DH2i DxEnterprise 20.0 Software:
MSSQL HA Instances for Linux Quick Start Guide

MSSQL HA Instances for Linux

This quick start guide describes how to set up MSSQL HA instance using DxEnterprise. Using this guide, the user will configure storage for a DxEnterprise Vhost, create the Vhost, and add a SQL instance to the Vhost.

Prerequisites

1. Install DxEnterprise and create a cluster of two or more VMs. For installation instructions, please reference the quick start guides for Ubuntu or RHEL/CentOS.
2. Install Microsoft SQL Server on each VM. For information about installing SQL Server on Linux, please reference Microsoft documentation.
3. One additional VM for iSCSI storage.

Configure the iSCSI Initiator

DxEnterprise supports any storage that is SCSI-3 Persistent Reservation compliant. These steps will guide the user through configuring an iSCSI Initiator so its storage will be presented to the servers (nodes) in the DxEnterprise cluster.

***NOTE:** Please ensure proper presentation of storage to nodes. If there is storage that is not managed by this DxEnterprise cluster presented to the nodes, it is possible to overwrite or corrupt the data held on that storage.*

For Ubuntu, do the following:

1. Install the iSCSI initiator by running the command `sudo apt install -y open-iscsi`.
2. Enable iSCSI by running `sudo systemctl enable open-iscsi`.
3. As root or using sudo, edit `/lib/systemd/system/open-iscsi.service` using a text editor such as nano or vim.
4. Ensure that the following entry is modified from `--loginall=automatic` to simply `-l`.
   - `ExecStart=-/sbin/iscsiadm -m node -l`

For RHEL/CentOS, do the following:

1. Enable the iSCSI initiator by running the command `sudo systemctl enable iscsi`.
2. As root or using sudo, edit `/usr/lib/systemd/system/iscsi.service` using a text editor such as nano or vim.
3. Within the file, ensure that the following entries are modified from `--loginall=automatic` to simply `-l`.
   - `ExecStart=-/sbin/iscsiadm -m node -l`
   - `ExecReload=-/sbin/iscsiadm -m node -l`

Be sure to obtain the IP address of the iSCSI initiator using a command such as `ip addr` before proceeding to the next section.

Configure iSCSI Target for All Nodes

On every node participating in the DxEnterprise cluster, run the following commands:

1. Discover the storage using `sudo iscsiadm –m discovery -t sendtargets -p <target_ip>`.
2. Login to the target using `sudo iscsiadm –m node –login`.
3. Locate the new storage device's path using `iscsiadm –m session –P3`. At or near the bottom the attached disk should be visible. Example output is given below:
   - `Attached scsi disk sdb        State: running`

Configure Disks in DxEnterprise

DxEnterprise leverages native file system (NTFS/ext4/xfs) and shared storage technology to coordinate access to a pool of disk resources in the cluster. Managing a disk puts that disk under DxEnterprise control.

1. After the shared disks are configured on each node, manage the disks using DxEnterprise. To manage a disk, use the `sudo dxcli add-disk` command. The `sudo dxcli get-disks` command can be used to look up a disk_id of interest.

   Syntax
   ```
   dxcli add-disk <disk_id> [label]
   ```

   Parameters

Name	Description	Required
disk_id	The ID of the disk.	True
label	The unique label for the disk.	False

   Example
   ```
   dxcli add-disk 50842140-be32-d3a7-45d4-3999bf3ad3a8 "Cluster Disk 1"
   ```

2. Create a volume on the managed disk using the `sudo dxcli create-volume` command.

   Syntax
   ```
   dxcli create-volume <disk_id>,<size_in_bytes>
   ```

   Parameters

Name	Description	Required
disk_id	The ID of the disk.	True
size_in_bytes	The size of the volume to create in bytes.	True

   Example
   ```
   dxcli create-volume 50842140-be32-d3a7-45d4-3999bf3ad3a8,1073741824
   ```

3. Format a volume on the managed disk using the `sudo dxcli format-volume` command. The `sudo dxcli get-disk-detail <disk_label>` command may be used to look up a volume ID of interest.

