



DH2i DxOdyssey 19.5 Software: IoT with SQL Edge on Azure 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 and SQL Edge IoT Modules in Azure

This quick start guide describes how to set up and configure IoT Modules on Azure running DxOdyssey and SQL Edge to communicate with a local machine running DxOdyssey using tunnels. This document assumes that the sites do not have direct network access to each another. Using this guide, the user will configure DxOdyssey on a local machine, create an Azure IoT Edge device, deploy DxOdyssey and SQL Edge modules, and configure the DxOdyssey module to communicate with the local machine using tunnels.

Install and Configure DxOdyssey On-Premises

1. Install system updates.
 - a. Run `sudo apt-get update` to refresh Ubuntu package information.
 - b. Run `sudo apt-get dist-upgrade` to install updates.
2. Ensure the system hostname is set and unique (e.g. dxo1).
 - a. Run `hostname` to display the current hostname.
 - b. Run `sudo hostnamectl set-hostname dxo1` to set the hostname for the machine.
3. Install .Net Core Runtime 3.1.
 - Refer to [Microsoft documentation](#) for the latest installation instructions for .NET Core.
4. Install DxOdyssey.
 - a. Place a copy of the DxOdyssey .deb package in a local directory.
 - b. Run `sudo dpkg -I dxodyssey_<version>_amd64.deb` to install the software.
5. Activate the DxOdyssey license using DxCli.

Syntax

```
dxcli activate-server <key> [node]
```

Parameters

Name	Description	Required
key	The license key.	True
node	The name of the node (comma separated list for multiples).	False

Example

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

6. Additional gateways may join the DxOdyssey gateway group via tunneling technology. To join another gateway using NAT, a One-Time PassKey (OTPK) must be set for the gateway group. Save the GUID output from this command.

Syntax

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

Parameters

Name	Description	Required
ttl	The time the key will expire.	False
otpk	The One-Time PassKey in base64.	False

Example

```
dxcli set-otpk
```

Create DxOdyssey and SQL Edge Modules on Azure

- Go to the Azure portal – <https://portal.azure.com>
- Log into the Azure account.
- Create an Azure IoT Edge Ubuntu VM, Resource Group, IoT Hub, and IoT Edge Device using [Microsoft's IoT Edge Quick Start Documentation](#).
- Set the connection string for the IoT Edge device in Cloud Shell. The name of the resource group, the hostname of the Azure VM, and the primary connection string for the IoT Edge Device are needed for this command. The primary connection string can be found under the properties of the IoT Edge Device on Azure.
 - ```
az vm run-command invoke -g <resource_group> -n <vm_name> --command-id RunShellScript --script "/etc/iotedge/configedge.sh '{device_connection_string}'"
```
- Deploy a DxOdyssey IoT Module.
  - Go to the Azure Marketplace and search for **DxOdyssey for IoT**.
  - Select **DxOdyssey for IoT**, then select **Create**.
  - Specify the Subscription, IoT Hub Name, and IoT Edge Device name, then select **Create**.
  - Select the module being created to see the properties and select **Environment Variables**.
    - Set the DX\_LICENSE variable with the DxOdyssey license key.
    - Set the DX\_OTPK variable with the One-Time PassKey generated in the previous section.
    - Set the DX\_PASSKEY variable to the gateway group passkey to use for DxOdyssey Client UI access.
    - Add a DX\_HOSTNAME (e.g. dxoedge1) variable to set a hostname for the module. This allows the user to redeploy the module while having it also retain membership in the gateway group when it attempts to rejoin. It also allows the alias to be used for tunnel origins and destinations.
  - Select **Review + Create**, then select **Create**
- Deploy a SQL Edge IoT Module.
  - Go to the Azure Marketplace and search for **Azure SQL Edge**.
  - Select **Azure SQL Edge**, then select **Create**.
  - Specify the Subscription, IoT Hub Name, and IoT Edge Device Name, then select **Create**.
  - Select **Review + Create**, then select **Create**.
- Connect to the Azure IoT Edge Ubuntu VM using `ssh azureuser@<public_ip>`.

- Verify the modules are active by running `sudo iotedget list` and note the name of the SQL Edge Module. This name will be used later to create tunnel that connects to the SQL instance.

## Add a Tunnel to the Gateway Group and Connect Using SSMS

DxOdyssey uses TCP tunnels to connect to connect gateways at remote locations. The following command example illustrates the creation of a tunnel name SQL-EDGE with an origin gateway at the on-premise location (dxo1) that accepts connection from any IP on the local network (0.0.0.0) on port 11433. The destination gateway is the DxOdyssey IoT Module in Azure (i.e. the DX\_HOST value of dxoedge1) and a destination of the Azure SQL Edge container on Azure (AzureSQLEdge) using the default port of 1433.

- To create a tunnel, run the following command:

### Syntax

```
dxcli add-tunnel <name> <enabled [true|false]>
<destination_gateway> <destination_address:destination_port>
<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 or hostname of the destination.                                            | True     |
| destination_port    | The port number for the destination.                                                      | 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 connections on the network or 127.0.0.1 for the gateway 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 (for tunnel access restriction).      | False    |

### Example

```
dxcli add-tunnel SQL-EDGE true dxoedge1 AzureSQLEdge:1433
dxo1,0.0.0.0:11433
```

- Launch SQL Server Management Studio (SSMS).
- Use the connection string `<dxo1_ip_address>,11433` to connect to the SQL Edge Module on Azure.

## References

- [DH2i Support Portal](#)
- [DxOdyssey v19.5 Documentation](#)
- [DxOdyssey v19.5 DxCli Guide](#)
- [.NET Core Installation Guide](#)

- [IoT Edge Linux Quick Start Guide](#)