Upgrading UFM on Docker Container

Table of contents

Upgrading UFM on Docker Container in Standalone Mode

Upgrading UFM Container in High Availability Mode

Note

Upgrade the UFM container based on the existing UFM configuration files that are mounted on the server. It is important to use that same directory as a volume for the UFM installation command.In the below example /opt/ufm_files is used.

Upgrading UFM on Docker Container in Standalone Mode

1. Stop the UFM Enterprise service. Run:

```
systemctl stop ufm-enterprise
```

2. Remove the existing docker image. Run:

```
docker rmi mellanox/ufm-enterprise:latest
```

3. Load the new UFM Enterprise docker image. Run:

```
docker pull mellanox/ufm-enterprise:latest
```

4. Run the docker upgrade command:

```
docker run -it --name=ufm_installer --rm \
-v /var/run/docker.sock:/var/run/docker.sock \
-v /etc/systemd/system/:/etc/systemd_files/ \
-v /opt/ufm/files/:/opt/ufm/shared_config_files/ \
mellanox/ufm-enterprise:latest --upgrade
```

5. Reload system manager configuration:

```
systemctl daemon-reload
```

6. Start UFM Enterprise service:

```
systemctl start ufm-enterprise
```

Upgrading UFM Container in High Availability Mode

> **Note**
>
> As of UFM version 6.14.0, UFM upgrade on HA supports in-service upgrade, meaning UFM can continue running during the steps of the upgrade, and there is no need to stop UFM before the upgrade (although this is also supported).

1. Remove the old docker image **from the standby server**. Run:

```
Stand-by# docker rmi mellanox/ufm-enterprise:latest
```

2. Pull the new UFM Enterprise docker image **on the standby server**. Run:

```
docker pull mellanox/ufm-enterprise:latest
```

> **Note**
>
> At this stage, the UFM container has been updated with the latest code. The UFM data, however, will be updated during the next UFM run.

3. Perform a failover to start UFM on the upgraded node. On the master node, run:

   ```
   ufm_ha_cluster failover
   ```

   > **Note**
   >
   > When UFM starts, it will automatically update the UFM configuration.

4. Repeat steps 1-2 on the un-upgraded node (previous Master node).

5. On both servers, download and extract the latest UFM HA package. Run:

   ```
   wget https://www.mellanox.com/downloads/UFM/ufm_ha_5.5.0-9.tgz
   ```

6. On both servers, run the upgrade command for the HA package:

   ```
   ./install.sh --upgrade
   ```

7. Configure HA. There are two methods:

   - Configure HA with SSH Trust - Requires passwordless SSH connection between the servers.
   - Configure HA without SSH Trust - Does not require passwordless SSH connection between the servers, but asks you to run configuration commands on both servers.

   Configure HA with SSH Trust

   1.

      1. On the **master server only**, configure the HA nodes. To do so, from /tmp, run the `configure_ha_nodes.sh` command as shown in the below example

```
configure_ha_nodes.sh
--cluster-password 12345678 \
--local-primary-ip 10.10.50.1 \
--peer-primary-ip 10.10.50.2 \
--local-secondary-ip 192.168.10.1 \
--peer-secondary-ip 192.168.10.2 \
--no-vip
```

ⓘ **Note**

The script `configure_ha_nodes.sh` is is located under /usr/local/bin/, therefore, by default, you do not need to use the full path to run it.

ⓘ **Note**

The `--cluster-password` must be at least 8 characters long.

ⓘ **Note**

To set up a Virtual IP for UFM and gain access to UFM through this IP, regardless of which server is running UFM, you may employ the `--no-vip` OR `--virtual-ip` command and provide an IP address as an argument. This can be achieved by navigating to https://<Virtual-IP>/ufm on your web browser.

ⓘ **Note**

When using back-to-back ports with local IP addresses for HA sync interfaces, ensure that you add your IP addresses and hostnames to the /etc/hosts file. This is needed to allow the HA configuration to resolve hostnames correctly based on the IP addresses you are using.

ⓘ **Note**

configure_ha_nodes.sh requires SSH connection to the standby server. If SSH trust is not configured, then you are prompted to enter the SSH password of the standby server during configuration runtime

2. Depending on the size of your partition, wait for the configuration process to complete and DRBD sync to finish.

Configure HA without SSH Trust

If you cannot establish an SSH trust between your HA servers, you can use **ufm_ha_cluster directly to configure HA. To configure HA, follow the below instructions:**

ⓘ **Note**

Please change the variables in the commands below based on your setup.

1.

1. [On Standby Server] Run the following command to configure Standby Server:

    ```
    ufm_ha_cluster config -r standby \
    --local-primary-ip 10.10.50.1 \
    --peer-primary-ip 10.10.50.2 \
    --local-secondary-ip 192.168.10.1 \
    --peer-secondary-ip 192.168.10.2 \
    --hacluster-pwd 123456789 \
    --no-vip
    ```

2. [On Master Server] Run the following command to configure Master Server:

    ```
    ufm_ha_cluster config -r master --local-primary-ip 10.10.50.1 \
    --peer-primary-ip 10.10.50.2 \
    --local-secondary-ip 192.168.10.1 \
    --peer-secondary-ip 192.168.10.2 \
    --hacluster-pwd 123456789 \
    --no-vip
    ```

    IPv6 Example:

    ```
    configure_ha_nodes.sh
    --cluster-password 12345678 \
    --local-primary-ip fcfc:fcfc:209:224:20c:29ff:fee7:d5f2 \
    --peer-primary-ip fcfc:fcfc:209:224:20c:29ff:fecb:4962 \
    --local-secondary-ip fe80::1270:fd03:17:6365 \
    --peer-secondary-ip fe80::1270:fd03:17:5375 \
    --no-vip
    ```

You must wait until after configuration for DRBD sync to finish depending on the size of your partition.

8. Start UFM HA cluster. Run:

```
ufm_ha_cluster start
```

© Copyright 2024, NVIDIA. PDF Generated on 06/06/2024