ThoughtSpot

AWS Installation Guide

Version 6.0 December 03, 2019

ThoughtSpot, Inc. 910 Hermosa Ct
Sunnyvale, CA 94085

Table of Contents

Installing Amazon Web Services

Summary: Learn how to install Amazon Web Services (AWS).

Installation Prerequisites

Ensure the successful creation of the virtual machines (VMs) before you install the ThoughtSpot cluster on AWS:

1. **Review configuration options** Refer to AWS configuration options for detailed instance specs.
2. **Create the instance** Refer to Set up AWS for ThoughtSpot to create and launch your instance.
3. **Review required ports** Refer to Network Policies to view the required ports for successful operation of ThoughtSpot.

Configure Nodes

After creating the instance, you must configure the nodes.

Step 1: Log into your cluster

Log into your cluster with admin credentials from Terminal on a Mac or a terminal emulator on Windows.

1. Run `ssh admin@clusterIP` or `ssh admin@hostname`, replacing 'clusterIP' or 'hostname' with your specific network information.
2. Enter your admin password.
    - Ask your network administrator if you don't know the password.

```
$ ssh admin@clusterIP
```

Step 2: Get a list of nodes to configure

Run the `tscli cluster get-config` command to get a list of the nodes to configure for the new cluster, and redirect it to the file `nodes.config`. You can find more information on this process in the nodes.config file reference.

```
$ tscli cluster get-config |& tee nodes.config
```

Step 3: Configure the network of nodes

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the autodiscovery of one node example.
2. Fill in the areas specified in Parameters of the `nodes.config` file with your specific network information.
    - If you have additional nodes, complete each node within the nodes.config file in the same way.

Make sure that you do not edit any part of the nodes.config file except the sections explained in Parameters of `nodes.config`. Deleting quotation marks, commas, or other parts of the code could cause setup to fail.

Step 4: Configure the nodes

Configure the nodes in the `nodes.config` file using the `set-config` command.

1. Disable `Firewalld` by running `sudo systemctl stop firewalld` in your terminal. `Firewalld` is a Linux firewall that must be off for ThoughtSpot installation. When the cluster installer reboots the nodes, `Firewalld` automatically turns back on.
2. Run `$ cat nodes.config | tscli cluster set-config`.
    - If the command returns an error, refer to set-config error recovery.

Set-config

```
$ sudo systemctl stop firewalld
$ cat nodes.config | tscli cluster set-config

Connecting to local node-scout
Setting up hostnames for all nodes
Setting up networking interfaces on all nodes
Setting up hosts file on all nodes
Setting up IPMI configuration
Setting up NTP Servers
Setting up Timezone
Done setting up ThoughtSpot
```

Set-config error recovery

If the set-config fails with the following warning, restart the node-scout service by running `sudo systemctl restart node-scout`.

Restart node-scout service

If you have this error, restart the node-scout:

```
Connecting to local node-scout WARNING: Detected 0 nodes, but f
ound configuration for only 1 nodes.
Continuing anyway. Error in cluster config validation: [] is no
t a valid link-local IPv6 address for node: 0e:86:e2:23:8f:76 C
onfiguration failed.
Please retry or contact support.
```

Restart node-scout with the following command, then retry the set-config command.

```
$ sudo systemctl restart node-scout
```

The command output should no longer have a warning:

```
$ cat nodes.config | tscli cluster set-config

Connecting to local node-scout
Setting up hostnames for all nodes
Setting up networking interfaces on all nodes
Setting up hosts file on all nodes
Setting up IPMI configuration
Setting up NTP Servers
Setting up Timezone
Done setting up ThoughtSpot
```

Step 5: Confirm node configuration with the `get-config` command

Run `tscli cluster get-config` in your terminal to confirm node configuration.

Confirm node configuration

```
$ tscli cluster get-config

{
  "ClusterId": "",
  "ClusterName": "",
  "DataNetmask": "255.255.252.0",
  "DataGateway": "192.168.4.1",
  "IPMINetmask": "255.255.252.0",
  "IPMIGateway": "192.168.4.1",
  "Timezone": "America/Los_Angeles",
  "NTPServers": "0.centos.pool.ntp.org,1.centos.pool.ntp.or
g,2.centos.pool.ntp.org,3.centos.pool.ntp.org",
  "DNS": "192.168.2.200,8.8.8.8",
  "SearchDomains": "example.company.com",
  "Nodes": {
        "ac:1f:6b:8a:77:f6": {
          "NodeId": "ac:1f:6b:8a:77:f6",
          "Hostname": "Thoughtspot-server1",
          "DataIface": {
            "Name": "eth2",
            "IPv4": "192.168.7.70"
          },
          "IPMI": {
            "IPv4": "192.168.5.70"
          }
        }
  }
}
```

Install Cluster

Next, install the cluster using the release tarball (est. time 1 hour).

