager.

Installing Updated Traffic Manager on 4.4.x Running in Docker

The following section gives the procedure to build updated Docker images on 4.4.x Docker.

Procedure

1. Save or backup the existing mashery-proxy-config.json.
2. Download TIB_mashery_local_4.4.3.HF001_docker.tar.gz and extract it to /opt/mashery/install directory.
3. Expand the images with

   ```
   build-docker.sh --extract
   ```

4. Update {install}/resources/mlce-runtime/MLCE_build_A/ml-docker/mashery-proxy-config.tmpl with values from the existing configuration.

   mashery-proxy-config.tmpl can be found in /opt/mashery/install/resources/mlce-runtime/MLCE_build_A/ml-docker/mashery-proxy-config.tmpl.

   Variables are parameter with enclosed %%.

5. Update the Cassandra client configuration section com.mashery.cassandra.client.cassandra-client with the properties from 5.5.1 cluster.

   This is to enable 4.4.x installation to access the Cassandra database for token access and migration.

   Boomi recommends that the client connection to Cassandra is a secure connection.

6. If you are using SSL, update the SSL configuration section factory:com.mashery.proxy.server.ssl. Otherwise, delete the section.
7. Update the following sections accordingly:
   - com.mashery.service.onprem.oauth.authenticator.oauth-service-authenticator
   - com.mashery.service.onprem.api-server
8. Build the images with

   ```
   build-docker.sh --build
   ```

   The build will not continue if mashery-proxy-config.tmpl still has variables that are not updated.

9. Update the images tag in docker-compose.yml.
10. Restart the cluster with updated docker-compose.yml.

Migrating Tokens from 4.4.x to 5.5.1

Token Migration Service is provided to enable migration of existing tokens in MySQL database to Apache Cassandra.

The following section gives the procedure to migrate tokens from 4.4.x to 5.5.1

Procedure

1. Setup a service endpoint to run the token migration service.
2. Optionally, login to ml-tm container to run the token migration service.
3. Initiate the token migration.
4. Repeat or restart the token migratiion if needed.

Migration API

The API for migration is /migrateTokens

For example:

```
curl -v "http://{ml-tm}:8082/migrateTokens?public_key={public_key}&noWrite&before=2022-10-10%2000:00:00"
-XPOST
```

Parameter	Description
before	The to date of the migration. By default, it is current time.
after	The from date of the migration. By default, it is the last token date in database. If the after value is earlier than the before parameter value, then the after value is reset to the before.
stop	Stops the current migration.
resume	Continue from the from date of the last migration.
noWrite	process read operation from SQL and no write to NOSQL
public_key	the key set in com.mashery.service.onprem.oauth.authenticator.oauth-service-authenticator

What to do next

After 4.4.x cluster has the updated configuration and traffic manager is restarted, any new token created is available in 5.5.1 cluster. After completing the token migration, all active tokens are migrated to Apache Cassandra and available in 5.5.1 cluster.

Any new token created in 5.5.1 cluster is available in 4.4.x cluster. To ensure that the tokens are consistant in both 4.4.x cluster and 5.5.1 cluster, any update to the token, for example updating the scope or user context of a token should be done in from 5.5.1 cluster.

The 5.5.1 cluster is ready for accepting traffic after the token migration is done. To allow for a smooth trasition from 4.4.x to 5.5.1, you can use a load balancer to slowly increase the load from 4.4.x to 5.5.1 cluster.

Upgrading Local Edition Cluster

The following sections describe upgrade paths for the currently available versions

- Upgrading Local Edition Cluster from 5.0 to 5.4.1
- Upgrading Local Edition cluster from 5.2.0, 5.3.0, 5.3.1 or 5.4.1 to 5.4.0
- Upgrading Local Edition cluster from 5.3.1 to 5.4.1 for Kubernetes in Tethered Mode
- Upgrading Local Edition cluster from 5.4.0 to 5.4.1 for Docker Swarm
- Upgrading Local Edition cluster from 5.4.0 to 5.4.1 for Kubernetes in Untethered Mode
- Upgrading Local Edition cluster from 5.3.1, 5.4.1 to 5.5.0
- Upgrading Local Edition cluster from 5.4.1 to 5.5.0
- Upgrading Local Edition cluster from 5.5.0 to 5.5.1
- Upgrading Local Editioncluster from 5.5.1 to 5.5.2
- Upgrading Local Edition cluster from 5.5.2 to 5.6.0
- Upgrading Local Edition cluster from 5.6.0 to 5.6.1
- Upgrading Local Edition cluster from 5.5.2 to 5.6.2
- Upgrading Local Edition cluster from 5.6.1 to 5.6.2

Upgrading Local Edition cluster from 5.0 to 5.4.1

The Local Edition cluster can be upgraded from a version 5.0 to version 5.4.1. The upgrade is not an in place upgrade.

A new cluster for version 5.4.1 must be created and then the existing configuration data namely services, packages, plans and other such data can be migrated from version 5.0 cluster to a 5.4.1 cluster using the Configuration Migration Tool.

Prerequisites

- TheLocal Edition 5.4.1 cluster must be created.
- Keep the cluster of version 5.0 running until migration of version 5.4.1 is complete.

The Boomi Cloud™ API Management - Local EditionConfiguration Migration Tool is bundled as tml-config-migration.tar.gz, which can be located inside the tml-upgrade folder after extracting theLocal Edition installer distribution. Extracting tml-config-migration.tar.gz results in the following files:

- export.sh
- import.sh
- config-migration-jar-with-dependencies.jar

Running the Migration

1. Copy the export.sh file from your machine into the MySQL pod of the running Local Edition 5.0 cluster.

   If the Local Edition 5.0 cluster is in Kubernetes:

   a. Use the following command to copy the file to the MySQL pod:

   ```
   kubectl cp export.sh mysql-set-0-0:/home/builder/
   ```

   b. Use the Kubectlx Exec command to access the MySQL pod using the following command:

   ```
   kubectl exec -it mysql-set-0-0 /bin/bash
   ```

   c. Locate the export.sh file at the default location (home/builder/), then run the following command:

   ```
   ./export.sh
   ```

   This produces the export.tar.gz file, which contains all of the exported .csv files and a manifest file.

   If Local Edition 5.0 is in Docker Swarm:

   a. Get the container ID of the tmgc-sql container and copy export.sh to it at the

default location:

```
docker cp export.sh <sql-container-id>:/home/builder
```

b. Use the Docker Exec command to access the tml-sql container:

```
docker exec -it <sql-container-id> /bin/bash
```

c. Locate the export.sh file at the default location (home/builder/), then run the following command:

```
./export.sh
```

This produces the export.tar.gz file, which contains all of the exported .csv files and a manifest file.

2. Download the export.tar.gz file from the Kubernetes/Swarm pod to your machine.
   a. If in a Kubernetes environment, use the following command to copy the file on your machine's current working directory:

```
kubectl cp mysql-set-0-0:/home/builder/export.tar.gz .
```

   b. If in a Docker Swarm environment, use the following command to copy the file on your machine's current working directory:

```
docker cp <sql-container-id>:/home/builder/export.tar.gz .
```

3. Copy the export.tar.gz, import.sh, and config-migration-jar-with-dependencies.jar files from your machine to the new 5.4.1 MySQL pod.

> **Note:** The MySQL pod must be ACTIVE.
>
> For more information, see the *Validating the Local Edition Cluster* topic in the *Boomi Cloud™ API Management - Local Edition Installation and Configuration Guide*.
>
> Installation and Configuration Guide

If in a Kubernetes environment, use the following commands:

```
kubectl cp export.tar.gz mysql-set-0-0:/home/builder/
kubectl cp import.sh mysql-set-0-0:/home/builder/
kubectl cp config-migration-jar-with-dependencies.jar mysql-set-0-0:/home/builder/
```

If in a Docker Swarm environment, use the following commands:

```
docker cp export.tar.gz <sql-container-id>:/home/builder
docker cp import.sh <sql-container-id>:/home/builder
docker cp config-migration-jar-with-dependencies.jar <sql-container-id>:/home/builder
```

4. Copy your latest and updated data.zip file, which is also in the MySQL container, to the same location as the other files.

   If in a Kubernetes environment, use the following command:

   ```
   kubectl cp data.zip mysql-set-0-0:/home/builder/
   ```

   If in a Docker Swarm environment, use the following command:

   ```
   docker cp data.zip <sql-container-id>:/home/builder
   ```

   > **Note:** If both your source and destination environments are Kubernetes, ensure that you configure kubectl per both environments.

5. Import into Local Edition 5.4.1.
   a. Log into the MySQL pod. All of the copied files in this step should be present

at the default login location (/home/builder/ in this case).

If in a Kubernetes environment, use the following command:

```
kubectl exec -it mysql-set-0-0 /bin/bash
```

If in a Docker Swarm environment, use the following command:

```
docker exec -it <sql-container-id> /bin/bash
```

b. Run the following command:

```
./import.sh
```

6. When the import has successfully completed, verify that all the data has been migrated successfully using the 5.4.1 Local Edition Configuration Manager. All of the services, packages, plans, applications, and so on should appear in their appropriate sections.
7. Use the docker exec/kubectl exec command to access the cache pod and run the following command to load configuration data into cache so that traffic calls can be made without delay.

```
/opt/javaproxy/proxy/cacheloader --service --mapi --devclass --packager --httpsclientsecurity
--env production --verbose
```

Upgrading Local Edition cluster from 5.2.0, 5.3.0,5.3.1, or 5.4.0 to 5.4.1

The Local Edition cluster can be upgraded in place from a version 5.2.0, 5.3.0,5.3,5.3.1 or 5.4 to version 5.4.1.

Note: The following steps are applicable to Swarm environment in Untethered mode.

Before you begin

- Local Edition 5.4.1 images must be built and pushed to the desired registry for deployment.
- For customization done to the JSON properties files in previous versions, the deployment scripts of Local Edition 5.4.1 must include these previously customized values.
- An admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.4.1 by running the compose command.
- For Docker Swarm deployment, if host were pinned in older versions then the deployment scripts are pinned to the same hosts for version 5.4.1.
- Carry out the configurations as follows:
    - Swarm: Configure the Docker client for communication with the manager node.

Note:
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **Log** components.
    a. Remove running Local Edition 5.4.0, 5.3.1, 5.3.0 or 5.2.0 Log container. Run the following command.
        - For single-zone deployment:

          For Docker Swarm:

          ```
          docker stack rm logstack
          ```

    b. Deploy Local Edition 5.4.1 Log containers. Run the following command.
        - For single-zone deployment:

          For Docker Swarm:

```
./deploy-log-pod.sh
```

2. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.4.0, 5.3.1, 5.3.0 or 5.2.0 NoSQL container. Run the following command:
        - For single-zone deployment:

          For Docker Swarm:

          ```
          docker stack rm nosqlstack
          ```

    b. Deploy Local Edition 5.4.1 NoSQL containers. Run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

3. Undeploy and redeploy **SQL** components.
    a. Remove running Local Edition 5.4.0, 5.3.1, 5.3.0 or 5.2.0 SQL container. Run the following command.
        - For single-zone deployment:
            - For Docker Swarm:

              ```
              docker stack rm sqlstack
              ```

    b. Deploy Local Edition 5.4.1 SQL containers. Run the following command.
        - For single-zone deployment:

          For Docker Swarm:

          ```
          ./deploy-sql-pod.sh
          ```

4. Upgrade **NoSQL** and **SQL**container consecutively. Run the upgrade scripts inside the container.
    a. Upgrade **NoSQL** container.

    For single zone deployment:
        - For Docker Swarm:

```
docker exec -it <nosql_container_id> bash
```

b. Navigate to the scripts folder using cd /opt/mashery/scripts/ and run the following commands as applicable.

- Step not required when upgrading from 5.3.1.
- For upgrading from 5.3.0:

```
./nosql-upgrade.sh 5.3.0
```

- For upgrading from 5.2.0:

```
./nosql-upgrade.sh 5.2.0
```

> **Note:** Execute the above upgrade step only in one NoSQL container. Changes will be reflected across all NoSQL instances.

The NoSQL container will be upgraded to 5.4.1.

c. Upgrade SQL container

For single zone deployment:

- For Docker Swarm:

```
docker exec -it <sql_container_id> bash
```

d. Navigate to the directory with the Boomi CAM scripts, for example: cd /opt/mashery/scripts/. Then, run sql-upgrade.sh and pass 5.4.0, 5.3.1, 5.3.0 or 5.2.0 as parameter to script as applicable.

- For upgrading from 5.4.0:

```
./sql-upgrade.sh 5.4.0
```

- For upgrading from 5.3.1:

```
./sql-upgrade.sh 5.3.1
```

- For upgrading from 5.3.0:

```
./sql-upgrade.sh 5.3.0
```

- For upgrading from 5.2.0:

```
./sql-upgrade.sh 5.2.0
```

> **Note:** Execute the above upgrade step only in one NoSQL container. Changes will be reflected across all NoSQL instances.

The SQL container will be upgraded to 5.4.1.

5. Undeploy and redeploy **Cache** components.
   a. Remove running Local Edition 5.4.0, 5.3.1, 5.3.0 or 5.2.0 Cache container. Run the following command.
      - For single-zone deployment:

        For Docker Swarm:

```
docker stack rm cachestack
```

   b. Deploy Local Edition 5.4.1 Cache containers. Run the following command.
      - For single-zone deployment:

        For Docker Swarm:

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.
   a. Remove running Local Edition 5.4.0, 5.3.1, 5.3.0 or 5.2.0 TM container. Run the following command.
      - For single-zone deployment:

        For Docker Swarm:

```
docker stack rm tmstack
```

b. Deploy Local Edition 5.4.1 TM containers. Run the following command.

- For single-zone deployment:

  For Docker Swarm:

```
./deploy-tm-pod.sh
```

7. Undeploy and redeploy CM components.

a. Remove running Local Edition 5.4.0, 5.3.1, 5.3.0 or 5.2.0 CM container. Run the following command.

- For single-zone deployment:

  For Docker Swarm:

```
docker stack rm cmstack
```

b. Deploy Local Edition 5.4.1 CM containers. Run the following command.

- For single-zone deployment:

  For Docker Swarm:

```
./deploy-cm-pod.sh
```

> **Note:** The following database migration step is applicable only for upgrading from ML 5.3.1.or 5.3.0.

c. Run a database migration in the CM container as follows:

-
- For single-zone deployment:

  For Docker Swarm:

```
docker exec -it <cm_container_id> bash
```

- To migrate data, use the following command:

```
service mlconfig database-upgrade --migrate
```

After deployment, test TM calls within the CM container.

Upgrading Local Edition cluster from 5.3.1 to 5.4.1 for Kubernetes in Tethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.3.1 to 5.4.1 for Kubernetes.

Note: The following steps are applicable to Kubernetes environment in tethered mode and Local Edition 5.3.1 HF 2.

Before you begin

- Local Edition 5.4.1 images must be built and pushed to the desired registry for deployment.
- For customization done to the JSON properties files in previous versions, the deployment scripts of Local Edition 5.4.1 must include these previously customized values.
- An admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.4.1 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Upgrade **NoSQL** container. Run the upgrade scripts inside the container.
    a. Upgrade **NoSQL** container using the migrate_nosql_from_5.3.1_to_5.4.1.sh file. Contact Boomi support for this file.

b. Run the migration script.

```
chmod +x /root/migrate_nosql_from_5.3.1_to_5.4.1.sh
```

Note: Ensure that the NoSQL container is running.

c. Navigate to the NoSQL docker container and copy the migration script to the NoSQL container.

```
kubectl cp /root/migrate_nosql_from_5.3.1_to_5.4.1.sh NOSQL_CONTAINER_
ID:/usr/local/bin/migrate_nosql_from_5.3.1_to_5.4.1.sh
```

d. Run the migration script to update Cassandra configuration.

```
kubectl exec NOSQL_CONTAINER_ID /usr/local/bin/migrate_nosql_from_5.3.1_to_
5.4.1.sh
```

2. Undeploy and redeploy NoSQL container.

a. For single zone undeployment:

- ```
  ./undeploy-nosql-pod.sh
  ```

b. To redeploy the container, run the following command.

```
./deploy-nosql-pod.sh
```

The NoSQL container will be upgraded to 5.4.1.

