rest-rdma Plugin

Table of contents

rest-rdma is a tool designed for sending requests over InfiniBand to the UFM server. These REST requests can fall into three categories:

1. UFM REST API requests
2. ibdiagnet requests
3. Telemetry requests

The rest-rdma utility is distributed as a Docker container, capable of functioning both as a server and a client.

Deployment Server

Deploy Plugin on UFM Appliance

1. Log into your UFM as admin.
2. Enter config mode. Run:

```
enable
config terminal
```

> **Note**
>
> Make sure that UFM is running with `show ufm status`. If UFM is down, then run with `ufm start`.

3. Ensure that rest-rdma plugin is disabled with the `show ufm plugin` command.
4. Pull the plugin container with `docker pull mellanox/ufm-plugin-rest-rdma:[version]`.
5. Run `ufm plugin rest-rdma add tag [version]` to enable the plugin.
6. Check that plugin is up and running with `docker pull mellanox/ufm-plugin-rest-rdma:[version]`

Deploy Plugin on Bare Metal Server

1. Verify that UFM is installed and running.

2. Pull image from docker hub:
   docker pull mellanox/ufm-plugin-rest-rdma:[version]

3. To load image run:
   /opt/ufm/scripts/manage_ufm_plugins.py add -p rest-rdma

Deployment Client

Run the following command to pull the image from the docker hub:

```
docker pull mellanox/ufm-plugin-rest-rdma:[version]
```

Verify that the /tmp/ibdiagnet directory exists on the client's computer. If not – create it.

To start container as client (on any host in the same fabric as UFM server) run:

```
docker run -d --network=host --privileged --name=ufm-plugin-rest-rdma --rm -v
/tmp/ibdiagnet:/tmp/ibdiagnet mellanox/ufm-plugin-rest-rdma:[version] client
```

To check that plugin is up and running, run:

```
docker ps
```

How to Run

Server

In server mode ufm_rdma.py is started automatically and is restarted if exited. If the ufm_rdma.py server is not running – enter to the docker and run the following commands to start the server:

```
cd /opt/ufm/src/ufm-plugin-ufm-rest
./ufm_rdma.py -r server
```

Client

There are three options to run client. Running the client from inside the Docker container, using a custom script from the hosting server to execute the client or using the "docker exec" command from the hosting server.

1. **Option 1:** Run the client from inside the Docker container
    1. Enter the docker container using docker exec -it ufm-plugin-rest-rdma bash
    2. Then, run cd /opt/ufm/src/ufm-plugin-rest-rdma
    3. Use the -h help option to see the available parameters
       ./ufm_rdma.py -h
2. **Option 2:** From the host server, the scripts can be located at /opt/ufm/ufm-plugin-ufm-rest/ directory inside the docker container. They can copied using the following command:

    > **Note**
    >
    > cp <containerId>:/opt/ufm/ufm-plugin-ufm-rest/[script name] /host/path/target

    Example:

    > **Note**

```
cp <containerId>:/opt/ufm/ufm-plugin-ufm-rest/ufm-rest-rdma_client.sh /host/path/target
```

1. To see the available options, run:
   ./ufm-rest-rdma_client.sh -h

3. **Option 3:** From hosting server, use the docker exec command.

> **Note**
>
> To run from inside docker, run:
>
> docker exec ufm-plugin-rest-rdma prior to the command.
>
> For example: docker exec ufm-plugin-rest-rdma /opt/ufm/ufm-plugin-ufm-rest/src/ufm_rdma.py -r client -u admin -p password -t simple -a GET -w ufmRest/app/ufm_version

Authentication Configuration

Telemetry and ibdiagnet request authentication options could be enabled or disabled (enabled by default – set to True) in ufm_rdma.ini file in [Server] section on the server. The rest_rdma server performs simple requests to UFM server, using supplied credentials to verify that the user is allowed to run telemetry or ibdiagnet requests.

```
[Server]
use_ufm_authentication=True
```

Remote ibdiagnet Request

The following two user scripts can run on the hosting server.

- remote_ibdiagnet_auth.sh
- remote_ibdiagnet.sh

These scripts should be copied from the container to the hosting server using the following command:

```
cp <containerId>:/opt/ufm/ufm-plugin-ufm-rest/[script name] /host/path/target
```

Example :

```
cp <containerId>:/opt/ufm/ufm-plugin-ufm-rest/remote_ibdiagnet_auth.sh /host/path/target
```

The remote_ibdiagnet.sh script does not require authentication as the server side can run on a machine which does not run UFM (which is responsible for the authentication). This means it can run from the hosting server.

```
/remote_ibdiagnet.sh [options]
```

Authenticated Remote ibdiagnet Request

The remote_ibdiagnet_auth.sh script can receive parameters as credentials for authentication with UFM server.

```
/remote_ibdiagnet_auth.sh [options]
```

To get all the options, run the following command:

```
/remote_ibdiagnet_auth.sh -h
```

Note

Important Note:

When using remote_ibdiagnet.sh, authentication is not required and the the ibdiagnet parameters should be sent in ibdiagnet format.

Example: ./remote_ibdiagnet.sh --get_phy_info

When using the remote_ibdiagnet_**auth**.sh, the ibdiagnet parameters should be sent using the –l key.

Example without credentials: ./remote_ibdiagnet_auth.sh -l '--get_phy_info'

Example with credentials: ./remote_ibdiagnet_auth.sh -u username -p password -l '–get_phy_info'

Please use the -h option to see the examples of credential usage.

Rest Request with Username/Password Authentication

To get the UFM version from inside the docker:

```
./ufm_rdma.py -r client -u admin -p admin_pwd -t simple -a GET -w ufmRest/app/ufm_version
```

To get the UFM version from hosting server using script:

```
./ufm_rest_rdma_client.sh -u admin -p admin_pwd -t simple -a GET -w ufmRest/app/ufm_version
```

For telemetry:

```
./ufm_rdma.py -r client -u admin -p admin_pwd -t telemetry -a GET -g 9001 -w /csv/enterprise
```

To get ibdiagnet run result using UFM REST API from inside the docker:

```
./ufm_rdma.py -r client -u admin -p admin_pwd -t ibdiagnet -a POST -w ufmRest/reports/ibdiagnetPeriodic -l '{"general": {"name": "IBDiagnet_CMD_1234567890_199_88", "location": "local", "running_mode": "once"}, "command_flags": {"--pc": ""}}'
```

Rest Request with Client Certificate Authentication

need to pass path to client certificate file and name of UFM server machine:
6. ./ufm_rdma.py -r client -t simple -a GET -w ufmRest/resources/modules -d /path/to/certificate/file/ufm-client.pfx -s ufm.azurehpc.core.azure-test.net
for telemetry if need authentication from inside the docker
./ufm_rdma.py -r client -t telemetry -a GET -g 9001 -w csv/enterprise -d /path/to/certificate/file/ufm-client.pfx -s ufm.azurehpc.core.azure-test.net

Note

Client certificate file should be located INSIDE the docker container.

Rest Request with Token Authentication

need to pass token for authentication
./ufm_rdma.py -r client -k OGUY7TwLvTmFkXyTkcsEWD9KKNvq6f -t simple -a GET -w ufmRestV3/app/ufm_version

```
for telemetry if need to perform authentication
./ufm_rdma.py -r client -k 4rQRf7i7wEeliuJEurGbeecc210V6G -t telemetry -a GET -g
9001 -w /csv/enterprise
```

Note

Token could be generated using UFM UI.

Note

If a token is used for client authentication, ufmRestV3 must be used.