DH2i DxOdyssey 21.0 Software: Container Quick Start Guide

DH2i Company

Support: +1 (800) 380-5405 ext. 2
https://dh2i.com/support/
eFax: +1 970-295-4505
Email: support@dh2i.com
https://www.dh2i.com

DxOdyssey Containers

This quick start guide describes how to use DxOdyssey containers to connect to a remote SQL instance via tunnels. Using this guide, the user will create and configure containers at Sites A and B using Docker, join the containers together into a gateway group, and create a tunnel between the sites that can be used to access the SQL instance.

Prerequisites

- A virtual machine at Site A running Microsoft SQL Server and listening on port 1433. The IP address of the VM will also be required in later steps. For information about installing SQL Server, view Microsoft documentation. The VM may be Linux or Windows.
- Linux Docker hosts at Site A and Site B. For information about installing Docker, please reference Docker documentation. The Docker host at Site B should not have direct network access to the SQL instance at Site A.
- SSMS or sqlcmd installed on a VM at Site B.

Create and Configure the DxOdyssey Container at Site A

1. Pull the latest DxOdyssey container image using the command `docker pull dh2i/dxo`.
2. Start a DxOdyssey container using the `docker run` command. Parameter explanations and a command example are given below.
    - DxOdyssey uses TCP port 7979 for gateway group administration with the DxOdyssey Client UI; this port is mapped using the `-p` parameter. The port number to the right of the colon must be 7979, but the numbers to the left of the colon – the port mapped to the host – does not have to be identical.
    - The `-d` flag runs the container in ‘detached’ (background) mode so that the host machine’s command line can still be used after launching the container.
    - The `-h` parameter assigns the internal container hostname used by DxOdyssey. This must be a unique name for each container participating in the gateway group.
    - The `--name` parameter specifies the name given to Docker for the container. The container name is not required to be the same as the hostname.
    - The `-v` parameter creates a named persistent data store for the /etc/dh2i folder that lasts beyond the lifetime of the container; if the container is removed, this parameter will save the gateway group configuration as a Docker volume that may be used by a new container.
    - The command example given below incorporates these parameters:
        - `docker run -p 7979:7979 -d -h dxodocker1 --name dxo1 -v dxo1vol:/etc/dh2i dh2i/dxo`
3. Connect to the container using the command `docker exec -it dxo1 bash`.
4. Activate the DxOdyssey license key using the command `dxcli activate-server`.

    Syntax

    ```
    dxcli activate-server <license_key> [gateway]
    ```

    Parameters

Name	Description	Required
license_key	The license key.	True
node	The name of the node.	False

Example

```
dxcli activate-server AAAA-BBBB-CCCC-DDDD
```

5. Set the DxOdyssey gateway group passkey using the command `dxcli gateway-group-set-secret`.

   Syntax

   ```
   dxcli gateway-group-set-secret
   ```

   Parameters

   Interactive only.

   Example

   ```
   dxcli gateway-group-set-secret
   ```

6. Set a One-Time PassKey (OTPK) for the gateway group using the command `dxcli set-otpk`. Save the output from this command; it will be used in the next section to join the second gateway to the first in a gateway group.

   Syntax

   ```
   dxcli set-otpk [ttl] [otpk]
   ```

   Parameters

Name	Description	Required
ttl	The time to live.	False
otpk	The One-Time PassKey in base64.	False

   Example

   ```
   dxcli set-otpk
   ```

Create and Configure the DxOdyssey Container at Site B

1. Pull the latest DxOdyssey container image using the command `docker pull dh2i/dxo`.
2. Start a DxOdyssey container using the `docker run` command. In addition to the parameter examples given in the previous section, a tunnel origin must also be mapped to the host machine; the origin will be used to access the SQL instance from Site B. In the command example below, the origin is mapped to port 11433 and the container names are unique.
   - `docker run -p 7979:7979 -p 11433:11433 -d -h dxodocker2 --name dxo2 -v dxo2vol:/etc/dh2i dh2i/dxo`
3. Connect to the container using the command `docker exec -it dxo2 bash`.
4. Join the gateway group using the command `dxcli join-gateway-group`. When prompted to join via NAT proxy, select **Yes**. When prompted to enter the NAT proxy, use the default value

of **match.dh2i.com**. When prompted for the passkey, use the OTPK saved from the previous section.

Syntax

```
dxcli join-gateway-group
```

Parameters

Interactive only.

Example

```
dxcli join-gateway-group
```

5. Activate the DxOdyssey license key using the command `dxcli activate-server`.
6. Create a tunnel using the command `dxcli add-tunnel`. The command example given below shows a tunnel with an origin at the Site B container (the currently-connected container). This origin accepts connections from any IP address that has access to the Docker host (0.0.0.0) on port 11433. The *destination gateway* of the tunnel is the DxOdyssey container at Site A, while the *destination IP* is the SQL Server VM at Site A using the default SQL listening port of 1433. In this example, the `<destination_IP>` should be substituted with the IP address of the SQL Server VM.

   Syntax

   ```
   dxcli add-tunnel <name> <enabled [true|false]> <destination_gateway>
   <destination_address:destination_port>
   <origin_gateway,origin_address:origin_port[,address_filter]>|<origin
   _gateway,origin_address:origin_port[,address_filter]>
   ```

   Parameters

Name	Description	Required
name	The name of the tunnel.	True
enabled	Enable the tunnel (true or false).	True
destination_gateway	The name of the destination gateway.	True
destination_address	The IP address of the destination.	True
destination_port	The port number for the destination IP.	True
origin_gateway	The name of the gateway where the listener is active.	True
origin_address	Set to 0.0.0.0 to allow all IP connections or 127.0.0.1 for local connections only.	True
origin_port	The port number for the origin gateway.	True
address_filter	The name of the address filter to add to the tunnel.	False

   Example

   ```
   dxcli add-tunnel SQL_TUNNEL true dxodocker1 <destination_IP>:1433
   dxodocker2,0.0.0.0:11433
   ```

Connect to the SQL Server Instance from Site B

1. Exit the container by running the command `exit`.
2. Connect to the machine running SSMS or sqlcmd.

3. Use one of following connection strings to access the tunnel, substituting in the Docker host's IP address or hostname:
    - <Docker Host IP>,11433
    - <Docker Host Hostname>,11433

References

- DH2i Support Portal
- DxOdyssey Documentation
- DxOdyssey DxCLI Guide
- Docker – CLI Reference Guide
- Docker – Install Docker Engine
- Microsoft – Install SQL Server