3. Undeploy and redeploy Log components.

Remove running Local Edition 5.3.1 Log container.

a. For single zone undeployment:

- ```
  ./undeploy-log-pod.sh
  ```

b. To redeploy the container, run the following command.

```
./deploy-log-pod.sh
```

c. Run the following command after all the log pods have been successfully deployed.

```
kubectl exec LOG_CONTAINER_ID chown -R td-agent:td-agent /mnt/log/td-agent
```

4. Undeploy and redeploy CM components.

Remove running Local Edition 5.3.1 CM container.

a. For single zone undeployment:

- ```
  ./undeploy-cm-pod.sh
  ```

b. To redeploy the container, run the following command.

```
./deploy-cm-pod.sh
```

5. Undeploy and redeploy SQL components.

Remove running Local Edition 5.3.1 SQL container.

a. For single zone undeployment:

```
./undeploy-sql-pod.sh
```

b. Delete SQL persistent volumes, persistent volume claims, and storage classes of 5.3.1 cluster.

- Get list of SQL persistent volumes and delete them.

```
$ kubectl get pv | grep default/sqlvol
pvc-07612b5f-ba8d-49dd-b87a-45bb66822735   10Gi      RWO          Delete
   Bound    default/sqlvol-mysql-set-2-0    sql-storage-2          18h
pvc-45bd1d23-1cde-4069-acf2-6a84790960e6   10Gi      RWO          Delete
  Bound    default/sqlvol-mysql-set-1-0    sql-storage-1          18h
pvc-a450f23a-e3bc-4618-8a85-ba1c34cf2d54   10Gi      RWO          Delete
  Bound    default/sqlvol-mysql-set-0-0    sql-storage-0          18h
```
```

```
Delete these PV
$ kubectl delete pv pvc-07612b5f-ba8d-49dd-b87a-45bb66822735
$ kubectl delete pv pvc-45bd1d23-1cde-4069-acf2-6a84790960e6
$ kubectl delete pv pvc-a450f23a-e3bc-4618-8a85-ba1c34cf2d54
```

- Delete SQL persistent volume claims.

```
$ kubectl get pvc | grep mysql
sqlvol-mysql-set-0-0    Bound    pvc-4d2003fe-8b11-48f1-aa00-5604266feb02
10Gi      RWO          sql-storage-0    4d2h
sqlvol-mysql-set-1-0    Bound    pvc-9eef5029-de94-4fd6-9402-b84f56fcaebb
10Gi      RWO          sql-storage-1    4d2h
sqlvol-mysql-set-2-0    Bound    pvc-b0032295-5326-4b02-99b7-087cbffc63db
10Gi      RWO          sql-storage-2    4d2h
Delete these PVC
$ kubectl delete pvc sqlvol-mysql-set-0-0
$ kubectl delete pvc sqlvol-mysql-set-1-0
$ kubectl delete pvc sqlvol-mysql-set-2-0
```

- Delete storage classes.

```
$ kubectl delete sc sql-storage-0
$ kubectl delete sc sql-storage-1
$ kubectl delete sc sql-storage-2
```

Note: This is an optional step. Reporting pod was introduced in 5.4.0 release.

c. Create storage-classes-sql-reporting.yaml.

Refer storage-classes-X.yaml in the 5.4.1 manifest folder, here X is zone number.

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: sql-storage-0
provisioner: kubernetes.io/aws-ebs
parameters:
  type: gp2
  zones: us-east-1a
```

```
---
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: sql-storage-1
provisioner: kubernetes.io/aws-ebs
parameters:
  type: gp2
  zones: us-east-1c
---
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: sql-storage-2
provisioner: kubernetes.io/aws-ebs
parameters:
  type: gp2
  zones: us-east-1d
---
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: reporting-storage-0
provisioner: kubernetes.io/aws-ebs
parameters:
  type: gp2
  zones: us-east-1a
```

Create storage classes using the following command

```
kubectl create -f storage-classes-sql-reporting.yaml
```

d. To redeploy the container, run the following command.

```
./deploy-sql-pod.sh
```

6. Undeploy and redeploy Cache components.

Remove running Local Edition 5.3.1 Cache container.

a. For single zone undeployment:

- ./undeploy-cache-pod.sh

b. To redeploy the container, run the following command.

```
./deploy-cache-pod.sh
```

7. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.3.1 **TM** container.

a. For single zone undeployment:

- ./undeploy-tm-pod.sh

b. Remove tm-svc service from 5.3.1 folder, run the following command.

```
./delete-tm-svs.sh
```

c. Create tm-svc service in 5.4.1 folder, run the following command.

```
./set-tm-svs.sh
```

d. Deploy 5.4.1 **TM** container, run the following command.

```
./deploy-tm-pod.sh
```

Upgrading Local Edition Cluster from 5.4.0 to 5.4.1

The Local Edition cluster can be upgraded in place from 5.4 to version 5.4.1.

- Upgrading cluster from 5.4.0 to 5.4.1 for Docker Swarm
- Upgrading cluster from 5.4.0 to 5.4.1 for Kubernetes in Untethered Mode

Upgrading Local Edition cluster from 5.4.0 to 5.4.1 for Docker Swarm

The following section provides information on upgrading Local Edition cluster from version 5.4.0 to 5.4.1 for Docker swarm.
The following steps are applicable to swarm environment in untethered mode.

Before you begin

- Local Edition 5.4.1 images must be built and pushed to the desired registry for deployment.
- For customization done to the JSON properties files in previous versions, the deployment scripts of Local Edition 5.4.1 must include these previously customized values.
- An admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.4.1 by running the compose command.
- For Docker Swarm deployment, if host were pinned in older versions then the deployment scripts must be pinned to the same hosts for version 5.4.1.
- Carry out the configurations as follows:

  Swarm: Configure the Docker client for communication with the manager node.

Note:
- When redeploying container in the following procedure, execute the scripts from the 5.4.1 manifest deployment folder.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Upgrade **NoSQL** container. Run the upgrade scripts inside the container.
   a. Upgrade **NoSQL** container using the migrate_nosql_from_5.4.0_to_5.4.1.sh file. Contact Boomi support for this file.
   b. Make the migration script executable as follows:

```
chmod +x /root/migrate_nosql_from_5.4.0_to_5.4.1.sh
```

Note: Ensure that the NoSQL container is running.

c. Navigate to the NoSQL docker container and copy the migration script to the NoSQL container.

```
docker cp /root/migrate_nosql_from_5.4.0_to_5.4.1.sh NOSQL_CONTAINER_
ID:/usr/local/bin
```

d. Run the migration script to update Cassandra configuration.

```
docker exec NOSQL_CONTAINER_ID /usr/local/bin/migrate_nosql_from_5.4.0_to_
5.4.1.sh
```

2. Undeploy and redeploy NoSQL container.
   a. For single zone undeployment:
      - `docker stack rm nosqlstack`
   b. For more than one NoSQL components, run the following command.
      - `docker stack rm nosqlnonseedstack`
   c. To redeploy the container, run the following command.

```
./deploy-nosql-pod.sh
```

   The NoSQL container is upgraded to 5.4.1.
3. Undeploy and redeploy Log components.
   a. Remove running Local Edition 5.4.0 Log container.
      - `docker stack rm logstack`
   b. To redeploy the container, run the following command.

```
./deploy-log-pod.sh
```

c. Run the following command after all the log pods have been successfully deployed.

```
docker exec LOG_CONTAINER_ID chown -R td-agent:td-agent /mnt/log/td-agent
```

4. Undeploy and redeploy CM components.
   a. Remove running Local Edition 5.4.0 CM container.
      - `docker stack rm cmstack`
   b. To redeploy the container, run the following command.

```
./deploy-cm-pod.sh
```

5. Undeploy and redeploy SQL components.
   a. Remove running Local Edition 5.4.0 SQL container.
      - `docker stack rm sqlstack`
   b. To redeploy the container, run the following command.

```
./deploy-sql-pod.sh
```

   c. Run the following command after the SQL pods have been successfully deployed.

```
docker exec SQL_CONTAINER_ID chown -R mysql:mysql /var/lib/mysql
docker exec SQL_CONTAINER_ID chmod 0751 /var/lib/mysql
```

6. Upgrade SQL. Run the upgrade scripts inside the container.
   a. Upgrade SQL container.

      For single zone deployment:

- docker exec -it <*sql_container_id*> bash

b. Navigate to the scripts folder using cd /opt/mashery/scripts/ and run the following commands as applicable.

For upgrading from 5.4.0:

```
./sql-upgrade.sh 5.4.0
```

The SQL container is upgraded to 5.4.1.

7. Undeploy and redeploy **Cache** components.
   a. Remove running Local Edition 5.4.0 **Cache** container.
      - docker stack rm cachestack
   b. To redeploy the container, run the following command.

```
./deploy-cache-pod.sh
```

8. Undeploy and redeploy **TM** components.
   a. Remove running Local Edition 5.4.0 **TM** container.
      - docker stack rm tmstack
   b. To redeploy the container, run the following command.

```
./deploy-tm-pod.sh
```

Upgrading Local Edition cluster from 5.4.0 to 5.4.1 for Kubernetes in Untethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.4.0 to 5.4.1 for Kubernetes.

Note: The following steps are applicable to Kubernetes environment in untethered mode.

Before you begin

- Local Edition 5.4.1 images must be built and pushed to the desired registry for deployment.
- For customization done to the JSON properties files in previous versions, the deployment scripts of Local Edition 5.4.1 must include these previously customized values.
- An admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.4.1 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Upgrade NoSQL container. Run the upgrade scripts inside the container.
    a. Upgrade NoSQL container using the migrate_nosql_from_5.4.0_to_5.4.1.sh file. Contact Boomi support for this file.
    b. Run the migration script.

```
chmod +x /root/migrate_nosql_from_5.4.0_to_5.4.1.sh
```

Note: Ensure that the NoSQL container is running.

    c. Navigate to the NoSQL docker container and copy the migration script to the NoSQL container.

```
kubectl cp /root/migrate_nosql_from_5.4.0_to_5.4.1.sh NOSQL_CONTAINER_
ID:/usr/local/bin/migrate_nosql_from_5.4.0_to_5.4.1.sh
```

    d. Run the migration script to update Cassandra configuration.

```
kubectl exec NOSQL_CONTAINER_ID /usr/local/bin/migrate_nosql_from_5.4.0_to_
```

```
5.4.1.sh
```

2. Undeploy and redeploy **NoSQL** container.
   a. For single zone undeployment:
      - `./undeploy-nosql-pod.sh`
   b. To redeploy the container, run the following command.

      ```
      ./deploy-nosql-pod.sh
      ```

   The **NoSQL** container is upgraded to 5.4.1.
3. Undeploy and redeploy **Log** components.
   Remove running Local Edition 5.4.0 **Log** container.
   a. For single zone undeployment:
      - `./undeploy-log-pod.sh`
   b. To redeploy the container, run the following command.

      ```
      ./deploy-log-pod.sh
      ```

   c. Run the following command after all the **log** pods have been successfully deployed.

      ```
      kubectl exec LOG_CONTAINER_ID chown -R td-agent:td-agent /mnt/log/td-agent
      ```

4. Undeploy and redeploy **CM** components.
   Remove running Local Edition 5.4.0 **CM** container.
   a. For single zone undeployment:
      - `./undeploy-cm-pod.sh`
   b. To redeploy the container, run the following command.

```
./deploy-cm-pod.sh
```

5. Undeploy and redeploy SQL components.

   Remove running Local Edition 5.4.0 SQL container.

   a. For single zone undeployment:

      - ./undeploy-sql-pod.sh

   b. To redeploy the container, run the following command.

      ```
      ./deploy-sql-pod.sh
      ```

   c. On successful deployment of SQL pods, run the following command from another terminal.

      ```
      kubectl -it exec sql-set-0-0 bash
      chown -R mysql:mysql /var/lib/mysql
      chmod 0751 /var/lib/mysql
      ```

6. Upgrade SQL. Run the upgrade scripts inside the container.

   a. Upgrade SQL container.

      For single zone deployment:

      - kubectl exec -it <*sql_container_id*> bash

   b. Navigate to the scripts folder using cd /opt/mashery/scripts/ and run the following commands as applicable.

      For upgrading from 5.4.0:

      ```
      ./sql-upgrade.sh 5.4.0
      ```

   The SQL container will be upgraded to 5.4.1.

7. Undeploy and redeploy Cache components.

   Remove running Local Edition 5.4.0 Cache container.

   a. For single zone undeployment:

- ./undeploy-cache-pod.sh

b. To redeploy the container, run the following command.

```
./deploy-cache-pod.sh
```

8. Undeploy and redeploy TM components.
   Remove running Local Edition 5.4.0 TM container.
   a. For single zone undeployment:
      - ./undeploy-tm-pod.sh
   b. Remove tm-svc service from 5.4.0 folder, run the following command.

```
./delete-tm-svs.sh
```

   c. Create tm-svc service in 5.4.1 folder, run the following command.

```
./set-tm-svs.sh
```

   d. Deploy 5.4.1 TM container, run the following command.

```
./deploy-tm-pod.sh
```

Upgrading Local Edition Cluster from 5.4.1 to 5.5.0

The Local Edition cluster can be upgraded from 5.4.1 to version 5.5.0.

- Upgrading cluster from 5.4.1 to 5.5.0 for Kubernetes in Tethered Mode
- Upgrading cluster from 5.4.1 to 5.5.0 for Docker Swarm in Tethered Mode
- Upgrading cluster from 5.4.1 to 5.5.0 for Kubernetes in Untethered Mode
- Upgrading cluster from 5.4.1 to 5.5.0 for Docker Swarm in Untethered Mode

Upgrading Local Edition Cluster from 5.4.1 to 5.5.0 for Kubernetes in Tethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.4.1 to 5.5. for Kubernetes in Tethered mode.

Before you begin

- Local Edition 5.5.0 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-logtml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files in earlier version must be ported to 5.5.0.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.4.1 by running the compose command.

Note:

- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.4.1 NoSQL container. Run the following command.
        - For single-zone deployment:

```
./undeploy-nosql-pod.sh
```

    b. Deploy Local Edition 5.5.0 NoSQL containers. Run the following command.
        - For single-zone deployment:

```
./deploy-nosql-pod.sh
```

2. Undeploy and redeploy **Log** components.
    a. Remove running Local Edition 5.4.1 Log container. Run the following command.
        - For single-zone deployment:

```
./undeploy-log-pod.sh
```

    b. Deploy Local Edition 5.5.0 Log containers. Run the following command.
        - For single-zone deployment:

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy **CM** components.
    a. Remove running Local Edition 5.4.1 CM container. Run the following command.
        -

```
./undeploy-cm-pod.sh
```

    b. Deploy Local Edition 5.5.0 CM containers. Run the following command.
        -

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy **SQL** components.

