



dbX MNT AWS Setup Guide

Rev 3.3
March 2017



Overview	3
Single-node Setup	4
1. Configure & Launch dbX instance	4
2. Initialize	4
3. Start cluster & dbX	4
4. Stopping the cluster	5
5. Re-starting the cluster	5
Multi-node Setup	6
1. Enable the Software	6
2. Launch dbX template	6
3. Login the first time	6
4. Stopping the cluster	7
5. Re-starting the cluster	7
Upgrading the software	8



Overview

This document describes the process to launch an XtremeData dbX system on Amazon Web Services. Stopping and re-starting of the cluster is also covered.

dbX may be deployed as a single node system with a combined virtual head and data node via the AWS marketplace console. A multi-node dbX system comprises one “head” node and multiple “data” node instances which are launched through an appropriate Cloud Formation template. Both configurations are referred to as “clusters”. **Launching the cluster from a template is the preferred method of deployment.** Launching a single node from the marketplace is discussed just for completeness.

It is assumed that the user has some familiarity with AWS and knows how to set up an account and establish keys for communication with the running instances.

Please refer to dbX documentation for detailed product descriptions, user guides and SQL reference manuals.



Single-node Setup

Step 1: Configure and launch the instance
Step 2: Login & Initialize configuration
Step 3: Start cluster & dbX

1. Configure & Launch dbX instance

Marketplace Image

Search the “[AWS Marketplace](#)” for “dbx”.

Select the image titled “[XtremeData dbx MNT \(HVM\)](#)”.

Security Group

The security group defaults should be used to allow ssh access over port 22 and GUI administrative access via port 2400.

Block storage (EBS)

The instance can run with only ephemeral drives. However, if the instance is ever stopped, all state and data will be lost. To avoid this, EBS drives should be used. All storage devices that are made available before initialization will be used for dbX. If additional storage is needed for other purposes, it should be added after the initialization process is complete. If EBS is used, it is recommended to have at least 4 devices with a Volume Size that is a minimum of 100GB each (they should all be the same size). The recommended class of storage is General Purpose SSD.

2. Initialize

WARNING: Initialization is done once after the instance is created. This is a destructive operation that will wipe-out all pre-existing DB servers and data.

Any EBS storage devices that are to be used by dbx should be attached before proceeding. If there are EBS devices that should not be formatted and used by dbx, they should be detached before proceeding. All local ephemeral drives will be formatted and used by dbx.

Using your preferred terminal window (e.g. putty), login to the head node as “ec2-user”.

Initialize cluster by typing:

```
cluster_init -p -i 0
```

Note that “-p” is an option to make the head node also function as a data node. The “-i 0” option indicates that there are no additional data nodes (correct for a single-node configuration).

This will initialize all storage devices that are attached and make them available for dbX.

- Final configuration tasks will be performed.
- This is a DESTRUCTIVE process – no previous data or configuration will be preserved!

When finished, the command will report: “cluster_init done”

3. Start cluster & dbX

Start the cluster by typing:

```
cluster_start
```



When finished, the command will report: “cluster_start done”

Now your instance is initialized and running. Next step is to start the dbX database engine.

Type:

```
dbx_start
```

You will see some messages displayed during the start-up of dbX. Take note of the message “starting dbx nodes, may take several minutes”. This is normal.

When finished, the command will report “dbx startup done”

Upon starting dbX for the first time, your “ec2-user” keys will be copied to user “dbxdba”. User “dbxdba” is preconfigured to allow the creation and management of all DB servers. This user will also be used to log into the administration tool.

Important note: You need to setup a system password for user “dbxdba” in order to use the administration tool. To do this, at a system prompt, type:

```
sudo passwd dbxdba
```

You are now up and running! You can now login to the head as user “dbxdba” to create servers and databases or login to the administration tool at: <https://<IP address>:2400/xdadm>

Please refer to dbX user documentation volumes I – IV for information on creating and managing servers and databases. Note that any references in the documentation to tasks managed by user “xdAdm” are performed by user “dbxdba”.

4. Stopping the cluster

To stop the cluster, login to the head node as “ec2-user” and type:

```
dbx_stop
```

All the running database servers will be stopped and main daemon “xdu” will be stopped. Wait until the stop process is complete.

Then in the AWS console select the instance and stop it.

Note: if your data is on ephemeral storage (no EBS drives were configured), it will be lost!

5. Re-starting the cluster

To restart the cluster, select the instance and start it. Note that the public IP address will probably change unless you have attached an Elastic IP.

Once the instance has been re-started, you need to re-start the cluster and dbX. NOTE: since this is a re-start, and your database might already contain data, you should NOT initialize storage, otherwise all data will be lost!

