



---

# ThoughtSpot Deployment Guide

*Release 6.0*

*February, 2020*

© COPYRIGHT 2015, 2020 THOUGHTSPOT, INC. ALL RIGHTS RESERVED.

910 Hermosa Court, Sunnyvale, California 94085

This document may not, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior consent in writing from ThoughtSpot, Inc.

All rights reserved. The ThoughtSpot products and related documentation are protected by U.S. and international copyright and intellectual property laws. ThoughtSpot and the ThoughtSpot logo are trademarks of ThoughtSpot, Inc. in the United States and certain other jurisdictions. ThoughtSpot, Inc. also uses numerous other registered and unregistered trademarks to identify its goods and services worldwide. All other marks used herein are the trademarks of their respective owners, and ThoughtSpot, Inc. claims no ownership in such marks.

Every effort was made to ensure the accuracy of this document. However, ThoughtSpot, Inc., makes no warranties with respect to this document and disclaims any implied warranties of merchantability and fitness for a particular purpose. ThoughtSpot, Inc. shall not be liable for any error or for incidental or consequential damages in connection with the furnishing, performance, or use of this document or examples herein. The information in this document is subject to change without notice.

# Table of Contents

Deploying on the Dell appliance .....	4
Prerequisites.....	5
Hardware requirements .....	6
Connect the appliance.....	9
Configure management settings .....	12
Configure nodes .....	15
Install cluster .....	18
The nodes.config file.....	27
Parameters of the nodes.config file .....	30
Using the cluster create command .....	33
Parameters of the cluster create command.....	41
Network policies.....	44
Cable reference .....	55
Deploying on the SMC appliance.....	60
Prerequisites.....	61
Hardware requirements .....	63
Connect the appliance.....	66
Configure nodes .....	70
Install cluster .....	75
The nodes.config file.....	27
Parameters of the nodes.config file .....	30
Using the cluster create command .....	33
Parameters of the cluster create command.....	41
Network policies.....	44
Cable reference .....	55
Overview.....	116
Hardware appliance	
Hardware appliance overview.....	117

<b>Deploying on the SMC appliance</b>	
<b>Deploying on the SMC appliance .....</b>	<b>60</b>
Prerequisites.....	61
Hardware requirements.....	63
Connect the appliance.....	66
Configure nodes.....	70
Install cluster .....	75
<b>Deploying on the Dell appliance</b>	
<b>Deploying on the Dell appliance .....</b>	<b>4</b>
Prerequisites.....	5
Hardware requirements.....	6
Connect the appliance.....	9
Configure management settings .....	12
Configure nodes.....	15
Install cluster .....	18
<b>Cloud</b>	
<b>Cloud overview .....</b>	<b>166</b>
<b>AWS</b>	
<b>Configuration options .....</b>	<b>169</b>
Set up AWS resources for ThoughtSpot.....	172
Prepare AWS VMs for ThoughtSpot.....	180
Configure ThoughtSpot nodes in AWS .....	182
Install ThoughtSpot clusters in AWS.....	189
Set up high availability.....	197
<b>Azure</b>	
<b>Configuration options .....</b>	<b>199</b>
Set up ThoughtSpot in Azure .....	200
Configure ThoughtSpot nodes in Azure.....	210
Install ThoughtSpot clusters in Azure .....	217
<b>GCP</b>	
<b>Configuration options .....</b>	<b>226</b>
Set up ThoughtSpot in GCP .....	228
Configure ThoughtSpot nodes in GCP .....	240
Install ThoughtSpot clusters in GCP .....	247
<b>VMware</b>	
<b>Configuration overview.....</b>	<b>255</b>

<b>Set up ThoughtSpot in VMware .....</b>	<b>258</b>
<b>Configure ThoughtSpot nodes in VMware.....</b>	<b>264</b>
<b>Install ThoughtSpot clusters in VMware .....</b>	<b>271</b>
<b>Network policies.....</b>	<b>44</b>
<b>Contact support.....</b>	<b>290</b>

# Deploying on the Dell appliance

**Summary:** Follow these steps to deploy ThoughtSpot on your Dell appliance.

Follow the steps in this checklist to deploy ThoughtSpot on your Dell appliance.

- Step 1: Complete installation prerequisites [See page 5]
- Step 2: Review hardware requirements [See page 6]
- Step 3: Connect your appliance [See page 9]
- Step 4: Configure management settings [See page 12]
- Step 5: Configure nodes [See page 15]
- Step 6: Install cluster [See page 18]

## Related information

Use these references to aid you in successful installation and administration of ThoughtSpot.

- the `nodes.config` file [See page 0]
- Parameters of the `nodes.config` file [See page 30]
- Using the `tscli cluster create` command [See page 33]
- Parameters of the `cluster create` command [See page 41]
- Cable Reference [See page 55]
- ThoughtSpot Documentation [See page 0]
- Contact Support [See page 290]

# Prerequisites

**Summary:** Complete these prerequisites to deploy ThoughtSpot on your Dell appliance.

## Installation prerequisites

Ensure that you have the following items, information, and understanding of policies before you begin installing your Dell 6420 appliance.

- 10gbE switch with IPv6 broadcast and multicast enabled. You need one switch for each cluster, with one port for each node on the cluster.
- Data center with proper environment controls, such as cooling.
- AC power
- 10G connection: SFP+ for the switch side
- 10GbE network cables, either direct attach copper (DAC) or fiber. See [Cable Reference \[See page 55\]](#) for more information to decide between the two types.
- 100Mbps or 1Gbps switch for connection to the iDRAC (Out of Band Management) port. You need one for each node in your cluster.
- Cat5 network cables, for iDRAC/ management port use. You need one for each node.
- Rack space of 2U or 3.5 inches for each appliance, and a power strip
- VGA Monitor and USB keyboard
- Networking information: IP addresses for data & management NICs, for up to 2 DNS servers, up to 4 NTP servers and for the default gateway. Ensure that you only configure two DNS servers. ThoughtSpot does not support configuration of three DNS servers. You also need to know the timezone for your cluster. Typically, your timezone is where most of the people who will use the product are. [Download \[See page 0\]](#) and fill out the ThoughtSpot site survey so that you have a quick reference. Contact your network administrator if you need help filling out the site survey.

## Review hardware requirements

Next, [review hardware requirements. \[See page 6\]](#)

# Hardware Requirements

**Summary:** Learn about the Dell hardware before deploying ThoughtSpot.

## About the Hardware

This table shows the hardware details and power requirements for your Dell C6420 appliance.

Details	Dell C6420
<b>Dimensions</b>	2 RU chassis (17.6" x 3.4" x 31.1" (WxHxD))
<b># of nodes</b>	Populated with 1 to 4 nodes
<b>Node specifications</b>	Each node is independent and consists of a server board (removable from rear), 1x 240GB SSD, 3x 2TB HDD
<b>Max power consumption</b>	2000 W
<b>Required power input</b>	200-240VAC / 11.5A max / 50 - 60 Hz (C19 / C20 power cords)

## Dell front and back views

These pictures show the front and back view of the Dell C6420 appliance.

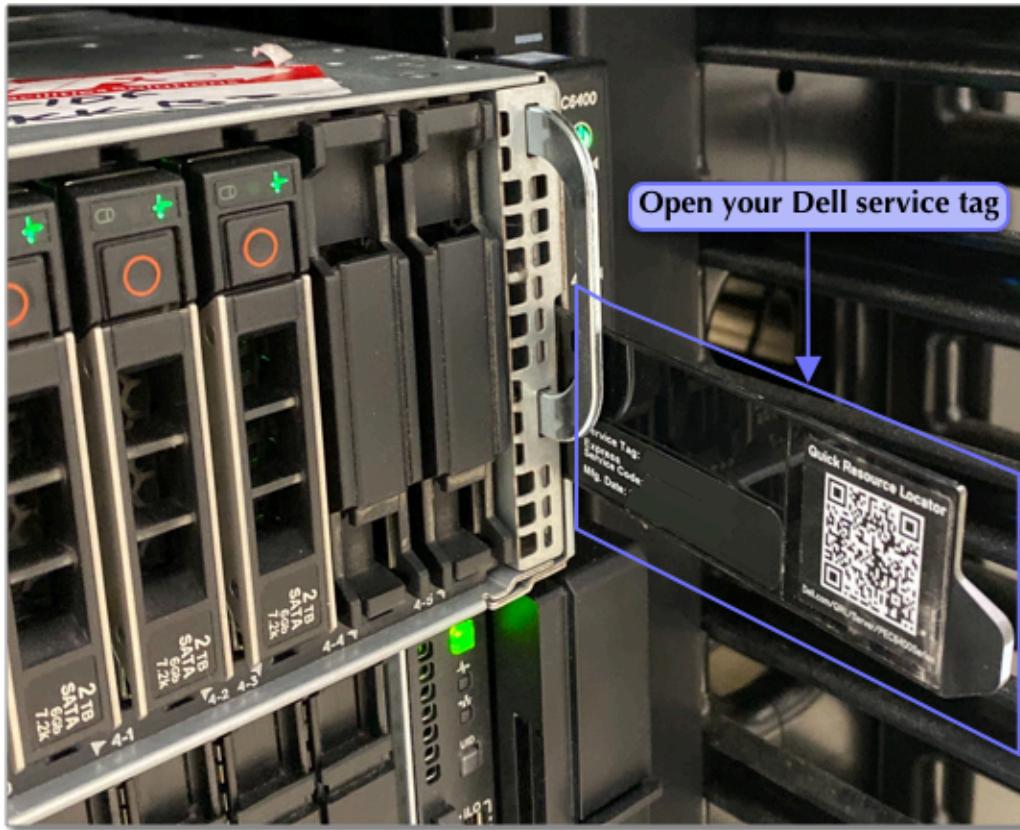




## Location of serial number

You may need to know your appliance's serial number, to be able to access online help from your appliance provider. Find your Dell appliance's serial number on the plastic tab next to the control panel for Node 4, on the front right side of your appliance. Pull out the tab to view your service tag and a QR code for Dell support.





## Connect appliance

Next, [connect the appliance](#). [See page 9]

# Connect the Appliance

**Summary:** Connect your Dell appliance before you can deploy ThoughtSpot.

After you rack and stack the appliance, it is time to configure it. Follow the steps in this checklist.

- Step 1: Connect switches to 10GbE ports [See page 0]
- Step 2: Connect iDRAC ports [See page 0]
- Step 3: Connect a keyboard and monitor [See page 0]
- Step 4: Turn on nodes [See page 0]

## Step 1: Connect switches to 10GbE ports

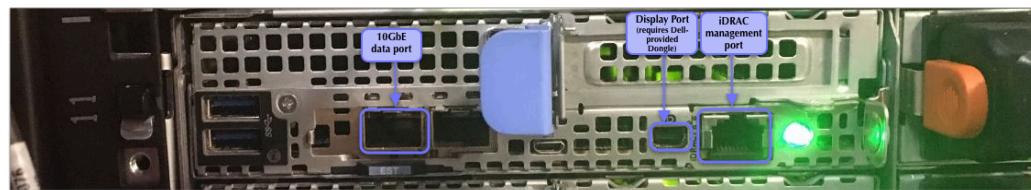
Connect the 10GbE port of each node, as illustrated in [Dell Port Location](#) [See page 9], to the 10GbE switches on your own rack, using either fiber or DAC cables.

Refer to the [Cable reference](#) [See page 55] for information on the cable types:

- [Fiber Cables](#) [See page 55]
- [DAC Cables](#) [See page 58]

**Note:** Ask your hardware vendor for more details about what they supply and what you need to buy.

The ports are on the back of the Dell appliance.



- Connect to switches **only** the appliances (4 nodes each) that you plan to use in your cluster.

## Connect the Appliance

- You must power off, or disconnect from the switch, any other nodes or appliances. This prevents accidental configuration of incorrect nodes.
- Connect all the nodes that you plan to use in your cluster to a 10G switch.

**Note:** You need at least three nodes for high availability (HA). Each appliance can have up to four nodes.

## Step 2: Connect iDRAC ports

Connect the iDRAC management ports of each node to the management switch.

See [Dell Port Location \[See page 9\]](#).

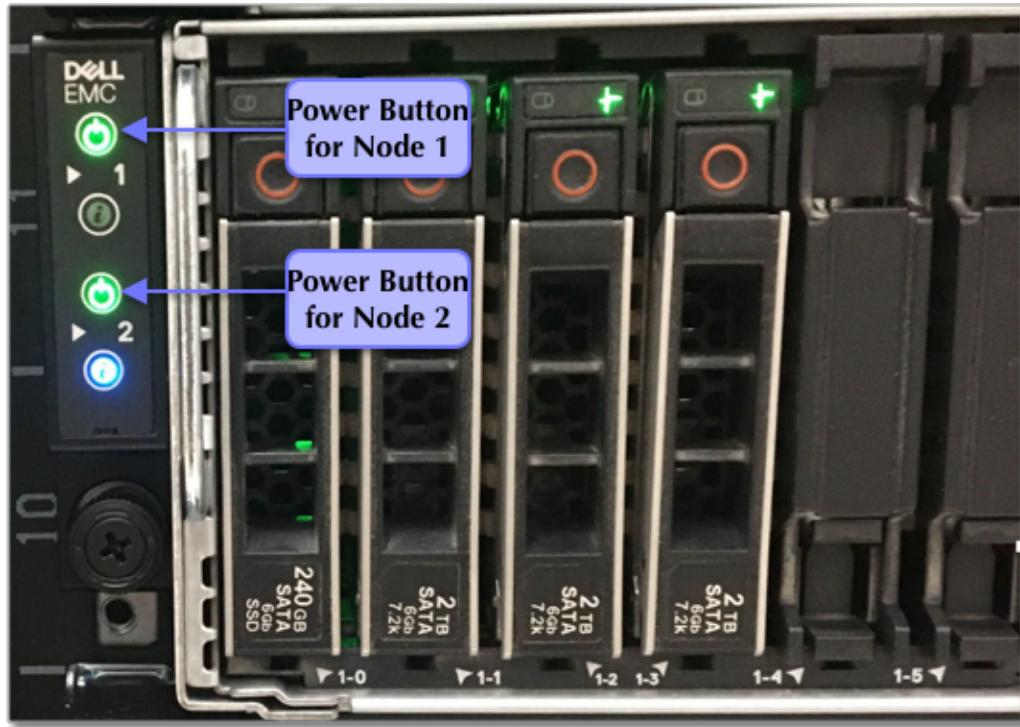
## Step 3: Connect a keyboard and monitor

Connect a keyboard and monitor to the appliance. You need these to initially configure the appliance, and you can disconnect them later. Use the adapter Dell provides. Plug it into the Display Port shown in [Dell Port Location \[See page 9\]](#), and plug the monitor in on the other side of the adapter.



## Step 4: Turn on nodes

Turn on power for the nodes by pressing the power button for each one; see [Dell Power Buttons \[See page 11\]](#).



There is one power button for each node. The image above shows two of the four power buttons.

## Configure the management settings

Next, [configure the management settings. \[See page 12\]](#)

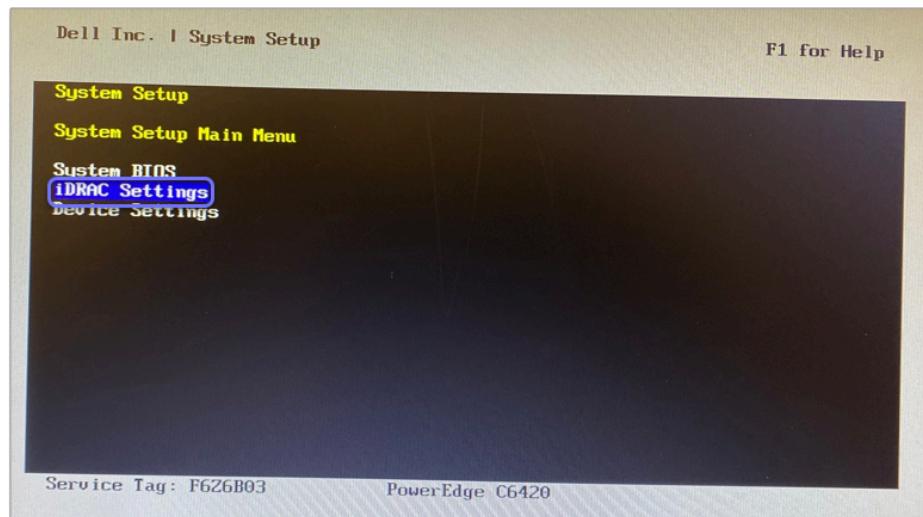
# Configure the Dell Management Settings

**Summary:** Configure the management settings for Dell before you can deploy ThoughtSpot.

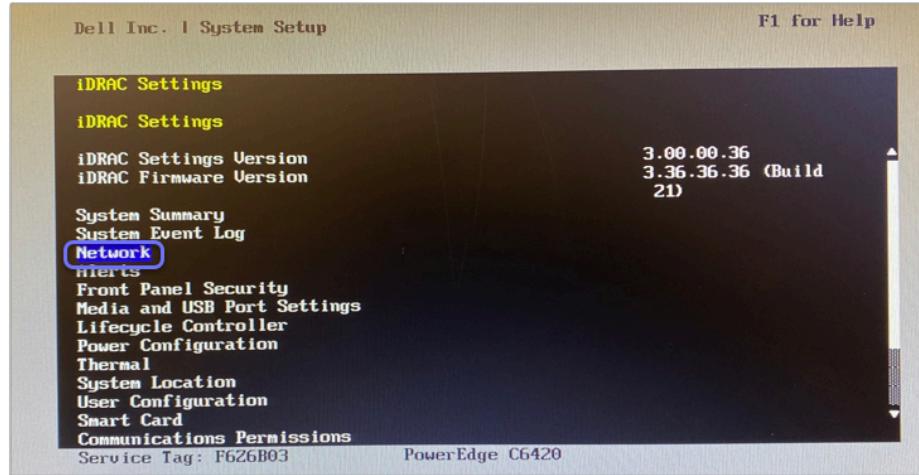
Input your specific network information to configure the management settings for your Dell appliance.

Refer to [Dell Management Configuration \[See page 14\]](#). If you need additional guidance, view [Dell Support \[See page 0\]](#) for this product.

1. **Open the iDRAC settings modal** Before the node boots, a screen appears on your monitor with several options. Click F2 to open the Bios setup menu.
2. **Select iDRAC** In the Bios setup screen, there are several options. Select **iDRAC settings** to configure your iDRAC management settings.



3. **Select network configuration** From the iDRAC settings options, select **network**.



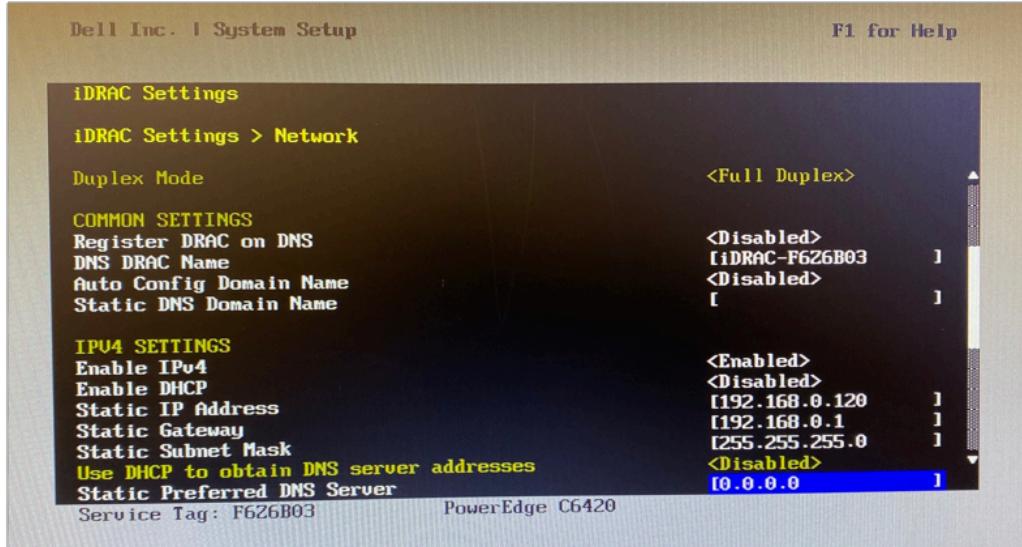
4. **Fill out the iDRAC settings form** Add your specific network information for the IP address, Gateway, and Netmask in the empty boxes. DNS information is optional. [Download \[See page 0\]](#) and fill out the ThoughtSpot site survey for a quick reference, and ask your network administrator for help if you have not filled out the site survey yet.

**● Warning:** If you configure DNS servers, you must only configure two servers.

ThoughtSpot does not support configuration of three DNS servers.

- For **Enable IPv4**, select **enabled**.
- For **Enable DHCP**, select **disabled**.

5. **Save changes and reboot** Follow the prompts on the monitor to save changes to the management settings form, exit, and reboot the system.
6. **Log in to ThoughtSpot** After the system reboots, the login page appears. Log in as an administrator. Ask your network administrator if you do not know the admin credentials.



## Configure nodes

Next, configure nodes. [See page 15]

# Configure Nodes

**Summary:** Configure ThoughtSpot nodes on your Dell appliance.

After you connect the appliance, a command line appears on your console. Configure the nodes on this command line. Follow the steps in this checklist.

- Step 1: Get a template for network configuration [See page 0]
- Step 2: Prepare node configuration [See page 0]
- Step 3: Configure the nodes [See page 0]
- Step 4: Confirm node configuration [See page 0]

## Step 1: Get a template for network configuration

Make sure you have logged into your cluster. If you have not, use admin credentials to log in to your cluster.

Run the `tscli cluster get-config` command to get a template for network configuration. Redirect it to the file `nodes.config`.

You can find more information on this process in the `nodes.config` file reference [See page 27].

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

## Step 2: Prepare node configuration

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the [autodiscovery of one node example \[See page 29\]](#). Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information.

If you have additional nodes, complete each node within the `nodes.config` file in the same way.

Do not edit any part of the `nodes.config` file except the sections described in [Parameters of the nodes.config file \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, it may cause setup to fail.

## Step 3: Configure the nodes

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

Run `$ cat nodes.config | tscli cluster set-config` in your terminal.

If the command returns an error, refer to [set-config error recovery \[See page 0\]](#).

```
$ 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 NTP Servers
Setting up Timezone
Done setting up ThoughtSpot
```

## Step 4: Confirm node configuration

Use the `get-config` command to confirm node configuration.

Your output may look similar to the following:

```
$ 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 your cluster

Next, [install your cluster. \[See page 18\]](#)

# Install Cluster

**Summary:** Install your ThoughtSpot cluster(s) on your Dell appliance.

Install the cluster using the ThoughtSpot software release bundle. Installation takes approximately one hour. Make sure you can connect to ThoughtSpot remotely. If you can, you can run the installer on your local computer.

Refer to your welcome letter from ThoughtSpot to find the link to download the release bundle. If you do not have a link to download the release bundle, open a support ticket at [ThoughtSpot Support \[See page 290\]](#) to access the release bundle.

Follow the steps in this checklist to install your cluster.

- Step 1: Run the installer [See page 0]
- Step 2: Check cluster health [See page 0]
- Step 3: Finalize installation [See page 0]

## Step 1: Run the installer

1. Copy the downloaded release bundle to `/export/sdb1/TS_TASKS/install`:

Run `scp <release-number> admin@<hostname>:/export/sdb1/TS_TASKS/install/<file-name>`. Note the following parameters:

- `release-number` is the release number of your ThoughtSpot installation, such as `6.0`, `5.3`, `5.3.1`, and so on.
- `hostname` is your network hostname. Ask your network administrator if you do not know your hostname.
- `file-name` is the name of the tarball file on your local computer.

```
$ scp <release-number>.tar.gz admin@<hostname>:/e  
xport/sdb1/TS_TASKS/install/<file-name>
```

**Note:** You can use another secure copy method, if you prefer a method other than the `scp` command.

2. Alternatively, use `tscli fileserver download-release` to download the release bundle.

You must [configure the fileserver \[See page 0\]](#) by running `tscli fileserver configure` before you can download the release.

```
$ tscli fileserver download-release <release-number> --user <username> --out <release-location>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 5.3.1, 6.0, and so on.
- `username` is the username for the fileserver that you set up earlier, when configuring the fileserver.
- `release-location` is the location path of the release bundle on your local machine. For example, `/export/sdb1/TS_TASKS/install/6.0.tar.gz`.

3. Verify the checksum to ensure you have the correct release.

Run `md5sum -c <release-number>.tar.gz.MD5checksum`.

```
$ md5sum -c <release-number>.tar.gz.MD5checksum
```

Your output says `ok` if you have the correct release.

4. Launch a [screen \[See page 0\]](#) session. Use screen to ensure that your installation does not stop if you lose network connectivity.

```
$ screen -S DEPLOYMENT
```

5. Create the cluster using `tscli cluster create <release-number>`.

```
$ tscli cluster create <release.number>.tar.gz
```

6. Edit the output using your specific cluster information. For more information on this process, refer to [Using the tscli cluster create command \[See page 33\]](#) and [Parameters of the cluster create command \[See page 41\]](#).

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

## Step 2: Check cluster health

After you install the cluster, check its status using the `tscli cluster status` command.

Your output may look something like the following:

```
$ 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
```

Ensure that the cluster is `RUNNING` and that the Database and Search Engine are `READY`.

```
$ tscli cluster check
Connecting to hosts...
[Wed Jan  8 23:15:47 2020] START Diagnosing ssh
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing connection
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing zookeeper
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing sage
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing timezone
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing disk
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing cassandra
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing hdfs
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing orion-oreo
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing memcheck
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing ntp
[Wed Jan  8 23:16:08 2020] SUCCESS
#####
```

```
#####
[Wed Jan  8 23:16:08 2020] START Diagnosing trace_vault
[Wed Jan  8 23:16:09 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:09 2020] START Diagnosing postgres
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing disk-health
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing falcon
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing orion-cgroups
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing callosum
/usr/lib/python2.7/site-packages/urllib3/connectionpool.py:85
2: InsecureRequestWarning: Unverified HTTPS request is being ma
de. Adding certificate verification is strongly advised. See: h
ttps://urllib3.readthedocs.io/en/latest/advanced-usage.html#ss
l-warnings
    InsecureRequestWarning)
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
```

Your output may look something like the above. Ensure that all diagnostics show `SUCCESS`.

**● Warning:** If `tscli cluster check` returns an error, it may suggest you run `tscli storage gc` to resolve the issue. If you run `tscli storage gc`, note that it restarts your cluster.

## Step 3: Finalize installation

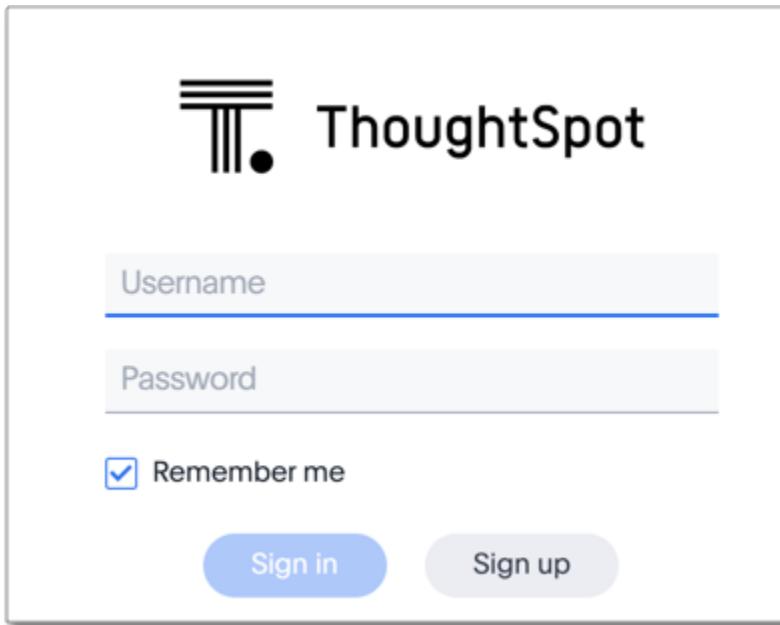
After the cluster status changes to `READY`, sign in to the ThoughtSpot application on your browser.

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 sign-in window \[See page 24\]](#), enter admin credentials, and click **Sign in**.  
ThoughtSpot recommends changing the default admin password.



## Lean configuration

**(For use with thin provisioning only)** If you have a [small or medium instance type \[See page 167\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 290\]](#) for assistance with this configuration.

## Error recovery

### Set-config error recovery

If you get a warning about node detection when you run the `set-config` command, restart the node-scout service.

Your error may look something like the following:

```
Connecting to local node-scout
WARNING: Detected 0 nodes, but found configuration for
only 1 nodes.
Continuing anyway. Error in cluster config validation: [] is not a valid link-local
IPv6 address for node: 0e:86:e2:23:8f:76 Configuration failed.
Please retry or contact support.
```

Restart the node-scout service with the following set of commands.

```
$ sudo vim /etc/systemd/system/node-scout.service
$ sudo systemctl daemon-reload
$ sudo systemctl restart node-scout
```

Ensure that you restarted the node-scout by running `sudo systemctl status node-scout`. Your output should specify that the node-scout service is active. It may look something like the following:

```
$ sudo systemctl status node-scout
● node-scout.service - Setup Node Scout service
  Loaded: loaded (/etc/systemd/system/node-scout.service; enabled; vendor preset: disabled)
  Active: active (running) since Fri 2019-12-06 13:56:29 PST; 4s ago
```

Next, retry the `set-config` command.

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

The command output should no longer have a warning.

# The nodes.config file

**Summary:** Learn how to use the get.config command and the nodes.config file to install your hardware or cloud appliance.

## Using the nodes.config file

As you install your appliance, you must configure the nodes.

1. Run the configuration command in your terminal.

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

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information, as shown in [Autodiscovery of one node example \[See page 0\]](#).

**Note:** Some of the information in the nodes.config file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your nodes.config file.

3. If you have additional nodes, complete each node within the nodes.config file as shown in the [Autodiscovery of one node example \[See page 0\]](#). [Autodiscovery of one node \[See page 0\]](#) shows the nodes.config file before you fill in your specific information.

Do not edit any part of the nodes.config file except the sections explained in [Parameters of the nodes.config file \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, setup may fail.

See [Parameters of the nodes.config file \[See page 30\]](#) to understand the parameters in the file.

### Autodiscovery of one node

```
$ tscli cluster get-config |& tee nodes.config
{
    "ClusterId": "",
    "ClusterName": "",
    "DataNetmask": "",
    "DataGateway": "",
    "IPMINetmask": "",
    "IPMIGateway": "",
    "Timezone": "",
    "NTPServers": "",
    "DNS": "",
    "SearchDomains": "",
    "Nodes": {
        "06:83:1f:f8:99:9e": {
            "NodeId": "06:83:1f:f8:99:9e",
            "Hostname": "",
            "DataIface": {
                "Name": "eth0",
                "IPv4": ""
            },
            "IPMI": {
                "IPv4": ""
            }
        }
    }
}
```

### Autodiscovery of one node example

```
$ vim nodes.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"
      }
    }
  }
}
```

## Related information

- [Deploying on the SMC appliance \[See page 60\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 182\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 240\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 264\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 210\]](#)
- [Deploying on the Dell Appliance \[See page 4\]](#)
- [Parameters of the nodes.config file \[See page 30\]](#)

# Parameters of the nodes.config file

**Summary:** Learn the parameters of the nodes.config file to install your cloud or hardware appliance.

## Using the nodes.config file

As you install your appliance, you must configure the nodes.

1. Run the configuration command in your terminal.

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

2. Add network information for your nodes in the `nodes.config` file output, as demonstrated in the [nodes.config file \[See page 27\]](#) example.

Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

Fill in the areas specified in [Parameters of nodes.config \[See page 31\]](#) with your specific network information.

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

3. If you have additional nodes, complete this process for each node.

Do not edit any part of the `nodes.config` file except the sections explained in [Parameters of nodes.config \[See page 31\]](#). If you delete quotation marks, commas, or other parts of the code, setup may fail.

See [Parameters of nodes.config \[See page 31\]](#) to understand the parameters in the file. Different hardware and cloud installations have different parameters. Your installation may not require all the listed parameters.

## Parameters of the nodes.config file

**ClusterId** and **Cluster Name**: Leave these two parameters blank. You fill them out later, when running `tscli cluster create .`

**DataNetmask** The IP of the data netmask, in the form 000.000.000.000. For example, 255.255.252.0.

**DataGateway** The IP of the data gateway, in the form 000.000.000.000. For example, 192.168.4.1.

**IPMINetmask** The IP of the Intelligent Platform Management Interface (IPMI) netmask, in the form 000.000.000.000. For example, 255.255.252.0.

**IPMIGateway** The IP of the Intelligent Platform Management Interface (IPMI) gateway, in the form 000.000.000.000. For example, 192.168.4.1.

**Timezone** The timezone the majority of your ThoughtSpot users are in, in the form Country/City. For example, America/Los\_Angeles. To find your timezone and a city you can use to identify it, use [this timezone list \[See page 0\]](#).

**NTPServers** The address of your company's Network Time Protocol (NTP) server. If your company does not have an NTP server, you can use one of ThoughtSpot's, as listed in [the nodes.config \[See page 29\]](#) example under NTPServers. For example, 0.centos.pool.ntp.org.

**DNS** The address of your company's Domain Name Server (DNS). For example, 192.168.2.200,8.8.8.8.

**⚠ Warning:** Configure only two DNS servers. ThoughtSpot does not support configuration of three DNS servers.

**⚠ Note:** You can only edit DNS settings with this command if you are deploying ThoughtSpot on hardware. ThoughtSpot does not support using `set-config` to edit your DNS settings for cloud deployment.

**SearchDomains** The domain of your company or organization, in the form example.company.com.

**Hostname** The name of the host server. For example, Thoughtspot-server1.

**IPv4** The main IP address, associated with DataNetmask and DataGateway, in the form 000.000.000.000. For example, 192.168.7.70.

**IPMI IPv4** A secondary IP address, associated with the IPMINetmask and IPMIGateway, in the form 000.000.000.000. For example, 192.168.5.70.

## Related information

- [Deploying on the SMC appliance \[See page 60\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 182\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 240\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 264\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 210\]](#)
- [Deploying on the Dell Appliance \[See page 4\]](#)
- [The nodes.config file \[See page 27\]](#)

# Using the tscli cluster create command

**Summary:** Learn how to use the cluster create command to install your appliance.

## Using the tscli cluster create command

To install your appliance, you must install the cluster using the release tarball (estimated time 1 hour).

1. **Download the release tarball** Download the release tarball from the download link sent by ThoughtSpot Support.  
Refer to your welcome letter to find a link to download the release tarball. If you do not have a download link, open a support ticket at [ThoughtSpot Support \[See page 0\]](#) to access the release tarball.
2. **Copy the release tarball** In your Terminal (Apple) or using [Winscp \[See page 0\]](#) (Windows application), copy the downloaded release tarball to `/home/admin/` and run the cluster installer as shown below.

Run the secure copy command: `scp <release-number> admin@<hostname>:/home/admin/<file-name>`. Note the following parameters:

- `release-number` is the version of ThoughtSpot you have on your cluster, such as `6.0` or `5.3`. `release-number` is of the form `0.0.tar.gz`.
- `hostname` is your network hostname. Ask your network administrator if you do not know your hostname.
- `file-name` is the name of the tarball file on your local machine. For example:

```
$ scp 6.0.tar.gz admin@ThoughtSpot:/home/admin/T  
S-tarball.
```

ThoughtSpot recommends that you start the session using [screen \[See page 0\]](#), so you do not lose your progress if you lose your connection at any point.

```
$ scp <release-number> admin@<hostname>:/home/adm  
in/<file-name>
```

3. **Run the `tscli cluster create` command** Run `tscli cluster create <release-number>` in your terminal.

If you are using an s3 or GCS bucket for object storage, include the flag `--enable_cloud_storage=s3` or `--enable_cloud_storage=gcs`. GCS is GCP's object storage, and s3 is AWS's object storage.

```
$ tscli cluster create 6.0.tar.gz --enable_cloud_storage=s3
```

```
$ tscli cluster create 6.0.tar.gz --enable_cloud_storage=gcs
```

4. **Specify your installation information** Fill out the cluster name, cluster ID, email alert preferences and the IP's of the nodes at the prompts specified in [Parameters of the tscli cluster create command \[See page 41\]](#).
5. **Wait for output** You may need to wait about 15 seconds before you see any output. The installer is unpacking files and copying them over to the nodes, which can take a few seconds.

Do not edit any part of the installer file except the sections specified in [Parameters of the tscli cluster create command \[See page 41\]](#). if you delete colons, commas, or other parts of the code, setup may fail.

Refer to [Parameters of the tscli cluster create command \[See page 41\]](#) for further information.