If you do not have a link to download the release tarball, open a support ticket at ThoughtSpot Support to access the release tarball.

Step 1. Run the Installer

1. Copy the downloaded release tarball to `/home/admin` with the command `scp 0.0.tar.gz admin@hostname:/home/admin/file-name`. Replace '0.0' with your release number. Replace 'hostname' and 'file-name' with your specific hostname and the name of the tarball file.

```
$ scp 0.0.tar.gz admin@hostname:/home/admin/file-name
```

2. Run `tscli cluster create <release>`.
    - If you are using an s3 bucket for object storage, include the flag `--enable_cloud_storage s3`.

```
 $ tscli cluster create 6.0.tar.gz --enable_clou
d_storage s3
```

3. Edit the output with your specific cluster information. For more information on this process, refer to Using the `cluster create` command and Parameters of the `cluster create` command.

The cluster installer automatically reboots all the nodes after the install. `Firewalld` automatically turns back on. Wait at least 15 minutes for the installation process to complete. The system is rebooting, which takes a few minutes. Log into any node to check the current cluster status, using the command `tscli cluster status`.

Step 2. Check Cluster Health

Once the cluster is installed, check its status with the `tscli cluster status` command (Cluster Status).

Cluster Status

```
$ tscli cluster status
Cluster: RUNNING
Cluster name    : thoughtspot
Cluster id      : 1234X11111
Number of nodes : 3
Release         : 6.0
Last update     = Wed Oct 16 02:24:18 2019
Heterogeneous Cluster : False
Storage Type    : HDFS

Database: READY
Number of tables in READY state: 2185
Number of tables in OFFLINE state: 0
Number of tables in INPROGRESS state: 0
Number of tables in STALE state: 0
Number of tables in ERROR state: 0

Search Engine: READY
Has pending tables. Pending time = 1601679ms
Number of tables in KNOWN_TABLES state: 1934
Number of tables in READY state: 1928
Number of tables in WILL_REMOVE state: 0
Number of tables in BUILDING_AND_NOT_SERVING state: 0
Number of tables in BUILDING_AND_SERVING state: 128
Number of tables in WILL_NOT_INDEX state: 0
```

Step 3. Finalize Installation

After the cluster status changes to “Ready,” log into the ThoughtSpot application on your browser.

Follow these steps:

1. Start a browser from your computer.
2. Enter your secure IP information on the address line.

   ```
   https:<IP-address>
   ```

3. If you don’t have a security certificate for ThoughtSpot, you must bypass the security warning

to proceed:

- Click **Advanced**
- Click **Proceed**

4. The ThoughtSpot login page appears.
5. In the ThoughtSpot login window, enter admin credentials, and click **Sign in**. ThoughtSpot recommends changing the default admin password.

References

Use these references for successful installation and administration of ThoughtSpot.

- The `nodes.config` file
- Parameters of the `nodes.config` file
- Using the `cluster create` command
- Parameters of the `cluster create` command
- ThoughtSpot Documentation
- Contact Support

AWS configuration options

Summary: Your specific instances require specific configurations of memory, CPU, storage, and networking capacity.

ThoughtSpot can be deployed in your AWS environment by deploying compute (VM) instances in your Amazon VPC as well as an underlying persistent storage infrastructure. Currently two configuration modes are supported by ThoughtSpot:

- Mode 1: Compute VMs + EBS-only persistent storage
- Mode 2: Compute VMs + EBS and S3 persistent storage

The cost of infrastructure for deploying ThoughtSpot is cheaper when using S3. However, there are differences in where data is loaded, as well as in the backup and restore procedure. For assistance in choosing the best mode for your organization, contact your ThoughtSpot representative. For more information on purchasing ThoughtSpot on AWS, see: ThoughtSpot Pricing.