Using your preferred terminal window, login to the head node as “ec2-user”

Start cluster by typing:

```
cluster_start
```

Start the dbX database engine by typing:

```
dbx_start
```



Multi-node Setup

Multi-node setup is done automatically with the proper template. Once deployed, the only additional step is to login to the head node and assign an administrative password.

Sample cloud formation templates:

For deployment into a new VPC:

https://s3.amazonaws.com/xtremedata-dbx-public/Marketplace-dbx_cluster-New_VPC-EBS_GP_SSD.template

For deployment into an existing VPC:

https://s3.amazonaws.com/xtremedata-dbx-public/Marketplace-dbx_cluster-Exist_VPC-EBS_GP_SSD.template

Note that the above templates implement 1 head node and 1 data node. To create a new template that deploys more data nodes, use the following Python-based scaling tool on one of the templates:

https://s3.amazonaws.com/xtremedata-dbx-public/dbx_template_scale_aws

```
$ ./dbx_template_scale_aws
usage: dbx_template_scale_aws [-h] [--ifile IFILE] [--ofile OFILE]
                             [--tab [TAB]] [--debug [DEBUG]]
                             nodes

$ ./dbx_template_scale_aws --ifile=Marketplace-dbx_cluster-New_VPC-
EBS_GP_SSD.template --ofile=my4nodes.template 4
```

Note that the number of nodes specified is the number of data nodes in addition to the head node.

1. Enable the Software

Before using the template, you must enable the software in Marketplace. Select the image titled “XtremeData dbx MNT (HVM)” and proceed like you plan to deploy it from Marketplace. You will be prompted to agree to the usage terms. After you have accepted the agreement, you can abort the deployment and then use the template. This process will only need to be done once for your account. If it is not done, you will get a deployment failure.

2. Launch dbX template

The multi-node template will configure the number of nodes and storage, run initialization and automatically start dbX. Note that this scenario utilizes a management framework for this automation. The framework will automatically shutdown dbx if there is a problem with the cluster and automatically bring it back up once the cluster is in a working state. This automated management supersedes any manual management of the service daemons associated with dbx, including xdadm and xdu.

3. Login the first time

Using your preferred terminal window (e.g. putty), login to the head node as “ec2-user”.

Check the status of the cluster:

```
xdc status
```

It should report: started. When the cluster is first launched, there may be several state transitions that may occur, including New, Stopped, Configured, Transitional or even Failed. This is normal and the cluster should transition to the “Started” state in a few minutes. If it does not reach that state, it may be due to a resourcing issue in the cloud infrastructure and may take some time to resolve. If it never reaches the “Started” state, the best course would be to delete the deployment and deploy a new stack.



Important note: You need to setup a system password for user “dbxdba” in order to use the administration tool. To do this, at a system prompt, type:

```
sudo passwd dbxdba
```

Upon starting dbX for the first time, your “ec2-user” keys will be copied to user “dbxdba”. User “dbxdba” is preconfigured to allow the creation and management of all DB servers. This user will also be used to log into the administration tool.

You are now up and running! You can now login to the head as user “dbxdba” to create servers and databases or login to the administration tool at: <https://<IP address>:2400/xdadm>

Please refer to dbX user documentation volumes I – IV for information on creating and managing servers and databases. Note that any references in the documentation to tasks managed by user “xdAdm” are performed by user “dbxdba” in this environment.

4. Stopping the cluster

To stop the cluster, login to the head node as “ec2-user” and type:

```
xdc stop
```

All the running database servers will be stopped and main daemon “xdu” will be stopped. Wait until the stop process is complete.

Then in the AWS console select the instances belonging to the cluster and stop them.

Note: if your data is on ephemeral storage (no EBS drives were configured in the template), it will be lost!

5. Re-starting the cluster

To restart the cluster, select the instances and start them.

Once all the instances have been re-started, the software will restart automatically.

Using your preferred terminal window, login to the head node as “ec2-user” to check the status of the cluster:

```
xdc status
```



Upgrading the software

It is always recommended that when launching a new cluster and periodically thereafter, that the software be upgraded to the latest version.

For the manually configured single node deployment, stop any running DB servers and run

```
yum update
```

For the multi-node deployment, upgrade the entire cluster from the head node as follows:

```
xdc stop
```

To upgrade all software packages, including the kernel:

```
xdc upgrade +all -verify
```

To upgrade only the DB software:

```
xdc upgrade -verify
```

Once the upgrade completes, run:

```
xdc verify +auto
```

On some systems there may be dependency errors that need to be manually addressed:

```
Required packages dependencies not met for:selinux-policy 3.7.19-292  
System state or settings are invalid. Xdc exit code:112
```

In this case, manually upgrade the dependency:

```
xdc upgrade selinux-policy
```

```
xdc verify +auto
```

If the kernel was upgraded, all the nodes need to be rebooted and they will automatically start the software. Otherwise run:

```
xdc start
```




[End of Document]