Your `tscli cluster create` output may look something like the following:

### Run the Installer

The output for a hardware installation:

```
[admin@jessi-gcs-test ~]$ tscli cluster create 6.0-145.tar.gz
Unpacking 6.0-145.tar.gz to /export/release_cache/e695feeecc6275
91dc644635c0d8ea03d
#####
#
#
#
#           Welcome to ThoughtSpot installer
#
#
#
#
#####
#
Enter a name for the cluster (alphanumeric characters only): jessi-ts-gcs-test
Enter cluster ID: 0x0000
Enter IP addresses of all hosts in the cluster (space separated): 10.116.0.66
Enter email addresses for alerts (space separated, "later" to skip): later
2019-11-03 21:26:47,959 Pushing /usr/local/scaligent/toolchain/
jolokia to localhost
2019-11-03 21:26:48,195 Rsync finished on localhost
2019-11-03 21:26:48,195 Rsync finished to all hosts
2019-11-03 21:26:48,195 Pushing /usr/local/scaligent/toolchain/
jvm to localhost
2019-11-03 21:26:48,461 Rsync finished on localhost
2019-11-03 21:26:48,461 Rsync finished to all hosts
2019-11-03 21:26:48,462 Pushing /usr/local/scaligent/toolchain/
hadoop to localhost
2019-11-03 21:26:48,738 Rsync finished on localhost
2019-11-03 21:26:48,738 Rsync finished to all hosts
2019-11-03 21:26:48,738 Pushing /usr/local/scaligent/toolchain/
zookeeper to localhost
2019-11-03 21:26:49,004 Rsync finished on localhost
2019-11-03 21:26:49,004 Rsync finished to all hosts
#####
#
Setup Hadoop
#####
#
Installing Zookeeper
First deleting existing Zookeeper service
Deleting Zookeeper
Zookeeper Deleted
```

```
Starting Zookeeper servers
Zookeeper Ready
Installing HDFS
First deleting existing HDFS service
Deleting HDFS
HDFS Deleted
Deploying configs
Formatting and starting Primary Name Node
Starting Data Nodes
Setting NFS configs
Waiting for HDFS
HDFS Ready
2019-11-03 21:27:06,212 Starting a secondary namenode for check
point...
2019-11-03 21:27:06,551 Cannot disable unknown service: hdfs_se
condary_namenode
2019-11-03 21:27:06,552 Failed to disable new service: hdfs_sec
ondary_namenode
2019-11-03 21:27:07,198 HDFS secondary namenode started success
fully.
Successfully Created zookeeper znodes
Successfully initialized HDFS files
Successfully saved hadoop layout proto
Successfully moved the release dir to export partition
Successfully installed orion on localhost
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Couldn't install application services
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Couldn't install application services
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Successfully installed application services
All nodes are now healthy
Restarted all services
Successfully pushed statsdb
Successfully refreshed alert metadata
Successfully enabled alerts
```

```
Successfully configured new ssh keys on cluster
Successfully recorded install event
Successfully deleted deployer service
2019-11-03 21:40:02,917 Cannot check for enabled status of unknown service: deploy
```

The output for a cloud installation, with GCS object storage:

```
admin@jessi-gcs-test ~]$ tscli cluster create ./6.0-145.tar.gz --enable_cloud_storage gcs
Unpacking ./6.0-145.tar.gz to /export/release_cache/e695feeeec627591dc644635c0d8ea03d
#####
#
#
#
#           Welcome to ThoughtSpot installer
#
#
#
#####
#
Enter a name for the cluster (alphanumeric characters only): jessi-ts-gcs-test
Enter cluster ID: 0x0000
Enter IP addresses of all hosts in the cluster (space separated): 10.116.0.66
Enter email addresses for alerts (space separated, "later" to skip): later
Enter the GCS bucket to be configured for the cluster: jessi-gcs-test-bucket
Bucket successfully validated.
2019-11-03 22:00:29,175 Pushing /usr/local/scaligent/toolchain/jolokia to localhost
2019-11-03 22:00:29,411 Rsync finished on localhost
2019-11-03 22:00:29,411 Rsync finished to all hosts
2019-11-03 22:00:29,412 Pushing /usr/local/scaligent/toolchain/jvm to localhost
2019-11-03 22:00:29,678 Rsync finished on localhost
2019-11-03 22:00:29,678 Rsync finished to all hosts
2019-11-03 22:00:29,678 Pushing /usr/local/scaligent/toolchain/hadoop to localhost
2019-11-03 22:00:29,945 Rsync finished on localhost
2019-11-03 22:00:29,945 Rsync finished to all hosts
2019-11-03 22:00:29,945 Pushing /usr/local/scaligent/toolchain/zookeeper to localhost
2019-11-03 22:00:30,211 Rsync finished on localhost
2019-11-03 22:00:30,211 Rsync finished to all hosts
#####
#
#           Setup Hadoop
#####
#
```

```
Installing Zookeeper
First deleting existing Zookeeper service
Deleting Zookeeper
Zookeeper Deleted
Starting Zookeeper servers
Zookeeper Ready
Installing HDFS
First deleting existing HDFS service
Deleting HDFS
HDFS Deleted
Deploying configs
Formatting and starting Primary Name Node
Starting Data Nodes
Setting NFS configs
Waiting for HDFS
HDFS Ready
2019-11-03 22:00:47,287 Starting a secondary namenode for check
point...
2019-11-03 22:00:47,629 Cannot disable unknown service: hdfs_se
condary_namenode
2019-11-03 22:00:47,630 Failed to disable new service: hdfs_sec
ondary_namenode
2019-11-03 22:00:48,282 HDFS secondary namenode started success
fully.
Successfully Created zookeeper znodes
Successfully initialized HDFS files
Successfully saved hadoop layout proto
Successfully moved the release dir to export partition
Successfully installed orion on localhost
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Couldn't install application services
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Couldn't install application services
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Successfully installed application services
All nodes are now healthy
```

```
Restarted all services
Successfully pushed statsdb
Successfully refreshed alert metadata
Successfully enabled alerts
Successfully configured new ssh keys on cluster
Successfully recorded install event
Successfully deleted deployer service
2019-11-03 22:11:54,571 Cannot check for enabled status of unknown service: deploy
```

## Related information

- [Deploying on the SMC appliance \[See page 60\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 182\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 240\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 264\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 210\]](#)
- [Deploying on the Dell Appliance \[See page 4\]](#)
- [Parameters of the tscli cluster create command \[See page 41\]](#)

# Parameters of the tscli cluster create command

**Summary:** Learn the parameters of the tscli cluster create command.

## Using the tscli cluster create command

To install your appliance, you must install the cluster.

1. **Copy the release tarball** In your Terminal (Apple) or using [Winscp \[See page 0\]](#) (Windows application), copy the downloaded release tarball to /home/admin/ and run the cluster installer as shown below.

Run the secure copy command: `scp <release-number> admin@<hostname>:/home/admin/<file-name>`. Note the following parameters:

- `release-number` is the version of ThoughtSpot you have on your cluster, such as `6.0` or `5.3`. `release-number` is of the form `0.0.tar.gz`.
- `hostname` is your network hostname. Ask your network administrator if you do not know your hostname.
- `file-name` is the name of the tarball file on your local machine. For example:

```
$ scp 6.0.tar.gz admin@ThoughtSpot:/home/admin/T  
S-tarball.
```

ThoughtSpot recommends that you start the session using [screen \[See page 0\]](#), so you do not lose your progress if you lose your connection at any point.

```
$ scp <release-number> admin@<hostname>:/home/ad  
min/<file-name>
```

2. **Run the cluster create command** Run `tscli cluster create <release-number>` in your terminal.

If you are using an s3 or GCS bucket for object storage, include the flag `--enable_cloud_storage=s3` or `--enable_cloud_storage=gcs`. GCS is GCP's object

storage, and s3 is AWS's object storage.

```
$ tscli cluster create 6.0.tar.gz --enable_cloud_storage=s3
```

```
$ tscli cluster create 6.0.tar.gz --enable_cloud_storage=gcs
```

3. **Fill out your specific information** Fill out the cluster name, cluster ID, email alert preferences and the IP's of the nodes at the prompts specified in [Parameters of ThoughtSpot Installer \[See page 42\]](#) below.

Refer to [Parameters of ThoughtSpot Installer \[See page 42\]](#) for further information. Refer to [Using the tscli cluster create command \[See page 33\]](#) to see the expected output of the install command, `tscli cluster create`.

Do not edit any part of the installer file except the sections specified in [Parameters of ThoughtSpot Installer \[See page 42\]](#). If you delete colons, commas, or other parts of the code, setup may fail.

## Parameters of ThoughtSpot Installer

**Cluster Name** Name your cluster based on the ThoughtSpot naming convention, in the form *company-clustertype-location-clusternumber*. For example, *ThoughtSpot-prod-Sunnyvale-12*.

**Cluster ID** Enter the ID of your cluster that [ThoughtSpot Support \[See page 290\]](#) provided for you. Open a ticket with ThoughtSpot Support if you do not have an ID.

**Host IPs** Enter the IP addresses of all cluster hosts, in the form 000.000.000.000. For example, 192.168.7.70. Use spaces instead of commas to separate multiple IP addresses.

**Email alerts** Enter the email addresses you would like to receive alerts about this cluster, in the form company@example.com. The address prod-alerts@thoughtspot.com appears automatically and should remain, so that ThoughtSpot can be aware of the status of your cluster. Separate email addresses using a space.

## Related information

- Deploying on the SMC appliance [See page 60]
- Configure ThoughtSpot nodes in AWS [See page 182]
- Configure ThoughtSpot nodes in GCP [See page 240]
- Configure ThoughtSpot nodes in VMware [See page 264]
- Configure ThoughtSpot nodes in Azure [See page 210]
- Deploying on the Dell Appliance [See page 4]
- Using the tscli cluster create command [See page 33]

# Network policies

**Summary:** Lists the required and optional ports for an installation.

For regular operations and for debugging, there are some ports you must keep open to network traffic from end users. Another, larger list of ports must be kept open for network traffic between the nodes in the cluster.

## Required ports for operations and debugging

The following ports must be open for requests from your user population.

Port	Protocol	Service Name	Direction	Source	Destination	Description
22	SSH	SSH	bidirectional	Administrators IP addresses	All nodes	Secure shell access. Also used for scp (secure copy).
443	HTTPS	HTTPS	bidirectional	All users IP addresses	All nodes	Secure HTTP.
12345	TCP	Simba	bidirectional	Administrators IP addresses	All nodes	Port used by ODBC and JDBC drivers when connecting to ThoughtSpot.

## Network Ports

This reference lists the potential ports to open when setting up your security group.

### Required ports for intracluster operation

Static ports are used for communication between services within the cluster. ThoughtSpot recommends that you open all ports within a cluster. This is not required, but it will ensure that cluster communication works properly if additional ports are used in a future software release.

If your organization does not allow you to open all ports, make sure you open the required intrACLuster ports listed in the following table. In addition, a number of ports are dynamically assigned to services, which change between runs. The dynamic ports come from the range of ports that are dynamically allocated by Linux (20K+).

Port	Protocol	Service Name	Direction	Source	Dest.	Description
80	TCP	nginx	inbound	All nodes	All nodes	Primary app HTTP port (nginx)
443	TCP	Secure nginx	inbound	All nodes	All nodes	Primary app HTTPS port (nginx)
2100	RPC	Oreo RPC port	bidirectional	All nodes	All nodes	Node daemon RPC
2101	HTTP	Oreo HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Node daemon HTTP
2181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
3181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
4181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
2200	RPC	Orion master RPC port	bidirectional	All nodes	All nodes	Internal communication with the cluster manager
2201	HTTP	Orion master HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the cluster manager

Port	Protocol	Service Name	Direction	Source	Dest.	Description
2205	TCP	Cluster update service TCP port	bidirectional	All nodes	All nodes	Internal communication with the cluster manager
2210	RPC	Cluster stats service RPC port	bidirectional	All nodes	All nodes	Internal communication with the stats collector
2211	HTTP	Cluster stats service HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the stats collector
2230	RPC	Callosum stats collector RPC port	bidirectional	All nodes	All nodes	Internal communication with the BI stats collector
2231	HTTP	Callosum stats collector HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the BI stats collector
2240	RPC	Alert manager	bidirectional	All nodes	All nodes	Port where alerting service receives alert events
2241	RPC	Alert manager	bidirectional	All nodes	All nodes	Port where alerting service receives alert events
2888	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
3181	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Dest.</b>	<b>Description</b>
3888	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
4000	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4001	HTTP	Falcon worker HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the data cache
4002	HTTP	Falcon worker HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the data cache
4003	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4004	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4021	RPC	Sage metadata service port (exported by Tomcat), Callosum services like meta-data services, medata-dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata

Port	Protocol	Service Name	Direction	Source	Dest.	Description
4181	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
4201	HTTP	Sage auto complete server HTTP interface port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4231	HTTP	Sage index server HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4232	RPC	Sage index server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4233	RPC	Sage index server RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4241	HTTP	Sage auto complete server HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4242	RPC	Sage auto complete server RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4243	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4244	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4245	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication

Port	Protocol	Service Name	Direction	Source	Dest.	Description
4243	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4251	RPC	Sage master RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4405	RPC	Diamond (graphite) port	bidirectional	All nodes	All nodes	Port used for communication with monitoring service
4406	RPC	Diamond (graphite) port	bidirectional	All nodes	All nodes	Port used for communication with monitoring service
4500	RPC	Trace vault service RPC port	bidirectional	All nodes	All nodes	Trace collection for ThoughtSpot services
4501	HTTP	Trace vault service HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug trace collection
4851	RPC	Graphite manager RPC port	bidirectional	All nodes	All nodes	Communication with graphite manager
4852	HTTP	Graphite manager HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug graphite manager
4853	RPC	Elastic search stack (ELK) manager RPC port	bidirectional	All nodes	All nodes	Communication with log search service
4853	HTTP	Elastic search stack (ELK) manager HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug log search service
9200	RPC	Elastic search (ELK)	bidirectional	All nodes	All nodes	Communication with log search service

Port	Protocol	Service Name	Direction	Source	Dest.	Description
5021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
5432	Postgres	Postgres database server port	bidirectional	All nodes	All nodes	Communication with Postgres database
6021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
7021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
8020	RPC	HDFS namenode server RPC port	bidirectional	All nodes	All nodes	Distributed file system (DFS) communication with clients
8021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
8080	HTTP	Tomcat	bidirectional	All nodes	All nodes	BI engine communication with clients
8081	HTTP	Callosum/Tomcat status	bidirectional	All nodes	All nodes	BI engine communication with clients
8787	HTTP	Periscope (UI) service HTTP port	bidirectional	All nodes	All nodes	Administration UI back end

Port	Protocol	Service Name	Direction	Source	Dest.	Description
8888	HTTP	HTTP proxy server (tinyproxy)	bidirectional	All nodes	All nodes	Reverse SSH tunnel
11211	Memcached	Memcached server port	bidirectional	All nodes	All nodes	BI engine cache
12345	ODBC	Simba server port	bidirectional	All nodes	All nodes	Port used for ETL (extract, transform, load)
8480	HTTP	HDFS journalnode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
8485	HTTP	HDFS journalnode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50070	HTTP	HDFS namenode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50090	HTTP	HDFS secondary namenode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50075	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
50010	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
50020	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
7000	TCP	Cassandra KV store database	bidirectional	All nodes	All nodes	Debug DFS data
7001	TCP	Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
9042	HTTP	Munshi server impression service, Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
9160	TCP	Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
4010	HTTP	Falcon moderator	bidirectional	All nodes	All nodes	Debug DFS data
4011	HTTP	Falcon moderator	bidirectional	All nodes	All nodes	Debug DFS data

Port	Protocol	Service Name	Direction	Source	Dest.	Description
20123 - 32768	TCP (dynamic)	Dynamic port in this range used for various services and ancillary services like atlas, caffeine, call-home, callosum, falcon, monitoring, munshi server, nlp, object_search, postgres, sage UBR, spo-tiq snapshot, timely	All nodes	Services		
5270	TCP	Cluster monitoring service (ELK)	bidirectional	All nodes	All nodes	Services
5271	TCP	Cluster monitoring service (ELK)	bidirectional	All nodes	All nodes	Services
5601	TCP	Kibana UI (ELK)	bidirectional	All nodes	All nodes	Services
6311	TCP	R service	bidirectional	All nodes	All nodes	Services
8008	TCP	Video recorder	bidirectional	All nodes	All nodes	Services
9090	TCP	Timely	bidirectional	All nodes	All nodes	Services
ICMPv4		Used for health check of cluster nodes	bidirectional	All nodes	All nodes	Services

### Required ports for inbound and outbound cluster access

ThoughtSpot uses static ports for inbound and outbound access to the cluster.

Port	Protocol	Service Name	Direction	Source	Dest.	Description
22	SCP	SSH	bidirectional	ThoughtSpot Support	All nodes	Secure shell access.
80	HTTP	HTTP	bidirectional	ThoughtSpot Support	All nodes	Hypertext Transfer Protocol for website traffic.

Port	Protocol	Service Name	Direction	Source	Dest.	Description
443	HTTPS	HTTPS	bidirectional	ThoughtSpot Support	All nodes	Secure HTTP.
12345	TCP	Simba	bidirectional	ThoughtSpot Support	All nodes	Port used by ODBC and JDBC drivers when connecting to ThoughtSpot.
2049	TCP	NFS: In case one needs to mount NFS share on TS node.	bidirectional	ThoughtSpot Support	All nodes	Port used by NFS.
123	UDP	NTP service	bidirectional	ThoughtSpot Support	All nodes	Port used by NTP service.

Port	Protocol	Service Name	Direction	Source	Destination	Description
443	TCP	HTTPS	outbound	All nodes	208.83.110.20	For transferring files to thoughtspot.egnyte.com.
443	TCP	HTTPS	outbound	All nodes	For transferring product usage data to mixpanel cloud.	outbound
443	TCP	HTTPS	outbound	All nodes	je8b47jfif.execute-api.us-east-2.amazonaws.com s3.us-west-1.amazonaws.com s3-us-west-1.amazonaws.com s3.dualstack.us-west-1.amazonaws.com	For transferring monitoring data to InfluxCloud. (Given address will resolve to point to AWS instances).
25 or 587	SMTP	SMTP or Secure SMTP	outbound	All nodes and SMTP relay (provided by customer)	All nodes	Allow outbound access for the IP address of whichever email relay server is in use. This is for sending alerts to ThoughtSpot Support.

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Destination</b>	<b>Description</b>
389 or 636	TCP	LDAP or LDAPS	outbound	All nodes and LDAP server (provided by cus- tomer)	All nodes	Allow outbound access for the IP address of the LDAP server in use.

### Required ports for IPMI (Intelligent Platform Management Interface)

ThoughtSpot uses static ports for out-of-band IPMI communications between the cluster and ThoughtSpot support.

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Dest.</b>	<b>Description</b>
80	HTTP	HTTP	bidirectional	ThoughtSpot Support	All nodes	Hypertext Transfer Protocol for website traffic.
443	TCP	S-HTTP	bidirectional	ThoughtSpot Support	All nodes	IPMI GUI and for HTML5-based IPMI console access.
623	UDP	Serial-over-LAN	bidirectional	ThoughtSpot Support	All nodes	IPMI GUI and for HTML5-based IPMI console access.

# Cable networking

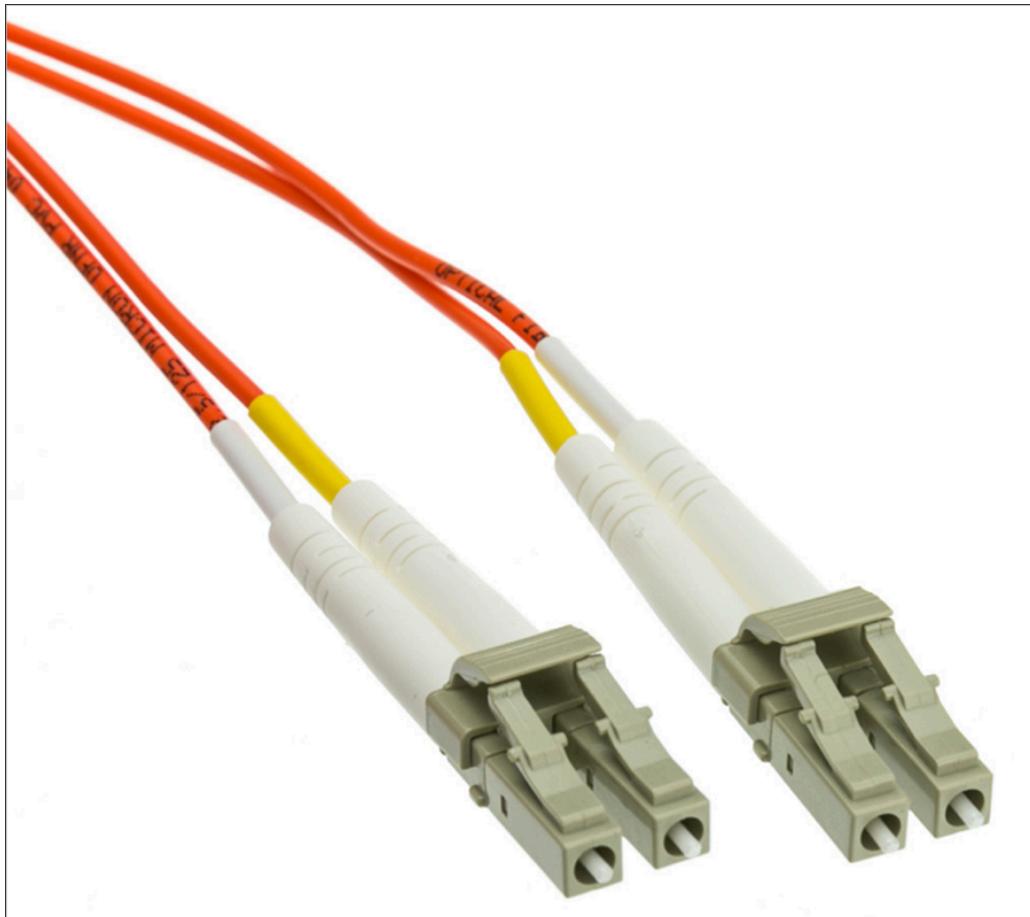
**Summary:** This section reviews the types of cables needed for 10GbE networking and how to plug them in.

There are three types of cables to consider for 10GbE networking:

- Fiber
- Direct Attach Copper (DAC)
- Category 6a (not supported by ThoughtSpot)

## Option 1 - Fiber cables

Fiber can be run long distances to the switch.

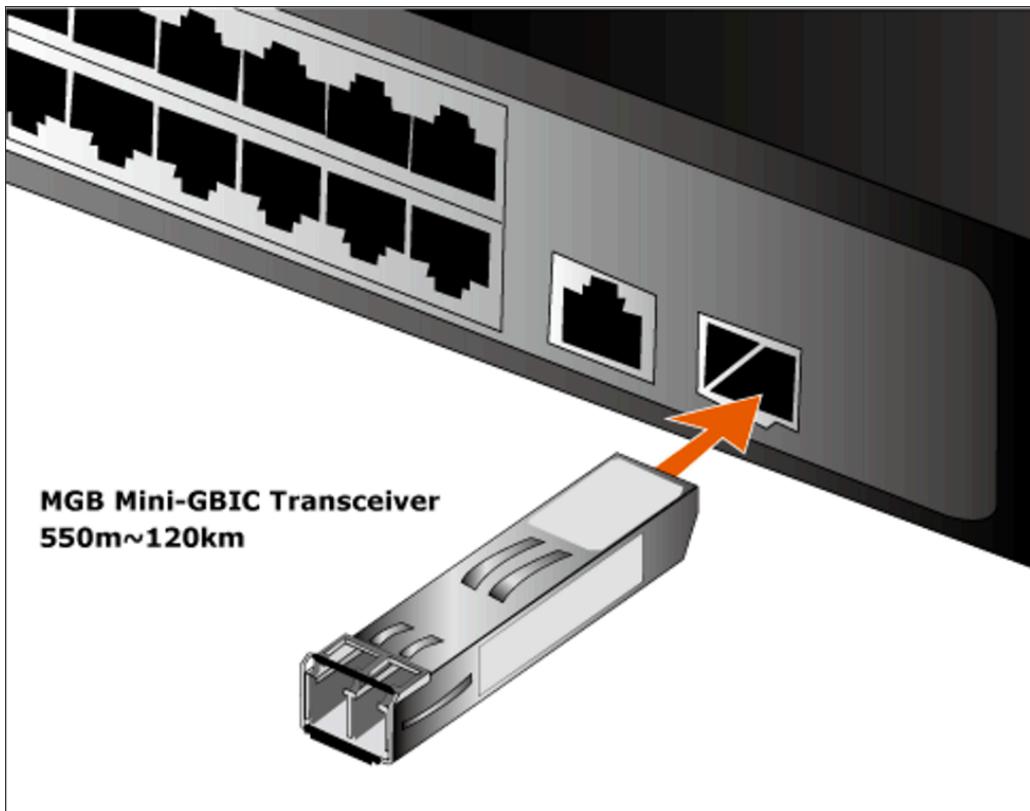


These cables require gigabit interface converters (GBICs), SFP+ form factor.

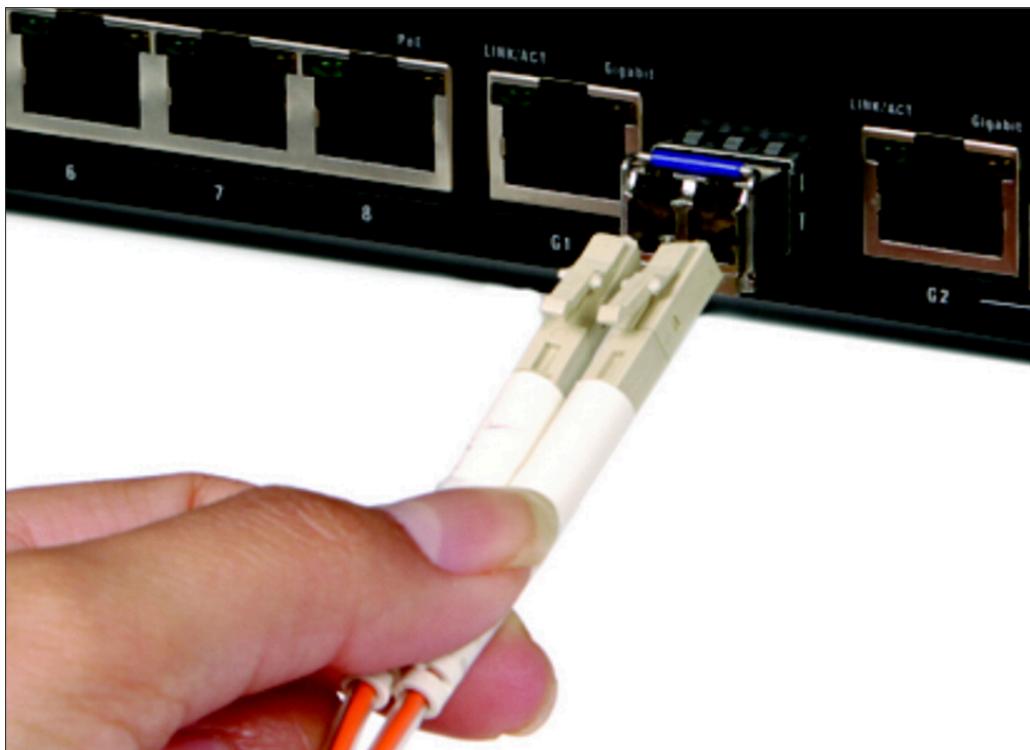
**Remember:** ThoughtSpot does not supply cables or GBICs



The GBIC must be plugged into a data port on the back of the appliance before plugging in the fiber cables.



The fiber cables must then be plugged into the GBIC.

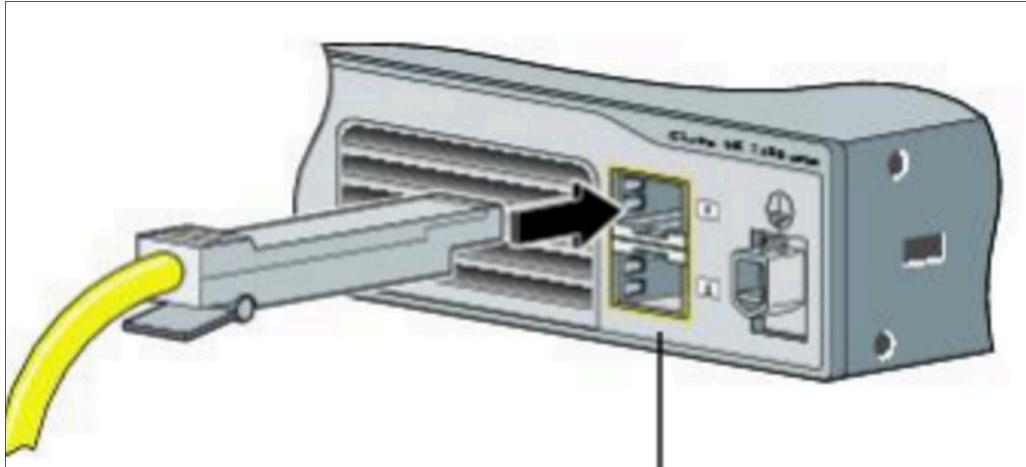


## Option 2 - DAC/Twinax cables

Copper can only be run short distances to the switch. An SFP+ is attached to the cable.



Here is how you would plug in a DAC cable.



## Non-option - Category 6a cables (not supported by ThoughtSpot)

There are no adapters for these cables. The 10GbE NIC (Network Interface Card) used on the ThoughtSpot appliance is not compatible with this type of cable/connection.



# Deploying on the SMC appliance

**Summary:** Follow these steps to deploy ThoughtSpot on your Super Micro Computer appliance.

Follow these steps to deploy ThoughtSpot on your Super Micro Computer (SMC) appliance.

- Step 1: Complete prerequisites [See page 61]
- Step 2: Review hardware requirements [See page 63]
- Step 3: Connect the SMC appliance [See page 66]
- Step 4: Configure nodes [See page 70]
- Step 5: Install cluster [See page 75]

## Related information

Use these references to aid you in successful installation and administration of ThoughtSpot.

- the `nodes.config` file [See page 0]
- Parameters of the `nodes.config` file [See page 30]
- Using the `tscli cluster create` command [See page 33]
- Parameters of the `cluster create` command [See page 41]
- Deployment Overview [See page 116]
- Contact Support [See page 290]

# Prerequisites

**Summary:** Complete these prerequisites before installing your ThoughtSpot clusters on the SMC appliance.

## Installation Prerequisites

Ensure that you have the following items, information, and understanding of policies before you begin deploying ThoughtSpot on your SMC appliance.

- Appliance Port Location [\[See page 0\]](#), to locate data and IPMI ports.
- Data center with proper environment controls, such as cooling.
- AC power
- 10GbE switch, with enabled IPv6 broadcast and multicast. You need one switch for each cluster, with one port for each node on the cluster.
- 10GbE network cables, either direct attach copper (DAC) or fiber. Refer to the [Cable reference \[See page 55\]](#) for more information to decide between the two types.
- 100Mbps or 1Gbps switch for IPMI, for Out of Band Management. You need one for each node in your cluster.
- Cat5 network cables, for IPMI management port use. You need one for each node.
- Rack space of 2U or 3.5 inches for each appliance, and a power strip
- VGA Monitor and USB keyboard
- 10G connection: SFP+ for the switch side
- Networking information: IP addresses for data & management NICs, for up to 2 DNS servers, up to 4 NTP servers and for the default gateway. Ensure that you configure only two DNS servers. ThoughtSpot does not support configuration of three DNS servers. You also need to know the timezone for your cluster. Typically, your timezone is where most of the people who will use the product are. [Download \[See page 0\]](#) and fill out the ThoughtSpot site survey so that you have a quick reference before beginning the install process. Ask your network administrator if you need help filling out the site survey.
- Network policies [\[See page 44\]](#), to determine the ports you need to have open for your cluster.

## Review hardware requirements

Next, [review hardware requirements \[See page 63\]](#).

# Hardware Requirements

**Summary:** Learn about your SMC hardware before deploying ThoughtSpot.

## About the Hardware

You can deploy ThoughtSpot on two different appliance hardware platforms: Haswell and Skylake. Both of the platforms provide the same performance. Refer to [Haswell and Skylake hardware details \[See page 63\]](#) for details on their physical differences.

Details	Haswell	Skylake
<b>Dimensions</b>	2 RU chassis (17.25" x 3.47" x 28.5" (WxHxD))	2 RU chassis (17.6" x 3.47" x 28.75" (WxHxD))
<b># of nodes</b>	Populated with 1 to 4 nodes	Populated with 1 to 4 nodes
<b>Node specifications</b>	Each node is independent and consists of a server board (removable from rear), 1x 200GB SSD, 3x .2TB HDD	Each node is independent and consists of a server board (removable from rear), 1x 240GB SSD, 3x 2TB HDD
<b>Max power consumption</b>	2000 W	2200 W
<b>Required power input</b>	200-240V / 11.8 - 9.8A / 50-60Hz (C13 / C14 power cords)	220-240 VAC 50-60 Hz (C13 / C14 power cords)

## Haswell front and back views

These images show the front and back views of each appliance.

The nodes on the front of both appliances go from A-D left to right. For this Haswell appliance, only Node D is populated.



The nodes on the back of both appliances are in a reverse N shape, with Node A at the bottom right and Node D at the top left.



The Haswell appliance shown here is not fully populated, as it only has three nodes. Your appliance may be populated with 1-4 nodes, depending on the ordered configuration. If you order less than four nodes, ThoughtSpot fills the empty slot with a filler panel.

## Skylake front and back views

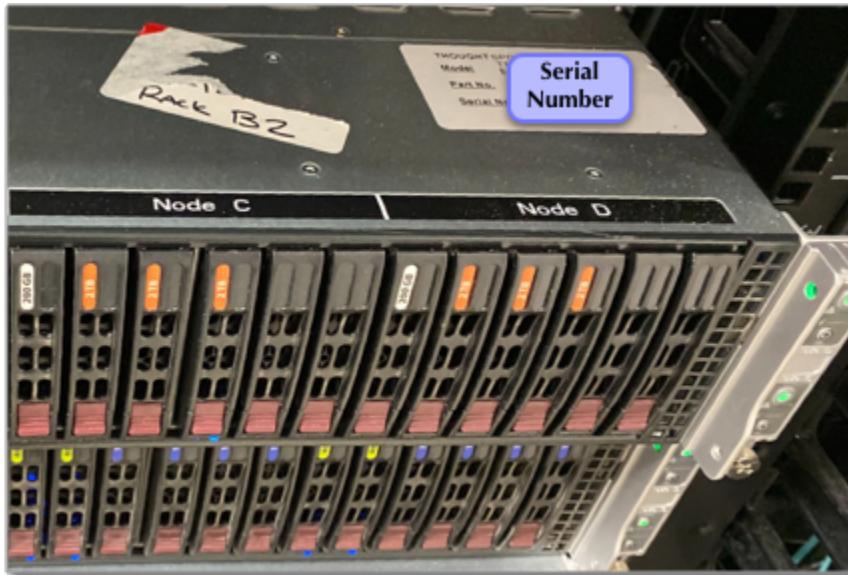




The Skylake appliance shown here is fully populated with four nodes.

## Location of serial number

You may need to know your appliance's serial number, to be able to access online help from your appliance provider. Find your Super Micro Computer's serial number on the top of the appliance, above the label for Node D at the front right corner.



## Connect the appliance

Next, [connect the appliance](#). [See page 66]

# Connect the SMC appliance

**Summary:** Connect the SMC appliance before you can deploy ThoughtSpot.

After you rack and stack the appliance, it is time to configure it. If necessary, review the [Hardware Appliance Overview \[See page 117\]](#). Follow the steps in this checklist.

- Step 1: Connect switches to 10GbE ports [See page 0]
- Step 2: Connect IPMI ports [See page 0]
- Step 3: Turn on nodes [See page 0]
- Step 4: Log in [See page 0]

## Step 1: Connect switches to 10GbE ports

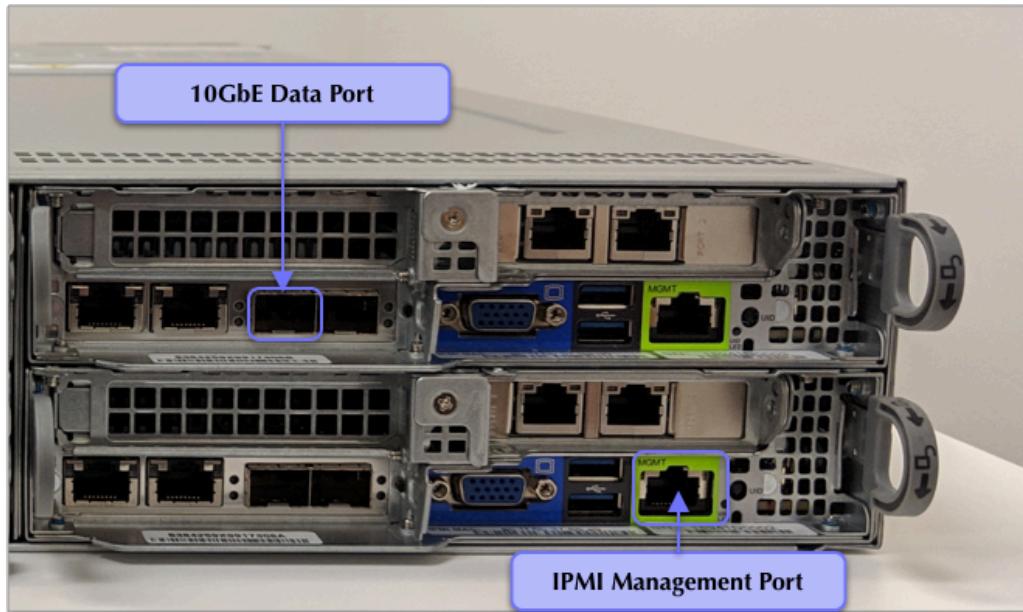
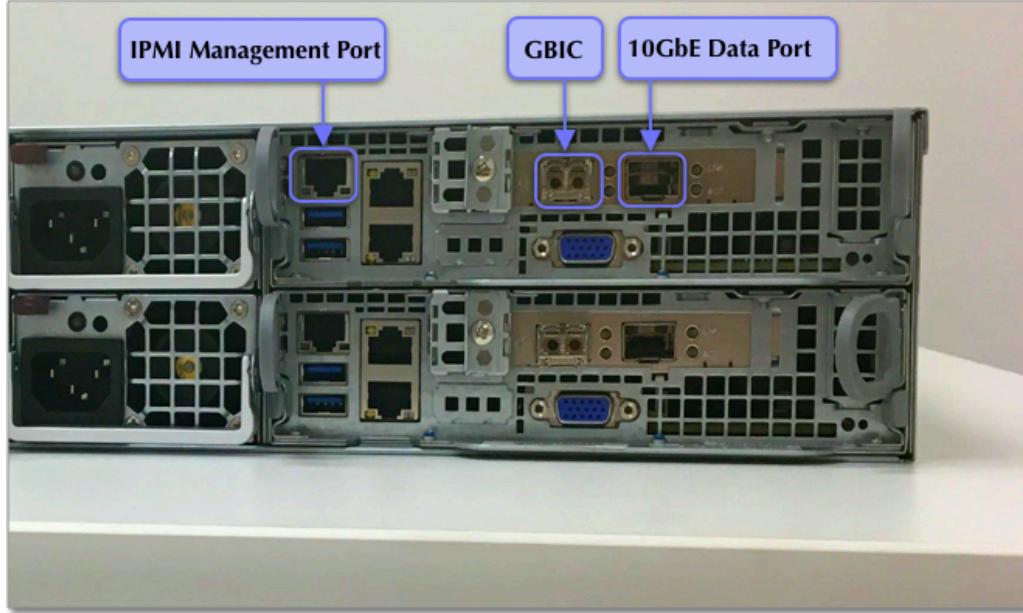
Connect the 10GbE port of each node, as illustrated in [Haswell port location \[See page 67\]](#) and [Skylake port location \[See page 67\]](#), to the 10GbE switches on your own rack, using either fiber or DAC cables.

