Issue

When backing up Docker EE, swarm must be backed up as a separate component. The UCP backup does not take care of the Swarm.

Prerequisites

Before performing these steps, you must meet the following requirements:

- Docker Engine >= v17.03
- Linux

Backup

To backup the Swarm on Linux using Docker Engine >= 17.03, you can use the following steps.

Since this will need you to stop the engine of a manager, your cluster need to be healthy with at least 3 managers.

> Be sure **to** maintain **the** quorum **of** swarm managers. During **the time that** a manager **is** shut down, your swarm **is** more vulnerable **to** losing **the** quorum **if** further nodes are lost. The **number of** managers you **run is** a trade-off. If you regularly take down managers **to** do backups, consider **running** a 5-manager swarm, so **that** you can lose an additional manager **while the** backup **is running**, **without** disrupting your services.

1. Select a manager node to do the operation. Try not to choose the leader one in order to avoid a new election inside the cluster:

   ```
   docker node ls -f "role=manager" | tail -n+2 | grep -vi leader
   ```

2. Optional: Store the Docker version to a variable for easy addition to your backup name.

   ```
   ENGINE=$(docker version -f '{{.Server.Version}}')
   ```

3. Stop the Docker Engine on the manager before backing up the data, so that no data is being changed during the backup:

   ```
   systemctl stop docker
   ```

4. Backup the entire Swarm folder:

   ```
   tar cvzf "/tmp/swarm-${ENGINE}-$(hostname -s)-$(date +%s%z).tgz" /var/lib/docker/swarm/
   ```

   Note: *You can decode the Unix epoch in the filename by typing date -d @timestamp.* For example:

   ```
   date -d @1531166143
   Mon Jul  9 19:55:43 UTC 2018
   ```

5. Restart the manager Docker Engine:

```
systemctl start docker
```

Restore

Before restoring the Swarm:

- You have to restore the backup on the same Docker Engine version.
- You must use the same IP as the node you made the backup with, this is mandatory. The command to force the new cluster does not reset the IP in the Swarm data.

1. Shutdown the Docker Engine on the node where you choose to restore:

   ```
   systemctl stop docker
   ```

2. Remove the contents of the swarm folder `/var/lib/docker/swarm`.

3. Restore the swarm folder `/var/lib/docker/swarm` with the content of the backup.

4. Restart the new node:

   ```
   systemctl start docker
   ```

5. Re-initialize the Swarm to isolate this node into his own new cluster. Avoid attempting to connect it to nodes that were part of the old swarm, and presumably no longer exist.

   ```
   docker swarm init --force-new-cluster
   ```

6. Add the manager and worker nodes to bring your new swarm up to operating capacity.

Additional Resources

- https://docs.docker.com/engine/swarm/admin_guide/#back-up-the-swarm (https://docs.docker.com/engine/swarm/admin_guide/#back-up-the-swarm)
- https://docs.docker.com/engine/swarm/admin_guide/#restore-from-a-backup (https://docs.docker.com/engine/swarm/admin_guide/#restore-from-a-backup)