    The MySQL Certificate/Identity data migration for tethered mode works for the following migration cases:

    - 5.3.1 to 5.4.1
    - 5.3.1 to 5.5.0
    - 5.4.1 to 5.5.0

    Please complete the Certificate/Identity data migration as soon as the new cluster is deployed, otherwise new certificates/identities created in the new6 /blop cluster will be overwritten by data migration.

a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
kubectl exec mysql-set-0-0 -- mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
```

b. Remove the SQL container from the Local Edition 5.4.1 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

Note: Delete SQL pvc after undeploying the SQL pod,

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?(@.metadata.labels.app=="mysql-svc")].metadata.name}')
```

c. Deploy Local Edition 5.5.0 SQL containers. Run the following command.

For single zone deployment,

```
./deploy-sql-pod.sh
```

5. Copy the MySQL data from Local Edition 5.4.0 cluster in the 5.5.0 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.5.0 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

Note: The mysql-set-0-0 is thetml-sql pod in the new cluster.

6. Import MySQL data dumped from the Local Edition 5.4.1 cluster to the Local Edition 5.5.0 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash531_data.sql
```

7. Run the SQL loader in sql pods in the new TML Cluster.

   Run the following command in eachtml-sql to run the sql loader.

   ```
   /opt/sqlloader/onpremloader --service --mapi --devclass --packager --httpsclientsecurity --
   env production --verbose
   ```

8. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.4.1 Cache container.

   a. For single zone undeployment:

      - ```
        ./undeploy-cache-pod.sh
        ```

   b. To redeploy the container, run the following command.

      ```
      ./deploy-cache-pod.sh
      ```

9. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.4.0 TM container.

   a. For single zone undeployment:

      - ```
        ./undeploy-tm-pod.sh
        ```

   b. Deploy TM container, run the following command.

      ```
      ./deploy-tm-pod.sh
      ```

   Note: This is applicable if reporting pod is deployed.

10. Undeploy and redeploy **Reporting** components.

    Remove running Local Edition 5.4.1 TM container.

    a. For single zone undeployment:

- ./undeploy-reporting-pod.sh

b. Deploy Local Edition 5.5.0 TM container and run the following command.

```
./deploy-reporting-pod.sh
```

Upgrading Local Edition cluster from 5.4.1 to 5.5.0 for Docker Swarm Tethered

The following section provides information on upgrading Local Edition cluster from version 5.4.1 to 5.5.0 for Docker swarm.
The following steps are applicable to swarm environment in tethered mode.

Before you begin

- Local Edition 5.4.1 images must be built and pushed to the desired registry for deployment.
- The customization done to the JSON properties files in earlier version must be ported to 5.5.0.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.4.1 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.4.1 NoSQL container. Run the following command.
        - For single-zone deployment:

```
./undeploy-nosql-pod.sh
```

b. Deploy Local Edition 5.5.0 NoSQL containers. Run the following command.

- For single-zone deployment:

```
./deploy-nosql-pod.sh
```

2. Undeploy and redeploy **Log** components.

a. Remove running Local Edition 5.4.1 Log container. Run the following command.

- For single-zone deployment:

```
./undeploy-log-pod.sh
```

b. Update the permissions for restricted user on Log volume. Run the following command on host machine where Log pod is installed.

```
chown 1001:1003 -R /var/lib/docker/volumes/logstack_log-1-vol/_data/data/tml-log/
```

c. Deploy Local Edition 5.5.0 Log containers. Run the following command.

- For single-zone deployment:

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy **CM** components.

a. Remove running Local Edition 5.4.1 CM container. Run the following command.

- For single-zone deployment:

```
./undeploy-cm-pod.sh
```

b. Deploy Local Edition 5.5.0 CM containers. Run the following command.

- For single-zone deployment:

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

   The MySQL Certificate/Identity data migration for tethered mode works for the following migration cases:

   - 5.3.1 to 5.4.1
   - 5.3.1 to 5.5.0
   - 5.4.1 to 5.5.0

   Please complete the Certificate/Identity data migration as soon as the new cluster is deployed, otherwise new certificates/identities created in the new6 /blop cluster will be overwritten by data migration.

   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

      ```
      docker exec -it <mysql pod> -- /bin/bash
      ```

   b. Run the following command to dump MySQL data. Update the MySQL password for the cluster.

      ```
      docker exec <sql-container-id> mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
      ```

   c. Copy the MySQL data file to the host.

      ```
      docker cp <mysql pod>:/tmp/mash_data.sql mash_data.sql
      ```

   d. Remove the SQL container from the Local Edition 5.4.1 cluster.

      For single zone deployment,

      ```
      ./undeploy-sql-pod.sh
      ```

> **Note:** Delete SQL pvc after undeploying the SQL pod,
>
> ```
> docker volume rm sqlstack_sql-1-vol
> ```

e. Deploy Local Edition 5.5.0 SQL containers. Run the following command.

For single zone deployment,

```
./deploy-sql-pod.sh
```

5. Copy the MySQL data from Local Edition 5.4.0 cluster in the Local Edition 5.5.0 cluster.

   Run the following command to copy MySQL data file to any tml-sql pod in TML 5.5.0 cluster

   ```
   docker cp mash_data.sql <mysql pod>:/tmp
   ```

   > **Note:** The "mysql-set-0-0" is thetml-sql pod in the new cluster.

6. Import MySQL data dumped from the Local Edition 5.4.1 cluster to the Local Edition 5.5.0 cluster.

   Run the following command in each tml-sql pod in the new cluster.

   ```
   mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
   mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
   mysql -u root -p'changeme' masherysolar < /tmp/mash531_data.sql
   ```

7. Run the SQL loader in sql pods in the new TML Cluster.

   Run the following command in each tml-sql to run the sql loader.

   ```
   /opt/sqlloader/onpremloader --service --mapi --devclass --packager --httpsclientsecurity --
   env production --verbose
   ```

8. Undeploy and redeploy Cache components.

   Remove running Local Edition 5.4.1 Cache container.

a. For single zone undeployment:

- ./undeploy-cache-pod.sh

b. To redeploy the container, run the following command.

```
./deploy-cache-pod.sh
```

9. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.4.0 TM container.

   a. For single zone undeployment:

   - ./undeploy-tm-pod.sh

   b. Deploy TM container, run the following command.

```
./deploy-tm-pod.sh
```

10. Undeploy and redeploy **Reporting** components.

    Remove running Local Edition 5.4.1 TM container.

    a. For single zone undeployment:

    - ./undeploy-reporting-pod.sh

    b. Update the permissions for restricted user on Reporting volume. Run the following command on Host Machine where Reporting pod installed.

```
chown 1001:1003 -R /var/lib/docker/volumes/reportingstack_reporting-1-vol/_
data/data/
```

    c. Deploy Local Edition 5.5.0 TM container, run the following command.

```
./deploy-reporting-pod.sh
```

Upgrading Local Edition Cluster from 5.4.1 to 5.5.0 for Kubernetes in Untethered mode

The following section provides information on upgrading Local Edition cluster from version 5.4.1 to 5.5. for Kubernetes in untethered mode.

Before you begin

- Local Edition 5.5.0 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-logtml-cacheare reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files in earlier version must be ported to 5.5.0.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.4.1 by running the compose command.

Note:

- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.4.1 NoSQL container. Run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.5.0 NoSQL containers. Run the following command.
        - For single-zone deployment:

```
./deploy-nosql-pod.sh
```

2. Undeploy and redeploy **Log** components.
    a. Remove running Local Edition 5.4.1 Log container. Run the following command.
        - For single-zone deployment:

```
./undeploy-log-pod.sh
```

    b. Deploy Local Edition 5.5.0 Log containers. Run the following command.
        - For single-zone deployment:

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy **CM** components.
    a. Remove running Local Edition 5.4.1 CM container. Run the following command.
        -

```
./undeploy-cm-pod.sh
```

    b. Deploy Local Edition 5.5.0 CM containers. Run the following command.
        -

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy **SQL** components.
    a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
docker exec -it <mysql pod> -- /bin/bash
```

    b. Run the following command to dump MySQL data. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
```

```
p'changeme' --ignore-table=masherysolar.area_config --ignore-
table=masherysolar.areas --ignore-table=masherysolar.package_key_audit_log --
ignore-table=masherysolar.application_audit_log --set-gtid-purged=OFF
masherysolar > /tmp/mash_data.sql
```

Note: In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.

- masherysolar.member_activity_log
- masherysolar.member_audit_log
- masherysolar.member_role_audit_log
- masherysolar.method_override_audit_log
- masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
kubectl cp mysql-set-0-0:/tmp/mash_data.sql mash_data.sql
```

d. Remove the SQL container from the Local Edition 5.4.1 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

Note: Delete SQL pvc after undeploying the SQL pod,

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

e. Deploy Local Edition 5.5.0 SQL containers. Run the following command.

For single zone deployment,

```
./deploy-sql-pod.sh
```

5. Copy the MySQL data from Local Edition 5.4.1 cluster in the Local Edition 5.5.0 cluster.

   Run the following command to copy MySQL data file to any tml-sql pod in TML 5.5.0 cluster

   ```
   kubectl cp mash_data.sql mysql-set-0-0:/tmp
   ```

6. Clear the MySQL data in the Local Edition 5.5.0 cluster.
   a. In the 5.5.0 cluster, login to the tml-sql pod where the MySQL data file was copied.

      ```
      kubectl exec -it mysql-set-0-0 -- /bin/bash
      ```

   b. Run the following command to clear data in MySQL.

      ```
      mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(area_
      config|areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u
      root -p'changeme' -e "truncate table $table" masherysolar; done
      ```

7. Import MySQL data dumped from the Local Edition 5.4.1 cluster to the Local Edition 5.5.0 cluster.

   Run the following command in each tml-sql pod in the new cluster.

   ```
   mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
   ```

8. Enable revert function.
   a. In the tml-sql pod run the following command.

      ```
      cd /opt/mashery/containeragent/resources/sql/scripts
      ./sql-upgrade.sh 5.4.1
      ```

   b. Connect to the masherysolar database in mysql and update the MySQL password of the cluster

```
./deploy-cache-pod.sh
```

```
SELECT id INTO @endpoint_id from `service_definition_endpoints` where epkey =
'fctgxsxh84vjpn8vr7zfh6kt';
```

```
UPDATE service_provider SET config = REPLACE(config, '"service_definition_
endpoint_id":171231' collate utf8_unicode_ci,
   concat('"service_definition_endpoint_id":', @endpoint_id) collate utf8_unicode_ci)
WHERE id = 6926;
```

9. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.4.1 Cache container.

   a. For single zone undeployment:

      - ```
        ./undeploy-cache-pod.sh
        ```

   b. To redeploy the container, run the following command.

      ```
      ./deploy-cache-pod.sh
      ```

10. Undeploy and redeploy **TM** components.

    Remove running Local Edition 5.4.0 TM container.

    a. For single zone undeployment:

       - ```
         ./undeploy-tm-pod.sh
         ```

    b. Deploy TM container, run the following command.

       ```
       ./deploy-tm-pod.sh
       ```

    Note: This is applicable if reporting pod is deployed.

11. Undeploy and redeploy **Reporting** components.

    Remove running Local Edition 5.4.1 TM container.

a. For single zone undeployment:

- ./undeploy-reporting-pod.sh

b. Deploy Local Edition 5.5.0 TM container, run the following command.

```
./deploy-reporting-pod.sh
```

Upgrading Local Edition cluster from 5.4.1 to 5.5.0 for Docker Swarm Untethered

The following section provides information on upgrading Local Edition cluster from version 5.4.1 to 5.5.0 for Docker swarm.
The following steps are applicable to swarm environment in tethered mode.

Before you begin

- Local Edition 5.5.0 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-logtml-cacheare reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files in earlier version must be ported to 5.5.0.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.4.1 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.4.1 NoSQL container. Run the following

command.

- For single-zone deployment:

```
./undeploy-nosql-pod.sh
```

b. Deploy Local Edition 5.5.0 NoSQL containers. Run the following command.

- For single-zone deployment:

```
./deploy-nosql-pod.sh
```

2. Undeploy and redeploy **Log** components.

a. Remove running Local Edition 5.4.1 Log container. Run the following command.

- For single-zone deployment:

```
./undeploy-log-pod.sh
```

b. Update the permissions for restricted user on Log volume. Run the following command on host machine where Log pod is installed.

```
chown 1001:1003 -R /var/lib/docker/volumes/logstack_log-1-vol/_data/data/tml-log/
```

c. Deploy Local Edition 5.5.0 Log containers. Run the following command.

- For single-zone deployment:

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy **CM** components.

a. Remove running Local Edition 5.4.1 CM container. Run the following command.

- For single-zone deployment:

```
./undeploy-cm-pod.sh
```

b. Deploy Local Edition 5.5.0 Log containers. Run the following command.

- For single-zone deployment:

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.
   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
docker exec -it <mysql pod> -- /bin/bash
```

   b. To dump MySQL data. Update the MySQL password for the cluster, run the following command .

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.area_config --ignore-
table=masherysolar.areas --ignore-table=masherysolar.package_key_audit_log --
ignore-table=masherysolar.application_audit_log --set-gtid-purged=OFF
masherysolar > /tmp/mash_data.sql
```

> **Note:** In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.
>
> - masherysolar.member_activity_log
> - masherysolar.member_audit_log
> - masherysolar.member_role_audit_log
> - masherysolar.method_override_audit_log
> - masherysolar.migration_log

   c. Copy the MySQL data file to the host.

```
docker cp <mysql pod>:/tmp/mash_data.sql mash_data.sql
```

   d. Remove the SQL container from the Local Edition 5.4.1 cluster.

   For single zone deployment,

```
./undeploy-sql-pod.sh
```

Note: Delete SQL pvc after undeploying the SQL pod,

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

e. To deploy Local Edition 5.5.0 SQL containers, run the following command.

For single zone deployment,

```
./deploy-sql-pod.sh
```

5. Copy the MySQL data from Local Edition 5.4.0 cluster in the Local Edition 5.5.0 cluster.

   To copy MySQL data file to any tml-sql pod in TML 5.5.0 cluster, run the following command:

```
docker cp mash_data.sql <mysql pod>:/tmp
```

6. Clear the MySQL data in the Local Edition 5.5.0 cluster.

   a. In the Local Edition 5.5.0 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
docker exec -it <mysql pod> /bin/bash
```

   b. To clear data in MySQL, run the following command:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(area_
config|areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u
root -p'changeme' -e "truncate table $table" masherysolar; done
```

7. Import MySQL data dumped from the Local Edition 5.4.1 cluster to the Local Edition 5.5.0 cluster.

   Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash531_data.sql
```

8. Enable revert function.
   a. In the tml-sql pod, run the following command.

      ```
      cd /opt/mashery/containeragent/resources/sql/scripts
      ./sql-upgrade.sh 5.4.1
      ```

   b. Connect to the masherysolar database in mysql and update the MySQL password of the cluster

      ```
      ./deploy-cache-pod.sh
      ```

      ```
      SELECT id INTO @endpoint_id from `service_definition_endpoints` where epkey =
      'fctgxsxh84vjpn8vr7zfh6kt';
      ```

      ```
      UPDATE service_provider SET config = REPLACE(config, '"service_definition_
      endpoint_id":171231' collate utf8_unicode_ci,
        concat('"service_definition_endpoint_id":', @endpoint_id) collate utf8_unicode_ci)
      WHERE id = 6926;
      ```

9. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.4.1 Cache container.
   a. For single zone undeployment:
      - ```
        ./undeploy-cache-pod.sh
        ```

   b. To redeploy the container, run the following command.

      ```
      ./deploy-cache-pod.sh
      ```

10. Undeploy and redeploy **TM** components.

    Remove running Local Edition 5.4.0 TM container.
    a. For single zone undeployment:
      ```

- ./undeploy-tm-pod.sh

b. To deploy TM container, run the following command:

```
./deploy-tm-pod.sh
```

Note: This is applicable if reporting pod is deployed.

11. Undeploy and redeploy **Reporting** components.

Remove running Local Edition 5.4.1 TM container.

a. For single zone undeployment:

- ./undeploy-reporting-pod.sh

b. To deploy Local Edition 5.5.0 TM container, run the following command:

```
./deploy-reporting-pod.sh
```

Upgrading Local Edition cluster from 5.5.0 to 5.5.1

The Local Edition cluster can be upgraded from 5.5.0 to version 5.5.1.

- Upgrading Local Edition cluster from 5.5.0 to 5.5.1 for Kubernetes in Tethered Mode
- Upgrading Local Edition cluster from 5.5.0 to 5.5.1 for Docker Swarm in Tethered Mode
- Upgrading Local Edition cluster from 5.5.0 to 5.5.1 for Kubernetes in Untethered Mode
- Upgrading Local Edition cluster from 5.5.0 to 5.5.1 for Docker Swarm in Untethered Mode

Upgrading Local Edition Cluster from 5.5.0 to 5.5.1 for Kubernetes in Tethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.5.0 to 5.5.1 for Kubernetes in Tethered mode.

Before you begin

- Local Edition 5.5.1 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.1.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.5.1 by running the compose command.

Note:

- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. To remove running Local Edition 5.5.0 NoSQL container, run the following command.

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:

a. To remove running Local Edition 5.5.0 Log container, run the following command.