Refer to the [Cable reference \[See page 55\]](#) for information on the cable types:

- [Fiber Cables \[See page 55\]](#)
- [DAC Cables \[See page 58\]](#)

**Note:** Ask your hardware vendor for more details about what they supply and what you need to buy.

Depending on which version of the SMC appliance you have, Haswell or Skylake, your 10GbE ports are in a different spot on the back of the appliance. Here is a picture of the back of each appliance.



- Connect to switches **only** the appliances (4 nodes each) that you plan to use in the cluster.
- You must power off, or disconnect from the switch, all other appliances or nodes.  
This prevents accidental configuration of incorrect nodes.
- Connect all the nodes that you plan to use in your cluster to a 10G switch.

## Step 2: Connect IPMI ports

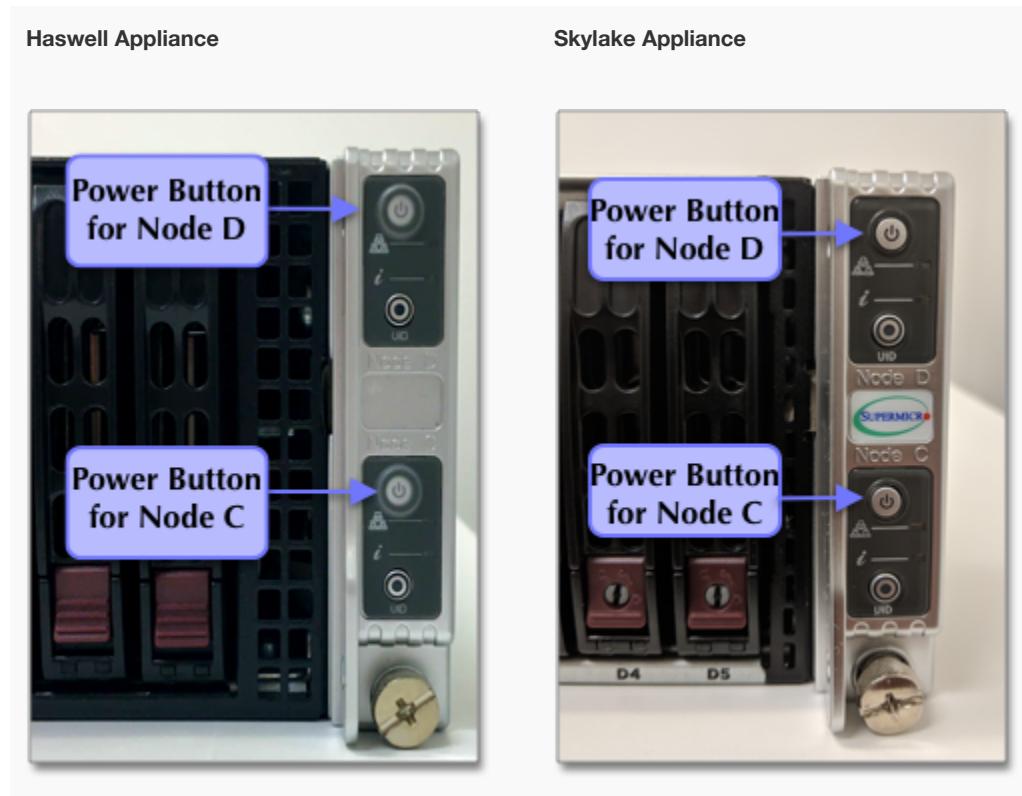
Connect the IPMI port of each node to the management switch.

See [Haswell port location \[See page 67\]](#) and [Skylake port location \[See page 67\]](#).

## Step 3: Turn on nodes

Turn on the power to the nodes by pressing the power button; see [Appliance Power Button \[See page 68\]](#).

There is one power button for each node. The images below show two of the four power buttons.



Check the health of the NIC by inspecting your appliance's LEDs. There should be a steady green light for the 100MB/s or 1Gb/s management NIC connection, a steady amber or orange light for the 10G/s connection, and a blinking LED on both NICs to indicate live traffic.

After the LEDs are the correct colors, verify that the connection is valid and the nodes are on by pinging the gateway:

Enter `ping <default-gateway-IP>`.

Ask your network administrator for your default gateway IP if you have not already listed it in your ThoughtSpot [site survey \[See page 0\]](#).

```
$ ping <default-gateway-IP>
```

## Step 4: Log in

1. Connect a keyboard and the mouse to each node on the appliance.
2. You should see a login prompt on the screen. If you don't see one or the screen isn't responsive, press the key combination **control-alt-F2** or **alt-F2** on your keyboard to bring up the login prompt.
3. Log in using the admin user credentials for the console. If you do not know the admin credentials, ask your network administrator.

## Configure nodes

Next, [configure nodes. \[See page 70\]](#)

# Configure ThoughtSpot Nodes on the SMC Appliance

**Summary:** Configure your nodes before you can install your cluster(s).

After you connect the appliance, configure the nodes in your Mac or Windows terminal emulator. Follow the steps in this checklist.

- Step 1: SSH into your cluster [See page 0]
- Step 2: Change to the `install` directory [See page 0]
- Step 3: Get a template for network configuration [See page 0]
- Step 4: Prepare node configuration [See page 0]
- Step 5: Configure the nodes [See page 0]
- Step 6: Confirm node configuration [See page 0]

If you completed ThoughtSpot's [site survey \[See page 0\]](#) form and returned it to [ThoughtSpot Support \[See page 290\]](#) before ThoughtSpot shipped the appliance, the appliance may be pre-configured for your network environment and ready to install and connect to your network.

If the network configuration was not pre-set, then this step must be done as part of the installation process.

Follow these steps to determine the configuration status of your appliance.

1. SSH into your cluster. Run `ssh admin@<nodeIP>`.  
Replace `nodeIP` with your specific network information.

```
$ ssh admin@<nodeIP>
```

2. Run `tscli cluster status`.

```
$ tscli cluster status
```

3. If the output shows READY, and looks like the [cluster status output \[See page 77\]](#) in the next article, your appliance is configured.
4. Skip to [Finalize installation \[See page 79\]](#).

If your status is not READY, continue with the installation process outlined below.

## Step 1: SSH into your cluster

SSH into your cluster with admin credentials.

1. Run the command `ssh admin@<cluster-IP>` or `ssh admin@<hostname>` on the command line.

Replace `clusterIP` or `hostname` with your specific network information.

```
$ ssh admin@<nodeIP>
```

2. Enter your admin password when prompted.

Ask your network administrator if you don't know the password.

## Step 2: Change to the install directory

In your terminal, change directory to `/home/admin/install` by running the command `cd /home/admin/install`. If your `/install` subdirectory does not exist, you may have to use the `/home/admin` directory.

```
$ cd /home/admin/install
```

## Step 3: Get a template for network configuration

Run the `tscli cluster get-config` command to get a template for network configuration. Redirect it to the file `nodes.config`. You can find more information on this process in the [nodes.config file reference \[See page 27\]](#).

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

## Step 4: Prepare node configuration

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the [autodiscovery of one node example \[See page 29\]](#). Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information.  
If you have additional nodes, complete each node within the `nodes.config` file in the same way.

Edit only the parts of the `nodes.config` file that are explicitly discussed in [Parameters of nodes.config \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, it may cause setup to fail.

## Step 5: Configure the nodes

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

Run `$ cat nodes.config | tscli cluster set-config` in your terminal.

If the command returns an error, refer to [set-config error recovery \[See page 81\]](#).

```
$ 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 6: Confirm node configuration

Use the `get-config` command to confirm node configuration.

Your output may look similar to the following:

```
$ 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 the cluster

Next, [install your cluster. \[See page 75\]](#)

# Install ThoughtSpot Clusters on the SMC Appliance

**Summary:** Install your clusters on the SMC appliance.

Install the cluster using the ThoughtSpot software release bundle. Installation takes approximately one hour. Make sure you can connect to ThoughtSpot remotely. If you can, you can run the installer on your local computer.

Refer to your welcome letter from ThoughtSpot to find the link to download the release bundle. If you have not received a link to download the release bundle, open a support ticket at [ThoughtSpot Support](#) [See page 0] to access the release bundle.

## Step 1. Run the Installer

1. Copy the downloaded release bundle to `/export/sdb1/TS_TASKS/install`. Run `scp <release-number>.tar.gz admin@<hostname>:/export/sdb1/TS_TASKS/install/<file-name>`.

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot installation, such as `6.0`, `5.3`, `5.3.1`, and so on.
- `hostname` is your specific hostname.
- `file-name` is the name of the tarball file on your local machine.

```
$ scp <release-number>.tar.gz admin@<hostname>:/export/sdb1/TS_TASKS/install/<file-name>
```

**Note:** You can use another secure copy method, if you prefer a method other than the `scp` command.

2. Alternatively, use `tscli fileserver download-release` to download the release bundle.

You must [configure the fileserver](#) [See page 0] by running `tscli fileserver configure`

before you can download the release.

```
$ tscli fileserver download-release <release-number> --  
user <username> --out <release-location>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 5.3.1, 6.0, and so on.
- `username` is the username for the fileserver that you set up earlier, when configuring the fileserver.
- `release-location` is the location path of the release bundle on your local machine. For example, `/export/sdb1/TS_TASKS/install/6.0.tar.gz`.

3. Verify the checksum to ensure you have the correct release.

Run `md5sum -c <release-number>.tar.gz.MD5checksum`.

```
$ md5sum -c <release-number>.tar.gz.MD5checksum
```

Your output says `ok` if you have the correct release.

4. Launch a [screen \[See page 0\]](#) session. Use screen to ensure that your installation does not stop if you lose network connectivity.

```
$ screen -S DEPLOYMENT
```

5. Create the cluster.

Run `tscli cluster create <release-number>`.

```
$ tscli cluster create <release-number>.tar.gz
```

6. Edit the output using your specific cluster information. For more information on this process, refer to [Using the tscli cluster create command \[See page 33\]](#) and [Parameters of the cluster create command \[See page 41\]](#).

The cluster installer automatically reboots all the nodes after the install. Wait at least 15 minutes for the installation process to complete. The system is rebooting, which takes a few minutes.

Log in to any node to check the current cluster status, using the command `tscli cluster status`.

## Step 2. Check Cluster Health

After you install the cluster, check its status using the `tscli cluster status` and `tscli cluster check` commands.

```
$ 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
```

Ensure that the cluster is `RUNNING` and that the Database and Search Engine are `READY`.

```
$ tscli cluster check
Connecting to hosts...
[Wed Jan  8 23:15:47 2020] START Diagnosing ssh
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing connection
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing zookeeper
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing sage
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing timezone
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing disk
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing cassandra
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing hdfs
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing orion-oreo
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing memcheck
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing ntp
[Wed Jan  8 23:16:08 2020] SUCCESS
#####
```

```
#####
[Wed Jan  8 23:16:08 2020] START Diagnosing trace_vault
[Wed Jan  8 23:16:09 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:09 2020] START Diagnosing postgres
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing disk-health
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing falcon
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing orion-cgroups
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing callosum
/usr/lib/python2.7/site-packages/urllib3/connectionpool.py:85
2: InsecureRequestWarning: Unverified HTTPS request is being ma
de. Adding certificate verification is strongly advised. See: h
ttps://urllib3.readthedocs.io/en/latest/advanced-usage.html#ss
l-warnings
    InsecureRequestWarning)
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
```

Your output may look something like the above. Ensure that all diagnostics show `SUCCESS`.

**● Warning:** If `tscli cluster check` returns an error, it may suggest you run `tscli storage gc` to resolve the issue. If you run `tscli storage gc`, note that it restarts your cluster.

## Step 3. Finalize Installation

After the cluster status changes to “Ready,” sign in to 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.

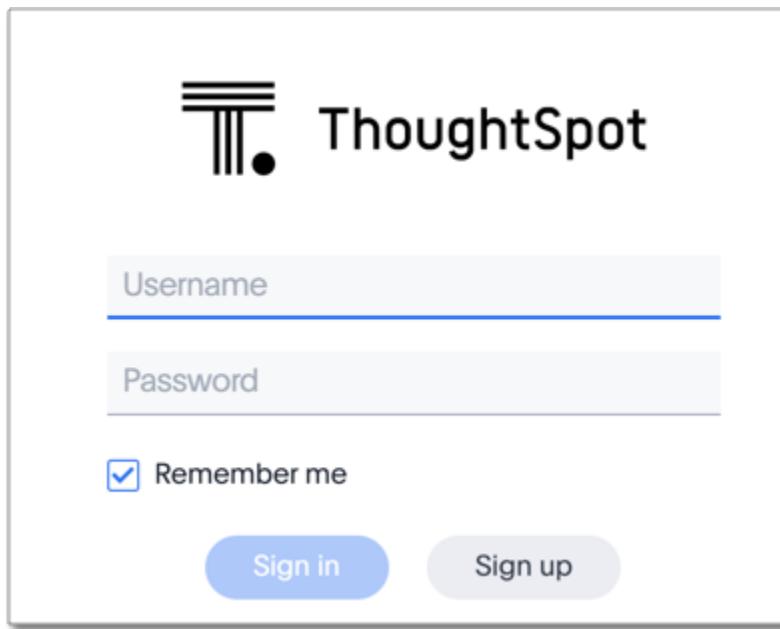
`https://<IP-address>`

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 sign-in page appears.
5. In the [ThoughtSpot sign-in window \[See page 80\]](#), enter admin credentials, and click **Sign in**.

If you do not know the admin credentials, ask your network administrator. ThoughtSpot recommends changing the default admin password.



## Lean configuration

(**For use with thin provisioning only**) If you have a [small or medium instance type \[See page 167\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 290\]](#) for assistance with this configuration.

## Error recovery

### Set-config error recovery

If you get a warning about node detection when you run the `set-config` command, restart the node-scout service.

Your error may look something like the following:

```
Connecting to local node-scout
WARNING: Detected 0 nodes, but found configuration for only 1 nodes.
Continuing anyway. Error in cluster config validation: [] is not a valid link-local IPv6 address for node: 0e:86:e2:23:8f:76
Configuration failed.
Please retry or contact support.
```

Restart the node-scout service with the following command.

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

Ensure that you restarted the node-scout by running `sudo systemctl status node-scout`. Your output should specify that the node-scout service is active. It may look something like the following:

```
$ sudo systemctl status node-scout
● node-scout.service - Setup Node Scout service
  Loaded: loaded (/etc/systemd/system/node-scout.service; enabled; vendor preset: disabled)
  Active: active (running) since Fri 2019-12-06 13:56:29 PS
          T; 4s ago
```

Next, retry the set-config command.

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

The command output should no longer have a warning.

# The nodes.config file

**Summary:** Learn how to use the get.config command and the nodes.config file to install your hardware or cloud appliance.

## Using the nodes.config file

As you install your appliance, you must configure the nodes.

1. Run the configuration command in your terminal.

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

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information, as shown in [Autodiscovery of one node example \[See page 0\]](#).

**Note:** Some of the information in the nodes.config file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your nodes.config file.

3. If you have additional nodes, complete each node within the nodes.config file as shown in the [Autodiscovery of one node example \[See page 0\]](#). [Autodiscovery of one node \[See page 0\]](#) shows the nodes.config file before you fill in your specific information.

Do not edit any part of the nodes.config file except the sections explained in [Parameters of the nodes.config file \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, setup may fail.

See [Parameters of the nodes.config file \[See page 30\]](#) to understand the parameters in the file.

### Autodiscovery of one node

```
$ tscli cluster get-config |& tee nodes.config
{
    "ClusterId": "",
    "ClusterName": "",
    "DataNetmask": "",
    "DataGateway": "",
    "IPMINetmask": "",
    "IPMIGateway": "",
    "Timezone": "",
    "NTPServers": "",
    "DNS": "",
    "SearchDomains": "",
    "Nodes": {
        "06:83:1f:f8:99:9e": {
            "NodeId": "06:83:1f:f8:99:9e",
            "Hostname": "",
            "DataIface": {
                "Name": "eth0",
                "IPv4": ""
            },
            "IPMI": {
                "IPv4": ""
            }
        }
    }
}
```

### Autodiscovery of one node example

```
$ vim nodes.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"
      }
    }
  }
}
```

## Related information

- [Deploying on the SMC appliance \[See page 60\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 182\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 240\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 264\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 210\]](#)
- [Deploying on the Dell Appliance \[See page 4\]](#)
- [Parameters of the nodes.config file \[See page 30\]](#)

# Parameters of the nodes.config file

**Summary:** Learn the parameters of the nodes.config file to install your cloud or hardware appliance.

## Using the nodes.config file

As you install your appliance, you must configure the nodes.

1. Run the configuration command in your terminal.

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

2. Add network information for your nodes in the `nodes.config` file output, as demonstrated in the [nodes.config file \[See page 27\]](#) example.

Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

Fill in the areas specified in [Parameters of nodes.config \[See page 31\]](#) with your specific network information.

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

3. If you have additional nodes, complete this process for each node.

Do not edit any part of the `nodes.config` file except the sections explained in [Parameters of nodes.config \[See page 31\]](#). If you delete quotation marks, commas, or other parts of the code, setup may fail.

See [Parameters of nodes.config \[See page 31\]](#) to understand the parameters in the file. Different hardware and cloud installations have different parameters. Your installation may not require all the listed parameters.

## Parameters of the nodes.config file

**ClusterId** and **Cluster Name**: Leave these two parameters blank. You fill them out later, when running `tscli cluster create .`

**DataNetmask** The IP of the data netmask, in the form 000.000.000.000. For example, 255.255.252.0.

**DataGateway** The IP of the data gateway, in the form 000.000.000.000. For example, 192.168.4.1.

**IPMINetmask** The IP of the Intelligent Platform Management Interface (IPMI) netmask, in the form 000.000.000.000. For example, 255.255.252.0.

**IPMIGateway** The IP of the Intelligent Platform Management Interface (IPMI) gateway, in the form 000.000.000.000. For example, 192.168.4.1.

**Timezone** The timezone the majority of your ThoughtSpot users are in, in the form Country/City. For example, America/Los\_Angeles. To find your timezone and a city you can use to identify it, use [this timezone list \[See page 0\]](#).

**NTPServers** The address of your company's Network Time Protocol (NTP) server. If your company does not have an NTP server, you can use one of ThoughtSpot's, as listed in [the nodes.config \[See page 29\]](#) example under NTPServers. For example, 0.centos.pool.ntp.org.

**DNS** The address of your company's Domain Name Server (DNS). For example, 192.168.2.200,8.8.8.8.

**⚠ Warning:** Configure only two DNS servers. ThoughtSpot does not support configuration of three DNS servers.

**⚠ Note:** You can only edit DNS settings with this command if you are deploying ThoughtSpot on hardware. ThoughtSpot does not support using `set-config` to edit your DNS settings for cloud deployment.

**SearchDomains** The domain of your company or organization, in the form example.company.com.

**Hostname** The name of the host server. For example, Thoughtspot-server1.

**IPv4** The main IP address, associated with DataNetmask and DataGateway, in the form 000.000.000.000. For example, 192.168.7.70.

**IPMI IPv4** A secondary IP address, associated with the IPMINetmask and IPMIGateway, in the form 000.000.000.000. For example, 192.168.5.70.

## Related information

- [Deploying on the SMC appliance \[See page 60\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 182\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 240\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 264\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 210\]](#)
- [Deploying on the Dell Appliance \[See page 4\]](#)
- [The nodes.config file \[See page 27\]](#)

# Using the tscli cluster create command

**Summary:** Learn how to use the cluster create command to install your appliance.

## Using the tscli cluster create command

To install your appliance, you must install the cluster using the release tarball (estimated time 1 hour).

1. **Download the release tarball** Download the release tarball from the download link sent by ThoughtSpot Support.  
Refer to your welcome letter to find a link to download the release tarball. If you do not have a download link, open a support ticket at [ThoughtSpot Support \[See page 0\]](#) to access the release tarball.
2. **Copy the release tarball** In your Terminal (Apple) or using [Winscp \[See page 0\]](#) (Windows application), copy the downloaded release tarball to `/home/admin/` and run the cluster installer as shown below.

Run the secure copy command: `scp <release-number> admin@<hostname>:/home/admin/<file-name>`. Note the following parameters:

- `release-number` is the version of ThoughtSpot you have on your cluster, such as `6.0` or `5.3`. `release-number` is of the form `0.0.tar.gz`.
- `hostname` is your network hostname. Ask your network administrator if you do not know your hostname.
- `file-name` is the name of the tarball file on your local machine. For example:

```
$ scp 6.0.tar.gz admin@ThoughtSpot:/home/admin/T  
S-tarball.
```

ThoughtSpot recommends that you start the session using [screen \[See page 0\]](#), so you do not lose your progress if you lose your connection at any point.

```
$ scp <release-number> admin@<hostname>:/home/adm  
in/<file-name>
```

3. **Run the `tscli cluster create` command** Run `tscli cluster create <release-number>` in your terminal.

If you are using an s3 or GCS bucket for object storage, include the flag `--enable_cloud_storage=s3` or `--enable_cloud_storage=gcs`. GCS is GCP's object storage, and s3 is AWS's object storage.

```
$ tscli cluster create 6.0.tar.gz --enable_cloud_storage=s3
```

```
$ tscli cluster create 6.0.tar.gz --enable_cloud_storage=gcs
```

4. **Specify your installation information** Fill out the cluster name, cluster ID, email alert preferences and the IP's of the nodes at the prompts specified in [Parameters of the tscli cluster create command \[See page 41\]](#).
5. **Wait for output** You may need to wait about 15 seconds before you see any output. The installer is unpacking files and copying them over to the nodes, which can take a few seconds.

Do not edit any part of the installer file except the sections specified in [Parameters of the tscli cluster create command \[See page 41\]](#). if you delete colons, commas, or other parts of the code, setup may fail.

Refer to [Parameters of the tscli cluster create command \[See page 41\]](#) for further information.

Your `tscli cluster create` output may look something like the following:

### Run the Installer

The output for a hardware installation:

```
[admin@jessi-gcs-test ~]$ tscli cluster create 6.0-145.tar.gz
Unpacking 6.0-145.tar.gz to /export/release_cache/e695feeecc6275
91dc644635c0d8ea03d
#####
#
#
#
#           Welcome to ThoughtSpot installer
#
#
#
#
#####
#
Enter a name for the cluster (alphanumeric characters only): jessi-ts-gcs-test
Enter cluster ID: 0x0000
Enter IP addresses of all hosts in the cluster (space separate
d): 10.116.0.66
Enter email addresses for alerts (space separated, "later" to s
kip): later
2019-11-03 21:26:47,959 Pushing /usr/local/scaligent/toolchain/
jolokia to localhost
2019-11-03 21:26:48,195 Rsync finished on localhost
2019-11-03 21:26:48,195 Rsync finished to all hosts
2019-11-03 21:26:48,195 Pushing /usr/local/scaligent/toolchain/
jvm to localhost
2019-11-03 21:26:48,461 Rsync finished on localhost
2019-11-03 21:26:48,461 Rsync finished to all hosts
2019-11-03 21:26:48,462 Pushing /usr/local/scaligent/toolchain/
hadoop to localhost
2019-11-03 21:26:48,738 Rsync finished on localhost
2019-11-03 21:26:48,738 Rsync finished to all hosts
2019-11-03 21:26:48,738 Pushing /usr/local/scaligent/toolchain/
zookeeper to localhost
2019-11-03 21:26:49,004 Rsync finished on localhost
2019-11-03 21:26:49,004 Rsync finished to all hosts
#####
#
Setup Hadoop
#####
#
Installing Zookeeper
First deleting existing Zookeeper service
Deleting Zookeeper
Zookeeper Deleted
```

```
Starting Zookeeper servers
Zookeeper Ready
Installing HDFS
First deleting existing HDFS service
Deleting HDFS
HDFS Deleted
Deploying configs
Formatting and starting Primary Name Node
Starting Data Nodes
Setting NFS configs
Waiting for HDFS
HDFS Ready
2019-11-03 21:27:06,212 Starting a secondary namenode for check
point...
2019-11-03 21:27:06,551 Cannot disable unknown service: hdfs_se
condary_namenode
2019-11-03 21:27:06,552 Failed to disable new service: hdfs_sec
ondary_namenode
2019-11-03 21:27:07,198 HDFS secondary namenode started success
fully.
Successfully Created zookeeper znodes
Successfully initialized HDFS files
Successfully saved hadoop layout proto
Successfully moved the release dir to export partition
Successfully installed orion on localhost
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Couldn't install application services
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Couldn't install application services
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Successfully installed application services
All nodes are now healthy
Restarted all services
Successfully pushed statsdb
Successfully refreshed alert metadata
Successfully enabled alerts
```

```
Successfully configured new ssh keys on cluster
Successfully recorded install event
Successfully deleted deployer service
2019-11-03 21:40:02,917 Cannot check for enabled status of unknown service: deploy
```

The output for a cloud installation, with GCS object storage:

```
admin@jessi-gcs-test ~]$ tscli cluster create ./6.0-145.tar.gz --enable_cloud_storage gcs
Unpacking ./6.0-145.tar.gz to /export/release_cache/e695feeeec627591dc644635c0d8ea03d
#####
#
#
#
#           Welcome to ThoughtSpot installer
#
#
#
#####
#
Enter a name for the cluster (alphanumeric characters only): jessi-ts-gcs-test
Enter cluster ID: 0x0000
Enter IP addresses of all hosts in the cluster (space separated): 10.116.0.66
Enter email addresses for alerts (space separated, "later" to skip): later
Enter the GCS bucket to be configured for the cluster: jessi-gcs-test-bucket
Bucket successfully validated.
2019-11-03 22:00:29,175 Pushing /usr/local/scaligent/toolchain/jolokia to localhost
2019-11-03 22:00:29,411 Rsync finished on localhost
2019-11-03 22:00:29,411 Rsync finished to all hosts
2019-11-03 22:00:29,412 Pushing /usr/local/scaligent/toolchain/jvm to localhost
2019-11-03 22:00:29,678 Rsync finished on localhost
2019-11-03 22:00:29,678 Rsync finished to all hosts
2019-11-03 22:00:29,678 Pushing /usr/local/scaligent/toolchain/hadoop to localhost
2019-11-03 22:00:29,945 Rsync finished on localhost
2019-11-03 22:00:29,945 Rsync finished to all hosts
2019-11-03 22:00:29,945 Pushing /usr/local/scaligent/toolchain/zookeeper to localhost
2019-11-03 22:00:30,211 Rsync finished on localhost
2019-11-03 22:00:30,211 Rsync finished to all hosts
#####
#
#           Setup Hadoop
#####
#
```

```
Installing Zookeeper
First deleting existing Zookeeper service
Deleting Zookeeper
Zookeeper Deleted
Starting Zookeeper servers
Zookeeper Ready
Installing HDFS
First deleting existing HDFS service
Deleting HDFS
HDFS Deleted
Deploying configs
Formatting and starting Primary Name Node
Starting Data Nodes
Setting NFS configs
Waiting for HDFS
HDFS Ready
2019-11-03 22:00:47,287 Starting a secondary namenode for check
point...
2019-11-03 22:00:47,629 Cannot disable unknown service: hdfs_se
condary_namenode
2019-11-03 22:00:47,630 Failed to disable new service: hdfs_sec
ondary_namenode
2019-11-03 22:00:48,282 HDFS secondary namenode started success
fully.
Successfully Created zookeeper znodes
Successfully initialized HDFS files
Successfully saved hadoop layout proto
Successfully moved the release dir to export partition
Successfully installed orion on localhost
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Couldn't install application services
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Couldn't install application services
Successfully connected to Orion Master
Successfully enabled data protection
Successfully enabled firewall
Successfully added machines to cluster
Successfully installed application services
All nodes are now healthy
```

```
Restarted all services
Successfully pushed statsdb
Successfully refreshed alert metadata
Successfully enabled alerts
Successfully configured new ssh keys on cluster
Successfully recorded install event
Successfully deleted deployer service
2019-11-03 22:11:54,571 Cannot check for enabled status of unknown service: deploy
```

## Related information

- [Deploying on the SMC appliance \[See page 60\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 182\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 240\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 264\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 210\]](#)
- [Deploying on the Dell Appliance \[See page 4\]](#)
- [Parameters of the tscli cluster create command \[See page 41\]](#)

# Parameters of the tscli cluster create command

**Summary:** Learn the parameters of the tscli cluster create command.

## Using the tscli cluster create command

To install your appliance, you must install the cluster.

1. **Copy the release tarball** In your Terminal (Apple) or using [Winscp \[See page 0\]](#) (Windows application), copy the downloaded release tarball to /home/admin/ and run the cluster installer as shown below.

Run the secure copy command: `scp <release-number> admin@<hostname>:/home/admin/<file-name>`. Note the following parameters:

- `release-number` is the version of ThoughtSpot you have on your cluster, such as `6.0` or `5.3`. `release-number` is of the form `0.0.tar.gz`.
- `hostname` is your network hostname. Ask your network administrator if you do not know your hostname.
- `file-name` is the name of the tarball file on your local machine. For example:

```
$ scp 6.0.tar.gz admin@ThoughtSpot:/home/admin/T  
S-tarball.
```

ThoughtSpot recommends that you start the session using [screen \[See page 0\]](#), so you do not lose your progress if you lose your connection at any point.

```
$ scp <release-number> admin@<hostname>:/home/ad  
min/<file-name>
```

2. **Run the cluster create command** Run `tscli cluster create <release-number>` in your terminal.

If you are using an s3 or GCS bucket for object storage, include the flag `--enable_cloud_storage=s3` or `--enable_cloud_storage=gcs`. GCS is GCP's object

storage, and s3 is AWS's object storage.