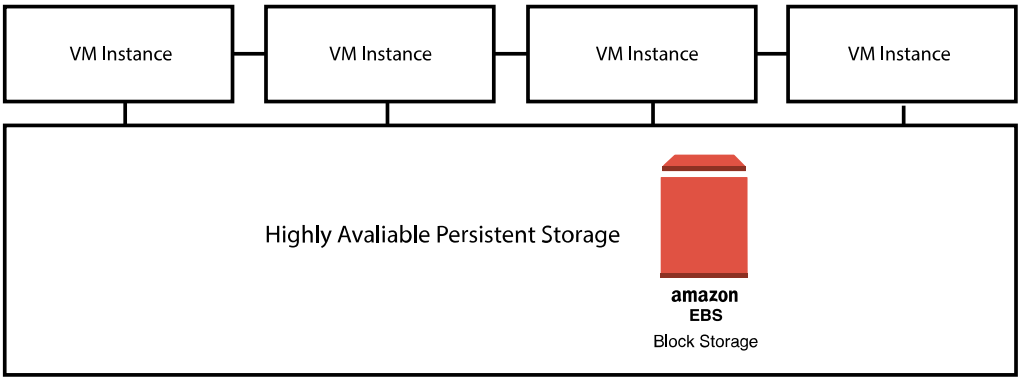
All AWS VMs in a ThoughtSpot cluster must be in the same availability zone (and therefore, also in the same region). ThoughtSpot does not support deploying VMs in the same cluster across availability zones. For more information, see Regions and Availability Zones in Amazon's AWS documentation.

ThoughtSpot AWS instance types

The following sections contain the supported and recommended instance types for a ThoughtSpot AWS deployment. When setting up your cluster in AWS, use the information here to select an instance type, configure the number of instances required for the storage you need, and add data volumes to your cluster.

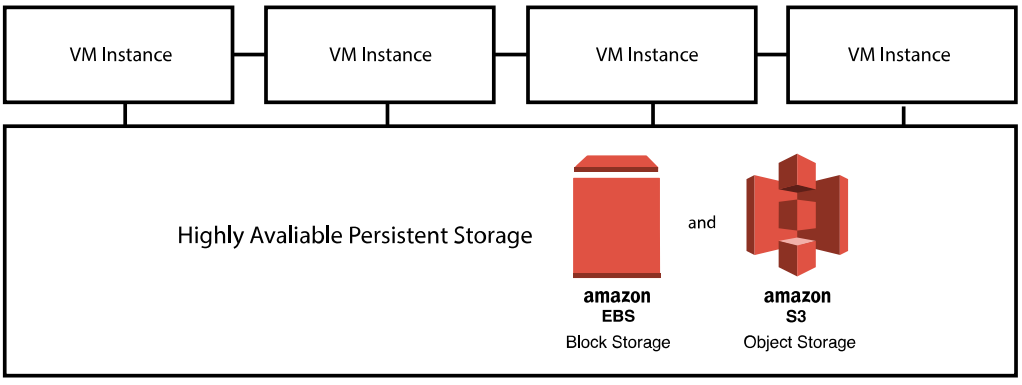
For example: If you were deploying a total cluster data size of 1 TB using the standard r5.16xlarge instance type, you would need 4 instances (VMs), because the per-VM user data capacity of that instance type is 250 GB. If you were deploying EBS-only data volumes, you would need 2x1 TB data volumes per VM.

VMs with EBS-only persistent storage



Per VM user data capacity	Instance type	CPU/RAM	Recommended per-VM EBS volume
20 GB	r4.4xlarge, r5.4xlarge	16/122, 16/128	2X 400 GB
100 GB	r4.8xlarge, r5.8xlarge	32/244, 32/256	2X 400 GB
192 GB	m5.24xlarge	96/384	2X 1 TB
250 GB	r4.16xlarge, r5.16xlarge	64/488, 64/512	2x 1 TB
384 GB	r5.24xlarge	96/768	2X 1.5 TB

VMs with EBS and S3 persistent storage



Per VM user data capacity	Instance type	CPU/RAM	Recommended per-VM EBS volume
20 GB	r4.4xlarge, r5.4xlarge	16/122, 16/128	1x 500 GB
100 GB	r4.8xlarge, r5.8xlarge	32/244, 32/256	1x 500 GB
192 GB	m5.24xlarge	96/384	1x 500 GB
250 GB	r4.16xlarge, r5.16xlarge	64/488, 64/512	1x 500 GB
384 GB	r5.24xlarge	96/768	1x 500 GB

Note: The S3 bucket size is approximately equal to the size of the user data.