```
./undeploy-log-pod.sh
```

b. To deploy Local Edition 5.5.1 Log containers, run the following command.

```
./deploy-log-pod.sh
```

3. To undeploy and redeploy CM components:

a. To remove running Local Edition 5.5.0 CM container, run the following command.

```
./undeploy-cm-pod.sh
```

b. To deploy Local Edition 5.5.0 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

The MySQL Certificate/Identity data migration for tethered mode works for the following migration case:

5.5.0 to 5.5.1

Complete the Certificate or Identity data migration as soon as the new cluster is deployed, otherwise new certificates or identities created in the new cluster is overwritten by data migration.

a. To dump MySQL data from the cluster created in the earlier version, login to any tml-sql pod running in the earlier cluster.

```
kubectl exec mysql-set-0-0 -- mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
```

i. To remove the SQL container from the 5.5.0 cluster for single zone deployment, run the following command:

```
./undeploy-sql-pod.sh
```

ii. Delete SQL pvc after undeploying the SQL pod:

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')
```

iii. To deploy Local Edition 5.5.1 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy the MySQL data fromLocal Edition 5.5.0 cluster to the Local Edition5.5.1 cluster.

To copy MySQL data file to any tml-sql pod in Local Edition 5.5.0 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

Note: The mysql-set-0-0 is thetml-sql pod in the new cluster.

c. Import MySQL data dumped from Local Edition 5.5.0 cluster to Local Edition 5.5.1 cluster. Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster. Run the following command in eachtml-sql to run the sql loader.

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.0 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To redeploy the container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.0 TM container.

   a. For single zone undeployment:

      - ./undeploy-tm-pod.sh

   b. Deploy Local Edition 5.5.1 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   a. Remove running Local Edition 5.5.0 TM container. For single zone undeployment, run the following command:

```
./undeploy-reporting-pod.sh
```

   b. To deploy Local Edition 5.5.1 TM container for single-zone deployment, run the following command.

```
source deploy-reporting-pod.sh "<zone-name>"
```

Upgrading Local Edition cluster from 5.5.0 to 5.5.1 for Docker Swarm Tethered

The following section provides information on upgrading Local Edition cluster from version 5.5.0 to 5.5.1 for Docker swarm.
The following steps are applicable to swarm environment in tethered mode.

Before you begin

- Local Edition 5.5.1 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade, you should maintain same number of tml-nosql, tml-log, and tml-cache pods configured as the previous release and all previous volumes are reused accordingly.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.1.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.5.1 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. To undeploy and redeploy **NoSQL** components:
    a. To remove running Local Edition 5.5.0 NoSQL container, run the following command for single-zone deployment:

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:
    a. To remove running Local Edition 5.5.0 Log container for single-zone deployment, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 Log containers for single-zone deployment, run the following command.

```
./deploy-log-pod.sh
```

3. To undeploy and redeploy CM components.
    a. To remove running Local Edition 5.5.0 CM container for single-zone deployment, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 CM containers for single-zone deployment, run the following command:

    ```
    ./deploy-cm-pod.sh
    ```

4. To undeploy and redeploy SQL components.

    The MySQL Certificate/Identity data migration for tethered mode works for the following migration cases:

    - 5.3.1 to 5.4.1
    - 5.3.1 to 5.5.1
    - 5.5.0 to 5.5.1

    Note: Complete the Certificate or Identity data migration as soon as the new Local Edition cluster is deployed, otherwise new certificates or identities created in the new Local Edition cluster is overwritten by data migration.

    a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

    ```
    docker exec <sql-container-id> mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
    ```

        i. To remove running Local Edition 5.5.0 SQL container for single-zone deployment, run the following command:

```
./undeploy-sql-pod.sh
```

ii. To delete SQL pvc after undeploying the SQL pod:

```
docker volume rm sqlstack_sql-1-vol
```

iii. To deploy Local Edition 5.5.1 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy MySQL Data dumped from the Old TML Cluster to the New TML Cluster

Copy MySQL data file to each tml-sql pod in the new TML cluster:

```
docker cp mash_data.sql <sql-container-id>:/tmp
```

Note: The mysql-set-0-0 is the tml-sql pod in the new cluster.

c. Import MySQL Data dumped from the Old TML Cluster to the new TML Cluster

Run the following command in each tml-sql pod in the new TML cluster:

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster

Run the following command in each tml-sql pod to run the sql loader:

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.0 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.5.1 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.0 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.5.1 TM container, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.5.0 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.5.1 TM container for single-zone deployment, run the following command.

   ```
   ./deploy-reporting-pod.sh
   ```

Upgrading Local Edition Cluster from 5.5.0 to 5.5.1 for Kubernetes in Untethered mode

The following section provides information on upgrading Local Edition cluster from version 5.5.0 to 5.5.1 for Kubernetes in untethered mode.

Before you begin

- Local Edition 5.5.1 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-logtml-cacheare reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.1.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.5.1 by running the compose command.

Note:

- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.5.0 NoSQL container,run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.5.1 NoSQL containers, run the following command.
        - For single-zone deployment:

        ```
        ./deploy-nosql-pod.sh
        ```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.5.0 Log container, run the following command.

- For single-zone deployment:

```
./undeploy-log-pod.sh
```

b. To deploy Local Edition 5.5.1 Log containers for single-zone deployment, run the following command.

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy CM components.
   a. To remove running Local Edition 5.5.0 CM container, run the following command.

```
./undeploy-cm-pod.sh
```

   b. To deploy Local Edition 5.5.1 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.
   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

   b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.area_config --ignore-
table=masherysolar.areas --ignore-table=masherysolar.package_key_audit_log --
ignore-table=masherysolar.application_audit_log --set-gtid-purged=OFF
masherysolar > /tmp/mash_data.sql
```

> **Note:** In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.
>
> - masherysolar.member_activity_log
> - masherysolar.member_audit_log
> - masherysolar.member_role_audit_log
> - masherysolar.method_override_audit_log
> - masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
kubectl cp mysql-set-0-0:/tmp/mash_data.sql mash_data.sql
```

i. Remove the SQL container from the Local Edition 5.5.0 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

ii. Delete SQL pvc after undeploying the SQL pod.

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

iii. Deploy Local Edition 5.5.1 SQL containers. Run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.5.0 cluster in the Local Edition 5.5.1 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.5.0 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

e. Clear the MySQL data in the Local Edition 5.5.0 cluster.

i. In the 5.5.0 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

ii. To clear data in MySQL:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(area_
config|areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do
mysql -u root -p'changeme' -e "truncate table $table" masherysolar; done
```

f. Import MySQL data dumped from the Local Edition 5.5.0 cluster to the Local Edition 5.5.1 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.0 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.5.1 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.5.0 TM container.

a. For single zone undeployment:

```
./undeploy-tm-pod.sh
```

b. To deploy Local Edition 5.5.1 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.5.0 TM container.

   a. For single zone undeployment:

```
./undeploy-reporting-pod.sh
```

   b. To deploy Local Edition 5.5.1 TM container, run the following command.

```
./deploy-reporting-pod.sh
```

Upgrading Local Edition cluster from 5.5.0 to 5.5.1 for Docker Swarm Untethered

The following section provides information on upgrading Local Edition cluster from version 5.5.0 to 5.5.1 for Docker swarm.
The following steps are applicable to swarm environment in untethered mode.

Before you begin

- Local Edition 5.5.1 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-sql, tml-log, and tml-cacheare reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.1. For example, tml_papi_properties.json in 5.5.0 has the same structure as 5.5.1, the customized tml_papi_properties.json in 5.5.0 can be copied to 5.5.1.
- Admin must generate the manifest deployment folder, namely the manifest-single-

zone, as per the deployment topology for version 5.5.1 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. To undeploy and redeploy **NoSQL** components:
    a. To remove running Local Edition 5.5.0 NoSQL container for single-zone deployment, run the following command.

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 NoSQL containers for single-zone deployment, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. Undeploy and redeploy **Log** components.
    a. Remove running Local Edition 5.5.0 Log container for single-zone deployment, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. Undeploy and redeploy **CM** components.
    a. To remove running Local Edition 5.5.0 CM container for single-zone deployment:

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 CM containers for single-zone deployment:

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.
    a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

    ```
    docker exec -it <mysql pod> -- /bin/bash
    ```

    b. To dump MySQL data. Update the MySQL password for the cluster, run the following command .

    ```
    mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' --ignore-table=masherysolar.area_config --ignore-table=masherysolar.areas --ignore-table=masherysolar.package_key_audit_log --ignore-table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar > /tmp/mash_data.sql
    ```

    Note: In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.

    - masherysolar.member_activity_log
    - masherysolar.member_audit_log
    - masherysolar.member_role_audit_log
    - masherysolar.method_override_audit_log
    - masherysolar.migration_log

    c. Copy the MySQL data file to the host.

    ```
    docker cp <mysql pod>:/tmp/mash_data.sql mash_data.sql
    ```

    d. Remove the SQL container from the Local Edition 5.5.0 cluster.

    For single zone deployment,

    ```
    ./undeploy-sql-pod.sh
    ```

Note: Delete SQL pvc after undeploying the SQL pod,

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

e. To deploy Local Edition 5.5.1 SQL containers, run the following command.

For single zone deployment,

```
./deploy-sql-pod.sh
```

5. Copy the MySQL data from Local Edition 5.5.0 cluster in the Local Edition 5.5.1 cluster.

   To copy MySQL data file to any tml-sql pod in TML 5.5.1 cluster, run the following command:

```
docker cp mash_data.sql <mysql pod>:/tmp
```

6. Clear the MySQL data in the Local Edition 5.5.1 cluster.
   a. In the Local Edition 5.5.1 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
docker exec -it <mysql pod> /bin/bash
```

   b. To clear data in MySQL, run the following command:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(area_
config|areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u
root -p'changeme' -e "truncate table $table" masherysolar; done
```

7. Clear the MySQL data in the Local Edition 5.5.1 cluster.
   a. In the Local Edition 5.5.1 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
docker exec -it <mysql pod> /bin/bash
```

   b. To clear data in MySQL, run the following command:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(area_
config|areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u
root -p'changeme' -e "truncate table $table" masherysolar; done
```

8. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.5.0 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To deploy the Local Edition 5.5.1 cache container for single-zone deployment:

   ```
   ./deploy-cache-pod.sh
   ```

9. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.0 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.5.1 TM container for single-zone deployment:

   ```
   ./deploy-tm-pod.sh
   ```

10. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

    Remove running Local Edition 5.5.0 TM container.

    a. For single-zone undeployment :

    ```
    ./undeploy-reporting-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 TM container:

    ```
    ./deploy-reporting-pod.sh
    ```

Upgrading Boomi Cloud API Management - Local Edition Cluster from 5.5.1 to 5.5.2

The Local Edition cluster can be upgraded from 5.5.1 to version 5.5.2.

- Upgrading Local Edition cluster from 5.5.1 to 5.5.2 for Kubernetes in Tethered Mode
- Upgrading Local Edition cluster from 5.5.1 to 5.5.2 for Docker Swarm in Tethered Mode
- Upgrading Local Edition cluster from 5.5.1 to 5.5.2 for Kubernetes in Untethered Mode
- Upgrading Local Edition cluster from 5.5.1 to 5.5.2 for Docker Swarm in Untethered Mode

Upgrading Local Edition Cluster from 5.5.1 to 5.5.2 for Kubernetes in Tethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.5.1 to 5.5.2 for Kubernetes in Tethered mode.

Before you begin

- Local Edition 5.5.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.2.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.5.2 by running the compose command.

Note:

- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. To remove running Local Edition 5.5.1 NoSQL container, run the following command.

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.5.2 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:
    a. To remove running Local Edition 5.5.1 Log container, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.5.2 Log containers, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. To undeploy and redeploy **CM** components:
    a. To remove running Local Edition 5.5.1 CM container, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.5.2 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

   The MySQL Certificate/Identity data migration for tethered mode works for the following migration case:

   5.5.1 to 5.5.2

   > **Note:** Complete the Certificate or Identity data migration as soon as the new cluster is deployed, otherwise new certificates or identities created in the new cluster is overwritten by data migration.

   a. To dump MySQL data from the cluster created in the earlier version, login to any tml-sql pod running in the earlier cluster.

   ```
   kubectl exec mysql-set-0-0 -- mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
   ```

      i. To remove the SQL container from the 5.5.1 cluster for single zone deployment, run the following command:

      ```
      ./undeploy-sql-pod.sh
      ```

      ii. Delete SQL pvc after undeploying the SQL pod:

      ```
      kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?(@.metadata.labels.app=="mysql-svc")].metadata.name}')
      ```

      iii. To deploy Local Edition 5.5.2 SQL containers for single zone deployment, run the following command.

      ```
      ./deploy-sql-pod.sh
      ```

   b. Copy the MySQL data fromLocal Edition 5.5.1 cluster to the Local Edition 5.5.2 cluster.

      To copy MySQL data file to any tml-sql pod in Local Edition 5.5.1 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

Note: The mysql-set-0-0 is thetml-sql pod in the new cluster.

c. Import MySQL data dumped from Local Edition 5.5.1 cluster to Local Edition 5.5.2 cluster. Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster. Run the following command in eachtml-sql to run the sql loader.

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.5.1 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To redeploy the 5.5.2 container, run the following command.

   ```
   ./deploy-cache-pod.sh
   ```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.1 TM container.

   a. For single zone undeployment:

   - ```
     ./undeploy-tm-pod.sh
     ```

   b. Deploy Local Edition 5.5.2 TM container, run the following command.
     ```

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.
    a. Remove running Local Edition 5.5.1 TM container. For single zone undeployment, run the following command:

    ```
    ./undeploy-reporting-pod.sh
    ```

    b. To deploy Local Edition 5.5.2 TM container for single-zone deployment, run the following command.

    ```
    source deploy-reporting-pod.sh "<zone-name>"
    ```

Upgrading Local Edition cluster from 5.5.1 to 5.5.2 for Docker Swarm Tethered

The following section provides information on upgrading Local Edition cluster from version 5.5.1 to 5.5.2 for Docker swarm.
The following steps are applicable to swarm environment in tethered mode.