```
$ tscli cluster create 6.0.tar.gz --enable_cloud_storage=s3
```

```
$ tscli cluster create 6.0.tar.gz --enable_cloud_storage=gcs
```

3. **Fill out your specific information** Fill out the cluster name, cluster ID, email alert preferences and the IP's of the nodes at the prompts specified in [Parameters of ThoughtSpot Installer \[See page 42\]](#) below.

Refer to [Parameters of ThoughtSpot Installer \[See page 42\]](#) for further information. Refer to [Using the tscli cluster create command \[See page 33\]](#) to see the expected output of the install command, `tscli cluster create`.

Do not edit any part of the installer file except the sections specified in [Parameters of ThoughtSpot Installer \[See page 42\]](#). If you delete colons, commas, or other parts of the code, setup may fail.

## Parameters of ThoughtSpot Installer

**Cluster Name** Name your cluster based on the ThoughtSpot naming convention, in the form *company-clustertype-location-clusternumber*. For example, *ThoughtSpot-prod-Sunnyvale-12*.

**Cluster ID** Enter the ID of your cluster that [ThoughtSpot Support \[See page 290\]](#) provided for you. Open a ticket with ThoughtSpot Support if you do not have an ID.

**Host IPs** Enter the IP addresses of all cluster hosts, in the form 000.000.000.000. For example, 192.168.7.70. Use spaces instead of commas to separate multiple IP addresses.

**Email alerts** Enter the email addresses you would like to receive alerts about this cluster, in the form company@example.com. The address prod-alerts@thoughtspot.com appears automatically and should remain, so that ThoughtSpot can be aware of the status of your cluster. Separate email addresses using a space.

## Related information

- Deploying on the SMC appliance [See page 60]
- Configure ThoughtSpot nodes in AWS [See page 182]
- Configure ThoughtSpot nodes in GCP [See page 240]
- Configure ThoughtSpot nodes in VMware [See page 264]
- Configure ThoughtSpot nodes in Azure [See page 210]
- Deploying on the Dell Appliance [See page 4]
- Using the tscli cluster create command [See page 33]

# Network policies

**Summary:** Lists the required and optional ports for an installation.

For regular operations and for debugging, there are some ports you must keep open to network traffic from end users. Another, larger list of ports must be kept open for network traffic between the nodes in the cluster.

## Required ports for operations and debugging

The following ports must be open for requests from your user population.

Port	Protocol	Service Name	Direction	Source	Destination	Description
22	SSH	SSH	bidirectional	Administrators IP addresses	All nodes	Secure shell access. Also used for scp (secure copy).
443	HTTPS	HTTPS	bidirectional	All users IP addresses	All nodes	Secure HTTP.
12345	TCP	Simba	bidirectional	Administrators IP addresses	All nodes	Port used by ODBC and JDBC drivers when connecting to ThoughtSpot.

## Network Ports

This reference lists the potential ports to open when setting up your security group.

### Required ports for intracluster operation

Static ports are used for communication between services within the cluster. ThoughtSpot recommends that you open all ports within a cluster. This is not required, but it will ensure that cluster communication works properly if additional ports are used in a future software release.

If your organization does not allow you to open all ports, make sure you open the required intrACLuster ports listed in the following table. In addition, a number of ports are dynamically assigned to services, which change between runs. The dynamic ports come from the range of ports that are dynamically allocated by Linux (20K+).

Port	Protocol	Service Name	Direction	Source	Dest.	Description
80	TCP	nginx	inbound	All nodes	All nodes	Primary app HTTP port (nginx)
443	TCP	Secure nginx	inbound	All nodes	All nodes	Primary app HTTPS port (nginx)
2100	RPC	Oreo RPC port	bidirectional	All nodes	All nodes	Node daemon RPC
2101	HTTP	Oreo HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Node daemon HTTP
2181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
3181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
4181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
2200	RPC	Orion master RPC port	bidirectional	All nodes	All nodes	Internal communication with the cluster manager
2201	HTTP	Orion master HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the cluster manager

Port	Protocol	Service Name	Direction	Source	Dest.	Description
2205	TCP	Cluster update service TCP port	bidirectional	All nodes	All nodes	Internal communication with the cluster manager
2210	RPC	Cluster stats service RPC port	bidirectional	All nodes	All nodes	Internal communication with the stats collector
2211	HTTP	Cluster stats service HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the stats collector
2230	RPC	Callosum stats collector RPC port	bidirectional	All nodes	All nodes	Internal communication with the BI stats collector
2231	HTTP	Callosum stats collector HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the BI stats collector
2240	RPC	Alert manager	bidirectional	All nodes	All nodes	Port where alerting service receives alert events
2241	RPC	Alert manager	bidirectional	All nodes	All nodes	Port where alerting service receives alert events
2888	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
3181	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Dest.</b>	<b>Description</b>
3888	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
4000	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4001	HTTP	Falcon worker HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the data cache
4002	HTTP	Falcon worker HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the data cache
4003	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4004	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4021	RPC	Sage metadata service port (exported by Tomcat), Callosum services like meta-data services, medata-dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata

Port	Protocol	Service Name	Direction	Source	Dest.	Description
4181	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
4201	HTTP	Sage auto complete server HTTP interface port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4231	HTTP	Sage index server HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4232	RPC	Sage index server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4233	RPC	Sage index server RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4241	HTTP	Sage auto complete server HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4242	RPC	Sage auto complete server RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4243	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4244	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4245	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication

Port	Protocol	Service Name	Direction	Source	Dest.	Description
4243	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4251	RPC	Sage master RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4405	RPC	Diamond (graphite) port	bidirectional	All nodes	All nodes	Port used for communication with monitoring service
4406	RPC	Diamond (graphite) port	bidirectional	All nodes	All nodes	Port used for communication with monitoring service
4500	RPC	Trace vault service RPC port	bidirectional	All nodes	All nodes	Trace collection for ThoughtSpot services
4501	HTTP	Trace vault service HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug trace collection
4851	RPC	Graphite manager RPC port	bidirectional	All nodes	All nodes	Communication with graphite manager
4852	HTTP	Graphite manager HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug graphite manager
4853	RPC	Elastic search stack (ELK) manager RPC port	bidirectional	All nodes	All nodes	Communication with log search service
4853	HTTP	Elastic search stack (ELK) manager HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug log search service
9200	RPC	Elastic search (ELK)	bidirectional	All nodes	All nodes	Communication with log search service

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Dest.</b>	<b>Description</b>
5021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
5432	Postgres	Postgres database server port	bidirectional	All nodes	All nodes	Communication with Postgres database
6021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
7021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
8020	RPC	HDFS namenode server RPC port	bidirectional	All nodes	All nodes	Distributed file system (DFS) communication with clients
8021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
8080	HTTP	Tomcat	bidirectional	All nodes	All nodes	BI engine communication with clients
8081	HTTP	Callosum/Tomcat status	bidirectional	All nodes	All nodes	BI engine communication with clients
8787	HTTP	Periscope (UI) service HTTP port	bidirectional	All nodes	All nodes	Administration UI back end

Port	Protocol	Service Name	Direction	Source	Dest.	Description
8888	HTTP	HTTP proxy server (tinyproxy)	bidirectional	All nodes	All nodes	Reverse SSH tunnel
11211	Memcached	Memcached server port	bidirectional	All nodes	All nodes	BI engine cache
12345	ODBC	Simba server port	bidirectional	All nodes	All nodes	Port used for ETL (extract, transform, load)
8480	HTTP	HDFS journalnode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
8485	HTTP	HDFS journalnode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50070	HTTP	HDFS namenode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50090	HTTP	HDFS secondary namenode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50075	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
50010	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
50020	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
7000	TCP	Cassandra KV store database	bidirectional	All nodes	All nodes	Debug DFS data
7001	TCP	Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
9042	HTTP	Munshi server impression service, Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
9160	TCP	Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
4010	HTTP	Falcon moderator	bidirectional	All nodes	All nodes	Debug DFS data
4011	HTTP	Falcon moderator	bidirectional	All nodes	All nodes	Debug DFS data

Port	Protocol	Service Name	Direction	Source	Dest.	Description
20123 - 32768	TCP (dynamic)	Dynamic port in this range used for various services and ancillary services like atlas, caffeine, call-home, callosum, falcon, monitoring, munshi server, nlp, object_search, postgres, sage UBR, spo-tiq snapshot, timely	All nodes	Services		
5270	TCP	Cluster monitoring service (ELK)	bidirectional	All nodes	All nodes	Services
5271	TCP	Cluster monitoring service (ELK)	bidirectional	All nodes	All nodes	Services
5601	TCP	Kibana UI (ELK)	bidirectional	All nodes	All nodes	Services
6311	TCP	R service	bidirectional	All nodes	All nodes	Services
8008	TCP	Video recorder	bidirectional	All nodes	All nodes	Services
9090	TCP	Timely	bidirectional	All nodes	All nodes	Services
ICMPv4		Used for health check of cluster nodes	bidirectional	All nodes	All nodes	Services

### Required ports for inbound and outbound cluster access

ThoughtSpot uses static ports for inbound and outbound access to the cluster.

Port	Protocol	Service Name	Direction	Source	Dest.	Description
22	SCP	SSH	bidirectional	ThoughtSpot Support	All nodes	Secure shell access.
80	HTTP	HTTP	bidirectional	ThoughtSpot Support	All nodes	Hypertext Transfer Protocol for website traffic.

Port	Protocol	Service Name	Direction	Source	Dest.	Description
443	HTTPS	HTTPS	bidirectional	ThoughtSpot Support	All nodes	Secure HTTP.
12345	TCP	Simba	bidirectional	ThoughtSpot Support	All nodes	Port used by ODBC and JDBC drivers when connecting to ThoughtSpot.
2049	TCP	NFS: In case one needs to mount NFS share on TS node.	bidirectional	ThoughtSpot Support	All nodes	Port used by NFS.
123	UDP	NTP service	bidirectional	ThoughtSpot Support	All nodes	Port used by NTP service.

Port	Protocol	Service Name	Direction	Source	Destination	Description
443	TCP	HTTPS	outbound	All nodes	208.83.110.20	For transferring files to thoughtspot.egnyte.com.
443	TCP	HTTPS	outbound	All nodes	For transferring product usage data to mixpanel cloud.	outbound
443	TCP	HTTPS	outbound	All nodes	je8b47jfif.execute-api.us-east-2.amazonaws.com s3.us-west-1.amazonaws.com s3-us-west-1.amazonaws.com s3.dualstack.us-west-1.amazonaws.com	For transferring monitoring data to InfluxCloud. (Given address will resolve to point to AWS instances).
25 or 587	SMTP	SMTP or Secure SMTP	outbound	All nodes and SMTP relay (provided by customer)	All nodes	Allow outbound access for the IP address of whichever email relay server is in use. This is for sending alerts to ThoughtSpot Support.

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Destination</b>	<b>Description</b>
389 or 636	TCP	LDAP or LDAPS	outbound	All nodes and LDAP server (provided by cus- tomer)	All nodes	Allow outbound access for the IP address of the LDAP server in use.

### Required ports for IPMI (Intelligent Platform Management Interface)

ThoughtSpot uses static ports for out-of-band IPMI communications between the cluster and ThoughtSpot support.

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Dest.</b>	<b>Description</b>
80	HTTP	HTTP	bidirectional	ThoughtSpot Support	All nodes	Hypertext Transfer Protocol for website traffic.
443	TCP	S-HTTP	bidirectional	ThoughtSpot Support	All nodes	IPMI GUI and for HTML5-based IPMI console access.
623	UDP	Serial-over-LAN	bidirectional	ThoughtSpot Support	All nodes	IPMI GUI and for HTML5-based IPMI console access.

# Cable networking

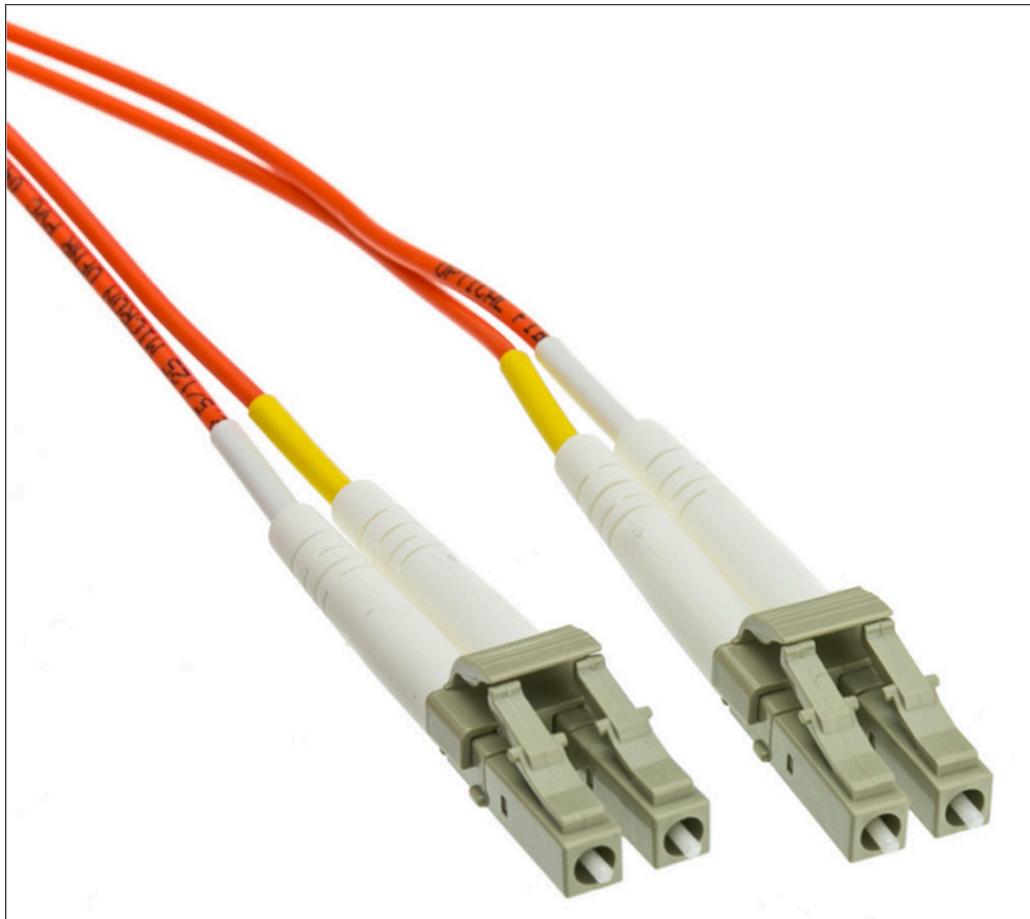
**Summary:** This section reviews the types of cables needed for 10GbE networking and how to plug them in.

There are three types of cables to consider for 10GbE networking:

- Fiber
- Direct Attach Copper (DAC)
- Category 6a (not supported by ThoughtSpot)

## Option 1 - Fiber cables

Fiber can be run long distances to the switch.

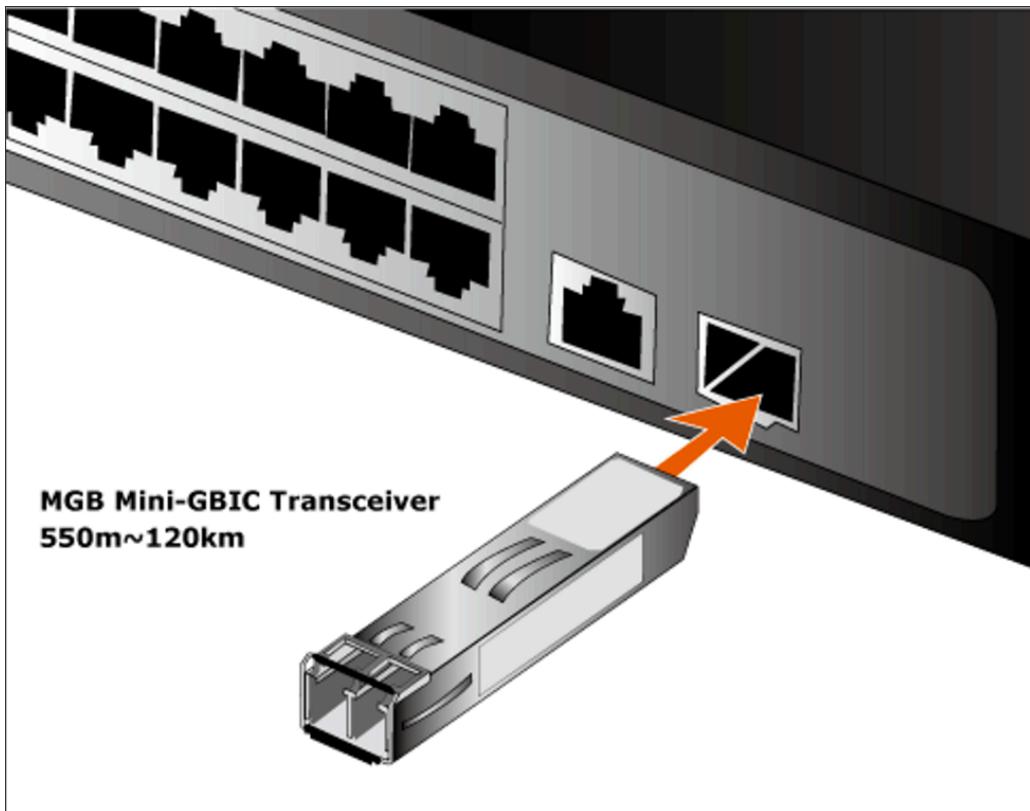


These cables require gigabit interface converters (GBICs), SFP+ form factor.

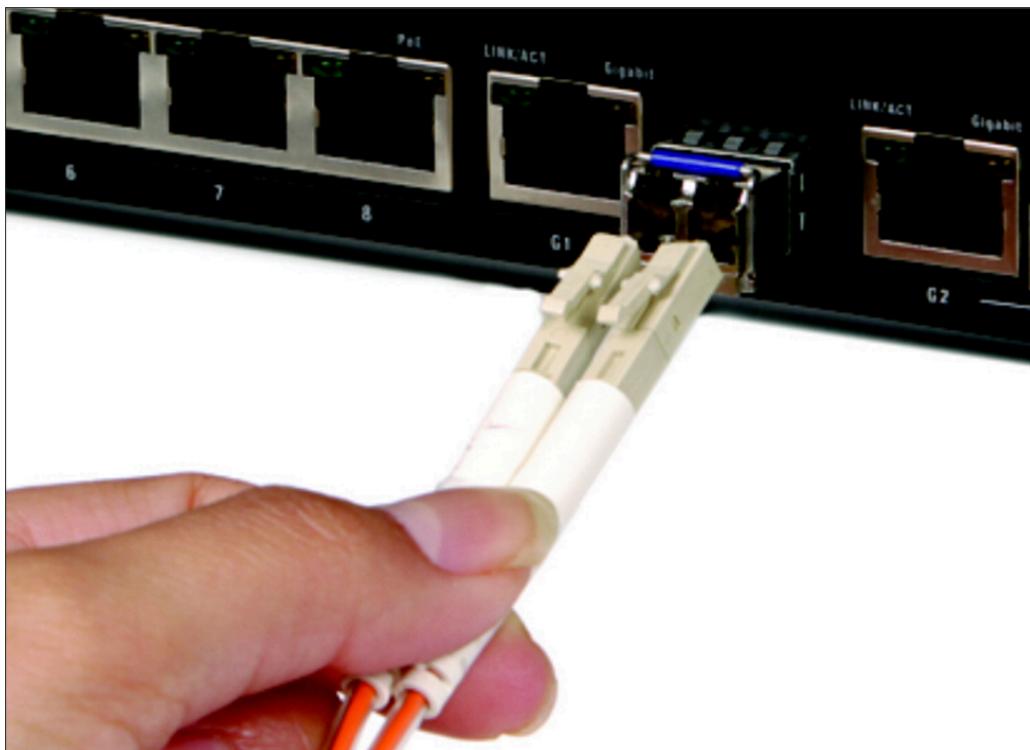
**Remember:** ThoughtSpot does not supply cables or GBICs



The GBIC must be plugged into a data port on the back of the appliance before plugging in the fiber cables.



The fiber cables must then be plugged into the GBIC.

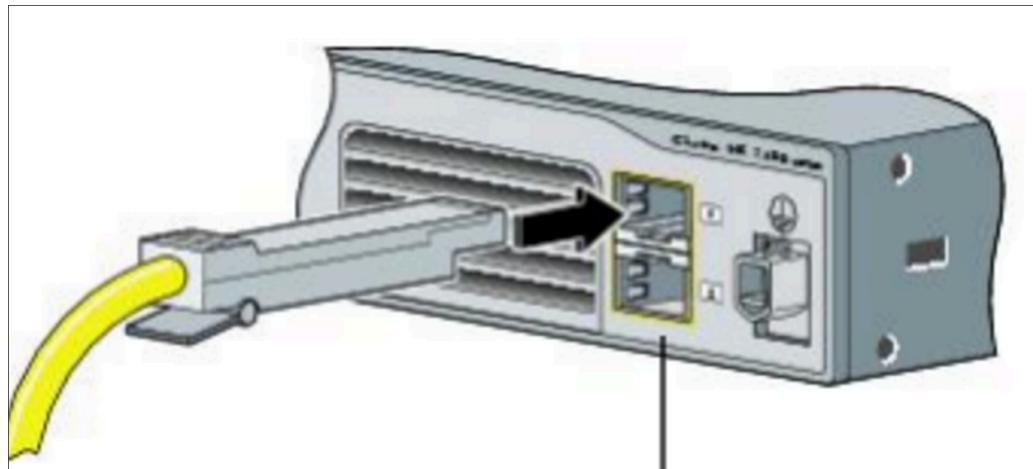


## Option 2 - DAC/Twinax cables

Copper can only be run short distances to the switch. An SFP+ is attached to the cable.



Here is how you would plug in a DAC cable.



## Non-option - Category 6a cables (not supported by ThoughtSpot)

There are no adapters for these cables. The 10GbE NIC (Network Interface Card) used on the ThoughtSpot appliance is not compatible with this type of cable/connection.



# Overview

**Summary:** You can install ThoughtSpot on a cloud provider, VMware, or on a hardware appliance.

You can install a ThoughtSpot cluster on a hardware appliance, cloud service, or VMware appliance. All ThoughtSpot nodes need to be on the same platform and same subnet with no connectivity blocked between any two nodes. You can have multiple clusters on multiple platforms, but you cannot host one cluster on multiple platforms.

This guide instructs you on how to prepare each of the following:

- [Hardware appliance \[See page 117\]](#)
- [Amazon Web Services \(AWS\) EC2 \[See page 169\]](#)
- [Microsoft Azure \[See page 199\]](#)
- [Google Cloud Platform \(GCP\) \[See page 226\]](#)
- [VMware \[See page 255\]](#)

After you configure your nodes, you can contact [ThoughtSpot Support \[See page 290\]](#) by phone, mail, email, or by filing a support ticket.

# Hardware appliance overview

## Summary: What is in the box.

The ThoughtSpot appliance hardware will be installed in a rack in your data center. This section describes the typical physical configuration.

## Hardware provided by ThoughtSpot

When your ThoughtSpot appliance arrives, the following items will be included:

Item Name	UOM	Qty
Round Hole to Sq Hole Adapter Kit (For Slide Rail Management)	Each	1
Power Cord, C13 to C14, 6 feet	Each	2
Power Cord, C13 to NEMA 5-15, 6 feet <sup>1</sup> <b>This power cord is not included with the Haswell platform.</b>	Each	2
Document, Rack Rail Installation, TS-2000	Each	1
TS-2000 Quick Start Guide	Each	1
Bezel Assembly, TS-2000	Each	1
Slide Rail Kit	Each	1
Appliance (containing 1-4 nodes, depending on ordered configuration)	Each	1
SFP+ Connector per ordered node (data connection)	Each	1
5m Fiber cable per ordered node (data connection)	Each	1
5m Network cable per ordered node (management connection)	Each	1

1: The supply voltage, 120 VAC, available when using a NEMA-15 power cord is an insufficient input to achieve the full power output required by the Haswell power supply. Only the C13 to C14 power cord should be used with the Haswell platform.



## Additional hardware requirements

You must supply the following items, as they will not be included with your ThoughtSpot appliance:

- Data center with proper cooling
- 2U of rack space per appliance (post depth 26.5" - 36.4")
- AC power **Attention:** Refer to [Hardware details \[See page 0\]](#) for power input requirements.
- 10GbE infrastructure (switch) - 1x port required / node
- 100MbE infrastructure (switch) - 1x port required /node
- Network cable Cat 5e/6 (node management)<sup>1</sup>
- 10G connection: SFP+ for switch side<sup>2</sup>

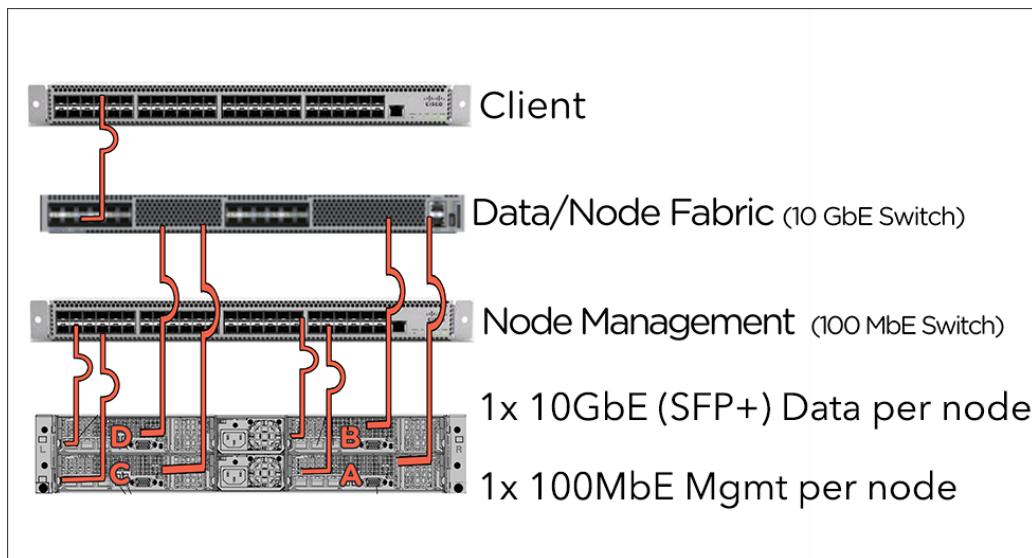
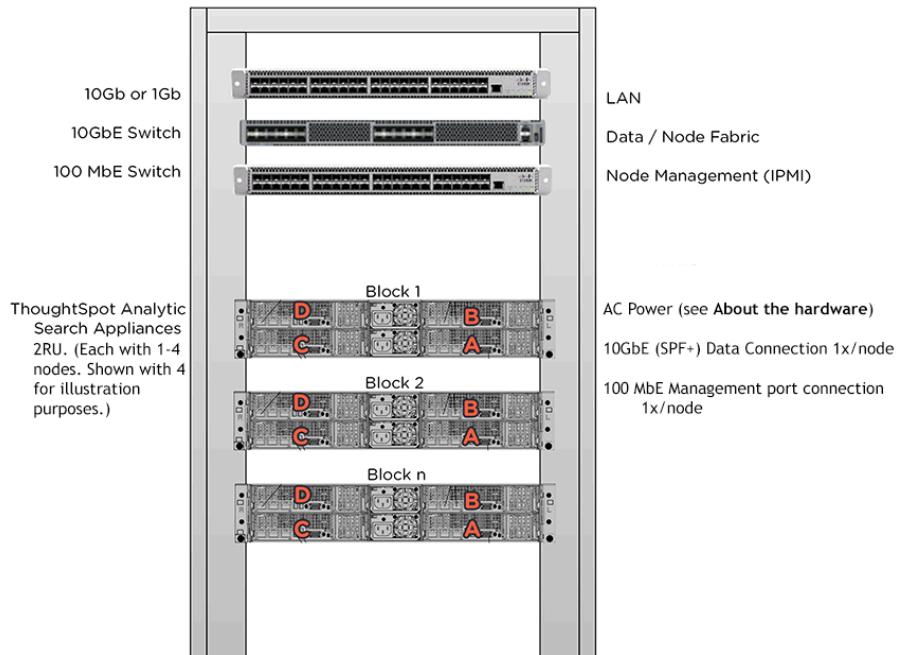
1. One 5m CAT 5e/6 network cable, per node, is provided with the appliance for management port connection. Customer supplied cable can be used if preferred.

2. One SFP+ connector is provided, per node, for the node side data connection. One 5m fiber cable is also provided. The customer must provide switch side SFP+ that is compatible with their switch. Customer supplied DAC cables or fiber cables can be used if preferred.

## Typical physical deployment

These diagrams show a physical configuration with three blocks of four nodes each. Your appliance can have 1-4 nodes, depending on the ordered configuration.

### Server Rack (42U) Back (Customer Supplied)



# Deploying on the SMC appliance

**Summary:** Follow these steps to deploy ThoughtSpot on your Super Micro Computer appliance.

Follow these steps to deploy ThoughtSpot on your Super Micro Computer (SMC) appliance.

- Step 1: Complete prerequisites [See page 61]
- Step 2: Review hardware requirements [See page 63]
- Step 3: Connect the SMC appliance [See page 66]
- Step 4: Configure nodes [See page 70]
- Step 5: Install cluster [See page 75]

## Related information

Use these references to aid you in successful installation and administration of ThoughtSpot.

- the `nodes.config` file [See page 0]
- Parameters of the `nodes.config` file [See page 30]
- Using the `tscli cluster create` command [See page 33]
- Parameters of the `cluster create` command [See page 41]
- Deployment Overview [See page 116]
- Contact Support [See page 290]

# Prerequisites

**Summary:** Complete these prerequisites before installing your ThoughtSpot clusters on the SMC appliance.

## Installation Prerequisites

Ensure that you have the following items, information, and understanding of policies before you begin deploying ThoughtSpot on your SMC appliance.

- Appliance Port Location [\[See page 0\]](#), to locate data and IPMI ports.
- Data center with proper environment controls, such as cooling.
- AC power
- 10GbE switch, with enabled IPv6 broadcast and multicast. You need one switch for each cluster, with one port for each node on the cluster.
- 10GbE network cables, either direct attach copper (DAC) or fiber. Refer to the [Cable reference \[See page 55\]](#) for more information to decide between the two types.
- 100Mbps or 1Gbps switch for IPMI, for Out of Band Management. You need one for each node in your cluster.
- Cat5 network cables, for IPMI management port use. You need one for each node.
- Rack space of 2U or 3.5 inches for each appliance, and a power strip
- VGA Monitor and USB keyboard
- 10G connection: SFP+ for the switch side
- Networking information: IP addresses for data & management NICs, for up to 2 DNS servers, up to 4 NTP servers and for the default gateway. Ensure that you configure only two DNS servers. ThoughtSpot does not support configuration of three DNS servers. You also need to know the timezone for your cluster. Typically, your timezone is where most of the people who will use the product are. [Download \[See page 0\]](#) and fill out the ThoughtSpot site survey so that you have a quick reference before beginning the install process. Ask your network administrator if you need help filling out the site survey.
- Network policies [\[See page 44\]](#), to determine the ports you need to have open for your cluster.

## Review hardware requirements

Next, [review hardware requirements \[See page 63\]](#).

# Hardware Requirements

**Summary:** Learn about your SMC hardware before deploying ThoughtSpot.

## About the Hardware

You can deploy ThoughtSpot on two different appliance hardware platforms: Haswell and Skylake. Both of the platforms provide the same performance. Refer to [Haswell and Skylake hardware details \[See page 63\]](#) for details on their physical differences.

Details	Haswell	Skylake
<b>Dimensions</b>	2 RU chassis (17.25" x 3.47" x 28.5" (WxHxD))	2 RU chassis (17.6" x 3.47" x 28.75" (WxHxD))
<b># of nodes</b>	Populated with 1 to 4 nodes	Populated with 1 to 4 nodes
<b>Node specifications</b>	Each node is independent and consists of a server board (removable from rear), 1x 200GB SSD, 3x .2TB HDD	Each node is independent and consists of a server board (removable from rear), 1x 240GB SSD, 3x 2TB HDD
<b>Max power consumption</b>	2000 W	2200 W
<b>Required power input</b>	200-240V / 11.8 - 9.8A / 50-60Hz (C13 / C14 power cords)	220-240 VAC 50-60 Hz (C13 / C14 power cords)

## Haswell front and back views

These images show the front and back views of each appliance.

The nodes on the front of both appliances go from A-D left to right. For this Haswell appliance, only Node D is populated.



The nodes on the back of both appliances are in a reverse N shape, with Node A at the bottom right and Node D at the top left.



The Haswell appliance shown here is not fully populated, as it only has three nodes. Your appliance may be populated with 1-4 nodes, depending on the ordered configuration. If you order less than four nodes, ThoughtSpot fills the empty slot with a filler panel.

## Skylake front and back views

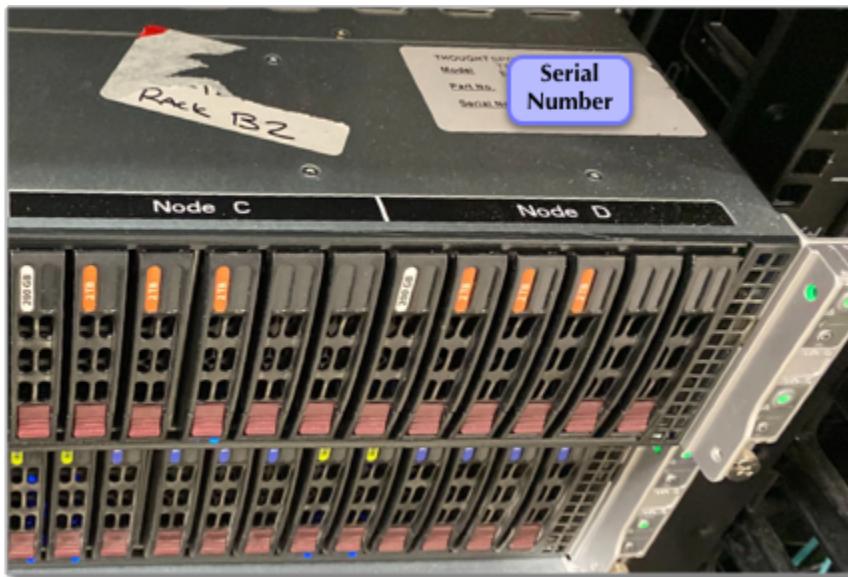




The Skylake appliance shown here is fully populated with four nodes.

## Location of serial number

You may need to know your appliance's serial number, to be able to access online help from your appliance provider. Find your Super Micro Computer's serial number on the top of the appliance, above the label for Node D at the front right corner.



## Connect the appliance

Next, [connect the appliance](#). [See page 66]

# Connect the SMC appliance

**Summary:** Connect the SMC appliance before you can deploy ThoughtSpot.

After you rack and stack the appliance, it is time to configure it. If necessary, review the [Hardware Appliance Overview \[See page 117\]](#). Follow the steps in this checklist.

- Step 1: Connect switches to 10GbE ports [See page 0]
- Step 2: Connect IPMI ports [See page 0]
- Step 3: Turn on nodes [See page 0]
- Step 4: Log in [See page 0]

## Step 1: Connect switches to 10GbE ports

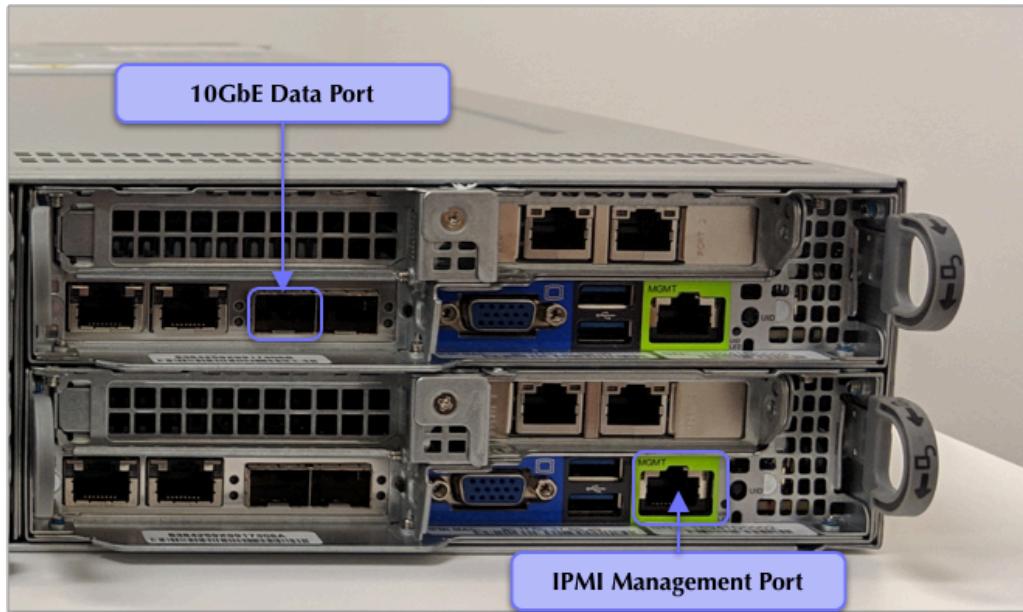
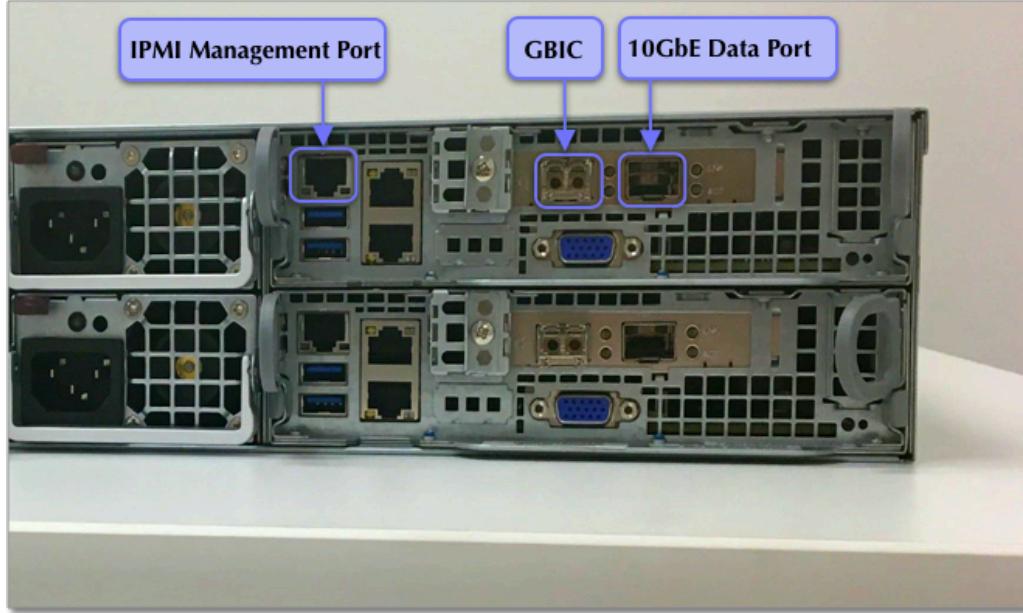
Connect the 10GbE port of each node, as illustrated in [Haswell port location \[See page 67\]](#) and [Skylake port location \[See page 67\]](#), to the 10GbE switches on your own rack, using either fiber or DAC cables.

Refer to the [Cable reference \[See page 55\]](#) for information on the cable types:

- [Fiber Cables \[See page 55\]](#)
- [DAC Cables \[See page 58\]](#)

**Note:** Ask your hardware vendor for more details about what they supply and what you need to buy.

Depending on which version of the SMC appliance you have, Haswell or Skylake, your 10GbE ports are in a different spot on the back of the appliance. Here is a picture of the back of each appliance.



- Connect to switches **only** the appliances (4 nodes each) that you plan to use in the cluster.
- You must power off, or disconnect from the switch, all other appliances or nodes.  
This prevents accidental configuration of incorrect nodes.
- Connect all the nodes that you plan to use in your cluster to a 10G switch.

## Step 2: Connect IPMI ports

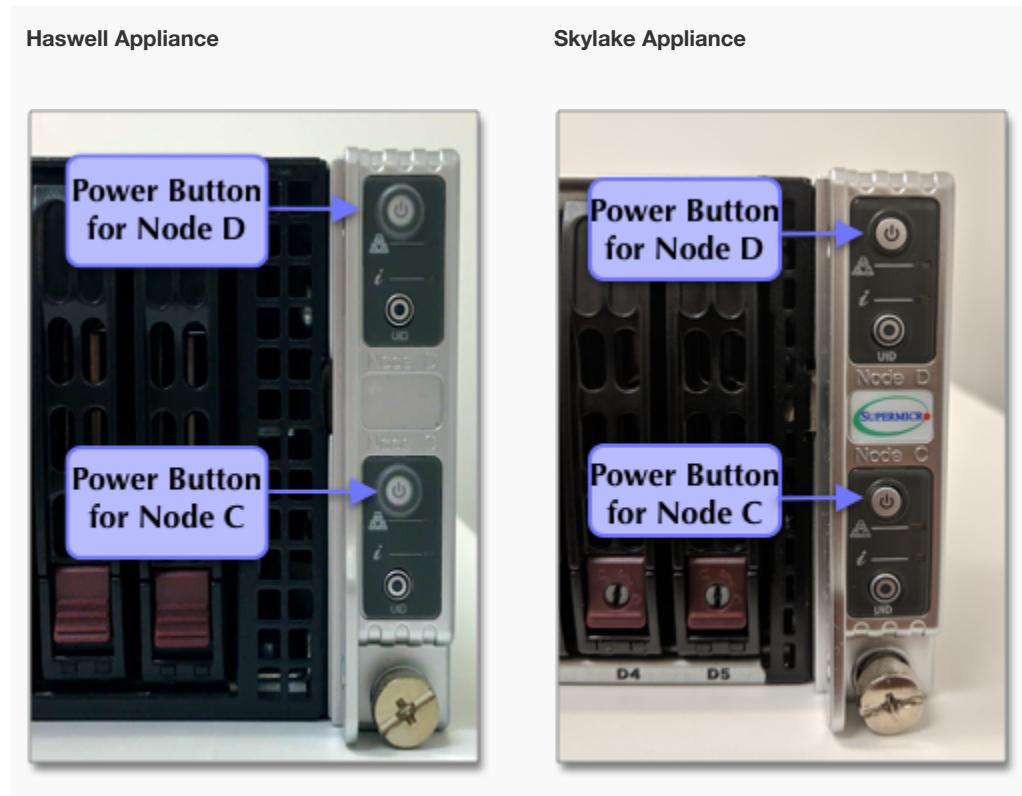
Connect the IPMI port of each node to the management switch.

See [Haswell port location \[See page 67\]](#) and [Skylake port location \[See page 67\]](#).

## Step 3: Turn on nodes

Turn on the power to the nodes by pressing the power button; see [Appliance Power Button \[See page 68\]](#).

There is one power button for each node. The images below show two of the four power buttons.



Check the health of the NIC by inspecting your appliance's LEDs. There should be a steady green light for the 100MB/s or 1Gb/s management NIC connection, a steady amber or orange light for the 10G/s connection, and a blinking LED on both NICs to indicate live traffic.

After the LEDs are the correct colors, verify that the connection is valid and the nodes are on by pinging the gateway:

Enter `ping <default-gateway-IP>`.

Ask your network administrator for your default gateway IP if you have not already listed it in your ThoughtSpot [site survey \[See page 0\]](#).

```
$ ping <default-gateway-IP>
```

## Step 4: Log in

1. Connect a keyboard and the mouse to each node on the appliance.
2. You should see a login prompt on the screen. If you don't see one or the screen isn't responsive, press the key combination **control-alt-F2** or **alt-F2** on your keyboard to bring up the login prompt.
3. Log in using the admin user credentials for the console. If you do not know the admin credentials, ask your network administrator.

## Configure nodes

Next, [configure nodes. \[See page 70\]](#)

# Configure ThoughtSpot Nodes on the SMC Appliance

**Summary:** Configure your nodes before you can install your cluster(s).

After you connect the appliance, configure the nodes in your Mac or Windows terminal emulator. Follow the steps in this checklist.

- Step 1: SSH into your cluster [See page 0]
- Step 2: Change to the `install` directory [See page 0]
- Step 3: Get a template for network configuration [See page 0]
- Step 4: Prepare node configuration [See page 0]
- Step 5: Configure the nodes [See page 0]
- Step 6: Confirm node configuration [See page 0]

If you completed ThoughtSpot's [site survey \[See page 0\]](#) form and returned it to [ThoughtSpot Support \[See page 290\]](#) before ThoughtSpot shipped the appliance, the appliance may be pre-configured for your network environment and ready to install and connect to your network.

If the network configuration was not pre-set, then this step must be done as part of the installation process.

Follow these steps to determine the configuration status of your appliance.

1. SSH into your cluster. Run `ssh admin@<nodeIP>`.  
Replace `nodeIP` with your specific network information.