Related information

- EC2 instance types
- EC2 pricing
- EBS pricing
- Placement groups

Set up AWS for ThoughtSpot

Summary: After you determine your configuration options, you must set up your virtual machines (VMs) on AWS using a ThoughtSpot Amazon Machine Image (AMI).

Overview of ThoughtSpot setup in AWS

The high-level process for setting up ThoughtSpot in AWS involves these steps:

1. Gain access to ThoughtSpot AMIs.
2. Choose a VM instance configuration recommended by ThoughtSpot.
3. Set up your Amazon S3 bucket (optional).
4. Set up your ThoughtSpot cluster in AWS.
5. Contact ThoughtSpot to finish setting up your cluster.
6. Open the required network ports for communication for the nodes in your cluster and end users.

About the ThoughtSpot AMI

An Amazon Machine image (AMI) is a preconfigured template that provides the information required to launch an instance. You must specify an AMI when you launch an instance in AWS.

To make deployment easy, the ThoughtSpot AMI includes a custom ThoughtSpot image, with the following components:

- A template for the root volume for the instance, such as an operating system, an appliance server, and applications.
- Launch permissions that control which AWS accounts can use the AMI to launch instances.
- A block device mapping that specifies the volumes to attach to the instance when it launches.

The ThoughtSpot AMI has specific applications on a CentOS base image. The AMI includes the EBS volumes necessary to install ThoughtSpot in AWS. When you launch an EC2 instance from this image, it automatically sizes and provisiones the EBS volumes. The base AMI includes 200 GB (xvda), 2X400 GB (xvdb), and SSD (gp2). It contains the maximum number of disks to handle a fully loaded VM.

Prerequisites

To install and launch ThoughtSpot, you must have the following:

- Familiarity with Linux administration, and a general understanding of cloud deployment models.
- The necessary AWS Identity and Access Management (IAM) users and roles assigned to you to access and deploy the various AWS resources and services as defined in the Required AWS components section that follows.

  For more information about IAM, see: What Is IAM? in Amazon's AWS documentation.

Required AWS components

- An AWS VPC. For details, see VPC and Subnets in Amazon's AWS documentation.
- A ThoughtSpot AMI. For details, see the next section.
- AWS security groups. For required open ports, see network policies.
- AWS VM instances. For instance type recommendations, see ThoughtSpot AWS instance types.
- EBS volumes.
- (Optional) If deploying with S3 persistent storage, one S3 bucket dedicated to each ThoughtSpot cluster.

Guidelines for setting up your EC2 instances

- Sign in to your AWS account.
- Copy the following ThoughtSpot public AMI which has been made available in N. California region to your AWS region:

  AMI Name: thoughtspot-image-20190718-dda1cc60a58-prod

  AMI ID: ami-0b23846e4761375f1

  Region: N. California

  > ℹ **Note:** The AMI is backward-compatible with ThoughtSpot releases 5.1.x - 5.2.x.

- Choose the appropriate EC2 instance type: See ThoughtSpot AWS instance types for supported instance types.
- Networking requirements: 10 GbE network bandwidth is needed between the VMs. This is the

default for the VM type recommended by ThoughtSpot.

- Security: The VMs that are part of a cluster need to be accessible by each other, which means they need to be on the same Amazon Virtual Private Cloud (VPC) and subnetwork. Additional external access may be required to bring data in/out of the VMs to your network.
- Number of EC2 instances needed: Based on the datasets, this number will vary. Please check ThoughtSpot AWS instance types for recommended nodes for a given data size.
- Staging larger datasets (> 50 GB per VM), may require provisioning additional attached EBS volumes that are SSD (gp2).

Setting up your Amazon S3 bucket

If you are going to deploy your cluster using the S3-storage option, you must set up that bucket before you set up your cluster. Contact ThoughtSpot Support to find out if your specific cluster size will benefit from the S3 storage option.

To set up an Amazon S3 bucket in AWS, do the following:

1. In AWS, navigate to the S3 service dashboard by clicking **Services**, then **S3**.
2. Make sure the selected region in the top-right corner of the dashboard is the same region in which you plan to set up your cluster.
3. Click **Create bucket**.
4. In the **Name and region** page, enter a name for your bucket, select the region where to set up the cluster, and click **Next**.
5. On the **Properties** page, click **Next**.
6. On the Configure options page, make sure **Block *all* public access** is selected and click **Next**.
7. On the Set permissions page, click **Create bucket**.