Before you begin

- Local Edition 5.5.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade, you should maintain same number of tml-nosql, tml-log, and tml-cache pods configured as the previous release and all previous volumes are reused accordingly.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.2.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.5.2 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. To undeploy and redeploy **NoSQL** components:
    a. To remove running Local Edition 5.5.1 NoSQL container, run the following command for single-zone deployment:

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.52 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:
    a. To remove running Local Edition 5.5.1 Log container for single-zone deployment, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.5.2 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. To undeploy and redeploy **CM** components.
    a. To remove running Local Edition 5.5.1 CM container for single-zone deployment, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.5.2 CM containers for single-zone deployment, run the following command:

```
./deploy-cm-pod.sh
```

4. To undeploy and redeploy SQL components.

   The MySQL Certificate/Identity data migration for tethered mode works for the following migration cases:

   - 5.3.1 to 5.4.1
   - 5.3.1 to 5.5.1
   - 5.5.0 to 5.5.1
   - 5.5.1 to 5.5.2

   Note: Complete the Certificate or Identity data migration as soon as the new Local Edition cluster is deployed, otherwise new certificates or identities created in the new Local Edition cluster is overwritten by data migration.

   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

   ```
   docker exec <sql-container-id> mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
   ```

      i. To remove running Local Edition 5.5.1 SQL container for single-zone deployment, run the following command:

      ```
      ./undeploy-sql-pod.sh
      ```

      ii. To delete SQL pvc after undeploying the SQL pod:

      ```
      docker volume rm sqlstack_sql-1-vol
      ```

      iii. To deploy Local Edition 5.5.2 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy MySQL Data dumped from the Old TML Cluster to the New TML Cluster

Copy MySQL data file to each tml-sql pod in the new TML cluster:

```
docker cp mash_data.sql <sql-container-id>:/tmp
```

Note: The mysql-set-0-0 is the tml-sql pod in the new cluster.

c. Import MySQL Data dumped from the Old TML Cluster to the new TML Cluster

Run the following command in each tml-sql pod in the new TML cluster:

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster

Run the following command in each tml-sql pod to run the sql loader:

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.1 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.5.2 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.1 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.5.2 TM container, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.5.1 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.5.2 TM container for single-zone deployment, run the following command.

   ```
   ./deploy-reporting-pod.sh
   ```

Upgrading Local Edition Cluster from 5.5.1 to 5.5.2 for Kubernetes in Untethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.5.1 to 5.5.2 for Kubernetes in untethered mode.

Before you begin

- Local Edition 5.5.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.2.

- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.5.2 by running the compose command.

Note:
- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.5.1 NoSQL container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.5.2 NoSQL containers, run the following command.
        - For single-zone deployment:

        ```
        ./deploy-nosql-pod.sh
        ```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.5.1 Log container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-log-pod.sh
        ```

    b. To deploy Local Edition 5.5.2 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. Undeploy and redeploy **CM** components.

a. To remove running Local Edition 5.5.1 CM container, run the following command.

```
./undeploy-cm-pod.sh
```

b. To deploy Local Edition 5.5.1 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.areas --ignore-
table=masherysolar.package_key_audit_log --ignore-
table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar >
/tmp/mash_data.sql
```

Note: In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.

- masherysolar.member_activity_log
- masherysolar.member_audit_log
- masherysolar.member_role_audit_log
- masherysolar.method_override_audit_log
- masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
kubectl cp mysql-set-0-0:/tmp/mash_data.sql mash_data.sql
```

i. Remove the SQL container from the Local Edition 5.5.1 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

ii. Delete SQL pvc after undeploying the SQL pod.

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

iii. Deploy Local Edition 5.5.2 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.5.1 cluster in the Local Edition 5.5.2 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.5.2 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

e. Clear the MySQL data in the Local Edition 5.5.2 cluster.

i. In the 5.5.2 cluster, login to the tml-sql pod where the MySQL data file was copied.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

ii. To clear data in MySQL:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b
(areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u
```

```
root -p'changeme' -e "truncate table $table" masherysolar; done
```

f. Import MySQL data dumped from the Local Edition 5.5.1 cluster to the Local Edition 5.5.2 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.1 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.5.2 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.5.1 TM container.

a. For single zone undeployment:

```
./undeploy-tm-pod.sh
```

b. To deploy Local Edition 5.5.2 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

Remove running Local Edition 5.5.1 TM container.

a. For single zone undeployment:

```
./undeploy-reporting-pod.sh
```

b. To deploy Local Edition 5.5.2 TM container, run the following command.

```
./deploy-reporting-pod.sh
```

Upgrading Local Edition Cluster from 5.5.1 to 5.5.2 for Docker Swarm Untethered

The following section provides information on upgrading Local Edition cluster from version 5.5.1 to 5.5.2 for Docker swarm.

The following steps are applicable to swarm environment in untethered mode.

Before you begin

- Local Edition 5.5.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-sql, tml-log, and tml-cacheare reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.2. For example, tml_papi_properties.json in 5.5.1 has the same structure as 5.5.1, the customized tml_papi_properties.json in 5.5.1 can be copied to 5.5.2.
- Admin must generate the manifest deployment folder, namely the manifest-single-zone, as per the deployment topology for version 5.5.1 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. To undeploy and redeploy **NoSQL** components:

a. To remove running Local Edition 5.5.1 NoSQL container for single-zone deployment, run the following command.

```
./undeploy-nosql-pod.sh
```

b. To deploy Local Edition 5.5.2 NoSQL containers for single-zone deployment, run the following command.

```
./deploy-nosql-pod.sh
```

2. Undeploy and redeploy **Log** components.
   a. Remove running Local Edition 5.5.1 Log container for single-zone deployment, run the following command.

```
./undeploy-log-pod.sh
```

   b. To deploy Local Edition 5.5.1 Log containers for single-zone deployment, run the following command.

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy **CM** components.
   a. To remove running Local Edition 5.5.1 CM container for single-zone deployment:

```
./undeploy-cm-pod.sh
```

   b. To deploy Local Edition 5.5.2 CM containers for single-zone deployment:

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy **SQL** components.
   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
docker exec -it <mysql pod> -- /bin/bash
```

b. To dump MySQL data. Update the MySQL password for the cluster, run the following command .

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.areas --ignore-
table=masherysolar.package_key_audit_log --ignore-
table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar >
/tmp/mash_data.sql
```

Note: In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.

- masherysolar.member_activity_log
- masherysolar.member_audit_log
- masherysolar.member_role_audit_log
- masherysolar.method_override_audit_log
- masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
docker cp <mysql pod>:/tmp/mash_data.sql mash_data.sql
```

d. Remove the SQL container from the Local Edition 5.5.1 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

Note: Delete SQL volume after undeploying the SQL pod,

```
docker volume rm  sqlstack_sql-1-vol
```

e. To deploy Local Edition 5.5.2 SQL containers, run the following command.

For single zone deployment,

```
./deploy-sql-pod.sh
```

5. Copy the MySQL data from Local Edition 5.5.1 cluster in the Local Edition 5.5.2 cluster.

   To copy MySQL data file to any tml-sql pod in TML 5.5.2 cluster, run the following command:

```
docker cp mash_data.sql <mysql pod>:/tmp
```

6. Clear the MySQL data in the Local Edition 5.5.2 cluster.

   a. In the Local Edition 5.5.2 cluster, login to the tml-sql pod where the MySQL data file was copied.

```
docker exec -it <mysql pod> /bin/bash
```

   b. To clear data in MySQL, run the following command:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u root -p'changeme' -e "truncate table $table" masherysolar; done
```

7. Import the MySQL data in the Local Edition 5.5.2 cluster.

   a. In the Local Edition 5.5.2 cluster, in the tml-sql pod where the MySQL data is copied, run the following command to import MySQL data dumped.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

8. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.5.1 Cache container.

   a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.5.2 cache container for single-zone deployment:

```
./deploy-cache-pod.sh
```

9. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.1 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.5.2 TM container for single-zone deployment:

   ```
   ./deploy-tm-pod.sh
   ```

10. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

    Remove running Local Edition 5.5.1 TM container.

    a. For single-zone undeployment :

    ```
    ./undeploy-reporting-pod.sh
    ```

    b. To deploy Local Edition 5.5.1 TM container for single-zone deployment:

    ```
    ./deploy-reporting-pod.sh
    ```

Upgrading Boomi Cloud API Management - Local Edition Cluster from 5.5.2 to 5.6.0

The Local Edition cluster can be upgraded from 5.5.2 to version 5.6.0.

- Upgrading Local Edition cluster from 5.5.2 to 5.6.0 for Kubernetes in Tethered Mode
- Upgrading Local Edition cluster from 5.5.2 to 5.6.0 for Docker Swarm in Tethered Mode

- Upgrading Local Edition cluster from 5.5.2 to 5.6.0 for Kubernetes in Untethered Mode
- Upgrading Local Edition cluster from 5.5.2 to 5.6.0 for Docker Swarm in Untethered Mode

Upgarding Local Edition Cluster from 5.5.2 to 5.6.0 for Kubernetes in Tethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.5.2 to 5.5.6 for Kubernetes in Tethered mode.

Before you begin

- Local Edition 5.6.0 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.0.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.0 by running the compose command.

Note:
- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. To remove running Local Edition 5.5.2 NoSQL container, run the following command.

```
./undeploy-nosql-pod.sh
```

b. To deploy Local Edition 5.6.0 NoSQL containers, run the following command.

```
./deploy-nosql-pod.sh
```

2. To undeploy and redeploy **Log** components:
   a. To remove running Local Edition 5.5.2 Log container, run the following command.

```
./undeploy-log-pod.sh
```

   b. To deploy Local Edition 5.6.0 Log containers, run the following command.

```
./deploy-log-pod.sh
```

3. To undeploy and redeploy **CM** components:
   a. To remove running Local Edition 5.5.2 CM container, run the following command.

```
./undeploy-cm-pod.sh
```

   b. To deploy Local Edition 5.6.0 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy **SQL** components.

   The MySQL Certificate/Identity data migration for tethered mode works for the following migration case:

   5.5.2 to 5.6.0

   Note: Complete the Certificate or Identity data migration as soon as the new cluster is deployed, otherwise new certificates or identities created in the new cluster is overwritten by data migration.

   a. To dump MySQL data from the cluster created in the earlier version, login to

any tml-sql pod running in the earlier cluster.

```
kubectl exec mysql-set-0-0 -- mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
```

i.
```
./undeploy-sql-pod.sh
```

ii. Delete SQL pvc after undeploying the SQL pod:

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?(@.metadata.labels.app=="mysql-svc")].metadata.name}')
```

iii. To deploy Local Edition 5.6.0 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy the MySQL data fromLocal Edition 5.5.2 cluster to the Local Edition 5.6.0 cluster.

To copy MySQL data file to any tml-sql pod in Local Edition 5.5.2 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

Note: The mysql-set-0-0 is thetml-sql pod in the new cluster.

c. Import MySQL data dumped from Local Edition 5.5.2 cluster to Local Edition 5.6.0 cluster. Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster. Run the following command in eachtml-sql to run the sql loader.

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.5.2 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To redeploy the 5.6.0 container, run the following command.

   ```
   ./deploy-cache-pod.sh
   ```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   - ```
     ./undeploy-tm-pod.sh
     ```

   b. Deploy Local Edition 5.6.0 TM container for single-zone deployment, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   a. Remove running Local Edition 5.5.2 TM container. For single zone undeployment, run the following command:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.0 TM container for single-zone deployment, run the following command.

   ```
   source deploy-reporting-pod.sh "<zone-name>"
   ```
   ```

Upgrading Local Edition Cluster from 5.5.2 to 5.6.0 for Docker Swarm Tethered

The following section provides information on upgrading Local Edition cluster from version 5.5.2 to 5.6.0 for Docker swarm.

The following steps are applicable to swarm environment in tethered mode.

Before you begin

- Local Edition 5.6.0 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade, you should maintain same number of tml-nosql, tml-log, and tml-cache pods configured as the previous release and all previous volumes are reused accordingly.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.0.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.0 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. To undeploy and redeploy **NoSQL** components:
    a. To remove running Local Edition 5.5.2 NoSQL container, run the following command for single-zone deployment:

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.6.0 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:

a. To remove running Local Edition 5.5.2 Log container for single-zone deployment, run the following command.

```
./undeploy-log-pod.sh
```

b. To deploy Local Edition 5.6.0 Log containers for single-zone deployment, run the following command.

```
./deploy-log-pod.sh
```

3. To undeploy and redeploy CM components.

a. To remove running Local Edition 5.5.2 CM container for single-zone deployment, run the following command.

```
./undeploy-cm-pod.sh
```

b. To deploy Local Edition 5.6.0 CM containers for single-zone deployment, run the following command:

```
./deploy-cm-pod.sh
```

4. To undeploy and redeploy SQL components.

The MySQL Certificate/Identity data migration for tethered mode works for the following migration cases:

- 5.3.1 to 5.4.1
- 5.3.1 to 5.5.1
- 5.5.0 to 5.5.1
- 5.5.1 to 5.5.2
- 5.5.2 to 5.6.0

Note: Complete the Certificate or Identity data migration as soon as the new Local Edition cluster is deployed, otherwise new certificates or identities created in the new Local Edition cluster is overwritten by data migration.

a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
docker exec <sql-container-id> mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
```

   i. To remove running Local Edition 5.5.2 SQL container for single-zone deployment, run the following command:

```
./undeploy-sql-pod.sh
```

   ii. To delete SQL pvc after undeploying the SQL pod:

```
docker volume rm sqlstack_sql-1-vol
```

   iii. To deploy Local Edition 5.6.0 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy MySQL Data dumped from the Old TML Cluster to the New TML Cluster

   Copy MySQL data file to each tml-sql pod in the new TML cluster:

```
docker cp mash_data.sql <sql-container-id>:/tmp
```

   Note: The mysql-set-0-0 is the tml-sql pod in the new cluster.

c. Import MySQL Data dumped from the Old TML Cluster to the new TML Cluster

   Run the following command in each tml-sql pod in the new TML cluster:

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
```

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster

Run the following command in each tml-sql pod to run the sql loader:

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.2 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.0 Cache container for single-zone, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.5.2 TM container.

a. For single zone undeployment:

```
./undeploy-tm-pod.sh
```

b. To deploy Local Edition 5.6.0 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Undeploy and redeploy **Reporting** components.

Remove running Local Edition 5.5.2 TM container.

a. For single zone undeployment:

```
./undeploy-reporting-pod.sh
```

b. To deploy Local Edition 5.6.0 TM container for single-zone deployment, run the following command.

```
./deploy-reporting-pod.sh
```

Post upgrade, the reporting pod IP will change. After reconfigurations of reporting log forward with new IP, reporting Dashboard will show data again.

Upgrading Local Edition Cluster from 5.5.2 to 5.6.0 for Kubernetes in Untethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.5.2 to 5.6.0 for Kubernetes in untethered mode.

Before you begin

- Local Edition 5.6.0 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.0.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.0 by running the compose command.

Note:
- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.5.2 NoSQL container, run the following

command.

- For single-zone deployment:

```
./undeploy-nosql-pod.sh
```

b. Deploy Local Edition 5.6.0 NoSQL containers, run the following command.

- For single-zone deployment:

```
./deploy-nosql-pod.sh
```

2. Undeploy and redeploy **Log** components.
   a. To remove running Local Edition 5.5.2 Log container, run the following command.
      - For single-zone deployment:

```
./undeploy-log-pod.sh
```

   b. To deploy Local Edition 5.6.0 Log containers for single-zone deployment, run the following command.

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy **CM** components.
   a. To remove running Local Edition 5.5.2 CM container, run the following command.