```
$ ssh admin@<nodeIP>
```

2. Run `tscli cluster status`.

```
$ tscli cluster status
```

3. If the output shows READY, and looks like the [cluster status output \[See page 77\]](#) in the next article, your appliance is configured.
4. Skip to [Finalize installation \[See page 79\]](#).

If your status is not READY, continue with the installation process outlined below.

## Step 1: SSH into your cluster

SSH into your cluster with admin credentials.

1. Run the command `ssh admin@<cluster-IP>` or `ssh admin@<hostname>` on the command line.

Replace `clusterIP` or `hostname` with your specific network information.

```
$ ssh admin@<nodeIP>
```

2. Enter your admin password when prompted.

Ask your network administrator if you don't know the password.

## Step 2: Change to the install directory

In your terminal, change directory to `/home/admin/install` by running the command `cd /home/admin/install`. If your `/install` subdirectory does not exist, you may have to use the `/home/admin` directory.

```
$ cd /home/admin/install
```

## Step 3: Get a template for network configuration

Run the `tscli cluster get-config` command to get a template for network configuration. Redirect it to the file `nodes.config`. You can find more information on this process in the [nodes.config file reference \[See page 27\]](#).

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

## Step 4: Prepare node configuration

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the [autodiscovery of one node example \[See page 29\]](#). Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information.  
If you have additional nodes, complete each node within the `nodes.config` file in the same way.

Edit only the parts of the `nodes.config` file that are explicitly discussed in [Parameters of nodes.config \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, it may cause setup to fail.

## Step 5: Configure the nodes

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

Run `$ cat nodes.config | tscli cluster set-config` in your terminal.

If the command returns an error, refer to [set-config error recovery \[See page 81\]](#).

```
$ 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 6: Confirm node configuration

Use the `get-config` command to confirm node configuration.

Your output may look similar to the following:

```
$ 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 the cluster

Next, [install your cluster.](#) [See page 75]

# Install ThoughtSpot Clusters on the SMC Appliance

**Summary:** Install your clusters on the SMC appliance.

Install the cluster using the ThoughtSpot software release bundle. Installation takes approximately one hour. Make sure you can connect to ThoughtSpot remotely. If you can, you can run the installer on your local computer.

Refer to your welcome letter from ThoughtSpot to find the link to download the release bundle. If you have not received a link to download the release bundle, open a support ticket at [ThoughtSpot Support](#) [See page 0] to access the release bundle.

## Step 1. Run the Installer

1. Copy the downloaded release bundle to `/export/sdb1/TS_TASKS/install`. Run `scp <release-number>.tar.gz admin@<hostname>:/export/sdb1/TS_TASKS/install/<file-name>`.

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot installation, such as `6.0`, `5.3`, `5.3.1`, and so on.
- `hostname` is your specific hostname.
- `file-name` is the name of the tarball file on your local machine.

```
$ scp <release-number>.tar.gz admin@<hostname>:/export/sdb1/TS_TASKS/install/<file-name>
```

**Note:** You can use another secure copy method, if you prefer a method other than the `scp` command.

2. Alternatively, use `tscli fileserver download-release` to download the release bundle.

You must [configure the fileserver](#) [See page 0] by running `tscli fileserver configure`

before you can download the release.

```
$ tscli fileserver download-release <release-number> --  
user <username> --out <release-location>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 5.3.1, 6.0, and so on.
- `username` is the username for the fileserver that you set up earlier, when configuring the fileserver.
- `release-location` is the location path of the release bundle on your local machine. For example, `/export/sdb1/TS_TASKS/install/6.0.tar.gz`.

3. Verify the checksum to ensure you have the correct release.

Run `md5sum -c <release-number>.tar.gz.MD5checksum`.

```
$ md5sum -c <release-number>.tar.gz.MD5checksum
```

Your output says `ok` if you have the correct release.

4. Launch a [screen \[See page 0\]](#) session. Use screen to ensure that your installation does not stop if you lose network connectivity.

```
$ screen -S DEPLOYMENT
```

5. Create the cluster.

Run `tscli cluster create <release-number>`.

```
$ tscli cluster create <release-number>.tar.gz
```

6. Edit the output using your specific cluster information. For more information on this process, refer to [Using the tscli cluster create command \[See page 33\]](#) and [Parameters of the cluster create command \[See page 41\]](#).

The cluster installer automatically reboots all the nodes after the install. Wait at least 15 minutes for the installation process to complete. The system is rebooting, which takes a few minutes.

Log in to any node to check the current cluster status, using the command `tscli cluster status`.

## Step 2. Check Cluster Health

After you install the cluster, check its status using the `tscli cluster status` and `tscli cluster check` commands.

```
$ 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
```

Ensure that the cluster is `RUNNING` and that the Database and Search Engine are `READY`.

```
$ tscli cluster check
Connecting to hosts...
[Wed Jan  8 23:15:47 2020] START Diagnosing ssh
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing connection
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing zookeeper
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing sage
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing timezone
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing disk
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing cassandra
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing hdfs
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing orion-oreo
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing memcheck
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing ntp
[Wed Jan  8 23:16:08 2020] SUCCESS
#####
```

```
#####
[Wed Jan  8 23:16:08 2020] START Diagnosing trace_vault
[Wed Jan  8 23:16:09 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:09 2020] START Diagnosing postgres
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing disk-health
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing falcon
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing orion-cgroups
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing callosum
/usr/lib/python2.7/site-packages/urllib3/connectionpool.py:85
2: InsecureRequestWarning: Unverified HTTPS request is being ma
de. Adding certificate verification is strongly advised. See: h
ttps://urllib3.readthedocs.io/en/latest/advanced-usage.html#ss
l-warnings
    InsecureRequestWarning)
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
```

Your output may look something like the above. Ensure that all diagnostics show `SUCCESS`.

**● Warning:** If `tscli cluster check` returns an error, it may suggest you run `tscli storage gc` to resolve the issue. If you run `tscli storage gc`, note that it restarts your cluster.

## Step 3. Finalize Installation

After the cluster status changes to “Ready,” sign in to 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.

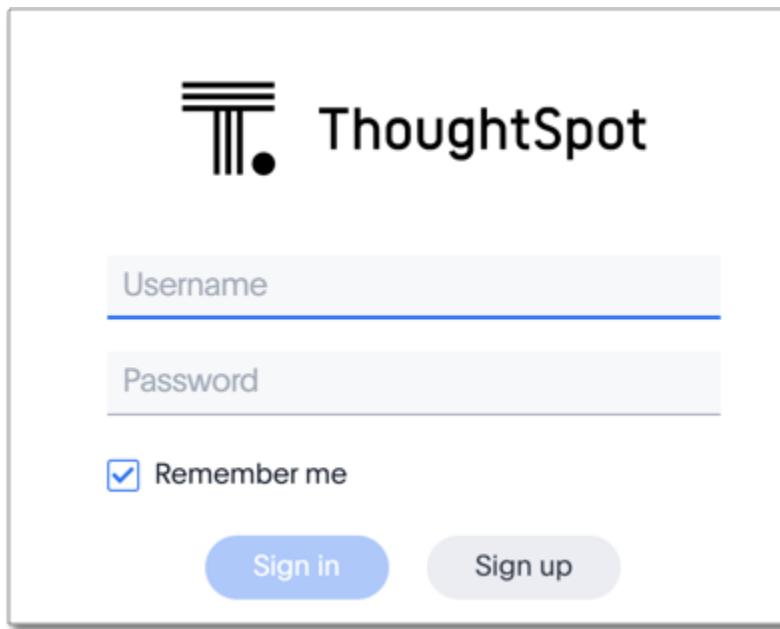
`https://<IP-address>`

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 sign-in page appears.
5. In the [ThoughtSpot sign-in window \[See page 80\]](#), enter admin credentials, and click **Sign in**.

If you do not know the admin credentials, ask your network administrator. ThoughtSpot recommends changing the default admin password.



## Lean configuration

(**For use with thin provisioning only**) If you have a [small or medium instance type \[See page 167\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 290\]](#) for assistance with this configuration.

## Error recovery

### Set-config error recovery

If you get a warning about node detection when you run the `set-config` command, restart the node-scout service.

Your error may look something like the following:

```
Connecting to local node-scout
WARNING: Detected 0 nodes, but found configuration for only 1 nodes.
Continuing anyway. Error in cluster config validation: [] is not a valid link-local IPv6 address for node: 0e:86:e2:23:8f:76
Configuration failed.
Please retry or contact support.
```

Restart the node-scout service with the following command.

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

Ensure that you restarted the node-scout by running `sudo systemctl status node-scout`. Your output should specify that the node-scout service is active. It may look something like the following:

```
$ sudo systemctl status node-scout
● node-scout.service - Setup Node Scout service
  Loaded: loaded (/etc/systemd/system/node-scout.service; enabled; vendor preset: disabled)
  Active: active (running) since Fri 2019-12-06 13:56:29 PS
          T; 4s ago
```

Next, retry the set-config command.

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

The command output should no longer have a warning.

# Deploying on the Dell appliance

**Summary:** Follow these steps to deploy ThoughtSpot on your Dell appliance.

Follow the steps in this checklist to deploy ThoughtSpot on your Dell appliance.

- Step 1: Complete installation prerequisites [See page 5]
- Step 2: Review hardware requirements [See page 6]
- Step 3: Connect your appliance [See page 9]
- Step 4: Configure management settings [See page 12]
- Step 5: Configure nodes [See page 15]
- Step 6: Install cluster [See page 18]

## Related information

Use these references to aid you in successful installation and administration of ThoughtSpot.

- the `nodes.config` file [See page 0]
- Parameters of the `nodes.config` file [See page 30]
- Using the `tscli cluster create` command [See page 33]
- Parameters of the `cluster create` command [See page 41]
- Cable Reference [See page 55]
- ThoughtSpot Documentation [See page 0]
- Contact Support [See page 290]

# Prerequisites

**Summary:** Complete these prerequisites to deploy ThoughtSpot on your Dell appliance.

## Installation prerequisites

Ensure that you have the following items, information, and understanding of policies before you begin installing your Dell 6420 appliance.

- 10gbE switch with IPv6 broadcast and multicast enabled. You need one switch for each cluster, with one port for each node on the cluster.
- Data center with proper environment controls, such as cooling.
- AC power
- 10G connection: SFP+ for the switch side
- 10GbE network cables, either direct attach copper (DAC) or fiber. See [Cable Reference \[See page 55\]](#) for more information to decide between the two types.
- 100Mbps or 1Gbps switch for connection to the iDRAC (Out of Band Management) port. You need one for each node in your cluster.
- Cat5 network cables, for iDRAC/ management port use. You need one for each node.
- Rack space of 2U or 3.5 inches for each appliance, and a power strip
- VGA Monitor and USB keyboard
- Networking information: IP addresses for data & management NICs, for up to 2 DNS servers, up to 4 NTP servers and for the default gateway. Ensure that you only configure two DNS servers. ThoughtSpot does not support configuration of three DNS servers. You also need to know the timezone for your cluster. Typically, your timezone is where most of the people who will use the product are. [Download \[See page 0\]](#) and fill out the ThoughtSpot site survey so that you have a quick reference. Contact your network administrator if you need help filling out the site survey.

## Review hardware requirements

Next, [review hardware requirements. \[See page 6\]](#)

# Hardware Requirements

**Summary:** Learn about the Dell hardware before deploying ThoughtSpot.

## About the Hardware

This table shows the hardware details and power requirements for your Dell C6420 appliance.

Details	Dell C6420
<b>Dimensions</b>	2 RU chassis (17.6" x 3.4" x 31.1" (WxHxD))
<b># of nodes</b>	Populated with 1 to 4 nodes
<b>Node specifications</b>	Each node is independent and consists of a server board (removable from rear), 1x 240GB SSD, 3x 2TB HDD
<b>Max power consumption</b>	2000 W
<b>Required power input</b>	200-240VAC / 11.5A max / 50 - 60 Hz (C19 / C20 power cords)

## Dell front and back views

These pictures show the front and back view of the Dell C6420 appliance.





## Location of serial number

You may need to know your appliance's serial number, to be able to access online help from your appliance provider. Find your Dell appliance's serial number on the plastic tab next to the control panel for Node 4, on the front right side of your appliance. Pull out the tab to view your service tag and a QR code for Dell support.





## Connect appliance

Next, [connect the appliance](#). [See page 9]

# Connect the Appliance

**Summary:** Connect your Dell appliance before you can deploy ThoughtSpot.

After you rack and stack the appliance, it is time to configure it. Follow the steps in this checklist.

- Step 1: Connect switches to 10GbE ports [See page 0]
- Step 2: Connect iDRAC ports [See page 0]
- Step 3: Connect a keyboard and monitor [See page 0]
- Step 4: Turn on nodes [See page 0]

## Step 1: Connect switches to 10GbE ports

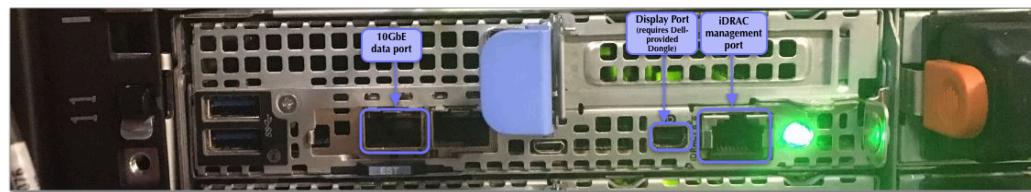
Connect the 10GbE port of each node, as illustrated in [Dell Port Location](#) [See page 9], to the 10GbE switches on your own rack, using either fiber or DAC cables.

Refer to the [Cable reference](#) [See page 55] for information on the cable types:

- [Fiber Cables](#) [See page 55]
- [DAC Cables](#) [See page 58]

**Note:** Ask your hardware vendor for more details about what they supply and what you need to buy.

The ports are on the back of the Dell appliance.



- Connect to switches **only** the appliances (4 nodes each) that you plan to use in your cluster.

- You must power off, or disconnect from the switch, any other nodes or appliances. This prevents accidental configuration of incorrect nodes.
- Connect all the nodes that you plan to use in your cluster to a 10G switch.

**Note:** You need at least three nodes for high availability (HA). Each appliance can have up to four nodes.

## Step 2: Connect iDRAC ports

Connect the iDRAC management ports of each node to the management switch.

See [Dell Port Location \[See page 9\]](#).

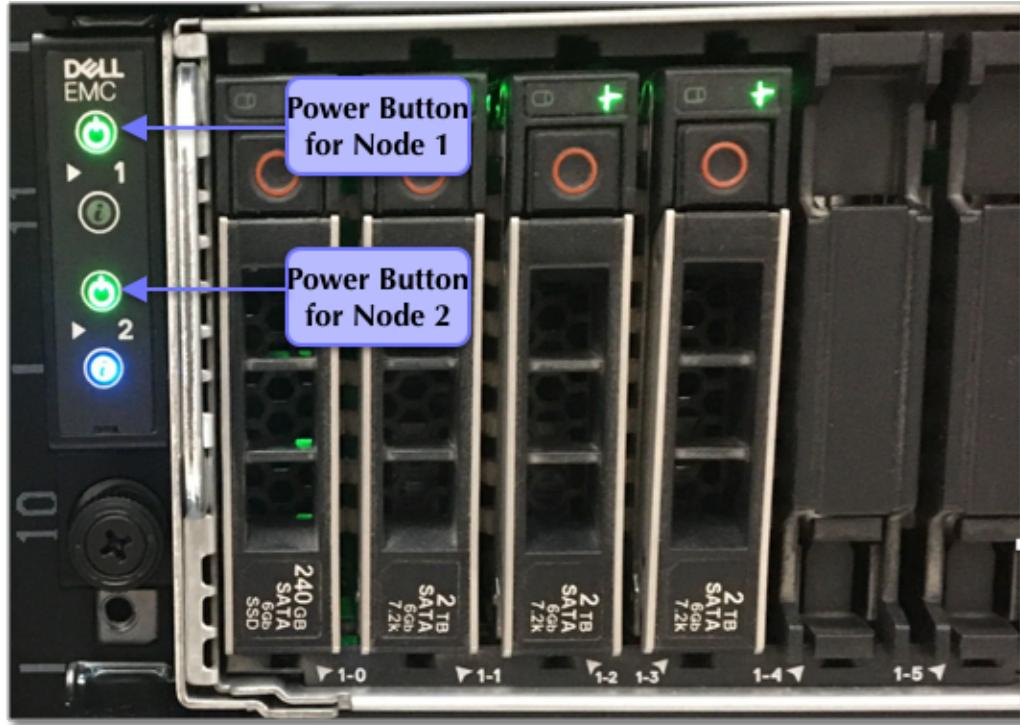
## Step 3: Connect a keyboard and monitor

Connect a keyboard and monitor to the appliance. You need these to initially configure the appliance, and you can disconnect them later. Use the adapter Dell provides. Plug it into the Display Port shown in [Dell Port Location \[See page 9\]](#), and plug the monitor in on the other side of the adapter.



## Step 4: Turn on nodes

Turn on power for the nodes by pressing the power button for each one; see [Dell Power Buttons \[See page 11\]](#).



There is one power button for each node. The image above shows two of the four power buttons.

## Configure the management settings

Next, [configure the management settings. \[See page 12\]](#)

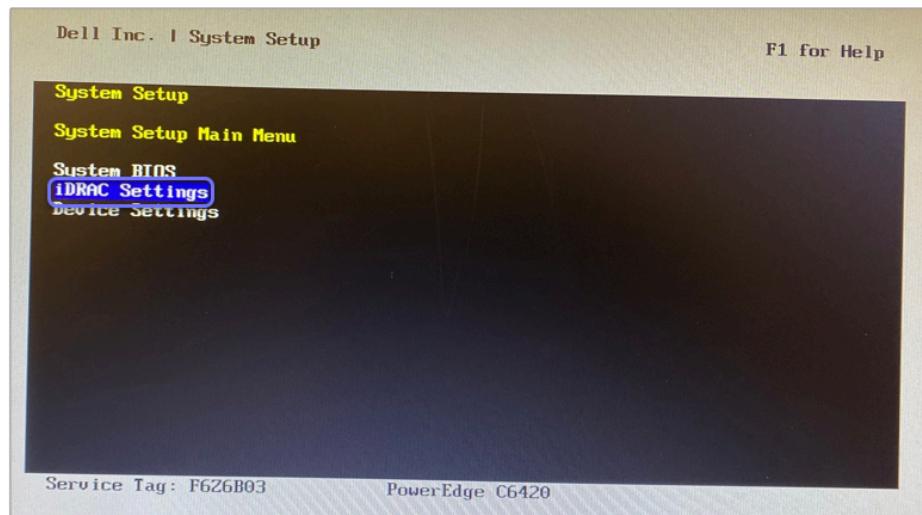
# Configure the Dell Management Settings

**Summary:** Configure the management settings for Dell before you can deploy ThoughtSpot.

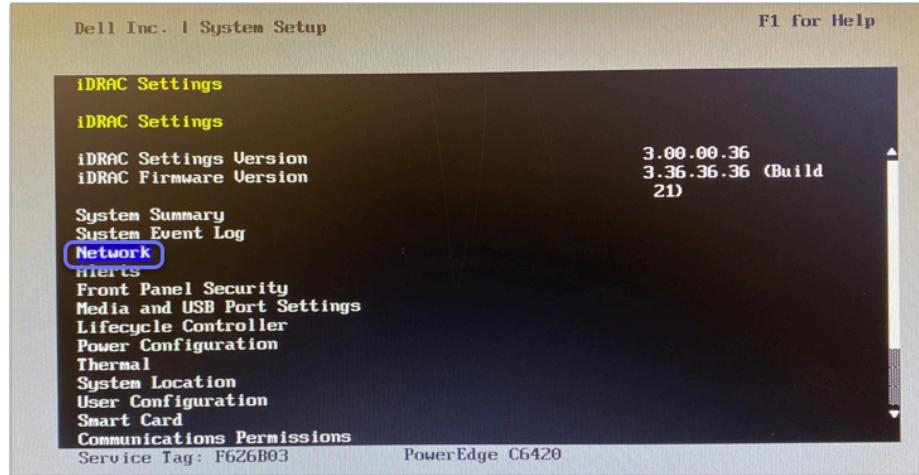
Input your specific network information to configure the management settings for your Dell appliance.

Refer to [Dell Management Configuration \[See page 14\]](#). If you need additional guidance, view [Dell Support \[See page 0\]](#) for this product.

1. **Open the iDRAC settings modal** Before the node boots, a screen appears on your monitor with several options. Click F2 to open the Bios setup menu.
2. **Select iDRAC** In the Bios setup screen, there are several options. Select **iDRAC settings** to configure your iDRAC management settings.



3. **Select network configuration** From the iDRAC settings options, select **network**.



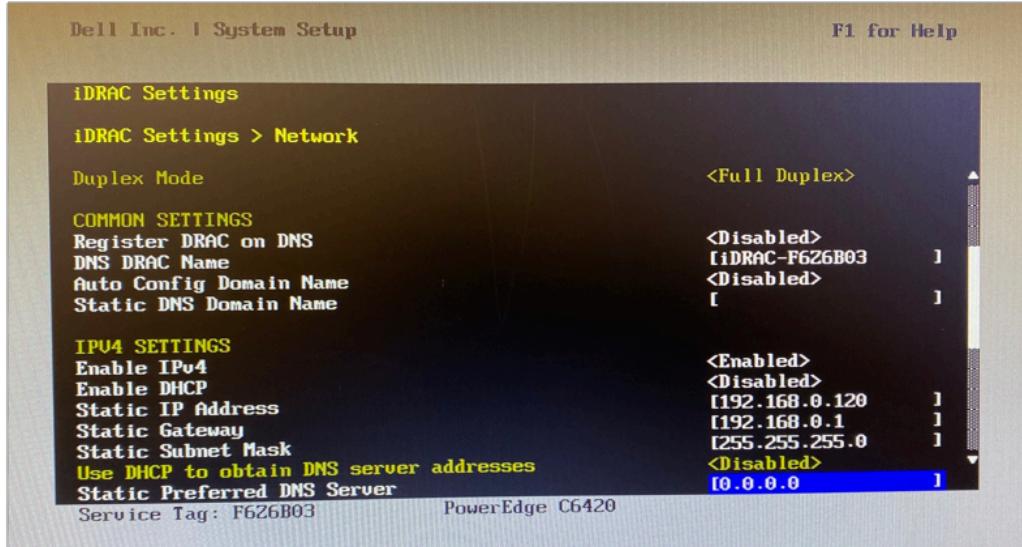
4. **Fill out the iDRAC settings form** Add your specific network information for the IP address, Gateway, and Netmask in the empty boxes. DNS information is optional. [Download \[See page 0\]](#) and fill out the ThoughtSpot site survey for a quick reference, and ask your network administrator for help if you have not filled out the site survey yet.

**● Warning:** If you configure DNS servers, you must only configure two servers.

ThoughtSpot does not support configuration of three DNS servers.

- For **Enable IPv4**, select **enabled**.
- For **Enable DHCP**, select **disabled**.

5. **Save changes and reboot** Follow the prompts on the monitor to save changes to the management settings form, exit, and reboot the system.
6. **Log in to ThoughtSpot** After the system reboots, the login page appears. Log in as an administrator. Ask your network administrator if you do not know the admin credentials.



## Configure nodes

Next, configure nodes. [See page 15]

# Configure Nodes

**Summary:** Configure ThoughtSpot nodes on your Dell appliance.

After you connect the appliance, a command line appears on your console. Configure the nodes on this command line. Follow the steps in this checklist.

- Step 1: Get a template for network configuration [See page 0]
- Step 2: Prepare node configuration [See page 0]
- Step 3: Configure the nodes [See page 0]
- Step 4: Confirm node configuration [See page 0]

## Step 1: Get a template for network configuration

Make sure you have logged into your cluster. If you have not, use admin credentials to log in to your cluster.

Run the `tscli cluster get-config` command to get a template for network configuration. Redirect it to the file `nodes.config`.

You can find more information on this process in the `nodes.config` file reference [See page 27].

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

## Step 2: Prepare node configuration

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the [autodiscovery of one node example \[See page 29\]](#). Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information.

If you have additional nodes, complete each node within the `nodes.config` file in the same way.

Do not edit any part of the `nodes.config` file except the sections described in [Parameters of the nodes.config file \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, it may cause setup to fail.

## Step 3: Configure the nodes

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

Run `$ cat nodes.config | tscli cluster set-config` in your terminal.

If the command returns an error, refer to [set-config error recovery \[See page 0\]](#).

```
$ 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 NTP Servers
Setting up Timezone
Done setting up ThoughtSpot
```

## Step 4: Confirm node configuration

Use the `get-config` command to confirm node configuration.

Your output may look similar to the following:

```
$ 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 your cluster

Next, [install your cluster. \[See page 18\]](#)

# Install Cluster

**Summary:** Install your ThoughtSpot cluster(s) on your Dell appliance.

Install the cluster using the ThoughtSpot software release bundle. Installation takes approximately one hour. Make sure you can connect to ThoughtSpot remotely. If you can, you can run the installer on your local computer.

Refer to your welcome letter from ThoughtSpot to find the link to download the release bundle. If you do not have a link to download the release bundle, open a support ticket at [ThoughtSpot Support \[See page 290\]](#) to access the release bundle.

Follow the steps in this checklist to install your cluster.

- Step 1: Run the installer [See page 0]
- Step 2: Check cluster health [See page 0]
- Step 3: Finalize installation [See page 0]

## Step 1: Run the installer

1. Copy the downloaded release bundle to `/export/sdb1/TS_TASKS/install`:

Run `scp <release-number> admin@<hostname>:/export/sdb1/TS_TASKS/install/<file-name>`. Note the following parameters:

- `release-number` is the release number of your ThoughtSpot installation, such as `6.0`, `5.3`, `5.3.1`, and so on.
- `hostname` is your network hostname. Ask your network administrator if you do not know your hostname.
- `file-name` is the name of the tarball file on your local computer.

```
$ scp <release-number>.tar.gz admin@<hostname>:/e  
xport/sdb1/TS_TASKS/install/<file-name>
```

**Note:** You can use another secure copy method, if you prefer a method other than the `scp` command.

2. Alternatively, use `tscli fileserver download-release` to download the release bundle.

You must [configure the fileserver \[See page 0\]](#) by running `tscli fileserver configure` before you can download the release.

```
$ tscli fileserver download-release <release-number> --user <username> --out <release-location>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 5.3.1, 6.0, and so on.
- `username` is the username for the fileserver that you set up earlier, when configuring the fileserver.
- `release-location` is the location path of the release bundle on your local machine. For example, `/export/sdb1/TS_TASKS/install/6.0.tar.gz`.

3. Verify the checksum to ensure you have the correct release.

Run `md5sum -c <release-number>.tar.gz.MD5checksum`.

```
$ md5sum -c <release-number>.tar.gz.MD5checksum
```

Your output says `ok` if you have the correct release.

4. Launch a [screen \[See page 0\]](#) session. Use screen to ensure that your installation does not stop if you lose network connectivity.

```
$ screen -S DEPLOYMENT
```

5. Create the cluster using `tscli cluster create <release-number>`.

```
$ tscli cluster create <release.number>.tar.gz
```

6. Edit the output using your specific cluster information. For more information on this process, refer to [Using the tscli cluster create command \[See page 33\]](#) and [Parameters of the cluster create command \[See page 41\]](#).

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

## Step 2: Check cluster health

After you install the cluster, check its status using the `tscli cluster status` command.

Your output may look something like the following:

```
$ 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
```

Ensure that the cluster is `RUNNING` and that the Database and Search Engine are `READY`.

```
$ tscli cluster check
Connecting to hosts...
[Wed Jan  8 23:15:47 2020] START Diagnosing ssh
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing connection
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing zookeeper
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing sage
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing timezone
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing disk
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing cassandra
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing hdfs
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing orion-oreo
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing memcheck
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing ntp
[Wed Jan  8 23:16:08 2020] SUCCESS
#####
```

```
#####
[Wed Jan  8 23:16:08 2020] START Diagnosing trace_vault
[Wed Jan  8 23:16:09 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:09 2020] START Diagnosing postgres
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing disk-health
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing falcon
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing orion-cgroups
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing callosum
/usr/lib/python2.7/site-packages/urllib3/connectionpool.py:85
2: InsecureRequestWarning: Unverified HTTPS request is being ma
de. Adding certificate verification is strongly advised. See: h
ttps://urllib3.readthedocs.io/en/latest/advanced-usage.html#ss
l-warnings
    InsecureRequestWarning)
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
```

Your output may look something like the above. Ensure that all diagnostics show `SUCCESS`.

**● Warning:** If `tscli cluster check` returns an error, it may suggest you run `tscli storage gc` to resolve the issue. If you run `tscli storage gc`, note that it restarts your cluster.

## Step 3: Finalize installation

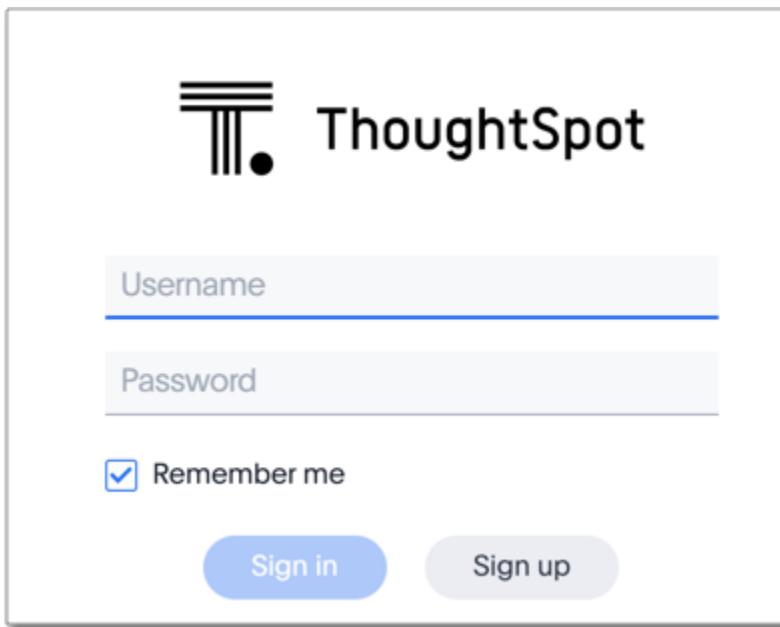
After the cluster status changes to `READY`, sign in to the ThoughtSpot application on your browser.

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 sign-in window \[See page 24\]](#), enter admin credentials, and click **Sign in**.  
ThoughtSpot recommends changing the default admin password.



## Lean configuration

**(For use with thin provisioning only)** If you have a [small or medium instance type \[See page 167\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 290\]](#) for assistance with this configuration.

## Error recovery

### Set-config error recovery

If you get a warning about node detection when you run the `set-config` command, restart the node-scout service.

Your error may look something like the following:

```
Connecting to local node-scout
WARNING: Detected 0 nodes, but found configuration for
only 1 nodes.
Continuing anyway. Error in cluster config validation: [] is not a valid link-local
IPv6 address for node: 0e:86:e2:23:8f:76 Configuration failed.
Please retry or contact support.
```

Restart the node-scout service with the following set of commands.

```
$ sudo vim /etc/systemd/system/node-scout.service
$ sudo systemctl daemon-reload
$ sudo systemctl restart node-scout
```

Ensure that you restarted the node-scout by running `sudo systemctl status node-scout`. Your output should specify that the node-scout service is active. It may look something like the following:

```
$ sudo systemctl status node-scout
● node-scout.service - Setup Node Scout service
  Loaded: loaded (/etc/systemd/system/node-scout.service; enabled; vendor preset: disabled)
  Active: active (running) since Fri 2019-12-06 13:56:29 PST; 4s ago
```

Next, retry the `set-config` command.

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

The command output should no longer have a warning.

# Cloud overview

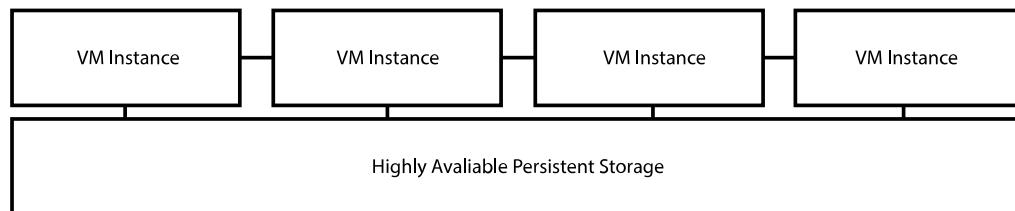
**Summary:** You can deploy ThoughtSpot on several cloud providers.

ThoughtSpot can currently be deployed in the following cloud provider environments:

- [Amazon Web Services \(AWS\) EC2 \[See page 169\]](#)
- [Microsoft Azure \[See page 199\]](#)
- [Google Cloud Platform \(GCP\) \[See page 226\]](#)

The ThoughtSpot cloud deployment consists of cloud compute (VM) instances and an underlying persistent storage layer. The number of instances required for a cloud deployment is based on the size of the data that needs to be analyzed in ThoughtSpot. The instances act as a distributed cluster of nodes to serve query responses.

	AWS	Azure	GCP
<b>Compute</b>	Virtual machines deployed in your AWS VPC	Virtual machines in your Azure VNET	Virtual machines in your GCP VPC
<b>Persistent storage</b>	Deployment options: 1. Elastic Block Storage 2. S3 + Elastic Block Storage	Premium SSD Managed Disks	Zonal SSD persistent disk



To determine the number of instances and the persistent storage requirements to provision your cluster, please refer to the available instance types for your cloud service provider in the next section.

## ThoughtSpot cloud instance types

Refer to the following guidelines for how to set up ThoughtSpot on each cloud service:

- [AWS instance types \[See page 0\]](#)
- [Azure instance types \[See page 199\]](#)
- [GCP instance types \[See page 226\]](#)

## Reducing your cloud infrastructure costs

ThoughtSpot recommends following these guidelines to help reduce the cost of your cloud deployment.

### Use small and medium instance types when applicable

ThoughtSpot has various instance types to suit your needs. For ThoughtSpot customers who are deploying their instance with lower data sizes (<=100 GB), ThoughtSpot supports “small” (20 GB data) and “medium” (100 GB data) instance types, as provided at the links above, to help reduce the costs of cloud infrastructure. These are instances with lower CPU/RAM sizes (16/32 vCPU and 128 GB/256 RAM). Advanced lean configuration is required before any data can be loaded onto these instances.

Please contact ThoughtSpot support for assistance with this configuration.

### Shut down and restart your cluster

If you do not need your ThoughtSpot cluster to be up and running 24/7, you can shut down your cluster and restart it during normal usage hours to save on the infrastructure costs of running ThoughtSpot instances in cloud provider environments.

To shut down and restart your cluster, do the following in the tscli:

1. Ensure there are no issues with the cluster by running: `$ tscli cluster check`

The above command should return no failure messages.

2. Stop the cluster by running: `$ tscli cluster stop`

Wait until you see the message: “Done stopping cluster”

3. Go to your cloud provider’s console and shut down all of the ThoughtSpot VMs in your cluster.

4. When you are ready to use ThoughtSpot again, start up your node VMs.

5. Restart your cluster by running: `$ tscli cluster start`

You should see the message: "Started pre-existing cluster"

Depending on the size of your cluster, you may need to wait several minutes before the system is up and running. Make sure you budget for this startup time to ensure that the system is fully operational before you expect people to use it.

6. Ensure that your cluster is ready for use by running: `$ tscli cluster status`

The following messages are displayed to indicate your cluster is up and running:

```
... Cluster: RUNNING  
Database: READY  
Search Engine: READY
```

## Automating your cloud deployment

You can automate your deployment, using the free tools in the [ThoughtSpot Cloud Deployment GitHub repository](#) [See page 0].

For more information about automating your cloud deployment, read [Deploying ThoughtSpot in the Cloud Using Terraform and Ansible](#) [See page 0].

# AWS configuration options

**Summary:** Your 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 procedures. For assistance in choosing the best mode for your organization, contact your ThoughtSpot representative. For more information on purchasing ThoughtSpot in AWS, see: [ThoughtSpot Pricing \[See page 0\]](#).

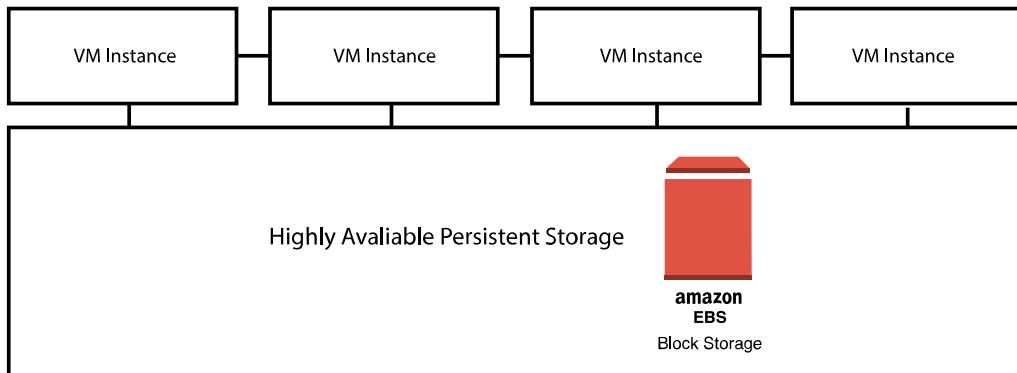
All AWS VMs in a ThoughtSpot cluster must be in the same availability zone (and therefore, also in the same region). ThoughtSpot does not support deploying VMs in the same cluster across availability zones. For more information, see [Regions and Availability Zones \[See page 0\]](#) in Amazon's AWS documentation.