Encrypting your data at rest on Amazon EBS or S3 in AWS

ThoughtSpot makes use of EBS for the data volumes to store persistent data (in the EBS deployment model) and the boot volume (in the EBS and S3 deployment models). ThoughtSpot recommends that you encrypt your data volumes prior to setting up your ThoughtSpot cluster. If you are using the S3 persistent storage model, you can encrypt the S3 buckets using SSE-S3. ThoughtSpot does not currently support AWS KMS encryption for AWS S3.

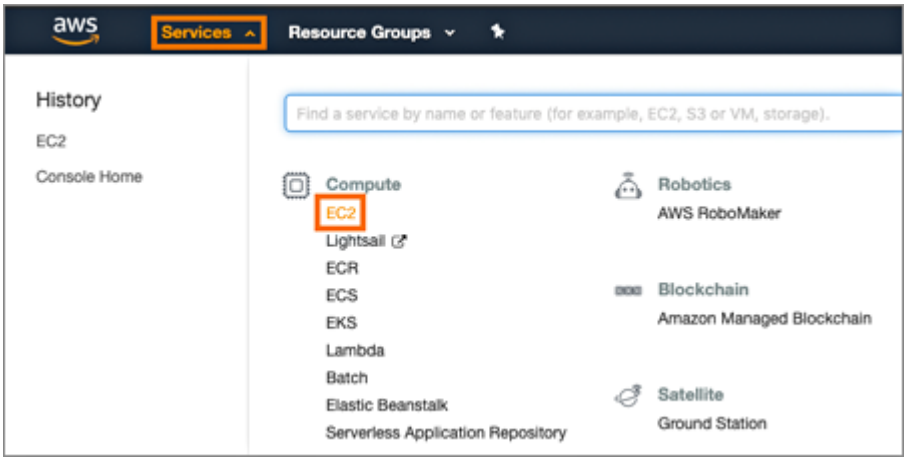
For more information on encryption supported with AWS:

- For EBS, see Amazon EBS Encryption in Amazon's AWS documentation.
- For S3, see Amazon S3 Default Encryption for S3 Buckets in Amazon's AWS documentation.

Setting up your ThoughtSpot cluster in AWS

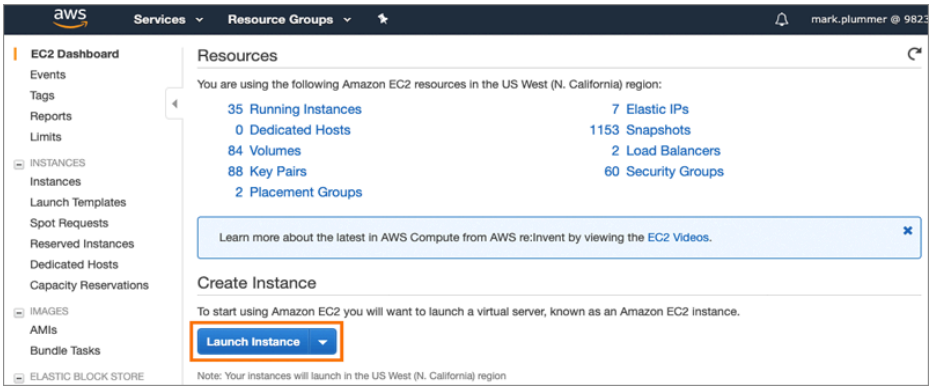
To set up a ThoughtSpot cluster in AWS, do the following:

1. In AWS, navigate to the EC2 service dashboard by clicking **Services**, then **EC2**.



2. Make sure your selected region is correct in the top-right corner of the dashboard. If not, select a different region you would like to launch your instance in. Let ThoughtSpot support know if you change your region.

3. Start the process of launching a VM by clicking **Launch Instance**.



4. Click the **My AMIs** tab, find the ThoughtSpot AMI from the list, and click **Select**.

5. On the Choose an Instance Type page, select a ThoughtSpot-supported instance type. (See ThoughtSpot AWS instance types.)

6. Click **Next: Configure Instance Details**.

7. Configure the instances by choosing the number of EC2 instances you need. The instances must be on the same VPC and subnetwork. ThoughtSpot will set up the instances to be in the same ThoughtSpot cluster.

   S3 storage setting: If you are going to use the S3 storage option, you must go to the **IAM role** menu and select **ec2rolewithfulls3access**. This setting gives your instance access to all S3 buckets in your account's region. If you want to restrict the access to a specific bucket, you must create a new IAM role that provides access to the specific bucket, and select it instead. For details on that, click **Create new IAM role**.