```
./undeploy-cm-pod.sh
```

   b. To deploy Local Edition 5.6.0 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy **SQL** components.
   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.areas --ignore-
table=masherysolar.package_key_audit_log --ignore-
table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar >
/tmp/mash_data.sql
```

> **Note:** In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.
>
> - masherysolar.member_activity_log
> - masherysolar.member_audit_log
> - masherysolar.member_role_audit_log
> - masherysolar.method_override_audit_log
> - masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
kubectl cp mysql-set-0-0:/tmp/mash_data.sql mash_data.sql
```

i. Remove the SQL container from the Local Edition 5.5.2 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

ii. Delete SQL pvc after undeploying the SQL pod.

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

iii. Deploy Local Edition 5.6.0 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.5.2 cluster in the Local Edition 5.6.0 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.6.0 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

e. Clear the MySQL data in the Local Edition 5.6.0 cluster.

i. In the 5.6.0 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

ii. To clear data in MySQL:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u root -p'changeme' -e "truncate table $table" masherysolar; done
```

f. Import MySQL data dumped from the Local Edition 5.5.2 cluster to the Local Edition 5.6.0 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.2 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.0 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.6.0 TM container, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.0 TM container, run the following command.

   ```
   source deploy-reporting-pod.sh "<zone-name>"
   ```

Upgrading Local Edition Cluster from 5.5.2 to 5.6.0 Docker Swarm Untethered

The following section provides information on upgrading Local Edition cluster from version 5.5.2 to 5.6.0 for Docker swarm.

The following steps are applicable to swarm environment in untethered mode.

Before you begin

- Local Edition 5.6.0 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.0. For example, tml_papi_properties.json in 5.5.2 has the same structure as 5.6.0, the customized tml_papi_properties.json in 5.5.2 can be copied to 5.6.0.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.0 by running the compose command.

Note:
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.5.2 NoSQL container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.6.0 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.5.2 Log container, run the following command.
        - For single-zone deployment:

```
./undeploy-log-pod.sh
```

b. To deploy Local Edition 5.6.0 Log containers for single-zone deployment, run the following command.

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy **CM** components.

a. To remove running Local Edition 5.5.2 CM container for single-zone deployment, run the following command.

```
./undeploy-cm-pod.sh
```

b. To deploy Local Edition 5.6.0 CM containers for single-zone deployment, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy **SQL** components.

a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
docker exec -it <mysql pod> -- /bin/bash
```

b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.areas --ignore-
table=masherysolar.package_key_audit_log --ignore-
table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar >
/tmp/mash_data.sql
```

> **Note:** In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.
>
> - masherysolar.member_activity_log
> - masherysolar.member_audit_log
> - masherysolar.member_role_audit_log
> - masherysolar.method_override_audit_log
> - masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
docker cp <mysql pod>:/tmp/mash_data.sql mash_data.sql
```

i. Remove the SQL container from the Local Edition 5.5.2 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

> **Note:** Delete SQL pvc after undeploying the SQL pod. For example, docker volume rm sqlstack_sql-1-vol

ii. Deploy Local Edition 5.6.0 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.5.2 cluster in the Local Edition 5.6.0 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.6.0 cluster

```
docker cp mash_data.sql <mysql pod>:/tmp
```

e. Clear the MySQL data in the Local Edition 5.6.0 cluster.

i. In the 5.6.0 cluster, login to the tml-sql pod where the MySQL data file was copied.

```
docker exec -it <mysql pod> /bin/bash
```

ii. To clear data in MySQL:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u root -p'changeme' -e "truncate table $table" masherysolar; done
```

f. Import MySQL data dumped from the Local Edition 5.5.2 cluster to the Local Edition 5.6.0 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.2 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.0 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.5.2 TM container.

a. For single zone undeployment:

```
./undeploy-tm-pod.sh
```

b. To deploy Local Edition 5.6.0 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.0 TM container, run the following command.

   ```
   ./deploy-reporting-pod.sh
   ```

   Post upgrade, the reporting pod IP will change. After reconfigurations of reporting log forward with new IP, reporting Dashboard will show data again.

Upgarding Boomi Cloud API Management - Local Edition Cluster from 5.5.2 to 5.6.2

The Local Edition cluster can be upgraded from version 5.5.2 to 5.6.2.

- Upgrading Local Edition cluster from 5.5.2 to 5.6.2 for Kubernetes in Tethered Mode
- Upgrading Local Edition cluster from 5.5.2 to 5.6.2 for Docker Swarm in Tethered Mode
- Upgrading Local Edition cluster from 5.5.2 to 5.6.2 for Kubernetes in Untethered Mode
- Upgrading cluster from 5.5.2 to 5.6.2 for Docker Swarm in Untethered Mode

Upgrading Local Edition Cluster from 5.5.2 to 5.6.2 for Kubernetes in Tethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.5.2 to 5.6.2 for Kubernetes in Tethered mode.

Before you begin

- Local Edition 5.6.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.2.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.2 by running the compose command.

Note:
- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. To remove running Local Edition 5.5.2 NoSQL container, run the following command.

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:
    a. To remove running Local Edition 5.5.2 Log container, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 Log containers, run the following command.

```
./deploy-log-pod.sh
```

3. To undeploy and redeploy **CM** components:
    a. To remove running Local Edition 5.5.2 CM container, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 CM containers, run the following command.

    ```
    ./deploy-cm-pod.sh
    ```

4. Undeploy and redeploy **SQL** components.

    The MySQL Certificate/Identity data migration for tethered mode works for the following migration case:

    5.5.2 to 5.6.2

    > **Note:** Complete the Certificate or Identity data migration as soon as the new cluster is deployed, otherwise new certificates or identities created in the new cluster is overwritten by data migration.

    a. To dump MySQL data from the cluster created in the earlier version, login to any tml-sql pod running in the earlier cluster.

    ```
    kubectl exec mysql-set-0-0 -- mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > masherysolar-552.sqldmp
    ```

        i. To remove the SQL container from the 5.5.2 cluster for single zone deployment, run the following command:

        ```
        ./undeploy-sql-pod.sh
        ```

        ii. Delete SQL pvc after undeploying the SQL pod:

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')
```

iii. Remove existing tml-cluster secret.

```
kubectl delete secret cluster-property
```

iv. Create tml-cluster secret.

```
kubectl create secret generic cluster-property --from-file=<5.6.2-manifest-
folder/tml_cluster_properties.json>
```

v. To deploy Local Edition 5.6.2 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy the MySQL data from Local Edition 5.5.2 cluster to the Local Edition 5.6.2 cluster.

To copy MySQL data file to any tml-sql pod in Local Edition 5.6.2 cluster

```
kubectl cp masherysolar-552.sqldmp mysql-set-0-0:/tmp
```

Note: The mysql-set-0-0 is the tml-sql pod in the new cluster.

c. Import MySQL data dumped from Local Edition 5.5.2 cluster to Local Edition 5.6.2 cluster. Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/masherysolar-552.sqldmp
```

d. Run the SQL loader in sql pods in the new TML Cluster. Run the following command in each tml-sql to run the sql loader.

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.5.2 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To redeploy the 5.5.2 container, run the following command.

   ```
   ./deploy-cache-pod.sh
   ```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   - ```
     ./undeploy-tm-pod.sh
     ```

   b. Deploy Local Edition 5.6.2 TM container for single-zone deployment, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   a. Remove running Local Edition 5.5.2 TM container. For single zone undeployment, run the following command:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container for single-zone deployment, run the following command.

   ```
   source deploy-reporting-pod.sh "<zone-name>"
   ```
   ```

Upgrading Local Edition Cluster from 5.5.2 to 5.6.2 for Docker Swarm Tethered

The following section provides information on upgrading Local Edition cluster from version 5.5.2 to 5.6.2 for Docker swarm.

The following steps are applicable to swarm environment in tethered mode.

Before you begin

- Local Edition 5.6.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade, you should maintain same number of tml-nosql, tml-log, and tml-cache pods configured as the previous release and all previous volumes are reused accordingly.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.2.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.2 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. To undeploy and redeploy NoSQL components:
    a. To remove running Local Edition 5.5.2 NoSQL container, run the following command for single-zone deployment:

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:
    a. To remove running Local Edition 5.5.2 Log container for single-zone deployment, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. To undeploy and redeploy **CM** components.
    a. To remove running Local Edition 5.5.2 CM container for single-zone deployment, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 CM containers for single-zone deployment, run the following command:

    ```
    ./deploy-cm-pod.sh
    ```

4. To undeploy and redeploy **SQL** components.

    The MySQL Certificate/Identity data migration for tethered mode works for the following migration cases:

    - 5.3.1 to 5.4.1
    - 5.3.1 to 5.5.1
    - 5.5.0 to 5.5.1
    - 5.5.1 to 5.5.2
    - 5.5.2 to 5.6.0
    - 5.6.0 to 5.6.1
    - 5.5.2 to 5.6.2

Note: Complete the Certificate or Identity data migration as soon as the new Local Edition cluster is deployed, otherwise new certificates or identities created in the new Local Edition cluster is overwritten by data migration.

a. Dump MySQL Data from the cluster created in the earlier version. Log-in to any tml-sql pod running in the earlier cluster.

```
docker exec <sql-container-id> mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > masherysolar-552.sqldmp
```

i. To remove running Local Edition 5.5.2 SQL container for single-zone deployment, run the following command:

```
./undeploy-sql-pod.sh
```

ii. To delete SQL pvc after undeploying the SQL pod:

```
docker volume rm sqlstack_sql-1-vol
```

iii. Remove tml-cluster secret.

```
docker secret rm sqlstack_cluster-property-vol
```

iv. To deploy Local Edition 5.6.2 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy MySQL Data dumped from the Old TML Cluster to the New TML Cluster

Copy MySQL data file to each tml-sql pod in the new TML cluster:

```
docker cp masherysolar-552.sqldmp <sql-container-id>:/tmp
```

> **Note:** The mysql-set-0-0 is the tml-sql pod in the new cluster.

c. Import MySQL Data dumped from the Old TML Cluster to the new TML Cluster

Run the following command in each tml-sql pod in the new TML cluster:

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/masherysolar-552.sqldmp
```

d. Run the SQL loader in sql pods in the new TML Cluster

Run the following command in each tml-sql pod to run the sql loader:

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.2 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.2 Cache container for single-zone, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.5.2 TM container.

a. For single zone undeployment:

```
./undeploy-tm-pod.sh
```

b. To deploy Local Edition 5.6.2 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container for single-zone deployment, run the following command.

   ```
   ./deploy-reporting-pod.sh
   ```

   Post upgrade, the reporting pod IP will change. After reconfigurations of reporting log forward with new IP, reporting Dashboard will show data again.

Upgrading Local Edition Cluster from 5.5.2 to 5.6.2 for Kubernetes in Untethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.5.2 to 5.6.2 for Kubernetes in untethered mode.

Before you begin

- Local Edition 5.6.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.2.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.2 by running the compose command.

Note:

- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.5.2 NoSQL container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.6.2 NoSQL containers, run the following command.
        - For single-zone deployment:

        ```
        ./deploy-nosql-pod.sh
        ```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.5.2 Log container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-log-pod.sh
        ```

    b. To deploy Local Edition 5.6.2 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. Undeploy and redeploy **CM** components.
    a. To remove running Local Edition 5.5.2 CM container, run the following command.

```
./undeploy-cm-pod.sh
```

b. To deploy Local Edition 5.6.2 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump masherysolar --add-drop-database --user masheryonprem -pchangeme --complete-insert --create-options --single-transaction --ignore-table=masherysolar.areas --set-gtid-purged=OFF > masherysolar-552.sqldmp
```

Note: In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.

- masherysolar.member_activity_log
- masherysolar.member_audit_log
- masherysolar.member_role_audit_log
- masherysolar.method_override_audit_log
- masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
kubectl cp mysql-set-0-0:/tmp/masherysolar-552.sqldmp masherysolar-552.sqldmp
```

i. Remove the SQL container from the Local Edition 5.5.2 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

ii. Delete SQL pvc after undeploying the SQL pod.

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

iii. Remove existing tml-cluster secret.

```
kubectl delete secret cluster-property
```

iv. Create tml-cluster secret.

```
kubectl create secret generic cluster-property --from-file=<5.6.2-manifest-
folder/tml_cluster_properties.json>
```

v. Deploy Local Edition 5.6.2 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.5.2 cluster in the Local Edition 5.6.2 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.6.2 cluster

```
kubectl cp masherysolar-552.sqldmp mysql-set-0-0:/tmp
```

e. Clear the MySQL data in the Local Edition 5.6.2 cluster.

i. In the 5.6.2 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

f. Import MySQL data dumped from the Local Edition 5.6.0 cluster to the Local Edition 5.6.2 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u masheryonprem -p'changeme' masherysolar < /tmp/masherysolar-552.sqldmp
```

g. Run 5.6.0, 5.6.1, and 5.6.2 sql files present at SQL-POD in the /opt/mashery/containeragent/resources/sql/scripts/ folder.

```
mysql -umasheryonprem -pchangeme masherysolar < /opt/mashery/containeragent/resources/sql/scripts/5.6.0/masherysolar_5.6.0_ddl.sql
mysql -umasheryonprem -pchangeme masherysolar < /opt/mashery/containeragent/resources/sql/scripts/5.6.1/masherysolar_5.6.1_ddl.sql
mysql -umasheryonprem -pchangeme masherysolar < /opt/mashery/containeragent/resources/sql/scripts/5.6.2/masherysolar_5.6.2_ddl.sql
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.5.2 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.2 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.5.2 TM container.

a. For single zone undeployment:

```
./undeploy-tm-pod.sh
```

b. To deploy Local Edition 5.6.2 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container, run the following command.

   ```
   source deploy-reporting-pod.sh "<zone-name>"
   ```

Upgarding Local Edition Cluster from 5.5.2 to 5.6.2 Docker Swarm Untethered

The following section provides information on upgrading Local Edition cluster from version 5.5.2 to 5.6.2 for Docker swarm.

The following steps are applicable to swarm environment in untethered mode.

Before you begin

- Local Edition 5.6.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.5.2. For example, tml_papi_properties.json in 5.5.2 has the same structure as 5.6.2, the customized tml_papi_properties.json in 5.5.2 can be copied to 5.6.2.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.2 by running the compose command.