## 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.

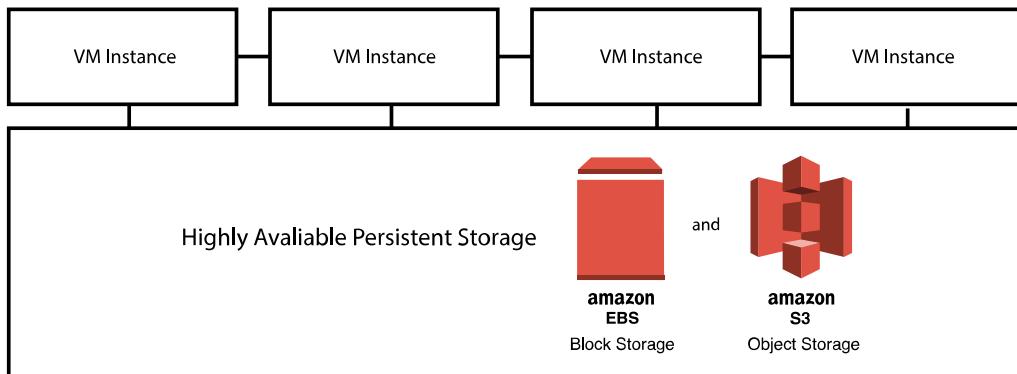
For example: If you were deploying a total cluster data size of 1 TB using the standard r5.16xlarge instance type, you would need 4 VM instances, because the instance type supports data capacity of 250 GB. The data volumes on the EBS would need to be provision with 2x1 TB volumes per VM.

### 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 \[See page 0\]](#)
- [EC2 pricing \[See page 0\]](#)
- [EBS pricing \[See page 0\]](#)
- [Placement groups \[See page 0\]](#)

# Set up AWS resources for ThoughtSpot

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

## Overview of ThoughtSpot setup in AWS

Follow these steps to set up your ThoughtSpot VMs in AWS.

- 1. Gain access to ThoughtSpot's AMI. [See page 0]
- 2. Choose a VM instance configuration recommended by ThoughtSpot. [See page 0]
- 3. Set up your Amazon S3 bucket (optional). [See page 0]
- 4. Set up your ThoughtSpot cluster in AWS. [See page 0]
- 5. Configure security groups. [See page 0]
- 6. Open the required network ports for communication for the nodes in your cluster and end users. [See page 0]
- 7. Prepare the VMs. [See page 0]

## 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 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 provisions 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.
- Networking information: [download \[See page 0\]](#) and fill out the ThoughtSpot site survey to have a quick reference point. Ask your network administrator if you need help filling out the site survey.

For more information about IAM, see: [What Is IAM? \[See page 0\]](#) in Amazon's AWS documentation.

### Required AWS components

- An AWS Virtual Private Cloud (VPC). An AWS VPC is a virtual network specifically for your AWS account. It exists in all availability zones in your region, but you can specify a local zone for even lower latency. For more details, see [VPCs and Subnets \[See page 0\]](#) in Amazon's AWS documentation.
- A ThoughtSpot AMI. For details, see [Setting up your EC2 instances \[See page 0\]](#).
- AWS security groups. For required open ports, see [Network Policies \[See page 0\]](#).
- AWS VM instances. For instance type recommendations, see [ThoughtSpot AWS instance types \[See page 0\]](#).
- EBS volumes for data storage.
- (Optional) If deploying with S3 persistent storage, you need one S3 bucket for each ThoughtSpot cluster.

## Setting up your EC2 instances

1. Sign in to your [AWS account \[See page 0\]](#).
2. Copy the following ThoughtSpot public AMI to your AWS region:

**AMI Name:** thoughtspot-image-20191031-8ae15008336-prod

**AMI ID:** ami-06276ece42ed96994

**Region:** N. California

**Note:** The AMI is based in the N. California region. You may have to temporarily switch to the N. California region on the AWS website to access it. Then you can return to your own region.

**Note:** The AMI is backward-compatible with ThoughtSpot releases 5.1.x - 6.0.x.

3. Choose the appropriate EC2 instance type: See [ThoughtSpot AWS instance types \[See page 169\]](#) for help choosing the correct instance type for your cluster.
4. Networking requirements: 10 GbE network bandwidth is needed between the VMs. Ensure that you have this bandwidth.
5. Ensure that all your VMs are on the same Amazon Virtual Private Cloud (VPC) and subnet. This is necessary because VMs that are part of a cluster need to be accessible by each other. Additional external access may be required to bring data in/out of the VMs to your network. Add all nodes in the same placement group.
6. Determine the number of EC2 instances you need: Based on the datasets, this number will vary. Refer to [ThoughtSpot AWS instance types \[See page 169\]](#) for recommended nodes for a given data size.

**Note:** Staging larger datasets (> 50 GB per VM), may require provisioning additional attached EBS volumes that are SSD (gp2).

## Setting up your Amazon S3 bucket (recommended)

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 \[See page 0\]](#) to find out if your specific cluster size can benefit from the S3 storage option.

Follow these steps to set up an S3 bucket in AWS.

1. On the AWS website, 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.
5. Select your region.
6. Click **Next**.
7. On the **Properties** page, click **Next**.
8. On the Configure options page, ensure that **Block all public access** is selected.
9. Click **Next**.
10. On the Set permissions page, click **Create bucket**.

## Encrypting your data at rest

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 or AWS KMS.

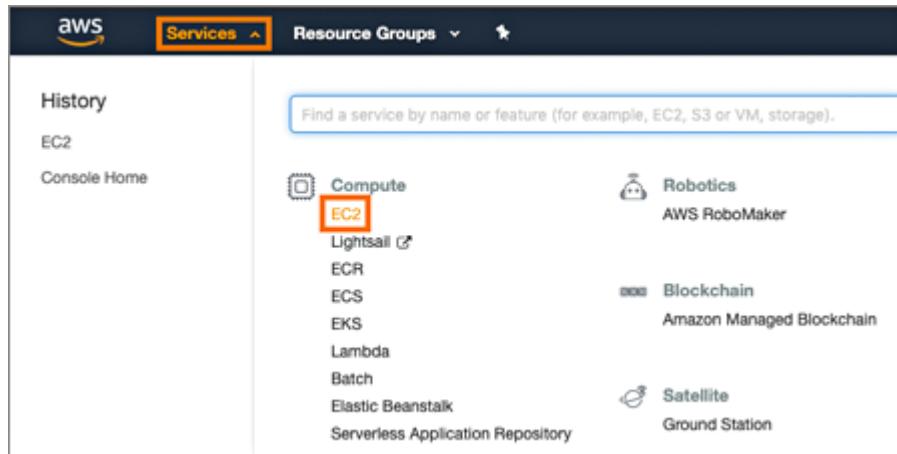
For more information on encryption supported with AWS:

- For EBS, see [Amazon EBS Encryption \[See page 0\]](#) in Amazon's AWS documentation.
- For S3, see [Amazon S3 Default Encryption for S3 Buckets \[See page 0\]](#) in Amazon's AWS documentation.

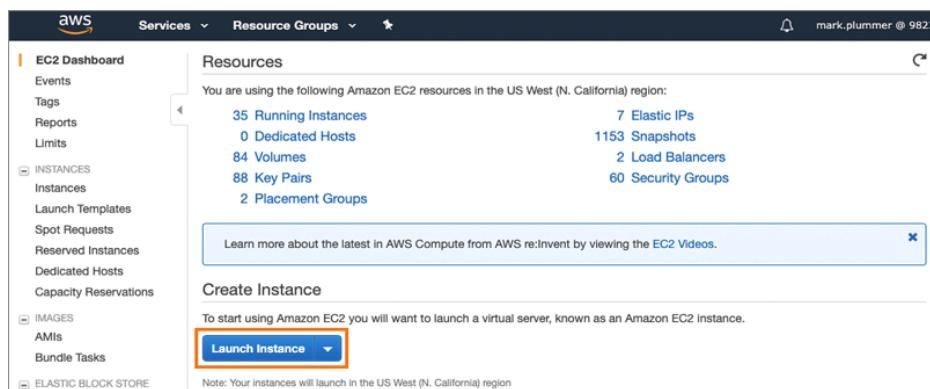
## Setting up your ThoughtSpot cluster

To set up a ThoughtSpot cluster in AWS, follow these steps:

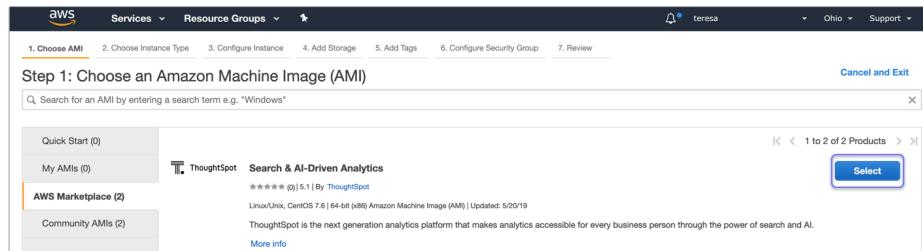
1. On the AWS website, navigate to the EC2 service dashboard by clicking **Services**, then **EC2**.



2. Make sure your selected region is correct in the top-right corner of the dashboard. If not, select your region. Let ThoughtSpot support know if you change your region.
3. Start the process of launching a VM by clicking **Launch Instance**.



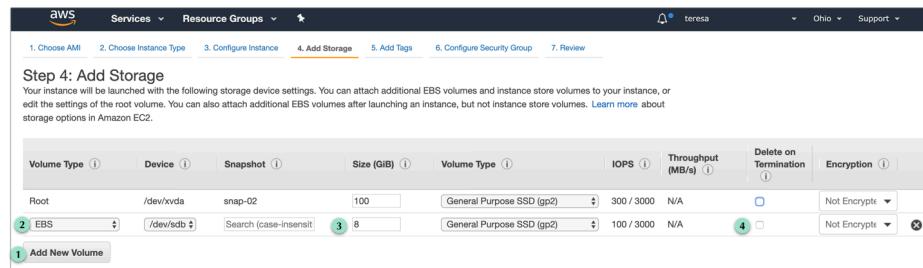
4. In the **My AMIs** tab under **1. Choose AMI**, search **ThoughtSpot** to find the ThoughtSpot AMI.
5. Click **Select**. Ensure that you select the ThoughtSpot AMI listed [above \[See page 174\]](#), which you entered earlier in this process.



6. On the **Choose an Instance Type** page, select a ThoughtSpot-supported instance type. (See [ThoughtSpot AWS instance types \[See page 169\]](#).)
7. Click **Next: Configure Instance Details**.
8. Configure the instances by choosing the number of EC2 instances you need. The instances must be on the same VPC and subnetwork. ThoughtSpot sets up the instances to be in the same ThoughtSpot cluster.

**S3 storage setting:** If you are going to use the S3 storage option, ThoughtSpot recommends that you restrict access to a specific S3 bucket. Create a new IAM role that provides read/write access to the specific bucket, and select it. For details on that, click **Create new IAM role**.

9. Click **Next: Add Storage**. Add the required storage based on your instance type (either EBS volumes or S3), and the amount of data you are deploying. For specific storage requirements, refer to [ThoughtSpot AWS instance types \[See page 169\]](#).



- 1 Click **Add new volume**.
- 2 Specify the type of storage, either EBS or S3.
- 3 Specify the size of the volume.
- 4 Ensure that you leave **Delete on termination** unchecked, to prevent potential loss of data if the VM is accidentally terminated.

10. When you are done modifying the storage size, click **Next: Add Tags**.
11. Set a name for tagging your instances. This tag allows you to identify your instance more easily.

## Configure security groups

1. Click **Next: Configure Security Group**.
2. 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 \[See page 0\]](#) to determine the minimum required ports you must open for your ThoughtSpot appliance.

3. Click **Review and Launch**.
4. After you have reviewed your instance launch details, click **Launch**.
5. 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. Without a key pair, you cannot SSH into the AWS instance later on.

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

## Prepare the VMs

Before installing a ThoughtSpot cluster, an administrator must [prepare the VMs](#). [See page 180]

# Prepare AWS VMs for ThoughtSpot

**Summary:** Prepare the VMs before installing your ThoughtSpot cluster(s).

1. SSH into a VM.

```
$ ssh -i key.pem admin@<VM-IP>
```

2. Run `sudo /usr/local/scaligent/bin/prepare_disks.sh` to configure the VMs.

**Warning:** Make sure you migrate any data off the disks. This command formats the disks, wiping them clean.

```
$ sudo /usr/local/scaligent/bin/prepare_disks.sh
```

3. Configure the VM based on your specific network information. Refer to your site-survey or ask your network administrator for that information.

4. Run `df -h` to confirm configuration.

```
$ df -h
```

5. Repeat these steps for each of your VMs.

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

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

## Open the required network ports

If you have not already opened the required network ports, see [Network policies \[See page 44\]](#) to determine which ports to open.

## Install Cluster

Next, you must configure your nodes and install your cluster. Follow the steps in [Installing AWS \[See page 182\]](#).

## Related information

[EC2 Best Practices \[See page 0\]](#)

[Loading data from an AWS S3 bucket \[See page 0\]](#)

# Configure ThoughtSpot nodes in AWS

**Summary:** Prepare to install your ThoughtSpot cluster by configuring nodes.

Before you can install a ThoughtSpot cluster in AWS, you must configure your nodes.

## Installation Prerequisites

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

1. **Review configuration options** Refer to [AWS configuration options \[See page 169\]](#) for detailed instance specs.
2. **Create the instance** Refer to [Set up AWS for ThoughtSpot \[See page 172\]](#) to create and launch your instance.
3. **Review required ports** Refer to [Network Policies \[See page 44\]](#) to view the required ports for successful operation of ThoughtSpot.

## Configure Nodes

After creating the instance, you must configure the nodes. Follow the steps in this checklist.

- Step 1: Log in to your cluster [\[See page 0\]](#)
- Step 2: Get a template for network configuration [\[See page 0\]](#)
- Step 3: Prepare node configuration [\[See page 0\]](#)
- Step 4: Configure the nodes [\[See page 0\]](#)
- Step 5: Confirm node configuration [\[See page 0\]](#)

### Step 1: Log in to your cluster

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

Ask your network administrator if you do not know the admin credentials.

1. Run `ssh admin@<nodeIP>`.

Replace `nodeIP` with your specific network information.

```
$ ssh admin@<nodeIP>
```

2. Enter your admin password at the prompt.

Ask your network administrator if you don't know the password.

**Note:** The password does not appear on the screen as you type it.

## Step 2: Get a template for network configuration

Run the `tscli cluster get-config` command to get a template for network configuration for the new cluster. Redirect it to the file `nodes.config`.

You can find more information on this process in the `nodes.config` [file reference \[See page 27\]](#).

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

## Step 3: Prepare node configuration

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the [autodiscovery of one node example \[See page 29\]](#). Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information.

If you have additional nodes, complete each node within the `nodes.config` file in the same

way.

Do not edit any part of the `nodes.config` file except the sections described in [Parameters of the nodes.config file \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, it may cause setup to fail.

## Step 4: Configure the nodes

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

1. Disable the `firewalld` service by running `sudo systemctl stop firewalld` in your terminal. The `firewalld` service is a Linux firewall that must be off for ThoughtSpot installation. After the cluster installer reboots the nodes, `firewalld` automatically turns back on.

```
$ sudo systemctl stop firewalld
```

2. To make sure you temporarily disabled `firewalld`, run `sudo systemctl status firewalld`. Your output should specify that `firewalld` is inactive. It may look something like the following:

```
$ sudo systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
    Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
    Active: inactive (dead)
```

3. Run the configuration command: `$ cat nodes.config | tscli cluster set-config`. If the command returns an error, refer to [set-config error recovery \[See page 186\]](#).

After you run the node configuration command, your output appears similar to the following:

```
$ 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 NTP Servers
Setting up Timezone
Done setting up ThoughtSpot
```

### Step 5: Confirm node configuration

Use the `get-config` command to confirm node configuration.

Your output may look similar to the following:

```
$ 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 ThoughtSpot software

Next, [install your ThoughtSpot clusters \[See page 189\]](#).

## Error recovery

### Set-config error recovery

If you get a warning about node detection when you run the `set-config` command, restart the node-scout service.

Your error may look something like the following:

```
Connecting to local node-scout
WARNING: Detected 0 nodes, but found 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 Configuration failed.
Please retry or contact support.
```

Restart the node-scout service with the following set of commands.

```
$ sudo vim /etc/systemd/system/node-scout.service
$ sudo systemctl daemon-reload
$ sudo systemctl restart node-scout
```

Ensure that you restarted the node-scout by running `sudo systemctl status node-scout`. Your output should specify that the node-scout service is active. It may look something like the following:

```
$ sudo systemctl status node-scout
● node-scout.service - Setup Node Scout service
  Loaded: loaded (/etc/systemd/system/node-scout.service; enabled; vendor preset: disabled)
  Active: active (running) since Fri 2019-12-06 13:56:29 PS
T; 4s ago
```

Next, retry the `set-config` command.

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

The command output should no longer have a warning.

## Related information

Use these references for successful installation and administration of ThoughtSpot:

- [The nodes.config file \[See page 0\]](#)

- Parameters of the `nodes.config` file [See page 30]
- Using the `tscli cluster create` command [See page 33]
- Parameters of the `cluster create` command [See page 41]
- Deployment Overview [See page 116]
- Contact Support [See page 290]

# Install ThoughtSpot clusters in AWS

**Summary:** Learn how to install ThoughtSpot clusters in AWS.

## Prerequisites

Before you can install your ThoughtSpot clusters in AWS, complete these prerequisites.

1. **Review configuration options** Refer to [AWS configuration options \[See page 169\]](#) for detailed instance specs.
2. **Create the instance** Refer to [Set up AWS for ThoughtSpot \[See page 172\]](#) to create and launch your instance.
3. **Review required ports** Refer to [Network Policies \[See page 44\]](#) to view the required ports for successful operation of ThoughtSpot.
4. **Configure nodes** Refer to [Configure ThoughtSpot nodes in AWS \[See page 182\]](#) to configure your nodes.

## Install ThoughtSpot Software

Install the cluster using the ThoughtSpot software release bundle. The estimated installation time is one hour. Follow the steps in this checklist.

- Step 1: Run the installer [\[See page 0\]](#)
- Step 2: Check cluster health [\[See page 0\]](#)
- Step 3: Finalize installation [\[See page 0\]](#)

Refer to your welcome letter from ThoughtSpot to find the link to download the release bundle. If you do not have a link, open a support ticket at [ThoughtSpot Support \[See page 0\]](#) to request access to the release bundle.

### Step 1: Run the installer

1. Copy the downloaded release bundle to `/export/xvdb1/TS_TASKS/install` using the following command:

```
$ scp -i key.pem <release-number>.tar.gz admin@<hostname>:/export/xvdb1/TS_TASKS/install/<file-name>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 6.0, and so on.
- `hostname` is your specific hostname.
- `file-name` is the name of the tarball file on your local computer.

**Note:** You can use another secure copy method, if you prefer a method other than the `scp` command.

2. Alternatively, use `tscli fileserver download-release` to download the release bundle.

You must [configure the fileserver \[See page 0\]](#) by running `tscli fileserver configure` before you can download the release.

```
$ tscli fileserver download-release <release-number> --user <username> --out <release-location>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 5.3.1, 6.0, and so on.
- `username` is the username for the fileserver that you set up earlier, when configuring the fileserver.
- `release-location` is the location path of the release bundle on your local machine. For example, `/export/xvdb1/TS_TASKS/install/6.0.tar.gz`.

3. Verify the checksum to ensure you have the correct release.

Run `md5sum -c <release-number>.tar.gz.MD5checksum`.

```
$ md5sum -c <release-number>.tar.gz.MD5checksum
```

Your output says `ok` if you have the correct release.

4. Launch a [screen \[See page 0\]](#) session. Use screen to ensure that your installation does not stop if you lose network connectivity.

```
$ screen -S DEPLOYMENT
```

5. Create the cluster.

Run `tscli cluster create` to create the cluster.

If you are using an s3 bucket for object storage, include the flag `--enable_cloud_storage=s3a`.

```
$ tscli cluster create <release-number>.tar.gz --enable_cloud_storage=s3a
```

6. Edit the output with your specific cluster information.

For more information on this process, refer to [Using the tscli cluster create command \[See page 33\]](#) and [Parameters of the cluster create command \[See page 41\]](#).

The cluster installer automatically reboots all the nodes after a successful install. The `firewalld` service automatically turns on. At this time, the system is rebooting, which may take approximately 15 minutes.

Log in to any node to check the current cluster status:

```
$ tscli cluster status
```

## Step 2: Check cluster health

After the cluster installs, check its status using the `tscli cluster status` command.

Your output may look similar to the following:

```
$ 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
```

Ensure that the cluster is `RUNNING` and that the Database and Search Engine are `READY`.

```
$ tscli cluster check
Connecting to hosts...
[Wed Jan  8 23:15:47 2020] START Diagnosing ssh
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing connection
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing zookeeper
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing sage
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing timezone
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing disk
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing cassandra
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing hdfs
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing orion-oreo
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing memcheck
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing ntp
[Wed Jan  8 23:16:08 2020] SUCCESS
#####
```

```
#####
[Wed Jan  8 23:16:08 2020] START Diagnosing trace_vault
[Wed Jan  8 23:16:09 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:09 2020] START Diagnosing postgres
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing disk-health
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing falcon
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing orion-cgroups
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing callosum
/usr/lib/python2.7/site-packages/urllib3/connectionpool.py:85
2: InsecureRequestWarning: Unverified HTTPS request is being ma
de. Adding certificate verification is strongly advised. See: h
ttps://urllib3.readthedocs.io/en/latest/advanced-usage.html#ss
l-warnings
    InsecureRequestWarning)
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
```

Your output may look something like the above. Ensure that all diagnostics show `SUCCESS`.

**● Warning:** If `tscli cluster check` returns an error, it may suggest you run `tscli storage gc` to resolve the issue. If you run `tscli storage gc`, note that it restarts your cluster.

### Step 3: Finalize installation

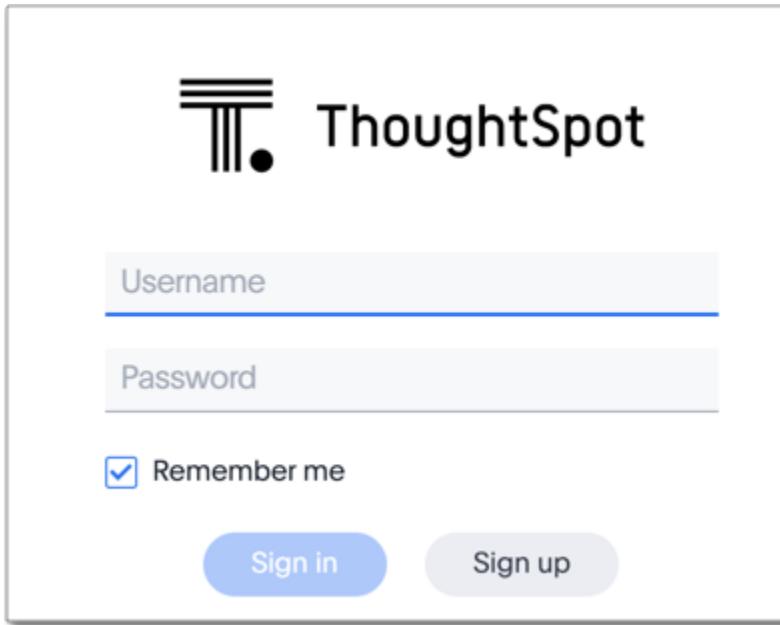
After the cluster status changes to `READY`, sign in to ThoughtSpot on your browser. Follow these steps:

1. Start a browser from your computer.

2. Enter your secure IP information on the address line.

`https://<VM-IP>`

3. If you don't have a security certificate for ThoughtSpot, you must bypass the security warning:
  - Click **Advanced**
  - Click **Proceed**
4. The ThoughtSpot sign-in page appears.
5. In the [ThoughtSpot sign-in window \[See page 195\]](#), enter admin credentials, and click **Sign in**.  
ThoughtSpot recommends changing the default admin password.



## Lean configuration

**(For use with thin provisioning only)** If you have a [small or medium instance type \[See page 167\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 290\]](#) for assistance with this configuration.

## Related information

Use these references for successful installation and administration of ThoughtSpot:

- [the nodes.config file \[See page 0\]](#)
- [Parameters of the nodes.config file \[See page 30\]](#)
- [Using the tscli cluster create command \[See page 33\]](#)
- [Parameters of the `cluster create` command \[See page 41\]](#)
- [Deployment Overview \[See page 116\]](#)
- [Contact Support \[See page 290\]](#)

# 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 availability

Follow these steps to set up High Availability (HA) for your ThoughtSpot cluster using the AWS Elastic File System (EFS).

1. Create an EFS File System that spans across different availability zones, and across different subnets. Refer to AWS documentation on [creating an EFS File System](#) [See page 0].
2. Create two ThoughtSpot clusters in each availability zone and in the subnets, where the new file system is.
3. Change the IP addresses of the cluster, if necessary.
4. Create an EFS directory in the `/home/admin` path.
5. Issue the following command to mount the new file system.

Modify the fields as necessary for your installation.

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

6. Modify permissions to ensure that all clusters with EFS mount points have read and write permissions.

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

7. On the first cluster, create a snapshot on to the EFS mount point, and back it up.

```
tscli snapshot create EfsTest HA 2
```

8. Create a backup on the cluster.

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

9. Ensure that the backup is successful, and that it can be accessed from all clusters where EFS is mounted, by listing all backups and looking for the new one.

```
tscli backup ls
```

10. Take down the first cluster instances.

11. On the second cluster, delete the existing cluster.

12. Create a new cluster 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 is now 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 \[See page 0\]](#).

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

# Azure configuration options

**Summary:** ThoughtSpot offers several Microsoft Azure instance types.

ThoughtSpot can be deployed in your Azure environment by deploying compute (VM) instances in your VNET as well as an underlying persistent storage infrastructure. Currently we support Premium SSD Managed Disks for persistent storage. For more information, see [Managed Disks pricing \[See page 0\]](#) in Microsoft's Azure documentation.

All Azure VMs (nodes) in a ThoughtSpot cluster must be in the same availability zone (and, therefore, also in the same region). ThoughtSpot does not support deploying VMs( nodes) of the same cluster across availability zones. For more information, see [What are Availability Zones in Azure? \[See page 0\]](#) in Microsoft's Azure documentation.

## ThoughtSpot Azure instance types

Per VM user data capacity	Instance type	CPU/RAM	Recommended per-VM Premium SSD Managed Disk volume
200 GB	E64sv3	64/432	2x1 TB
100 GB	E32sv3	32/256	2X 400 GB
20 GB	E16sv3	16/128	2X 400 GB
120 GB	D64v3	64/256	2X 1 TB

# Set up ThoughtSpot in Azure

**Summary:** After you determine your configuration options, you must set up your virtual machines using a ThoughtSpot image for Azure.

## About the ThoughtSpot image

To provision ThoughtSpot in the Azure portal, access the ThoughtSpot Virtual Machine in the Azure Marketplace.

The ThoughtSpot Virtual Machine comes provisioned with the custom ThoughtSpot image to make hosting simple. A virtual machine is a preconfigured template that provides the information required to launch an instance of ThoughtSpot. It includes a root disk for the instance, which contains an operating system, application server, and other necessary software.

The ThoughtSpot Virtual Machine has the ThoughtSpot software installed and configured, on a base image. Check with your ThoughtSpot contact to learn about the latest version of the ThoughtSpot Virtual Machine.

Due to security restrictions, the ThoughtSpot Virtual Machine does not have default passwords for the administrator users. When you are ready to obtain the password, contact [ThoughtSpot Support \[See page 290\]](#).

## Set up ThoughtSpot in Azure

Follow these steps to provision and set up the VMs and launch ThoughtSpot.

### Prerequisites

Complete these steps before launching your ThoughtSpot Virtual Machine:

1. Obtain an Azure login account.
2. Set up usage payment details with Microsoft Azure.
3. Find your company's [Resource Group \[See page 0\]](#). (optional—you can also create one while creating your virtual machines.)
4. [Download \[See page 0\]](#) and fill out the ThoughtSpot site survey to have a quick reference for

any networking information you may need to fill out. Ask your network administrator if you need help filling out the site survey.

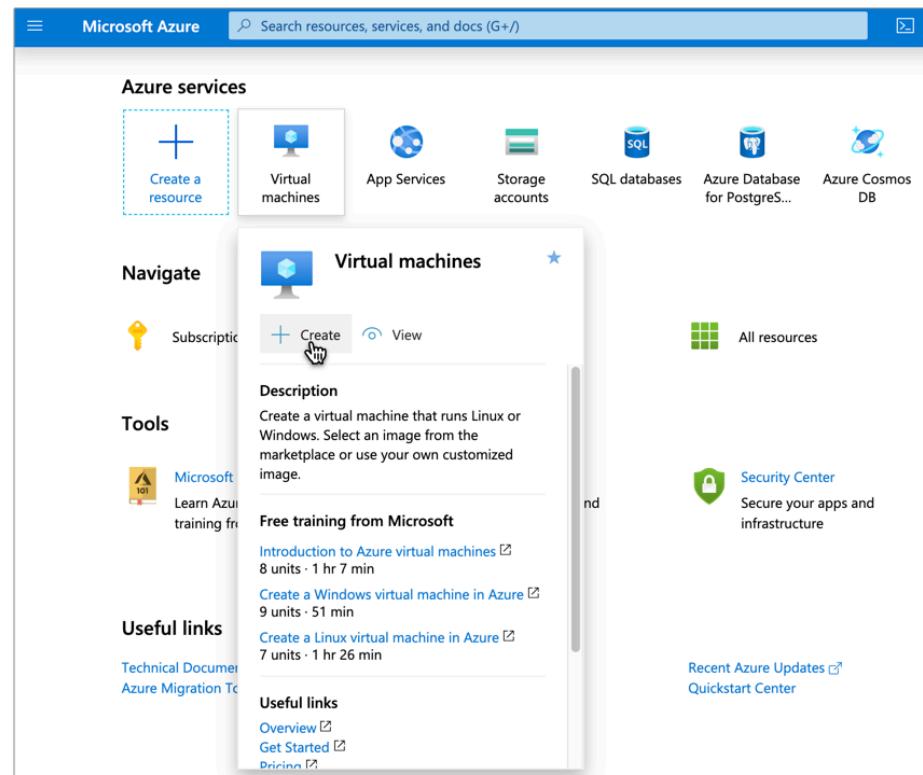
## Create an instance

Create your virtual machines based on the [ThoughtSpot Virtual Machine \[See page 0\]](#).

1. Log in to the Azure portal.

In a browser, go to [https://portal.azure.com/#home \[See page 0\]](https://portal.azure.com/#home), and log in to your Azure account.

2. On the Azure portal homepage, hover over **Virtual machines**, and click **Create**.



3. Specify information under **Basics**.

**Microsoft Azure** Search resources, services, and docs (G+)

Home > Create a virtual machine

## Create a virtual machine

**Basics** **Disks** **Networking** **Management** **Advanced** **Tags** **Review + create**

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image.

Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization.

Looking for classic VMs? [Create VM from Azure Marketplace](#)

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

**Subscription \*** (1)

**Resource group \*** (2)  [Select existing...](#) [Create new](#)

**Instance details**

**Virtual machine name \*** (3)

**Region \*** (4)  (US) East US

**Availability options** (5)  No infrastructure redundancy required

**Image \*** (6)  [Browse all public and private images](#)  
Standard D2s v3  
2 vcpus, 8 GiB memory (Price unavailable)  
[Change size](#)

**Azure Spot instance** (7)  Yes  No

**Size \*** (8)  Standard D2s v3  
2 vcpus, 8 GiB memory (Price unavailable)  
[Change size](#)

**Administrator account**

**Authentication type** (9)  Password  SSH public key

**Username \*** (10)

**SSH public key \*** (11)   
[Learn more about creating and using SSH keys in Azure](#)

**Inbound port rules**

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

**Public inbound ports \*** (12)  None  Allow selected ports

**Select inbound ports \*** (13)  SSH (22)

**Warning:** This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

- 1 Choose a subscription type from the dropdown menu.
- 2 If your company already has a resource group, select *existing*. If not, *create new*.
- 3 Specify a name for your virtual machine.

- 4 Specify the region in which you are creating the VM.
- 5 Click **Browse all public and private images**, and search for the ThoughtSpot image. Click on it.
- 6 Refer to [Azure configuration options \[See page 199\]](#) to choose a size for your VM that works for your cluster needs.
- 7 Select **SSH public key** and specify a username.
- 8 Enter your SSH public key. [Contact ThoughtSpot support \[See page 290\]](#) to obtain a public key, if you do not have one. Note that this SSH public key is different from the SSH private key you use later, to ssh into your VM from the command line.
- 9 Choose **allow selected ports**.
- 10 Open the necessary Inbound and Outbound ports to ensure that the ThoughtSpot processes do not get blocked. See the [minimum port requirements. \[See page 205\]](#)

4. Specify information under **Disks**.

The screenshot shows the 'Create a virtual machine' wizard in Microsoft Azure. The 'Disks' tab is selected. A dropdown menu for 'OS disk type' is open, showing options: Premium SSD (highlighted with a green circle), Standard HDD, Standard SSD, and Premium SSD. Below the dropdown, there's a section for 'Data disks' with a table header: LUN, Name, Size (GiB), Disk type, Host caching. At the bottom of the 'Disks' section, there are two buttons: 'Create and attach a new disk' (highlighted with a green circle) and 'Attach an existing disk'. Under the 'Advanced' section, there are two checkboxes: 'Use managed disks' (highlighted with a green circle) and 'Use ephemeral OS disk'. The 'Use managed disks' checkbox has 'Yes' selected.

- 1 Choose a disk type from the dropdown menu. ThoughtSpot recommends the Premium SSD disks.

- 2 Click **Create and attach a new disk**. Add two data disks. Refer to [Azure configuration options \[See page 0\]](#) to see what size they should be.

- 3 Under **Advanced**, click **yes to use managed disks**.

**Tip:** The new Standard SSD disk types are only available for particular regions. Make sure this disk type is supported in the region you chose for your VM before selecting it.

See [Standard SSD Disks for Virtual Machine workloads \[See page 0\]](#) for more on SSD disks.

ThoughtSpot recommends the Premium SSD disks.

5. Specify information under **Networking**.

The screenshot shows the 'Networking' tab of the Azure VM creation interface. It includes fields for 'Virtual network', 'Public IP', 'NIC network security group', 'Configure network security group', and 'Accelerated networking'. A note at the bottom states: 'The selected VM size does not support accelerated networking.'

**1** Filter virtual networks  
Create new  
The value must not be empty.

**2** None  
Create new

**3** None Basic Advanced

**4** Create new

- 1 Find your company's virtual network and select it, or **create new**.

- 2 Find your company's public IP, or **create new**.

- 3 Select **Advanced** for NIC network security group.
- 4 After you select **Advanced**, the **Configure network security group** option appears. Find your company's security group, or **create new**. When creating your security group, ensure that the required ports are open. Refer to the [minimum port requirements](#). [See page 205]
6. Under **Management**, configure your monitoring and management preferences. If you have no preferences, you can leave them at their default settings.
7. Under **Advanced**, configure your advanced settings preferences. If you have no preferences, you can leave them at their default settings.
8. Under **Tags**, tag your virtual machine with a human-readable string to help you identify it.
9. Click **Review + create** in the bottom left corner of your screen.
10. Review your changes, and click **create**. Azure does the final validation check.

### Minimum required ports

Open the following ports between the User/ETL server and ThoughtSpot nodes. This ensures that the ThoughtSpot processes do not get blocked.

The minimum ports needed are:

Port	Protocol	Service
22	SSH	Secure Shell access
443	HTTPS	Secure Web access
12345	TCP	ODBC and JDBC drivers access

**Note:** Nodes purchased from Azure must be reachable to each other so that they can communicate and form a distributed environment. ThoughtSpot requires that these ports be accessible between nodes within a cluster. Use your discretion about whether to restrict public access or not for all nodes and all ports.

Refer to [network policies \[See page 44\]](#) for more information.

## Prepare for starting up ThoughtSpot

*Prerequisite:* To log in to the VM, you need the private key that is available in the image. You can obtain this from your ThoughtSpot contact.

1. Obtain the VM's public and private IP addresses.
  - To see the public IP, click the VM name link. This will show the public IP of the VM.
  - To see the private IP, select **more services** from the Microsoft Azure homepage. Select **Networking** from the list on the left side of the screen.
2. In a terminal application, connect to the VM through SSH. When prompted, enter the private key provided for the admin user.
  - You must file a support ticket to obtain this private key; it is necessary for the first login.
  - This key is different from the credentials, or the public keys supplied in earlier steps, which do not work in this context.

```
$ ssh admin@<VM-IP>
```

3. Update the password for both the `admin` and the `thoughtspot` users.

The command prompts you to type in a new password, and then to confirm the password.

```
$ sudo passwd admin  
Changing password for user admin  
$ sudo passwd thoughtspot  
Changing password for user thoughtspot
```

**⚠ Warning:** If you do not change the password, you cannot log back into your Azure VMs. Your private key does not work after initial installation.

4. Update the file `/etc/hosts` with all the node IP addresses for the other VMs that will be part of the ThoughtSpot cluster.

## Verify storage disks

Verify the existence of your data disks, created in Step 4 of [create an instance \[See page 201\]](#), by issuing `lsblk` in your terminal application:

```
$ lsblk
```

Your result may look something like the following:

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
sda	8:0	0	200G	0	disk	
└─sda1	8:1	0	1G	0	part	/mntboot
└─sda2	8:2	0	20G	0	part	/
└─sda3	8:3	0	20G	0	part	/update
└─sda4	8:4	0	159G	0	part	/export
sdb	8:16	0	1T	0	disk	
└─sb1	8:17	0	1T	0	part	/mnt/resource
sdc	8:32	0	1T	0	disk	
sdd	8:48	0	1T	0	disk	
sr0	11:0	1	628K	0	rom	

1. Unmount the temporary disk by issuing the following command:

```
$ sudo umount /mnt/resource
```

**⚠ Warning:** The `/mnt/resource` disk, which is mounted on the `/dev/sdb` disk, is temporary. Any data on it will be wiped if the VM is shut down. You must unmount the `/mnt/resource` disk.

2. Prepare the disks `/dev/sdc` and `/dev/sdd` for ThoughtSpot by issuing the following command:

**⚠ Warning:** Do not use the disk /dev/sdb. Any data on it will be wiped if the VM is shut down.