8. Click **Next: Add Storage**. Add the required storage based on the storage requirements of the instance type you have selected, and the amount of data you are deploying. For specific storage requirements, refer to ThoughtSpot AWS instance types.

9. When you are done modifying the storage size, Click **Next: Add Tags**.

10. Set a name for tagging your instances and click **Next: Configure Security Group**.

11. Select an existing security group to attach new security groups to so that it meets the security requirements for ThoughtSpot.

    ☑ Tip: Security setting for ThoughtSpot

    - The VMs need intragroup security, i.e. every VM in a cluster must be accessible from one another. For easier configuration, ThoughtSpot recommends that you enable full access between VMs in a cluster.
    - Additionally, more ports must be opened on the VM to provide data staging capabilities to your network. Check Network policies to determine the minimum required ports that must be opened for your ThoughtSpot appliance.

12. Click **Review and Launch**. After you have reviewed your instance launch details, click **Launch**.

13. Choose a key pair. A key pair consists of a public and private key used to encrypt and decrypt login information. If you don't have a key pair, you must create one, otherwise you won't be able to SSH into the AWS instance later on.

14. Click **Launch Instances**. Wait a few minutes for it to fully start up. After it starts, it will appear on the EC2 console.

Prepare the VMs (ThoughtSpot Systems Reliability Team)

⚠ **Important:** This procedure is typically done by a ThoughtSpot Systems Reliability Engineer (SRE). Please consult with your ThoughtSpot Customer Service or Support Engineer on these steps.

Before we can install a ThoughtSpot cluster, an administrator must log into each VM through SSH as user "admin", and complete the following preparation steps:

1. Run `sudo /usr/local/scaligent/bin/prepare_disks.sh` on every machine.
2. Configure each VM based on the site-survey.

When complete, your storage is mounted and ready for use with your cluster.

Launch the cluster

Upload the TS tarball to one of the VMs and proceed with the normal cluster creation process, using tscli cluster create.

If you are going to use S3 as your persistent storage, you must enable it when running this command, using the **enable_cloud_storage** flag. Example: `tscli cluster create 6.0-167.tar.gz --enable_cloud_storage=s3a`

When the setup is complete, you can load data into ThoughtSpot for search analytics.

Open the required network ports

To determine which network ports to open for a functional ThoughtSpot cluster, see Network policies.

Related information

EC2 Best Practices

Loading data from an AWS S3 bucket

Set up high availability for AWS

Summary: This article explains how to set up High Availability (HA) for your ThoughtSpot cluster using the AWS Elastic File System (EFS)

Configure high availabiity

To set up High Availability (HA) for your ThoughtSpot cluster using the AWS Elastic File System (EFS), follow these steps:

1. Create an EFS File System that spans across different availability zones, and across different subnets.

2. Create two ThoughtSpot clusters in each availability zone and in the subnets, where the file system was created.

3. Change the IP addresses of the cluster, if necessary.

4. Create an EFS directory in the `/home/admin` path, and issue the following command to mount the previously created file system.

   Modify the fields as necessary for your installation.

   ```
   sudo mount -t nfs -o nfsvers=4.1,rsize=1048576,wsize=104
   8576,hard,timeo=600,retrans=2,
                        noresvport fs-f756f1ee.efs.us-wes
   t-1.amazonaws.com:/ /home/admin/efs/
   ```

   To ensure that all clusters with EFS mount points have read and write permissions, modify permissions:

   ```
   chmod 777 /home/admin/efs
   ```

5. On the first cluster, create a snapshot on to the EFS mount point, and backup it.

```
tscli snapshot create EfsTest HA 2
tscli backup create --mode full --type full
                    --storage_type local EfsTest /home/a
dmin/efs/Efs-backup
```

6. Ensure that the backup is successful, and that it can be accessed from all clusters where EFS is mounted.

7. Take down the first cluster instances.

8. On the second cluster, delete the existing cluster, and create a new one by restoring from the first cluster backup. This is accessible from the EFS mount point.

```
tscli cluster restore /home/admin/EFS/Efs-backup
```

Your cluster should now be successfully restored to the second cluster from the backup on the EFS, achieving HA for ThoughtSpot clusters.

Replace a cluster

For information on how to recover from infrastructure failure scenarios, see: Cluster replacement.

Note: At this time, ThoughtSpot does not support AWS Auto Scaling or deployment across AWS availability zones or regions.