   Syntax

```
dxcli format-volume <volume_id> <fstype> <label> <block_size>
<quick_format:true|false> <compression:true|false>
[optional_parameters]
```

Parameters

Name	Description	Required
volume_id	The ID of the volume	True
fstype	The file system type. [FAT|FAT32|exFAT|EXT3|EXT4|NTFS|UDF|ReFS|XFS]	True
label	The label for the volume.	True
block_size	The block size in bytes. [512|1024|2048|4096|8192|16384|32768|65536]	True
quick_format:true|false	Whether or not to perform a quick format.	True
compression:true|false	Whether or not to enable compression.	True
optional_parameters	Optional parameters from format.com	False

Example
```
dxcli format-volume 3409ed39-60c1-4f49-8186-dfface26e2a1 EXT4
Volume1 4096 quick_format:true compression:false
```

4. Assign a mount point to the volume using the `sudo dxcli set-mountpoint` command.

Syntax
```
dxcli set-mountpoint <volume_id> <mount_point>
```

Parameters

Name	Description	Required
volume_id	The ID of the volume.	True
mount_point	The mount path.	True

Example
```
dxcli set-mountpoint 3409ed39-60c1-4f49-8186-dfface26e2a1
/mnt/volume1
```

Configure the Vhost

DxEnterprise uses Virtual Hosts (Vhosts) to provide failover support and high availability. A Vhost virtualizes the network name and IP address associated to a particular SQL Server Instance, file share, and/or service. Instead of using the network name and IP address of a physical server, a Vhost is created and assigned a unique name/virtual IP-address pair. Clients access the databases associated with an instance via the Vhost name or IP address; they do not need to know which node is running the SQL instance.

1. To add a Vhost to the DxEnterprise cluster, use the `sudo dxcli cluster-add-vhost` command.

When a Vhost is configured, the user will need to specify at least one node to participate in the Vhost. It is recommended to create A and PTR records for each Vhost in DNS for resolution as well as add Vhost entries to each node's local hosts file.

Syntax
```
dxcli cluster-add-vhost <vhost> <vips> <nodes> [autofailback]
[priority[1-5]] [ilb_ports]
```

Parameters

Name	Description	Required
vhost	The name of the Vhost.	True
vips	The virtual IP(s) for the Vhost (comma separated list for multiples). The use of a loopback address (127.0.0.1) is supported, but must be preceded by an asterisk (*).	True
nodes	The node(s) to add to the Vhost (comma-separated list for multiples).	True
autofailback	Set autofailback option or leave blank if autofailback is not desired.	False
priority	The priority order of failover between Vhosts (1 = highest, 5 = lowest)	False
ilb_ports	Comma-separated list of ports to use for internal load balancer probing.	False

Example
```
dxcli cluster-add-vhost vhost1 192.168.1.10 dxemssql1, dxemssql2
```

2. Assign the disks to the Vhost using the `sudo dxcli vhost-set-diskgroup` command.

   A diskgroup is a logical set of disks that are added to a Vhost. When a disk is added to a Vhost diskgroup, the disk will be set online on the active node and offline on all other nodes. If there is a failure of a disk within the diskgroup, the Vhost will failover onto the next available node in the cluster.

   Syntax
   ```
   dxcli vhost-set-diskgroup <vhost> <disk_ids>
   ```

   Parameters

Name	Description	Required
vhost	The name of the Vhost.	True
disk_ids	A full list of pipe-delimited (|) disk IDs to add to the diskgroup.	True

   Example
   ```
   dxcli vhost-set-diskgroup VHOST1 50842140-be32-d3a7-45d4-
   3999bf3ad3a8
   ```

3. Add a SQL instance to the Vhost using the `sudo dxcli add-instance` command.

Assigning a SQL Server instance to a Vhost creates a managed instance. When a SQL Server instance is added to a Vhost, DxEnterprise virtualizes the network name and IP address associated with the SQL Server instance creating a Virtual SQL Server instance. Clients can then access the Virtual SQL Server instance via the Vhost\instance name.

Syntax
```
add-instance <vhost>\<instance> <port> <sql_data_path>
<sql_log_path> [sql_login] [sql_pass] [keep]
```

Parameters

Name	Description	Required
vhost	The name of the Vhost.	True
instance	The name of the SQL instance.	True
port	The port to be used.	True
sql_data_path	The path for the SQL data	True
sql_log_path	The path for the SQL logs	True
sql_login	A sysadmin user for the SQL instance. If using Windows authentication, the credential must be supplied in UPN format (e.g. user@domain.com).	False
sql_pass	The password for the SQL sysadmin user. "dxcli encrypt-text <pass>" command can be used to generate encrypted string.	False
keep	Specify this parameter to keep the current SQL data.	False

Example
```
dxcli add-instance vhost1\mssqlserver 1433 /mnt/volume1/data
/mnt/volume1/log sa Gks+GJplFmUbTLlBy4wPmw==
```

References

- DH2i Support Portal
- DxEnterprise v20.0 Admin Guide
- DxEnterprise v20.0 DxCli Guide
- Microsoft – Install SQL Server on Linux