```
$ sudo /usr/local/scaligent/bin/prepare_disks.sh /dev/sdc /dev/sdd
```

3. Check the disks' status by issuing the following command:

```
$ df -h
```

4. Repeat the steps in this section for each node in your cluster.

#### Create network support settings

**☒ Tip:** All changes in this section must be re-applied each time after a cluster is created or updated. If these changes are not present, a reboot of the VMs will not have network access. So when updating these files, keep a backup to copy after any subsequent cluster creation or update.

1. SSH into one of your VMs.

```
ssh admin@<VM-IP>
```

2. Update the VM's hostname:

```
$ sudo hostnamectl set-hostname <HOSTNAME>
```

If you are using a static name, you can issue:

```
sudo hostnamectl set-hostname <HOSTNAME> --static
```

3. Update `/etc/sysconfig/network-scripts/ifcfg-eth0` with the IP and hostname:

```
$ sudo vi /etc/sysconfig/network-scripts/ifcfg-eth0  
  
DEVICE=eth0 ONBOOT=yes BOOTPROTO=dhcp HWADDR=<Add eth0 MAC> TYPE=Ethernet USERCTL=no PEERDNS=yes IPV6INIT=no
```

4. Modify permissions for `/etc/sysconfig/network-scripts/ifcfg-eth0`. This command allows the root user to retain read/write permissions, and grants read-only permissions to other users.

```
$ sudo chmod 644 /etc/sysconfig/network-scripts/ifcfg-eth0
```

5. Repeat this process (steps 1 through 4) for each node.

# Configure ThoughtSpot nodes in Azure

**Summary:** Prepare to install your ThoughtSpot cluster by configuring nodes.

Before you can install a ThoughtSpot cluster in Azure, you must configure your nodes.

## Installation Prerequisites

Ensure the successful creation of the virtual machines (VMs) before you install the ThoughtSpot cluster in Azure.

1. **Review configuration options** Refer to [Azure configuration options \[See page 199\]](#) for detailed instance specs.
2. **Create the instance** Refer to [Set up Azure for ThoughtSpot \[See page 200\]](#) to create and launch your instance.
3. **Review required ports** Refer to [Network Policies \[See page 44\]](#) to view the required ports for successful operation of ThoughtSpot.

## Configure Nodes

After creating the instance, you must configure the nodes. Follow the steps in this checklist.

- Step 1: Log in to your cluster [\[See page 0\]](#)
- Step 2: Get a template for network configuration [\[See page 0\]](#)
- Step 3: Prepare node configuration [\[See page 0\]](#)
- Step 4: Configure the nodes [\[See page 0\]](#)
- Step 5: Confirm node configuration [\[See page 0\]](#)

## Step 1: Log in to your cluster

Use Terminal on a Mac or a terminal emulator on Windows to log in to your cluster. Log in using the ssh private key provided by ThoughtSpot.

If you do not have a private key, contact [ThoughtSpot Support \[See page 290\]](#) by email or through the support portal.

To log in to your cluster, run `ssh -i <private-key> admin@<public-vm-ip>`.

```
$ ssh -i <private_key> admin@<public-vm-ip>
```

## Step 2: Get a template for network configuration

Run the `tscli cluster get-config` command to get a template for network configuration for the new cluster. Redirect it to the file `nodes.config`.

You can find more information on this process in the [nodes.config file reference \[See page 27\]](#).

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

## Step 3: Prepare node configuration

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the [autodiscovery of one node example \[See page 29\]](#). Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information.

If you have additional nodes, complete each node within the nodes.config file in the same way.

Do not edit any part of the `nodes.config` file except the sections described in [Parameters of the nodes.config file \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, it may cause setup to fail.

#### Step 4: Configure the nodes

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

1. Disable the `firewalld` service by running `sudo systemctl stop firewalld` in your terminal. The `firewalld` service is a Linux firewall that must be off for ThoughtSpot installation. After the cluster installer reboots the nodes, `firewalld` automatically turns back on.

```
$ sudo systemctl stop firewalld
```

2. To make sure you temporarily disabled `firewalld`, run `sudo systemctl status firewalld`. Your output should specify that `firewalld` is inactive. It may look something like the following:

```
$ sudo systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
  Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
  Active: inactive (dead)
```

3. Run the configuration command: `$ cat nodes.config | tscli cluster set-config`.

If the command returns an error, refer to [set-config error recovery \[See page 214\]](#).

After you run the node configuration command, your output appears similar to the following:

```
$ 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 NTP Servers
Setting up Timezone
Done setting up ThoughtSpot
```

### Step 5: Confirm node configuration

Use the `get-config` command to confirm node configuration.

Your output may look similar to the following:

```
$ 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 ThoughtSpot software

Next, [install your ThoughtSpot clusters \[See page 217\]](#).

## Error recovery

### Set-config error recovery

If you get a warning about node detection when you run the `set-config` command, restart the node-scout service.

Your error may look something like the following:

```
Connecting to local node-scout
WARNING: Detected 0 nodes, but found 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 Configuration failed.
Please retry or contact support.
```

Restart the node-scout service with the following set of commands.

```
$ sudo vim /etc/systemd/system/node-scout.service
$ sudo systemctl daemon-reload
$ sudo systemctl restart node-scout
```

Ensure that you restarted the node-scout by running `sudo systemctl status node-scout`. Your output should specify that the node-scout service is active. It may look something like the following:

```
$ sudo systemctl status node-scout
● node-scout.service - Setup Node Scout service
  Loaded: loaded (/etc/systemd/system/node-scout.service; enabled; vendor preset: disabled)
  Active: active (running) since Fri 2019-12-06 13:56:29 PS
T; 4s ago
```

Next, retry the `set-config` command.

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

The command output should no longer have a warning.

## Related information

Use these references for successful installation and administration of ThoughtSpot.

- [the nodes.config file \[See page 0\]](#)

- Parameters of the `nodes.config` file [See page 30]
- Using the `tscli cluster create` command [See page 33]
- Parameters of the `cluster create` command [See page 41]
- ThoughtSpot Documentation [See page 0]
- Contact Support [See page 290]

# Install ThoughtSpot clusters in Azure

**Summary:** Learn how to install ThoughtSpot clusters in Azure.

## Prerequisites

Before you can install your ThoughtSpot clusters in Azure, complete these prerequisites.

1. **Review configuration options** Refer to [Azure configuration options \[See page 199\]](#) for detailed instance specs.
2. **Create the instance** Refer to [Set up Azure for ThoughtSpot \[See page 200\]](#) to create and launch your instance.
3. **Review required ports** Refer to [Network Policies \[See page 44\]](#) to view the required ports for successful operation of ThoughtSpot.
4. **Configure nodes** Refer to [Configure ThoughtSpot nodes in Azure \[See page 210\]](#) to configure your nodes.

## Install ThoughtSpot Software

Install the cluster using the ThoughtSpot software release bundle. The estimated installation time is one hour. Follow the steps in this checklist.

- Step 1: Run the installer [\[See page 0\]](#)
- Step 2: Check cluster health [\[See page 0\]](#)
- Step 3: Finalize installation [\[See page 0\]](#)

Refer to your welcome letter from ThoughtSpot to find the link to download the release bundle. If you do not have a link, open a support ticket at [ThoughtSpot Support \[See page 0\]](#) to request access to the release bundle.

### Step 1: Run the installer

1. Copy the downloaded release bundle to `/export/sdc1/TS_TASKS/install` using the following command:

```
$ scp <release-number>.tar.gz admin@<hostname>:/export/  
sdc1/TS_TASKS/install/<file-name>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 6.0, and so on.
- `hostname` is your specific hostname.
- `file-name` is the name of the tarball file on your local computer.

**Note:** You can use another secure copy method, if you prefer a method other than the `scp` command.

2. Alternatively, use `tscli fileserver download-release` to download the release bundle.

You must [configure the fileserver \[See page 0\]](#) by running `tscli fileserver configure` before you can download the release.

```
$ tscli fileserver download-release <release-number> --  
user <username> --out <release-location>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 5.3.1, 6.0, and so on.
- `username` is the username for the fileserver that you set up earlier, when configuring the fileserver.
- `release-location` is the location path of the release bundle on your local machine. For example, `/export/sdc1/TS_TASKS/install/6.0.tar.gz`.

3. Verify the checksum to ensure you have the correct release.

Run `md5sum -c <release-number>.tar.gz.MD5checksum`.

```
$ md5sum -c <release-number>.tar.gz.MD5checksum
```

Your output says `ok` if you have the correct release.

4. Launch a [screen \[See page 0\]](#) session. Use screen to ensure that your installation does not stop if you lose network connectivity.

```
$ screen -S DEPLOYMENT
```

5. Create the cluster.

Run `tscli cluster create` to create the cluster.

```
$ tscli cluster create <release-number>.tar.gz
```

6. Edit the output with your specific cluster information.

For more information on this process, refer to [Using the tscli cluster create command \[See page 33\]](#) and [Parameters of the cluster create command \[See page 41\]](#).

The cluster installer automatically reboots all the nodes after a successful install. The `firewalld` service automatically turns on. At this time, the system is rebooting, which may take approximately 15 minutes.

7. Make this change to each node to ensure that you can log back in:

- a. Open the grub file `/update/etc/default/grub`:

```
$ sudo vi /update/etc/default/grub
```

- b. Change the line:

```
GRUB_CMDLINE_LINUX="console=tty0 console=ttyS1,115200n8"
```

to:

```
GRUB_CMDLINE_LINUX="console=tty0 console=ttyS1,115200n  
8 net.ifnames=0"
```

c. Save your changes.

8. Issue these commands:

```
$ sudo cp /update/etc/default/grub /etc/default/  
$ rm /usr/local/scaligent/bin/setup-net-devices.sh
```

Log in to any node to check the current cluster status:

```
$ tscli cluster status
```

## Step 2: Check cluster health

After the cluster installs, check its status using the `tscli cluster status` command.

Your output may look similar to the following:

```
$ 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
```

Ensure that the cluster is `RUNNING` and that the Database and Search Engine are `READY`.

```
$ tscli cluster check
Connecting to hosts...
[Wed Jan  8 23:15:47 2020] START Diagnosing ssh
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing connection
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing zookeeper
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing sage
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing timezone
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing disk
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing cassandra
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing hdfs
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing orion-oreo
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing memcheck
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing ntp
[Wed Jan  8 23:16:08 2020] SUCCESS
#####
```

```
#####
[Wed Jan  8 23:16:08 2020] START Diagnosing trace_vault
[Wed Jan  8 23:16:09 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:09 2020] START Diagnosing postgres
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing disk-health
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing falcon
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing orion-cgroups
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing callosum
/usr/lib/python2.7/site-packages/urllib3/connectionpool.py:85
2: InsecureRequestWarning: Unverified HTTPS request is being ma
de. Adding certificate verification is strongly advised. See: h
ttps://urllib3.readthedocs.io/en/latest/advanced-usage.html#ss
l-warnings
    InsecureRequestWarning)
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
```

Your output may look something like the above. Ensure that all diagnostics show `SUCCESS`.

**● Warning:** If `tscli cluster check` returns an error, it may suggest you run `tscli storage gc` to resolve the issue. If you run `tscli storage gc`, note that it restarts your cluster.

### Step 3: Finalize installation

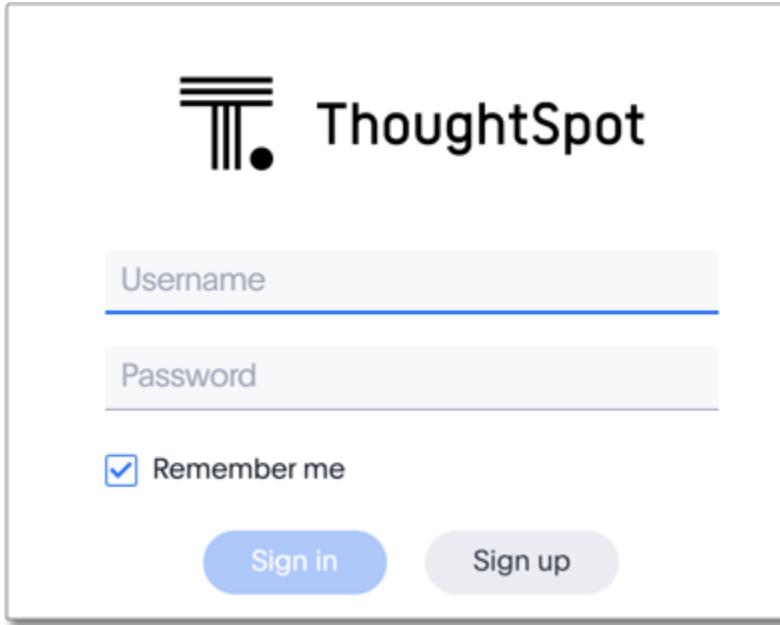
After the cluster status changes to `READY`, sign in to ThoughtSpot on your browser. Follow these steps:

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:
  - Click **Advanced**
  - Click **Proceed**
4. The ThoughtSpot sign-in page appears.
5. In the [ThoughtSpot sign-in window \[See page 224\]](#), enter admin credentials, and click **Sign in**.  
ThoughtSpot recommends changing the default admin password.



## Lean configuration

**(For use with thin provisioning only)** If you have a [small or medium instance type \[See page 167\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 290\]](#) for assistance with this configuration.

## Related information

Use these references for successful installation and administration of ThoughtSpot:

- [The nodes.config file \[See page 0\]](#)
- [Parameters of the nodes.config file \[See page 30\]](#)
- [Using the tscli cluster create command \[See page 33\]](#)
- [Parameters of the `cluster create` command \[See page 41\]](#)
- [Deployment Overview \[See page 116\]](#)
- [Contact Support \[See page 290\]](#)

# GCP configuration options

**Summary:** ThoughtSpot can be deployed using several different GCP instance types.

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

- Mode 1: Compute VMs + SSD Persistent Disk storage-only
- Mode 2: Compute VMs + SSD Persistent Disk and Google Cloud Storage (GCS).

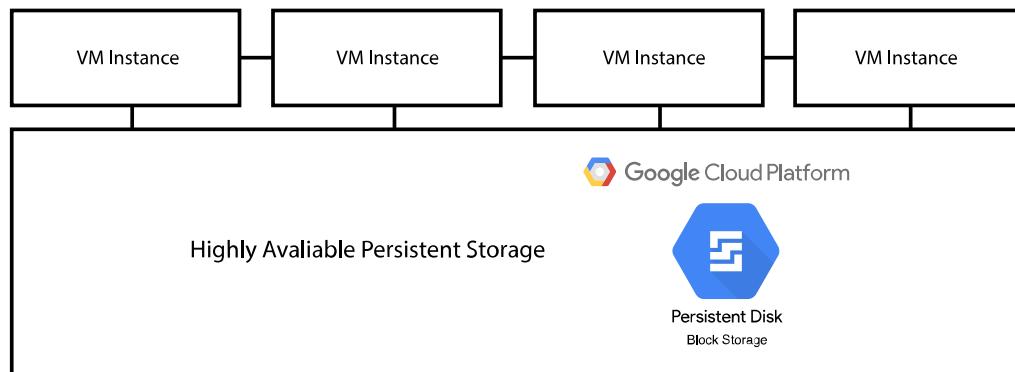
For more information about Persistent Storage, see [Zonal Persistent SSD disks \[See page 0\]](#) in Google's Cloud documentation.

For more information about Google Cloud Storage, see [Cloud Storage Buckets \[See page 0\]](#) in Google's Cloud documentation.

All GCP VMs (nodes) in a ThoughtSpot cluster must be in the same zone (and, therefore, also in the same region). ThoughtSpot does not support deploying VMs (nodes) of the same cluster across different zones. For more information, see [Regions and Zones \[See page 0\]](#) in Google's Cloud documentation.

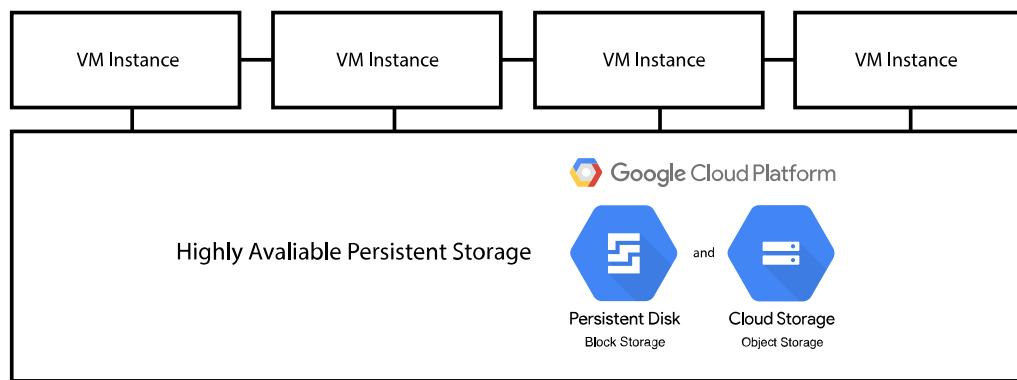
## ThoughtSpot GCP instance types

VMs with Persistent Disk-only storage



Per VM user data capacity	Instance type	CPU/RAM	Recommended per-VM Zonal Persistent SSD Disk volume
208 GB	n1-highmem-64	64/416	2x 1 TB
312 GB	n1-highmem-96	96/624	2x 1.5 TB
100 GB	n1-highmem-32	32/208	2X 400 GB
20 GB	n1-highmem-16	16/122	2X 400 GB
180 GB	n1-standard-96	96/330	2X 1 TB

## VMs with Persistent Disk and Google Cloud storage



Per VM user data capacity	Instance type	CPU/RAM	Recommended per-VM Zonal Persistent SSD Disk volume
208 GB	n1-highmem-64	64/416	1X 500 GB
312 GB	n1-highmem-96	96/624	1X 500 GB
100 GB	n1-highmem-32	32/208	1X 500 GB
20 GB	n1-highmem-16	16/122	1X 500 GB
180 GB	n1-standard-96	96/330	1X 500 GB

# Set up ThoughtSpot in GCP

## Summary: Set up your GCP virtual machines.

After you determine your configuration options, set up your virtual machines (VMs). ThoughtSpot will share the ThoughtSpot base image for booting the VMs and some other aspects of system setup with you on [the GCP platform \[See page 0\]](#).

## About the ThoughtSpot and Google Cloud Platform

ThoughtSpot uses a custom image to populate VMs in GCP. To find the ThoughtSpot custom image, refer to step 13 under [create an instance \[See page 230\]](#).

Ask your ThoughtSpot contact for access to this image. We need the Google account/email ID of the individual who will be signed into your organization's GCP console. We will share ThoughtSpot's GCP project with them so they can use the contained boot disk image to create ThoughtSpot VMs.

## Overview

Before you can create a ThoughtSpot cluster, you must set up your VMs. Use the Google Compute Engine (GCP) platform to create and run VMs.

The following topics walk you through this process.

## Prerequisites

1. Ensure that your **Network Service Tier** on the [Google Cloud Console \[See page 0\]](#) is set to **Premium** for the best performance of all your VMs.
2. A ThoughtSpot cluster requires 10 Gb/s bandwidth (or better) between any two nodes. You must ensure this *before* creating a new cluster.
3. [Download \[See page 0\]](#) and fill out the ThoughtSpot site survey to have a quick reference for your networking information. Ask your network administrator if you need help filling out the site survey.

## Setting up your Google Cloud Storage (GCS) bucket

If you are going to deploy your cluster using the GCS-storage option, you must set up that bucket before you set up your cluster. Contact [ThoughtSpot Support \[See page 0\]](#) to find out if your specific cluster size will benefit from the GCS storage option. If you are not using GCS, skip this step and [create an instance \[See page 230\]](#).

1. Sign in to the [Google Cloud Console \[See page 0\]](#).
2. Go to the **Storage** dashboard from the navigation bar on the side of your screen.
3. Click **CREATE BUCKET** on the top menu bar.
4. Enter a name for your bucket, and click **CONTINUE**.
5. For location type, select **Region**.
6. Use the Location drop-down menu to select the region where you are going to set up your instance.
7. Click **CONTINUE**.
8. For default storage class, select **Standard**.
9. Click **CONTINUE**.
10. Under **Access Control**, select **Uniform** to ensure uniform access to all objects in the storage bucket.
11. Click **CONTINUE**.
12. Do not edit the advanced settings.  
Leave Encryption set to **Google-managed key** and do not set a retention policy.
13. Click **CREATE**.

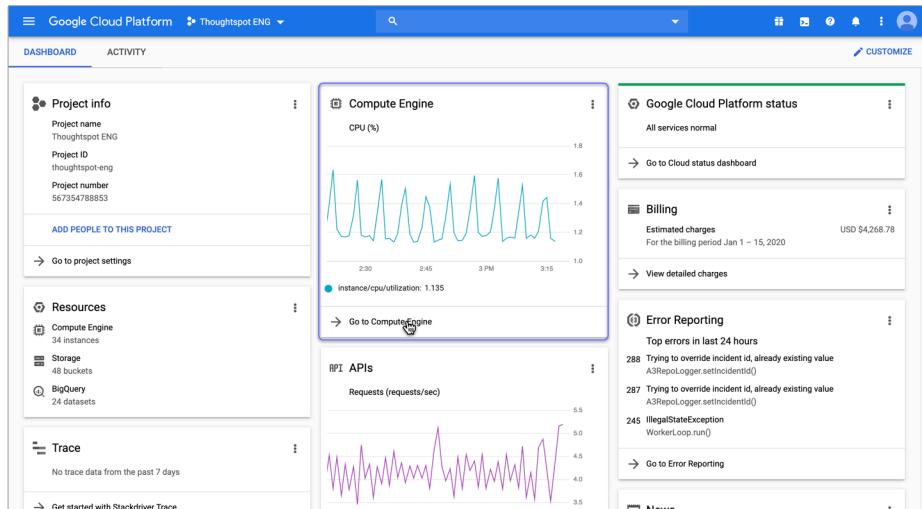
When you create your instance, make sure you set Storage to **Read Write** access.

## Create an instance

1. Sign in to the [Google Cloud Console](#) [See page 0].
2. Click **Select a Project** from the top bar.



3. Under **Select From**, pick your company's project.
4. Go to the Compute Engine dashboard.



5. Select **VM instances** on the left panel.
6. Click **CREATE INSTANCE** from the top menu bar.
7. Provide a name for the instance.
8. Select the region you are creating the instance in.
9. Select the zone you are creating your region in.
10. Under **Machine type**, select **custom**.

11. Select the number of CPUs you need.

Refer to [ThoughtSpot GCP instance types \[See page 226\]](#) to determine the number of CPUs your cluster needs.

12. Specify your memory requirements and CPU platform. Refer to [ThoughtSpot GCP instance types \[See page 226\]](#) to determine the memory your cluster needs.

Your configuration may look something like the following, but with your specific information.

Setting	Value
Cores	64 vCPU
Memory	416 GB
CPU platform	Automatic (or select either one of the preferred CPU platforms, Intel Skylake or Intel Broadwell, if available.)

**Machine family**

[General-purpose](#) [Memory-optimized](#)

Machine types for common workloads, optimized for cost and flexibility

**Series**

N1

Powered by Intel Skylake CPU platform or one of its predecessors

**Machine type**

Custom

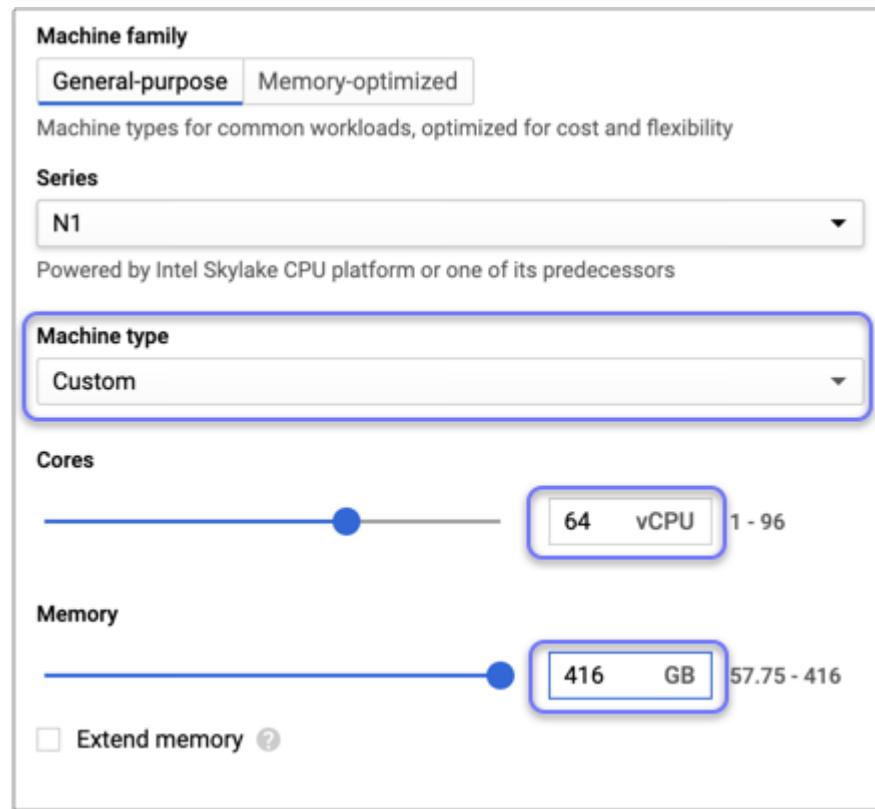
**Cores**

64 vCPU 1 - 96

**Memory**

416 GB 57.75 - 416

Extend memory ?



Automatic

Intel Skylake or later

Intel Broadwell or later

Either of these are preferred platforms

13. Configure the Boot disk.

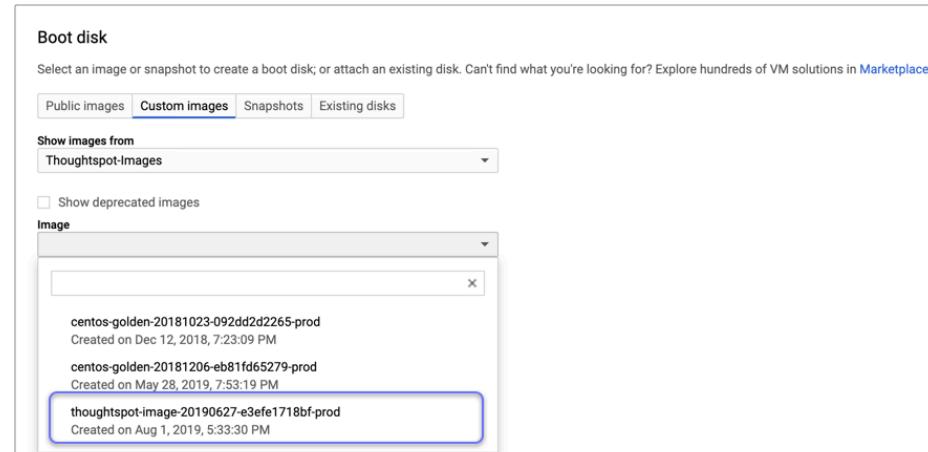
a. Scroll down to the **Boot disk** section and click **Change**.



b. Click **Custom Images** from the options under **Boot disk**.

c. Select **ThoughtSpot-images** under **Show images from**.

d. Select one of the ThoughtSpot base images. Under the name of the image, you can see when it was created. Select the latest image. ThoughtSpot may have directly sent you an image to use through the console. If so, use that image.



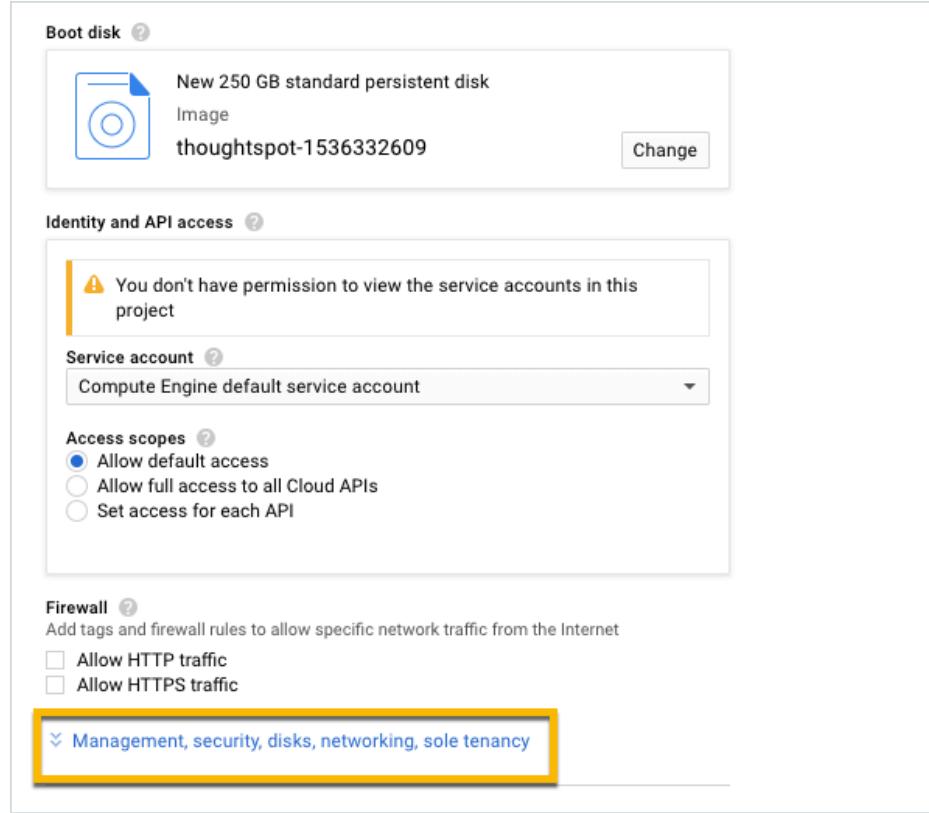
**Note:** ThoughtSpot updates these base images with patches and enhancements. If more than one image is available, select the latest one by looking at the dates of creation. Each image will work, but we recommend using the latest image because it typically contains the latest security and maintenance patches. Contact ThoughtSpot Support if you are unsure which image to use.

e. Configure the boot disk as follows:

Setting	Value
Image	ThoughtSpot
Boot disk type	Standard persistent disk
Size (GB)	250

f. Click **Select** to save the boot disk configuration.

14. Back on the main configuration page, click to expand the advanced configuration options (**Management, security, disks, networking, sole tenancy**).



15. Attach two 1 TB SSD drives for data storage. Refer to [SSD-only persistent storage \[See page 226\]](#). If you are using GCS, attach only 1 SSD drive, with 500 GB instead of 1 TB. Refer to [GCS and SSD persistent storage \[See page 227\]](#).

- a. Click the **Disks** tab, and click **Add new disk**.

Management Security Disks Networking Sole Tenancy

**Boot disk**

**Deletion rule**

Delete boot disk when instance is deleted

**Encryption**

Data is encrypted automatically. Select an encryption key management solution.

- Google-managed key  
No configuration required
- Customer-managed key  
Manage via Google Cloud Key Management Service
- Customer-supplied key  
Manage outside of Google Cloud

**Additional disks** (Optional)

**+ Add new disk** + Attach existing disk

[Less](#)

You can select or unselect the **Deletion rule**, depending on your preferences.

- b. Configure the following settings for each disk. Refer to [ThoughtSpot GCP instance types](#) [See page 227] to determine the size in GB when you have GCS. Ensure the disks have read/write access.

Setting	Value
Type	SSD persistent disk
Source type	Blank disk
Size (GB)	1024

Under **Deletion rule**, select either **keep disk** or **delete disk**, depending on your preference.

vmb-ts-data-disk (Blank, 1024 GB)

Name (Optional)

Type

Source type

Mode  
 Read/write  
 Read only

Deletion rule  
When deleting instance  
 Keep disk  
 Delete disk

Size (GB)

Estimated performance

Operation type	Read	Write
Sustained random IOPS limit	30,720.00	30,000.00
Sustained throughput limit (MB/s)	491.52	400.00

Encryption  
Data is encrypted automatically. Select an encryption key management solution.

Google-managed key  
No configuration required

Customer-managed key  
Manage via Google Cloud Key Management Service

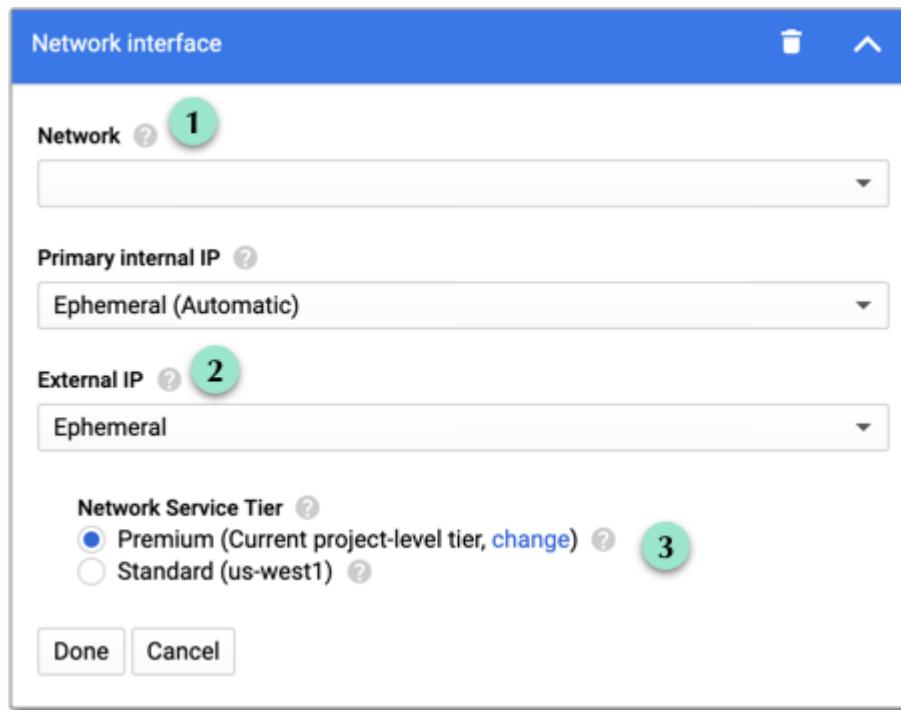
Customer-supplied key  
Manage outside of Google Cloud

This new disk will be added once you create the new instance

16. (For use with GCS only) In the Identity and API access section, make sure Service account is set to **Compute Engine default service account**. Under Access scopes, select **Set access for each API**.

17. (For use with GCS only) After you click **Set access for each API**, scroll down to the **Storage** dropdown menu in the Identity and API access section. Set it to one of the following options:
  - To use Google Cloud Storage (GCS) as persistent storage for your instance, select **Read Write**.
  - To only use GCS to load data into ThoughtSpot, select **Read Only**.
18. Under **Networking**, customize the network settings as needed. Use your default VPC settings, if you know them. Ask your network administrator if you do not know your default VPC settings.

Update the network interface with your specific information or create a new one.



- 1 Add an existing VPC network, or create a new one by clicking **VPC network** from the main menu. Ensure that this network has a **firewall rule** attached, with the minimum ports required for ThoughtSpot operation open. Refer to the [minimum port requirements](#). [See page 238] See Google's [using firewalls](#) [See page 0] and [using VPCs](#) [See page 0] documentation for assistance creating a firewall rule and a VPC network.
- 2 Set the external IP as either ephemeral or static, depending on your preference.
- 3 Ensure that **network service tier** is set to **premium**.

19. Repeat these steps to create the necessary number of VMs for your cluster.

#### Minimum required ports

Open the following ports between the User/ETL server and ThoughtSpot nodes. This ensures that the ThoughtSpot processes do not get blocked. Refer to [network policies \[See page 0\]](#) for more information on what ports to open for intracluster operation, so that your clusters can communicate.

The minimum ports needed are:

Port	Protocol	Service
22	SSH	Secure Shell access
443	HTTPS	Secure Web access
12345	TCP	ODBC and JDBC drivers access

## Prepare the VMs

Before you can install your ThoughtSpot cluster, an administrator must log in to each VM through SSH as user “admin”, and complete the following preparation steps:

1. Open a terminal application on your machine and ssh into one of your VMs.

```
ssh admin@<VM-IP>
```

2. Run `sudo /usr/local/scaligent/bin/prepare_disks.sh`.

```
$ sudo /usr/local/scaligent/bin/prepare_disks.sh
```

3. Configure the VM based on the site-survey.
4. Repeat this process for each of your VMs.

## Install cluster

To install your ThoughtSpot cluster, complete the installation process outlined in [Installing ThoughtSpot in GCP \[See page 240\]](#).

## Related information

[Connecting to Google Cloud Storage buckets \[See page 0\]](#)

[Loading data from a GCP GCS bucket \[See page 0\]](#)

# Configure ThoughtSpot nodes in GCP

**Summary:** Prepare to install your ThoughtSpot cluster by configuring nodes.

Before you can install a ThoughtSpot cluster in GCP, you must configure your nodes.

## Installation Prerequisites

Ensure the successful creation of the virtual machines (VMs) before you install the ThoughtSpot cluster in GCP.