Note:
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.5.2 NoSQL container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.6.2 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.5.2 Log container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-log-pod.sh
        ```

    b. To deploy Local Edition 5.6.2 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. Undeploy and redeploy **CM** components.
    a. To remove running Local Edition 5.5.2 CM container for single-zone deployment, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

b. To deploy Local Edition 5.6.2 CM containers for single-zone deployment, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

a. Dump MySQL Data from the cluster created in the earlier version. Log-in to any tml-sql pod running in the earlier cluster.

```
docker exec -it <mysql pod> -- /bin/bash
```

b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump masherysolar --add-drop-database --user masheryonprem -pchangeme --complete-insert --create-options --single-transaction --ignore-table=masherysolar.areas --set-gtid-purged=OFF > /tmp/masherysolar-552.sqldmp
```

Note:

- masherysolar.member_activity_log
- masherysolar.member_audit_log
- masherysolar.member_role_audit_log
- masherysolar.method_override_audit_log
- masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
docker cp <mysql pod>:/tmp/masherysolar-552.sqldmp  masherysolar-552.sqldmp
```

i. Remove the SQL container from the Local Edition 5.5.2 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

ii. To delete SQL pvc after undeploying the SQL pod:

```
docker volume rm sqlstack_sql-1-vol
```

iii. Remove tml-cluster secret.

```
docker secret rm sqlstack_cluster-property-vol
```

iv. Deploy 5.6.2 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.5.2 cluster in the Local Edition 5.6.2 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.6.2 cluster

```
docker cp masherysolar-552.sqldmp <mysql pod>:/tmp
```

e. Clear the MySQL data in the Local Edition 5.6.2 cluster.

i. In the 5.6.2 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
docker exec -it <mysql pod> /bin/bash
```

f. Import MySQL data dumped from the Local Edition 5.5.2 cluster to the Local Edition 5.6.2 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u masheryonprem -p'changeme' masherysolar < /tmp/masherysolar-552.sqldmp
```

g. Run 5.6.0, 5.6.1, and 5.6.2 sql files present at SQL-POD in the /opt/mashery/containeragent/resources/sql/scripts/ folder.

```
mysql -umasheryonprem -pchangeme masherysolar <
/opt/mashery/containeragent/resources/sql/scripts/5.6.0/masherysolar_5.6.0_ddl.sql
mysql -umasheryonprem -pchangeme masherysolar <
/opt/mashery/containeragent/resources/sql/scripts/5.6.1/masherysolar_5.6.1_ddl.sql
mysql -umasheryonprem -pchangeme masherysolar <
/opt/mashery/containeragent/resources/sql/scripts/5.6.2/masherysolar_5.6.2_ddl.sql
```

5. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.5.2 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To deploy the Local Edition 5.6.2 Cache container, run the following command.

   ```
   ./deploy-cache-pod.sh
   ```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.5.2 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container, run the following command.

```
./deploy-reporting-pod.sh
```

Post upgrade, the reporting pod IP will change. After reconfigurations of reporting log forward with new IP, reporting Dashboard will show data again.

Upgrading Boomi Cloud API Management - Local Edition Cluster from 5.6.0 to 5.6.1

The Local Edition cluster can be upgraded from 5.6.0 to version 5.6.1.

- Upgrading Local Edition cluster from 5.6.0 to 5.6.1 for Kubernetes in Tethered Mode
- Upgrading Local Edition cluster from 5.6.0 to 5.6.1 for Docker Swarm in Tethered Mode
- Upgrading Local Edition cluster from 5.6.0 to 5.6.1 for Kubernetes in Untethered Mode
- Upgrading Local Edition cluster from 5.6.0 to 5.6.1 for Docker Swarm in Untethered Mode

Upgrading Local Edition Cluster from 5.6.0 to 5.6.1 for Kubernetes in Tethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.6.0 to 5.6.1 for Kubernetes in Tethered mode.

Before you begin

- Local Edition 5.6.1 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.1.

- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.1 by running the compose command.

Note:
- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. To remove running Local Edition 5.6.0 NoSQL container, run the following command.

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.6.1 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:
    a. To remove running Local Edition 5.6.0 Log container, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.6.1 Log containers, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. To undeploy and redeploy **CM** components:
    a. To remove running Local Edition 5.6.0 CM container, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

b. To deploy Local Edition 5.6.1 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

   The MySQL Certificate/Identity data migration for tethered mode works for the following migration case:

   5.6.0 to 5.6.1

   Note: Complete the Certificate or Identity data migration as soon as the new cluster is deployed, otherwise new certificates or identities created in the new cluster is overwritten by data migration.

   a. To dump MySQL data from the cluster created in the earlier version, login to any tml-sql pod running in the earlier cluster.

   ```
   kubectl exec mysql-set-0-0 -- mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
   ```

      i.
      ```
      ./undeploy-sql-pod.sh
      ```

      ii. Delete SQL pvc after undeploying the SQL pod:

      ```
      kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?(@.metadata.labels.app=="mysql-svc")].metadata.name}')
      ```

      iii. To deploy Local Edition 5.6.1 SQL containers for single zone deployment, run the following command.

      ```
      ./deploy-sql-pod.sh
      ```

   b. Copy the MySQL data from Local Edition 5.6.0 cluster to the Local Edition 5.6.1 cluster.

      To copy MySQL data file to any tml-sql pod in Local Edition 5.6.1 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

Note: The mysql-set-0-0 is thetml-sql pod in the new cluster.

c. Import MySQL data dumped from Local Edition 5.6.0 cluster to Local Edition 5.6.1 cluster. Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster. Run the following command in each tml-sql to run the sql loader.

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.6.0 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To redeploy the 5.6.0 container, run the following command.

   ```
   ./deploy-cache-pod.sh
   ```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.6.0 TM container.

   a. For single zone undeployment:

   - ```
     ./undeploy-tm-pod.sh
     ```

   b. Deploy Local Edition 5.6.1 TM container for single-zone deployment, run the following command.
     ```

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.
   a. Remove running Local Edition 5.6.0 TM container. For single zone undeployment, run the following command:

      ```
      ./undeploy-reporting-pod.sh
      ```

   b. To deploy Local Edition 5.6.1 TM container for single-zone deployment, run the following command.

      ```
      source deploy-reporting-pod.sh "<zone-name>"
      ```

Upgrading Local Edition Cluster from 5.6.0 to 5.6.1 for Docker Swarm Tethered

The following section provides information on upgrading Local Edition cluster from version 5.6.0 to 5.6.1 for Docker swarm.

The following steps are applicable to swarm environment in tethered mode.

Before you begin

- Local Edition 5.6.1 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade, you should maintain same number of tml-nosql, tml-log, and tml-cache pods configured as the previous release and all previous volumes are reused accordingly.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.1.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.1 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. To undeploy and redeploy NoSQL components:
    a. To remove running Local Edition 5.6.0 NoSQL container, run the following command for single-zone deployment:

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.6.1 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy Log components:
    a. To remove running Local Edition 5.6.0 Log container for single-zone deployment, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.6.1 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. To undeploy and redeploy CM components.
    a. To remove running Local Edition 5.6.0 CM container for single-zone deployment, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.6.1 CM containers for single-zone deployment, run the following command:

```
./deploy-cm-pod.sh
```

4. To undeploy and redeploy SQL components.

   The MySQL Certificate/Identity data migration for tethered mode works for the following migration cases:

   - 5.3.1 to 5.4.1
   - 5.3.1 to 5.5.1
   - 5.5.0 to 5.5.1
   - 5.5.1 to 5.5.2
   - 5.5.2 to 5.6.0
   - 5.6.0 to 5.6.1

   Note: Complete the Certificate or Identity data migration as soon as the new Local Edition cluster is deployed, otherwise new certificates or identities created in the new Local Edition cluster is overwritten by data migration.

   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

   ```
   docker exec <sql-container-id> mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
   ```

      i. To remove running Local Edition 5.6.0 SQL container for single-zone deployment, run the following command:

      ```
      ./undeploy-sql-pod.sh
      ```

      ii. To delete SQL pvc after undeploying the SQL pod:

      ```
      docker volume rm sqlstack_sql-1-vol
      ```

iii. To deploy Local Edition 5.6.1 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy MySQL Data dumped from the Old TML Cluster to the New TML Cluster

Copy MySQL data file to each tml-sql pod in the new TML cluster:

```
docker cp mash_data.sql <sql-container-id>:/tmp
```

Note: The mysql-set-0-0 is the tml-sql pod in the new cluster.

c. Import MySQL Data dumped from the Old TML Cluster to the new TML Cluster

Run the following command in each tml-sql pod in the new TML cluster:

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster

Run the following command in each tml-sql pod to run the sql loader:

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy Cache components.

Remove running Local Edition 5.6.0 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.1 Cache container for single-zone, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.6.0 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.6.1 TM container, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.6.0 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.1 TM container for single-zone deployment, run the following command.

   ```
   ./deploy-reporting-pod.sh
   ```

   Post upgrade, the reporting pod IP will change. After reconfigurations of reporting log forward with new IP, reporting Dashboard will show data again.

Upgarding Local Edition Cluster from 5.6.0 to 5.6.1 for Kubernetes in Untethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.6.0 to 5.6.1 for Kubernetes in untethered mode.

Before you begin

- Local Edition 5.6.1 images must be built and pushed to the desired registry for

deployment.

- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.1.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.1 by running the compose command.

Note:
- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.6.0 NoSQL container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.6.1 NoSQL containers, run the following command.
        - For single-zone deployment:

        ```
        ./deploy-nosql-pod.sh
        ```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.6.0 Log container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-log-pod.sh
        ```

b. To deploy Local Edition 5.6.1 Log containers for single-zone deployment, run the following command.

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy CM components.
   a. To remove running Local Edition 5.6.0 CM container, run the following command.

```
./undeploy-cm-pod.sh
```

   b. To deploy Local Edition 5.6.1 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.
   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

   b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.areas --ignore-
table=masherysolar.package_key_audit_log --ignore-
table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar >
/tmp/mash_data.sql
```

> **Note:** In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.
>
> - masherysolar.member_activity_log
> - masherysolar.member_audit_log
> - masherysolar.member_role_audit_log
> - masherysolar.method_override_audit_log
> - masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
kubectl cp mysql-set-0-0:/tmp/mash_data.sql mash_data.sql
```

i. Remove the SQL container from the Local Edition 5.6.0 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

ii. Delete SQL pvc after undeploying the SQL pod.

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

iii. Deploy Local Edition 5.6.1 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.6.0 cluster in the Local Edition 5.6.1 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.6.1 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

e. Clear the MySQL data in the Local Edition 5.6.1 cluster.

i. In the 5.6.1 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

ii. To clear data in MySQL:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u root -p'changeme' -e "truncate table $table" masherysolar; done
```

f. Import MySQL data dumped from the Local Edition 5.6.0 cluster to the Local Edition 5.6.1 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.6.0 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.1 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.6.0 TM container.

a. For single zone undeployment:

```
./undeploy-tm-pod.sh
```

b. To deploy Local Edition 5.6.1 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.6.0 TM container.

   a. For single zone undeployment:

```
./undeploy-reporting-pod.sh
```

   b. To deploy Local Edition 5.6.1 TM container, run the following command.

```
source deploy-reporting-pod.sh "<zone-name>"
```

Upgrading Local Edition Cluster from 5.6.0 to 5.6.1 Docker Swarm Untethered

The following section provides information on upgrading Local Edition cluster from version 5.6.0 to 5.6.1 for Docker swarm.

The following steps are applicable to swarm environment in untethered mode.

Before you begin

- Local Edition 5.6.1 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.0. For example, tml_papi_properties.json in 5.6.0 has the same structure as 5.6.1, the customized tml_papi_properties.json in 5.5.2 can be copied to 5.6.1.

- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.1 by running the compose command.

Note:
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.6.0 NoSQL container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.6.1 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.6.0 Log container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-log-pod.sh
        ```

    b. To deploy Local Edition 5.6.1 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. Undeploy and redeploy **CM** components.
    a. To remove running Local Edition 5.6.0 CM container for single-zone deployment, run the following command.

```
./undeploy-cm-pod.sh
```

b. To deploy Local Edition 5.6.1 CM containers for single-zone deployment, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
docker exec -it <mysql pod> -- /bin/bash
```

b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.areas --ignore-
table=masherysolar.package_key_audit_log --ignore-
table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar >
/tmp/mash_data.sql
```

Note: In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.

- masherysolar.member_activity_log
- masherysolar.member_audit_log
- masherysolar.member_role_audit_log
- masherysolar.method_override_audit_log
- masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
docker cp <mysql pod>:/tmp/mash_data.sql mash_data.sql
```

i. Remove the SQL container from the Local Edition 5.5.2 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

> **Note:** Delete SQL pvc after undeploying the SQL pod. For example, docker volume rm sqlstack_sql-1-vol

ii. Deploy Local Edition 5.6.1 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.6.0 cluster in the Local Edition 5.6.1 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.6.1 cluster

```
docker cp mash_data.sql <mysql pod>:/tmp
```

e. Clear the MySQL data in the Local Edition 5.6.1 cluster.

i. In the 5.6.1 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
docker exec -it <mysql pod> /bin/bash
```

ii. To clear data in MySQL:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u root -p'changeme' -e "truncate table $table" masherysolar; done
```

f. Import MySQL data dumped from the Local Edition 5.6.0 cluster to the Local Edition 5.6.1 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

5. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.6.0 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To deploy the Local Edition 5.6.1 Cache container, run the following command.

   ```
   ./deploy-cache-pod.sh
   ```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.6.0 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.6.1 TM container, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.6.0 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.1 TM container, run the following command.

   ```
   ./deploy-reporting-pod.sh
   ```

   Post upgrade, the reporting pod IP will change. After reconfigurations of

reporting log forward with new IP, reporting Dashboard will show data again.

Upgrading Boomi Cloud API Management - Local Edition Cluster from 5.6.1 to 5.6.2

The Local Edition cluster can be upgraded from 5.6.1 to version 5.6.2.

- Upgrading Local Edition cluster from 5.6.1 to 5.6.2 for Kubernetes in Tethered Mode
- Upgrading Local Edition cluster from 5.6.1 to 5.6.2 for Docker Swarm in Tethered Mode
- Upgrading Local Edition cluster from 5.6.1 to 5.6.2 for Kubernetes in Untethered Mode
- Upgrading Local Edition cluster from 5.6.1 to 5.6.2 for Docker Swarm in Untethered Mode

Upgrading Local Edition Cluster from 5.6.1 to 5.6.2 for Kubernetes in Tethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.6.1 to 5.6.2 for Kubernetes in Tethered mode.

Before you begin

- Local Edition 5.6.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.2.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.2 by running the compose command.

Note:

- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. To remove running Local Edition 5.6.1 NoSQL container, run the following command.

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy **Log** components:
    a. To remove running Local Edition 5.6.1 Log container, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 Log containers, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. To undeploy and redeploy **CM** components:
    a. To remove running Local Edition 5.6.1 CM container, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

   The MySQL Certificate/Identity data migration for tethered mode works for the following migration case:

   5.6.1 to 5.6.2

   > **Note:** Complete the Certificate or Identity data migration as soon as the new cluster is deployed, otherwise new certificates or identities created in the new cluster is overwritten by data migration.

   a. To dump MySQL data from the cluster created in the earlier version, login to any tml-sql pod running in the earlier cluster.

   ```
   kubectl exec mysql-set-0-0 -- mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
   ```

      i.
      ```
      ./undeploy-sql-pod.sh
      ```

      ii. Delete SQL pvc after undeploying the SQL pod:

      ```
      kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?(@.metadata.labels.app=="mysql-svc")].metadata.name}')
      ```

      iii. To deploy Local Edition 5.6.2 SQL containers for single zone deployment, run the following command.

      ```
      ./deploy-sql-pod.sh
      ```

   b. Copy the MySQL data from Local Edition 5.6.1 cluster to the Local Edition 5.6.2 cluster.

      To copy MySQL data file to any tml-sql pod in Local Edition 5.6.2 cluster

      ```
      kubectl cp mash_data.sql mysql-set-0-0:/tmp
      ```

> **Note:** The mysql-set-0-0 is thetml-sql pod in the new cluster.

c. Import MySQL data dumped from Local Edition 5.6.1 cluster to Local Edition 5.6.2 cluster. Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster. Run the following command in each tml-sql to run the sql loader.

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.6.1 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To redeploy the 5.6.1 container, run the following command.

   ```
   ./deploy-cache-pod.sh
   ```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.6.1 TM container.

   a. For single zone undeployment:

   - ```
     ./undeploy-tm-pod.sh
     ```

   b. Deploy Local Edition 5.6.2 TM container for single-zone deployment, run the following command.
     ```

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.
    a. Remove running Local Edition 5.6.1 TM container. For single zone undeployment, run the following command:

    ```
    ./undeploy-reporting-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 TM container for single-zone deployment, run the following command.

    ```
    source deploy-reporting-pod.sh "<zone-name>"
    ```

Upgrading Local Edition Cluster from 5.6.1 to 5.6.2 for Docker Swarm Tethered

The following section provides information on upgrading Local Edition cluster from version 5.6.1 to 5.6.2 for Docker swarm.

The following steps are applicable to swarm environment in tethered mode.

Before you begin

- Local Edition 5.6.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, and tml-cache are reused, so while doing upgrade, you should maintain same number of tml-nosql, tml-log, and tml-cache pods configured as the previous release and all previous volumes are reused accordingly.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.2.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.2 by running the compose command.

Note: Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade theLocal Edition components sequentially, explained as follows.

Procedure

1. To undeploy and redeploy NoSQL components:
    a. To remove running Local Edition 5.6.1 NoSQL container, run the following command for single-zone deployment:

    ```
    ./undeploy-nosql-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. To undeploy and redeploy Log components:
    a. To remove running Local Edition 5.6.1 Log container for single-zone deployment, run the following command.

    ```
    ./undeploy-log-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. To undeploy and redeploy CM components.
    a. To remove running Local Edition 5.6.1 CM container for single-zone deployment, run the following command.

    ```
    ./undeploy-cm-pod.sh
    ```

    b. To deploy Local Edition 5.6.2 CM containers for single-zone deployment, run the following command:

```
./deploy-cm-pod.sh
```

4. To undeploy and redeploy SQL components.

   The MySQL Certificate/Identity data migration for tethered mode works for the following migration cases:

   - 5.3.1 to 5.4.1
   - 5.3.1 to 5.5.1
   - 5.5.0 to 5.5.1
   - 5.5.1 to 5.5.2
   - 5.5.2 to 5.6.0
   - 5.6.0 to 5.6.1
   - 5.6.1 to 5.6.2

   Note: Complete the Certificate or Identity data migration as soon as the new Local Edition cluster is deployed, otherwise new certificates or identities created in the new Local Edition cluster is overwritten by data migration.

   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

      ```
      docker exec <sql-container-id> mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash_data.sql
      ```

      i. To remove running Local Edition 5.6.1 SQL container for single-zone deployment, run the following command:

         ```
         ./undeploy-sql-pod.sh
         ```

      ii. To delete SQL pvc after undeploying the SQL pod:

         ```
         docker volume rm sqlstack_sql-1-vol
         ```

iii. To deploy Local Edition 5.6.2 SQL containers for single zone deployment, run the following command.

```
./deploy-sql-pod.sh
```

b. Copy MySQL Data dumped from the Old TML Cluster to the New TML Cluster

Copy MySQL data file to each tml-sql pod in the new TML cluster:

```
docker cp mash_data.sql <sql-container-id>:/tmp
```

Note: The mysql-set-0-0 is the tml-sql pod in the new cluster.

c. Import MySQL Data dumped from the Old TML Cluster to the new TML Cluster

Run the following command in each tml-sql pod in the new TML cluster:

```
mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

d. Run the SQL loader in sql pods in the new TML Cluster

Run the following command in each tml-sql pod to run the sql loader:

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --
httpsclientsecurity --env production --verbose
```

5. Undeploy and redeploy Cache components.

Remove running Local Edition 5.6.1 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.2 Cache container for single-zone, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.6.1 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.6.1 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container for single-zone deployment, run the following command.

   ```
   ./deploy-reporting-pod.sh
   ```

   Post upgrade, the reporting pod IP will change. After reconfigurations of reporting log forward with new IP, reporting Dashboard will show data again.

Upgrading Local Edition Cluster from 5.6.1 to 5.6.2 for Kubernetes in Untethered Mode

The following section provides information on upgrading Local Edition cluster from version 5.6.1 to 5.6.2 for Kubernetes in untethered mode.

Before you begin

- Local Edition 5.6.2 images must be built and pushed to the desired registry for

deployment.

- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.2.
- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.2 by running the compose command.

Note:
- The following procedure is applicable for single and multi zone clusters.
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.6.1 NoSQL container, run the following command.
        - For single-zone deployment:

```
./undeploy-nosql-pod.sh
```

    b. Deploy Local Edition 5.6.2 NoSQL containers, run the following command.
        - For single-zone deployment:

```
./deploy-nosql-pod.sh
```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.6.1 Log container, run the following command.
        - For single-zone deployment:

```
./undeploy-log-pod.sh
```

b. To deploy Local Edition 5.6.2 Log containers for single-zone deployment, run the following command.

```
./deploy-log-pod.sh
```

3. Undeploy and redeploy CM components.
   a. To remove running Local Edition 5.6.1 CM container, run the following command.

```
./undeploy-cm-pod.sh
```

   b. To deploy Local Edition 5.6.2 CM containers, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.
   a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

   b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.areas --ignore-
table=masherysolar.package_key_audit_log --ignore-
table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar >
/tmp/mash_data.sql
```

> **Note:** In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.
>
> - masherysolar.member_activity_log
> - masherysolar.member_audit_log
> - masherysolar.member_role_audit_log
> - masherysolar.method_override_audit_log
> - masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
kubectl cp mysql-set-0-0:/tmp/mash_data.sql mash_data.sql
```

i. Remove the SQL container from the Local Edition 5.6.1 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

ii. Delete SQL pvc after undeploying the SQL pod.

```
kubectl delete pvc $(kubectl get pvc -o=jsonpath='{.items[?
(@.metadata.labels.app=="mysql-svc")].metadata.name}')l
```

iii. Deploy Local Edition 5.6.2 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.6.1 cluster in the Local Edition 5.6.2 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.6.2 cluster

```
kubectl cp mash_data.sql mysql-set-0-0:/tmp
```

e. Clear the MySQL data in the Local Edition 5.6.2 cluster.

i. In the 5.6.2 cluster, login to the tml-sql pod where the MySQL data file was copied.

```
kubectl exec -it mysql-set-0-0 -- /bin/bash
```

ii. To clear data in MySQL:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u root -p'changeme' -e "truncate table $table" masherysolar; done
```

f. Import MySQL data dumped from the Local Edition 5.6.0 cluster to the Local Edition 5.6.2 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

5. Undeploy and redeploy **Cache** components.

Remove running Local Edition 5.6.1 Cache container.

a. For single zone undeployment:

```
./undeploy-cache-pod.sh
```

b. To deploy the Local Edition 5.6.2 Cache container, run the following command.

```
./deploy-cache-pod.sh
```

6. Undeploy and redeploy **TM** components.

Remove running Local Edition 5.6.1 TM container.

a. For single zone undeployment:

```
./undeploy-tm-pod.sh
```

b. To deploy Local Edition 5.6.2 TM container, run the following command.

```
./deploy-tm-pod.sh
```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.6.1 TM container.

   a. For single zone undeployment:

```
./undeploy-reporting-pod.sh
```

   b. To deploy Local Edition 5.6.2 TM container, run the following command.

```
source deploy-reporting-pod.sh "<zone-name>"
```

Upgrading Local Edition Cluster from 5.6.1 to 5.6.2 Docker Swarm Untethered

The following section provides information on upgrading Local Edition cluster from version 5.6.1 to 5.6.2 for Docker swarm.

The following steps are applicable to swarm environment in untethered mode.

Before you begin

- Local Edition 5.6.2 images must be built and pushed to the desired registry for deployment.
- The volumes for tml-nosql, tml-log, tml-cache are reused, so while doing upgrade the number of these pods must be consistent with the earlier version.
- The customization done to the JSON properties files at docker-deploy/properties/*.json in earlier version must be ported to 5.6.1. For example, tml_papi_properties.json in 5.6.1 has the same structure as 5.6.2, the customized tml_papi_properties.json in 5.6.1 can be copied to 5.6.2.

- Admin must generate the manifest deployment folder, namely the manifest, as per the deployment topology for version 5.6.2 by running the compose command.

Note:
- Some traffic disruptions are expected during upgrade. To minimize disruptions, upgrade the Local Edition components sequentially, explained as follows.

Procedure

1. Undeploy and redeploy **NoSQL** components.
    a. Remove running Local Edition 5.6.1 NoSQL container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-nosql-pod.sh
        ```

    b. Deploy Local Edition 5.6.2 NoSQL containers, run the following command.

    ```
    ./deploy-nosql-pod.sh
    ```

2. Undeploy and redeploy **Log** components.
    a. To remove running Local Edition 5.6.1 Log container, run the following command.
        - For single-zone deployment:

        ```
        ./undeploy-log-pod.sh
        ```

    b. To deploy Local Edition 5.6.2 Log containers for single-zone deployment, run the following command.

    ```
    ./deploy-log-pod.sh
    ```

3. Undeploy and redeploy **CM** components.
    a. To remove running Local Edition 5.6.1 CM container for single-zone deployment, run the following command.

```
./undeploy-cm-pod.sh
```

b. To deploy Local Edition 5.6.2 CM containers for single-zone deployment, run the following command.

```
./deploy-cm-pod.sh
```

4. Undeploy and redeploy SQL components.

a. Dump MySQL Data from the cluster created in the earlier version. Login to any tml-sql pod running in the earlier cluster.

```
docker exec -it <mysql pod> -- /bin/bash
```

b. To dump MySQL data, run the following command. Update the MySQL password for the cluster.

```
mysqldump --no-create-info --complete-insert --single-transaction -u root -
p'changeme' --ignore-table=masherysolar.areas --ignore-
table=masherysolar.package_key_audit_log --ignore-
table=masherysolar.application_audit_log --set-gtid-purged=OFF masherysolar >
/tmp/mash_data.sql
```

Note: In case duplication of data occurs then redo the MySQL data dump by ignoring the duplicate entries. The following are few cases where duplication of entries is observed.

- masherysolar.member_activity_log
- masherysolar.member_audit_log
- masherysolar.member_role_audit_log
- masherysolar.method_override_audit_log
- masherysolar.migration_log

c. Copy the MySQL data file to the host.

```
docker cp <mysql pod>:/tmp/mash_data.sql mash_data.sql
```

i. Remove the SQL container from the Local Edition 5.6.1 cluster.

For single zone deployment,

```
./undeploy-sql-pod.sh
```

> **Note:** Delete SQL pvc after undeploying the SQL pod. For example, docker volume rm sqlstack_sql-1-vol

ii. Deploy Local Edition 5.6.2 SQL container, run the following command.

For single zone deployment:

```
./deploy-sql-pod.sh
```

d. Copy the MySQL data from Local Edition 5.6.1 cluster in the Local Edition 5.6.2 cluster.

Run the following command to copy MySQL data file to any tml-sql pod in TML 5.6.2 cluster

```
docker cp mash_data.sql <mysql pod>:/tmp
```

e. Clear the MySQL data in the Local Edition 5.6.2 cluster.

i. In the 5.6.1 cluster, login to thetml-sql pod where the MySQL data file was copied.

```
docker exec -it <mysql pod> /bin/bash
```

ii. To clear data in MySQL:

```
mysql -u root -p'changeme' -Nse 'show tables' masherysolar | sed -r 's/\b(areas|package_key_audit_log)\b//g' | sed '/^$/d' | while read table; do mysql -u root -p'changeme' -e "truncate table $table" masherysolar; done
```

f. Import MySQL data dumped from the Local Edition 5.6.1 cluster to the Local Edition 5.6.2 cluster.

Run the following command in each tml-sql pod in the new cluster.

```
mysql -u root -p'changeme' masherysolar < /tmp/mash_data.sql
```

5. Undeploy and redeploy **Cache** components.

   Remove running Local Edition 5.6.1 Cache container.

   a. For single zone undeployment:

   ```
   ./undeploy-cache-pod.sh
   ```

   b. To deploy the Local Edition 5.6.2 Cache container, run the following command.

   ```
   ./deploy-cache-pod.sh
   ```

6. Undeploy and redeploy **TM** components.

   Remove running Local Edition 5.6.1 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-tm-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container, run the following command.

   ```
   ./deploy-tm-pod.sh
   ```

7. Optional. This step is applicable if reporting pod is deployed. Undeploy and redeploy **Reporting** components.

   Remove running Local Edition 5.6.1 TM container.

   a. For single zone undeployment:

   ```
   ./undeploy-reporting-pod.sh
   ```

   b. To deploy Local Edition 5.6.2 TM container, run the following command.

   ```
   ./deploy-reporting-pod.sh
   ```

   Post upgrade, the reporting pod IP will change. After reconfigurations of

reporting log forward with new IP, reporting Dashboard will show data again.

Migrating Certificate Data in MySQL in Tethered mode

The following procedure is applicable for

- 5.3.1 to 5.4.1
- 5.3.1 to 5.5.0
- 5.4.1 to 5.5.0

Before you begin

Complete the data migration as soon as the new TML cluster is deployed, otherwise new certificates or identities created in the new TML cluster are overwritten by data migration.

Procedure

1. To dump MySQL data from the old TML cluster, run the following command against a tml-sql pod in the old TML cluster:

   ```
   kubectl exec mysql-set-0-0 -- mysqldump --no-create-info --complete-insert --single-transaction -u root -p'changeme' masherysolar trust_store identity_store > mash531_data.sql
   ```

2. To copy MySQL data file that is dumped from the old TML cluster to each tml-sql pod in the new TML cluster, run the following command:

   ```
   kubectl cp mash531_data.sql mysql-set-0-0:/tmp
   ```

   The mysql-set-0-0 is the tml-sql pod in the new TML cluster.

3. To import MySQL data dumped from the old TML cluster to the new TML cluster, run the following command in each tml-sql pod in the new TML cluster:

```
 mysql -u root -p'changeme' -e "truncate table masherysolar.trust_store"
mysql -u root -p'changeme' -e "truncate table masherysolar.identity_store"
mysql -u root -p'changeme' masherysolar < /tmp/mash531_data.sql
```

4. To run the SQL loader in sql pods in the new TML Cluster, run the following command in each tml-sql pod to run the sql loader:

```
/opt/sqlloader/onpremloader --service --mapi --devclass --packager --httpsclientsecurity --
env production --verbose
```

5. To refresh Cache in the new TML Cluster, run the following command in each tml-cache pod in the new TML cluster:

```
/opt/javaproxy/proxy/cacheloader --service --mapi --devclass --packager --httpsclientsecurity
--env production --verbose
```

Downgrading Local Edition to an Earlier Version

The operation to rollback an upgrade to an earlier version of Local Edition are similar to the upgrade path, except the sql-rollback.sh is used in the case of rollback path.

Notes for Downgrading Local Edition 5.4.1 to an Earlier Version

Kubernetes Notes

In place upgrade of a supported Local Edition cluster to Local Edition 5.4.1 is not possible on Kubernetes deployments, due to the fact that the group replication is ON with the MySQL and the upgrade involves configurations on the my.cnf.

In place upgrade, in case of Kubernetes, is not limited to DDLs and DMLs alone, hence the limitation. However, in case of Swarm deployment of Local Edition 5.4.1, it is possible to perform an in-place upgrade because MySQL group replication is OFF in that form factor.

As a corollary, in place rollback from Local Edition 5.4.1 to an earlier supported version is also not possible on Kubernetes deployment. It is possible with Swarm deployment.

MySQL Notes

For MySQL rollback, roll back from Local Edition 5.4.1 to 5.4.0, then roll back from 5.4.0 to 5.3.1, and then roll back from 5.3.1 to 5.3.0.

Boomi References

Refer to these links to learn more about Boomi privacy policy, terms of service, and Boomi help documentation:

Privacy Policy
Terms of Service
Help Documentation