1. **Review configuration options** Refer to [GCP configuration options \[See page 226\]](#) for detailed instance specs.
2. **Create the instance** Refer to [Set up GCP for ThoughtSpot \[See page 228\]](#) to create and launch your instance.
3. **Review required ports** Refer to [Network Policies \[See page 44\]](#) to view the required ports for successful operation of ThoughtSpot.

## Configure Nodes

After creating the instance, you must configure the nodes. Follow the steps in this checklist.

- Step 1: Log in to your cluster [\[See page 0\]](#)
- Step 2: Get a template for network configuration [\[See page 0\]](#)
- Step 3: Prepare node configuration [\[See page 0\]](#)
- Step 4: Configure the nodes [\[See page 0\]](#)
- Step 5: Confirm node configuration [\[See page 0\]](#)

### Step 1: Log in to your cluster

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

Ask your network administrator if you do not know the admin credentials.

1. Run `ssh admin@<nodeIP>`.

Replace `nodeIP` with your specific network information.

```
$ ssh admin@<nodeIP>
```

2. Enter your admin password at the prompt.

Ask your network administrator if you don't know the password.

**Note:** The password does not appear on the screen as you type it.

## Step 2: Get a template for network configuration

Run the `tscli cluster get-config` command to get a template for network configuration for the new cluster. Redirect it to the file `nodes.config`.

You can find more information on this process in the `nodes.config` [file reference \[See page 27\]](#).

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

## Step 3: Prepare node configuration

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the [autodiscovery of one node example \[See page 29\]](#). Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

2. Fill in the areas specified in [Parameters of the nodes.config file \[See page 30\]](#) with your specific network information.

If you have additional nodes, complete each node within the `nodes.config` file in the same

way.

Do not edit any part of the `nodes.config` file except the sections described in [Parameters of the nodes.config file \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, it may cause setup to fail.

#### Step 4: Configure the nodes

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

1. Disable the `firewalld` service by running `sudo systemctl stop firewalld` in your terminal. The `firewalld` service is a Linux firewall that must be off for ThoughtSpot installation. After the cluster installer reboots the nodes, `firewalld` automatically turns back on.

```
$ sudo systemctl stop firewalld
```

2. To make sure you temporarily disabled `firewalld`, run `sudo systemctl status firewalld`. Your output should specify that `firewalld` is inactive. It may look something like the following:

```
$ sudo systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
          Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
          Active: inactive (dead)
```

3. Run the configuration command: `$ cat nodes.config | tscli cluster set-config`.

If the command returns an error, refer to [set-config error recovery \[See page 244\]](#).

After you run the node configuration command, your output appears similar to the following:

```
$ 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 NTP Servers
Setting up Timezone
Done setting up ThoughtSpot
```

### Step 5: Confirm node configuration

Use the `get-config` command to confirm node configuration.

Your output may look similar to the following:

```
$ 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 ThoughtSpot software

Next, [install your ThoughtSpot clusters \[See page 247\]](#).

## Error recovery

### Set-config error recovery

If you get a warning about node detection when you run the `set-config` command, restart the node-scout service.

Your error may look something like the following:

```
Connecting to local node-scout
WARNING: Detected 0 nodes, but found 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 Configuration failed.
Please retry or contact support.
```

Restart the node-scout service with the following set of commands.

```
$ sudo vim /etc/systemd/system/node-scout.service
$ sudo systemctl daemon-reload
$ sudo systemctl restart node-scout
```

Ensure that you restarted the node-scout by running `sudo systemctl status node-scout`. Your output should specify that the node-scout service is active. It may look something like the following:

```
$ sudo systemctl status node-scout
● node-scout.service - Setup Node Scout service
  Loaded: loaded (/etc/systemd/system/node-scout.service; enabled; vendor preset: disabled)
  Active: active (running) since Fri 2019-12-06 13:56:29 PST;
  4s ago
```

Next, retry the `set-config` command.

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

The command output should no longer have a warning.

## Related information

Use these references for successful installation and administration of ThoughtSpot.

- [The nodes.config file \[See page 0\]](#)

- Parameters of the `nodes.config` file [See page 30]
- Using the `tscli cluster create` command [See page 33]
- Parameters of the `cluster create` command [See page 41]
- ThoughtSpot Documentation [See page 0]
- Contact Support [See page 290]

# Install ThoughtSpot clusters in GCP

**Summary:** Learn how to install ThoughtSpot clusters in GCP.

## Prerequisites

Before you can install your ThoughtSpot clusters in GCP, complete these prerequisites.

1. **Review configuration options** Refer to [GCP configuration options \[See page 226\]](#) for detailed instance specs.
2. **Create the instance** Refer to [Set up ThoughtSpot in GCP \[See page 228\]](#) to create and launch your instance.
3. **Review required ports** Refer to [Network Policies \[See page 44\]](#) to view the required ports for successful operation of ThoughtSpot.
4. **Configure nodes** Refer to [Configure ThoughtSpot nodes in GCP \[See page 240\]](#) to configure your nodes.

## Install ThoughtSpot Software

Install the cluster using the ThoughtSpot software release bundle. The estimated installation time is one hour. Follow the steps in this checklist.

- Step 1: Run the installer [\[See page 0\]](#)
- Step 2: Check cluster health [\[See page 0\]](#)
- Step 3: Finalize installation [\[See page 0\]](#)

Refer to your welcome letter from ThoughtSpot to find the link to download the release bundle. If you do not have a link, open a support ticket at [ThoughtSpot Support \[See page 0\]](#) to request access to the release bundle.

### Step 1: Run the installer

1. Copy the downloaded release bundle to `/export/sdc1/TS_TASKS/install` using the following command:

```
$ scp <release-number>.tar.gz admin@<hostname>:/export/  
sdc1/TS_TASKS/install/<file-name>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 6.0, and so on.
- `hostname` is your specific hostname.
- `file-name` is the name of the tarball file on your local computer.

**Note:** You can use another secure copy method, if you prefer a method other than the `scp` command.

2. Alternatively, use `tscli fileserver download-release` to download the release bundle.

You must [configure the fileserver \[See page 0\]](#) by running `tscli fileserver configure` before you can download the release.

```
$ tscli fileserver download-release <release-number> --  
user <username> --out <release-location>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 5.3.1, 6.0, and so on.
- `username` is the username for the fileserver that you set up earlier, when configuring the fileserver.
- `release-location` is the location path of the release bundle on your local machine. For example, `/export/sdc1/TS_TASKS/install/6.0.tar.gz`.

3. Verify the checksum to ensure you have the correct release.

Run `md5sum -c <release-number>.tar.gz.MD5checksum`.

```
$ md5sum -c <release-number>.tar.gz.MD5checksum
```

Your output says `ok` if you have the correct release.

4. Launch a [screen \[See page 0\]](#) session. Use screen to ensure that your installation does not stop if you lose network connectivity.

```
$ screen -S DEPLOYMENT
```

5. Create the cluster.

Run `tscli cluster create` to create the cluster.

If you are using a gcs bucket for object storage, include the flag `--enable_cloud_storage=gcs`.

```
$ tscli cluster create <release-number>.tar.gz --enable_cloud_storage=gcs
```

6. Edit the output with your specific cluster information.

For more information on this process, refer to [Using the tscli cluster create command \[See page 33\]](#) and [Parameters of the cluster create command \[See page 41\]](#).

The cluster installer automatically reboots all the nodes after a successful install. The `firewalld` service automatically turns on. At this time, the system is rebooting, which may take approximately 15 minutes.

Log in to any node to check the current cluster status:

```
$ tscli cluster status
```

## Step 2: Check cluster health

After the cluster installs, check its status using the `tscli cluster status` command.

Your output may look similar to the following:

```
$ 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
```

Ensure that the cluster is `RUNNING` and that the Database and Search Engine are `READY`.

```
$ tscli cluster check
Connecting to hosts...
[Wed Jan  8 23:15:47 2020] START Diagnosing ssh
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing connection
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing zookeeper
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing sage
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing timezone
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing disk
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing cassandra
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing hdfs
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing orion-oreo
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing memcheck
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing ntp
[Wed Jan  8 23:16:08 2020] SUCCESS
#####
```

```
#####
[Wed Jan  8 23:16:08 2020] START Diagnosing trace_vault
[Wed Jan  8 23:16:09 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:09 2020] START Diagnosing postgres
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing disk-health
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing falcon
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing orion-cgroups
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing callosum
/usr/lib/python2.7/site-packages/urllib3/connectionpool.py:85
2: InsecureRequestWarning: Unverified HTTPS request is being ma
de. Adding certificate verification is strongly advised. See: h
ttps://urllib3.readthedocs.io/en/latest/advanced-usage.html#ss
l-warnings
    InsecureRequestWarning)
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
```

Your output may look something like the above. Ensure that all diagnostics show `SUCCESS`.

**● Warning:** If `tscli cluster check` returns an error, it may suggest you run `tscli storage gc` to resolve the issue. If you run `tscli storage gc`, note that it restarts your cluster.

### Step 3: Finalize installation

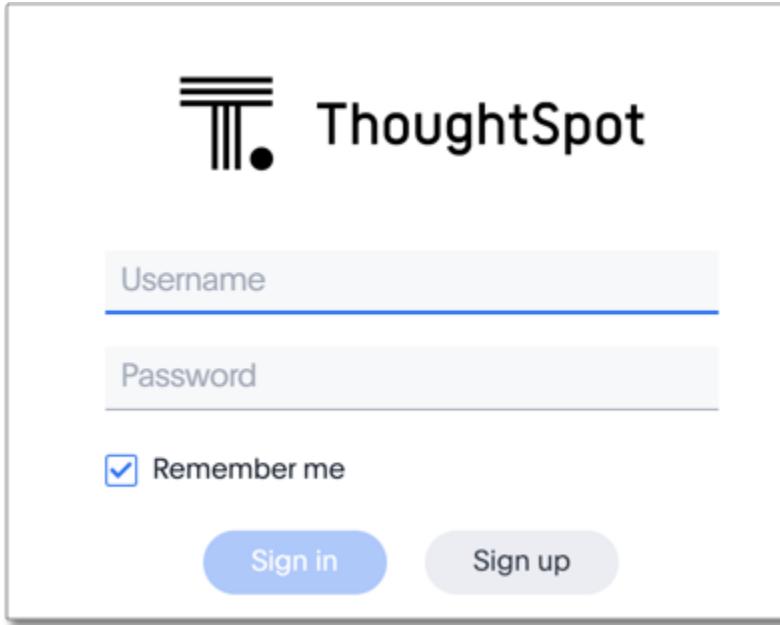
After the cluster status changes to `READY`, sign in to ThoughtSpot on your browser. Follow these steps:

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:
  - Click **Advanced**
  - Click **Proceed**
4. The ThoughtSpot sign-in page appears.
5. In the [ThoughtSpot sign-in window \[See page 253\]](#), enter admin credentials, and click **Sign in**.  
ThoughtSpot recommends changing the default admin password.



## Lean configuration

**(For use with thin provisioning only)** If you have a [small or medium instance type \[See page 167\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 290\]](#) for assistance with this configuration.

## Related information

Use these references for successful installation and administration of ThoughtSpot:

- [the nodes.config file \[See page 0\]](#)
- [Parameters of the nodes.config file \[See page 30\]](#)
- [Using the tscli cluster create command \[See page 33\]](#)
- [Parameters of the `cluster create` command \[See page 41\]](#)
- [Deployment Overview \[See page 116\]](#)
- [Contact Support \[See page 290\]](#)

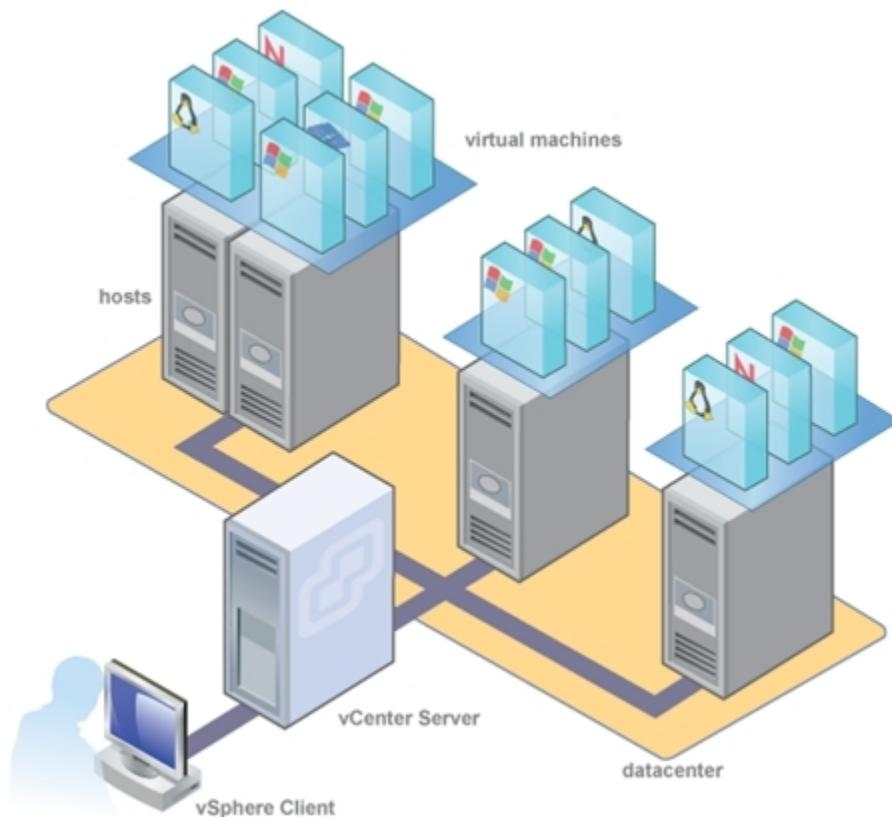
# VMware configuration overview

**Summary:** You can host ThoughtSpot on VMware.

This section is an overview of the ThoughtSpot AI-Driven analytics platform hosted on the VMware vSphere Hypervisor (ESXi) 6.5 environment.

## About ThoughtSpot in VMware

The VMware virtualization platform provides highly scalable and efficient memory and CPU resources management that can be used by ThoughtSpot instances. Additionally, the VMware virtualization environment is an easy transition between development and production environments. The following diagram shows the components of a VMware and ThoughtSpot architecture:



**Note:** This is a generic representation; Only CentOS-based virtual machines are supported with

ThoughtSpot.

Your database capacity will determine the number of ThoughtSpot instances and the instance network/storage requirements. In addition, you can scale your ThoughtSpot VMs as your dataset size grows.

## Supported configurations

ThoughtSpot Engineering has performed extensive testing of the ThoughtSpot platform in VMware for the best performance, load balancing, scalability, and reliability. Based on this testing, ThoughtSpot recommends the following *minimum specifications* for an individual VMware ESXi host machine:

Per VM user data capacity	CPU/RAM	Data disk
20 GB	16/128 GB	800 GB
100 GB	32/256 GB	800 GB
256 GB	72/512 GB	6 TB

**Note:** All cores must be hyperthreaded. 200GB SSD boot disk required for all configurations.

Locally attached storage provides the best performance.

SAN can be used, but must comply with the following requirements:

- 136 MBps minimum random read bandwidth
- 240 random IOPS (~4ms seek latency)

NAS/NFS is not supported since its latency is so high that it tends to be unreliable.

All virtualization hosts should have VMware vSphere Hypervisor (ESXi) 6.5 installed.

ThoughtSpot provides a VMware template (OVF) together with a VMDK (Virtual Machine Disk) file for configuring a VM. VMDK is a file format that describes containers for virtual hard disk drives to be used in virtual machines like VMware Workstation or VirtualBox. OVF is a platform-independent, efficient, extensible, and open packaging distribution format for virtual machines.

The ThoughtSpot VM configuration uses thin provisioning and sets the recommended reserved memory, among other important specifications. You can obtain these files from your ThoughtSpot Customer Success Engineer.

## Questions or comments?

We hope your experience with ThoughtSpot is excellent. Please let us know how it goes, and what we can do to make it better. You can [contact ThoughtSpot \[See page 290\]](#) by email, phone, or by filing a support ticket.

# Set up ThoughtSpot in VMware

**Summary:** Learn how to install a ThoughtSpot cluster in a VMware environment.

This page explains how to install a ThoughtSpot cluster in a VMware VSphere Hypervisor (ESXi) 6.5 environment.

**Note:** Older versions of ESXi aren't supported due to hardware/driver incompatibility issues.

For each hardware node, you must:

- Complete the prerequisites
- Use the ThoughtSpot Open Virtualization Format (OVF) file to create a virtual machine (VM)
- Add hard disks to the VM

## Prerequisites

This installation process assumes you have already acquired your host machines. You can install on a cluster with any number of nodes. A one node cluster is suitable for a sandbox environment but is insufficient for a production environment. You need at least three nodes for high availability (HA), but there is no limit on the number of nodes.

1. Make sure you have installed the Hypervisor on each of your nodes.

The VM template, by default, captures a 72-core configuration. If your physical host has more than 72 cores, you may want to edit VM to have ( n-2 ) cores (for a physical host with n cores) to fully take advantage of computing power of the physical host. Extra cores help performance.

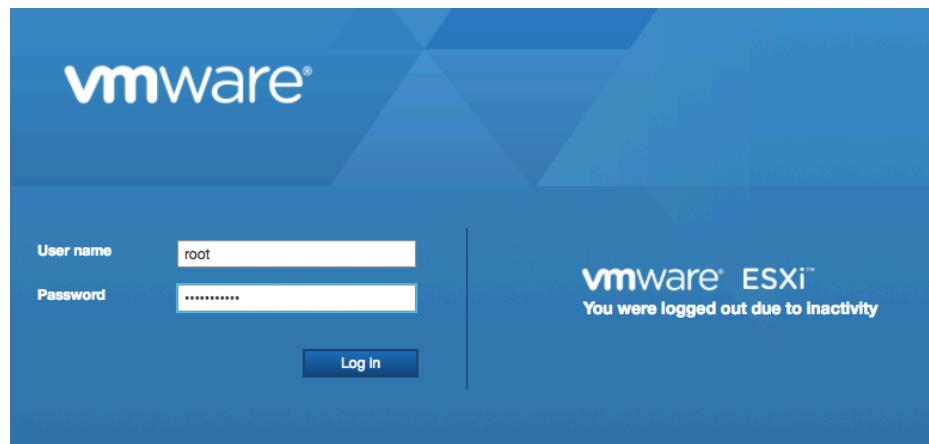
You should aim to allocate 490 GB or more RAM.

2. Create datastores for all solid-state drive (SSD) and hard drive devices.

3. Download [See page 0] and fill out the ThoughtSpot site survey to have a quick reference for any networking information you may need to fill out as you install ThoughtSpot. Ask your network administrator if you need help filling out the site survey.

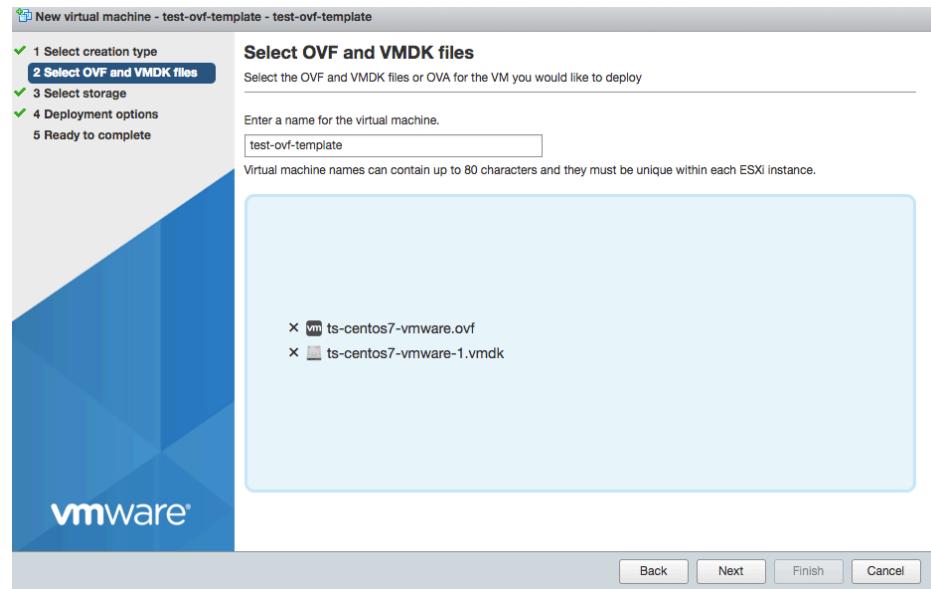
## Use the OVF to Create a VM

1. Download [See page 0] the ThoughtSpot OVF to a location on an accessible disk.
2. Log in to the ESXi web portal.



3. Select **Virtual Machines > Create/Register VM**.

The system displays the dialog for selecting an OVF template.



4. Choose the OVF template and click **Next**.

The system prompts you to select a storage.

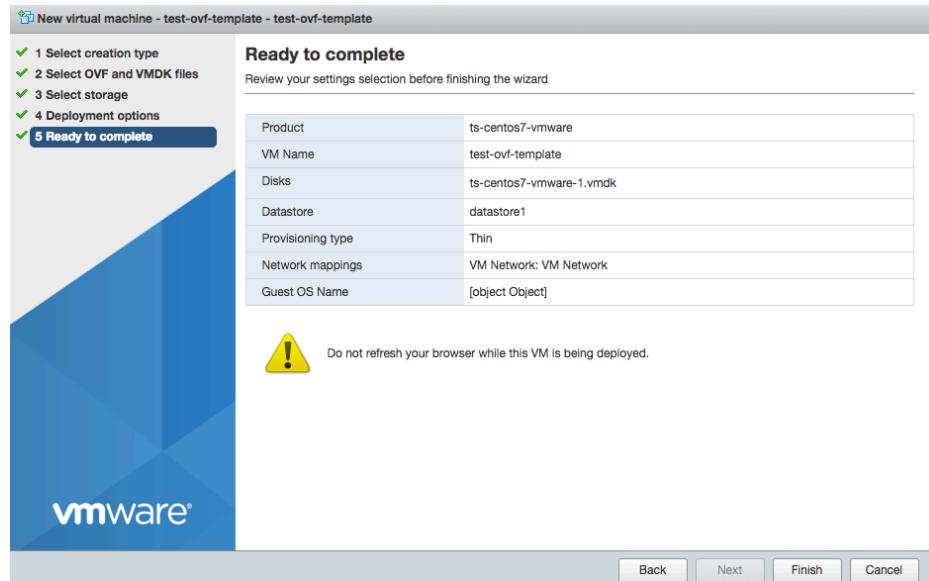
5. Choose the SSD as the destination and click **Next**.

The system displays the **Deployment Options** dialog.

6. Enter the options and click **Next**.

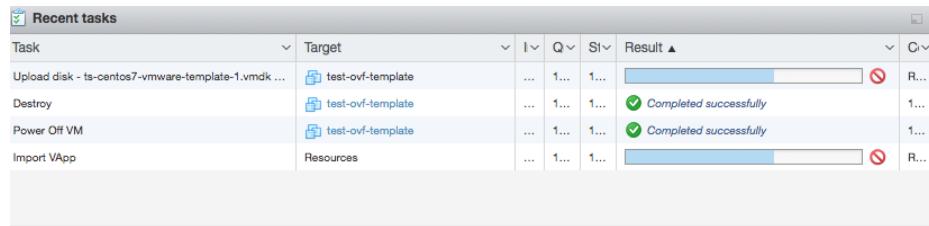
Setting	Value
Network mappings	Select the correct network for your installation.
Disk provisioning	Choose Thin.
Power on automatically	Check this box.

7. Review your selection and click **Finish**.



- Wait for the template to be loaded.

Depending on your network speed, loading can take several minutes.



- Make sure that VM is powered off.

## Add hard disks to the VM

### Use Case    HDFS Disk Requirements

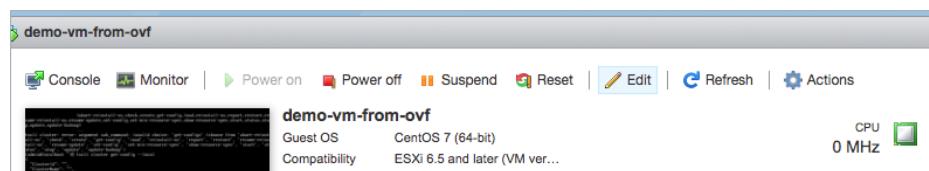
POC      2 x 1 TB on HDD

Production    3 x 2 TB on HDD

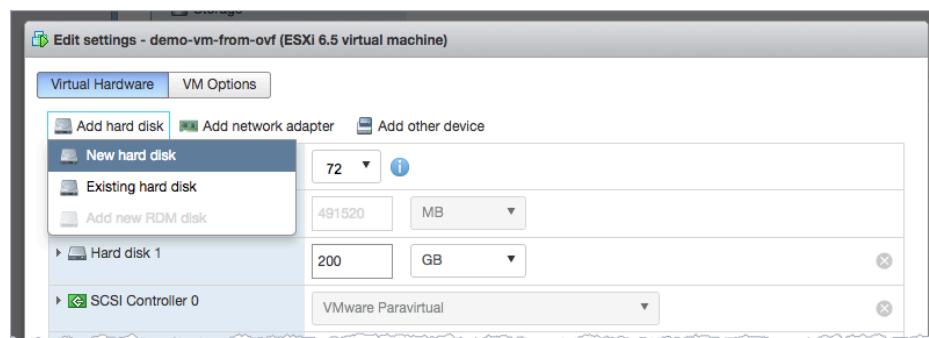
For a proof of concept (POC), follow these steps to create two 1TB HDFS disks on HDD storage, as shown here (2 x 1TB).

For production deployments, ThoughtSpot requires you to have three 2TB HDFS disks on HDD (3 x 2TB). For this use case, follow these same steps to create the additional, larger capacity disks.

1. Edit the VM you just created.



2. Select **Add hard disk > New hard disk**.



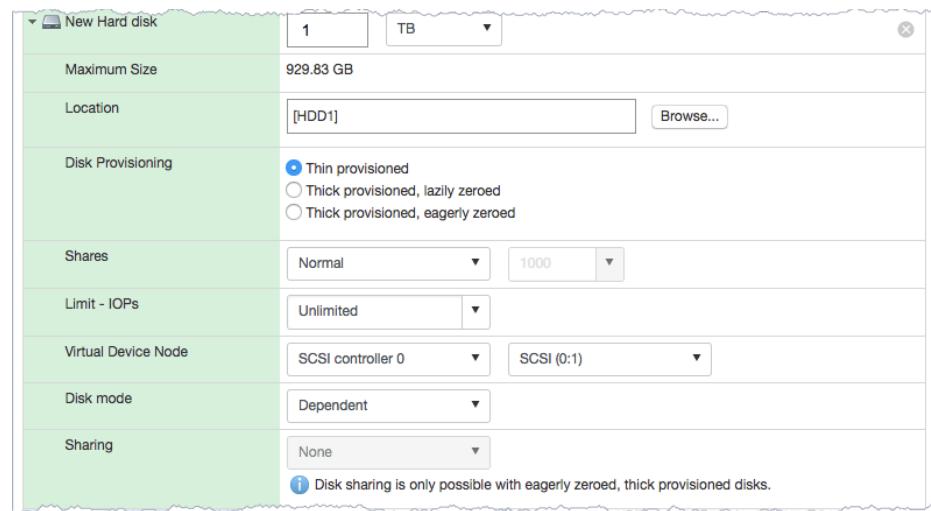
The new disk appears as a new row under the only existing SSD row.

3. Click the **New Hard disk** to expand the detailed configuration options.
4. For a proof of concept, set the options as follows. (For production deployments, set the size to 2TB.)

Setting	Value
size	1 TB
Location	Use the <b>Browse</b> button to select the hard disk store.

**Thin provisioned** Check this box.

You should see something similar to the following:



5. Save your changes.
6. Repeat steps 1-5 to create more hard disks.
7. Power on the VM
8. After the VM is online, run the following command to prepare the HDFS disks:

```
$ sudo /usr/local/scaligent/bin/prepare_disks.sh
```

## Next steps

There is no network at this point on your VMs. As a prerequisite:

1. Verify that Network Adapter type is set to VMware vmxnet3 (Recommended).
2. Verify that all ESXi hosts in your VMware farm for ThoughtSpot have been trunks to the VLAN assigned to your ThoughtSpot VMs.
3. Verify that the console of all ThoughtSpot VMs is accessible in VMware vCenter Server.

# Configure ThoughtSpot nodes in VMware

**Summary:** Prepare to install your ThoughtSpot cluster by configuring nodes.

Before you can install a ThoughtSpot cluster in VMware, you must configure your nodes.

## Installation Prerequisites

Ensure the successful creation of the virtual machines (VMs) before you install the ThoughtSpot cluster in VMware.

1. **Download the OVF** Download the [Open Virtualization Format \[See page 0\]](#) (OVF) file.
2. **Review configuration overview** Refer to [VMware configuration overview \[See page 255\]](#) for detailed instance specs.
3. **Create the instance** Refer to [Set up VMware for ThoughtSpot \[See page 258\]](#) to create and launch your instance.
4. **Review required ports** Refer to [Network Policies \[See page 44\]](#) to view the required ports for successful operation of ThoughtSpot.

## Configure Nodes

After creating the instance, you must configure the nodes. Follow the steps in this checklist.

- Step 1: Log in to your cluster [See page 0]
- Step 2: Get a template for network configuration [See page 0]
- Step 3: Prepare node configuration [See page 0]
- Step 4: Configure the nodes [See page 0]
- Step 5: Confirm node configuration [See page 0]

## Step 1: Log in to your cluster

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

Ask your network administrator if you do not know the admin credentials.

1. Run `ssh admin@<nodeIP>`.

Replace `nodeIP` with your specific network information.

```
$ ssh admin@<nodeIP>
```

2. Enter your admin password at the prompt.

Ask your network administrator if you don't know the password.

**❶ Note:** The password does not appear on the screen as you type it.

## Step 2: Get a template for network configuration

Run the `tscli cluster get-config` command to get a template for network configuration for the new cluster. Redirect it to the file `nodes.config`.

You can find more information on this process in the `nodes.config` [file reference \[See page 27\]](#).

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

## Step 3: Prepare node configuration

1. Add your specific network information for the nodes in the `nodes.config` file, as demonstrated in the [autodiscovery of one node example \[See page 29\]](#). Run `vim nodes.config` to edit the file.

```
$ vim nodes.config
```

**Note:** Some of the information in the `nodes.config` file may be pre-populated from earlier steps. For example, if you specified an IP address while creating VMs, that IP address might already be present in your `nodes.config` file.

2. Fill in the areas specified in [Parameters of the `nodes.config` file \[See page 30\]](#) with your specific network information.

If you have additional nodes, complete each node within the `nodes.config` file in the same way.

Do not edit any part of the `nodes.config` file except the sections described in [Parameters of the `nodes.config` file \[See page 30\]](#). If you delete quotation marks, commas, or other parts of the code, it may cause setup to fail.

#### Step 4: Configure the nodes

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

1. Disable the `firewalld` service by running `sudo systemctl stop firewalld` in your terminal. The `firewalld` service is a Linux firewall that must be off for ThoughtSpot installation. After the cluster installer reboots the nodes, `firewalld` automatically turns back on.

```
$ sudo systemctl stop firewalld
```

2. To make sure you temporarily disabled `firewalld`, run `sudo systemctl status firewalld`. Your output should specify that `firewalld` is inactive. It may look something like the following:

```
$ sudo systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
          Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset: enabled)
          Active: inactive (dead)
```

3. Run the configuration command: `$ cat nodes.config | tscli cluster set-config`.

If the command returns an error, refer to [set-config error recovery \[See page 268\]](#).

After you run the node configuration command, your output appears similar to the following:

```
$ 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 NTP Servers
Setting up Timezone
Done setting up ThoughtSpot
```

## Step 5: Confirm node configuration

Use the `get-config` command to confirm node configuration.

Your output may look similar to the following:

```
$ 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 ThoughtSpot software

Next, [install your ThoughtSpot clusters \[See page 271\]](#).

## Error recovery

### Set-config error recovery

If you get a warning about node detection when you run the `set-config` command, restart the node-scout service.

Your error may look something like the following:

```
Connecting to local node-scout
WARNING: Detected 0 nodes, but found 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 Configuration failed.
Please retry or contact support.
```

Restart the node-scout service with the following set of commands.

```
$ sudo vim /etc/systemd/system/node-scout.service
$ sudo systemctl daemon-reload
$ sudo systemctl restart node-scout
```

Ensure that you restarted the node-scout by running `sudo systemctl status node-scout`. Your output should specify that the node-scout service is active. It may look something like the following:

```
$ sudo systemctl status node-scout
● node-scout.service - Setup Node Scout service
  Loaded: loaded (/etc/systemd/system/node-scout.service; enabled; vendor preset: disabled)
  Active: active (running) since Fri 2019-12-06 13:56:29 PS
T; 4s ago
```

Next, retry the `set-config` command.

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

The command output should no longer have a warning.

## Related information

Use these references for successful installation and administration of ThoughtSpot.

- [the nodes.config file \[See page 0\]](#)

- Parameters of the `nodes.config` file [See page 30]
- Using the `tscli cluster create` command [See page 33]
- Parameters of the `cluster create` command [See page 41]
- ThoughtSpot Documentation [See page 0]
- Contact Support [See page 290]

# Install ThoughtSpot clusters in VMware

**Summary:** Learn how to install ThoughtSpot clusters in VMware.

## Prerequisites

Before you can install your ThoughtSpot clusters in VMware, complete these prerequisites.

1. **Download the OVF** Download the [Open Virtualization Format \[See page 0\]](#) (OVF) file.
2. **Review configuration overview** Refer to [VMware configuration overview \[See page 255\]](#) for detailed instance specs.
3. **Create the instance** Refer to [Set up VMware for ThoughtSpot \[See page 258\]](#) to create and launch your instance.
4. **Review required ports** Refer to [Network Policies \[See page 44\]](#) to view the required ports for successful operation of ThoughtSpot.
5. **Configure nodes** Refer to [Configure ThoughtSpot nodes in VMware \[See page 264\]](#) to configure your nodes.

## Install ThoughtSpot Software

Install the cluster using the ThoughtSpot software release bundle. The estimated installation time is one hour. Follow the steps in this checklist.

- Step 1: Run the installer [\[See page 0\]](#)
- Step 2: Check cluster health [\[See page 0\]](#)
- Step 3: Finalize installation [\[See page 0\]](#)

Refer to your welcome letter from ThoughtSpot to find the link to download the release bundle. If you do not have a link, open a support ticket at [ThoughtSpot Support \[See page 0\]](#) to request access to the release bundle.

### Step 1: Run the installer

1. Copy the downloaded release bundle to `/export/sdb1/TS_TASKS/install` using the

following command:

```
$ scp <release-number>.tar.gz admin@<hostname>:/export/sdb1/TS_TASKS/install/<file-name>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 6.0, and so on.
- `hostname` is your specific hostname.
- `file-name` is the name of the tarball file on your local computer.

**Note:** You can use another secure copy method, if you prefer a method other than the `scp` command.

2. Alternatively, use `tscli fileserver download-release` to download the release bundle.

You must [configure the fileserver \[See page 0\]](#) by running `tscli fileserver configure` before you can download the release.

```
$ tscli fileserver download-release <release-number> --user <username> --out <release-location>
```

Note the following parameters:

- `release-number` is the release number of your ThoughtSpot instance, such as 5.3, 5.3.1, 6.0, and so on.
- `username` is the username for the fileserver that you set up earlier, when configuring the fileserver.
- `release-location` is the location path of the release bundle on your local machine. For example, `/export/sdb1/TS_TASKS/install/6.0.tar.gz`.

3. Verify the checksum to ensure you have the correct release.

Run `md5sum -c <release-number>.tar.gz.MD5checksum`.

```
$ md5sum -c <release-number>.tar.gz.MD5checksum
```

Your output says `ok` if you have the correct release.

4. Launch a [screen \[See page 0\]](#) session. Use screen to ensure that your installation does not stop if you lose network connectivity.

```
$ screen -S DEPLOYMENT
```

5. Create the cluster.

Run `tscli cluster create` to create the cluster.

```
$ tscli cluster create <release-number>.tar.gz
```

6. Edit the output with your specific cluster information.

For more information on this process, refer to [Using the tscli cluster create command \[See page 33\]](#) and [Parameters of the cluster create command \[See page 41\]](#).

The cluster installer automatically reboots all the nodes after a successful install. The `firewalld` service automatically turns on. At this time, the system is rebooting, which may take approximately 15 minutes.

Log in to any node to check the current cluster status:

```
$ tscli cluster status
```

## Step 2: Check cluster health

After the cluster installs, check its status using the `tscli cluster status` command.

Your output may look similar to the following:

```
$ 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
```

Ensure that the cluster is `RUNNING` and that the Database and Search Engine are `READY`.

```
$ tscli cluster check
Connecting to hosts...
[Wed Jan  8 23:15:47 2020] START Diagnosing ssh
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing connection
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing zookeeper
[Wed Jan  8 23:15:47 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:47 2020] START Diagnosing sage
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing timezone
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing disk
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing cassandra
[Wed Jan  8 23:15:48 2020] SUCCESS
#####
#####
[Wed Jan  8 23:15:48 2020] START Diagnosing hdfs
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing orion-oreo
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing memcheck
[Wed Jan  8 23:16:02 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:02 2020] START Diagnosing ntp
[Wed Jan  8 23:16:08 2020] SUCCESS
#####
```

```
#####
[Wed Jan  8 23:16:08 2020] START Diagnosing trace_vault
[Wed Jan  8 23:16:09 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:09 2020] START Diagnosing postgres
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing disk-health
[Wed Jan  8 23:16:11 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:11 2020] START Diagnosing falcon
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing orion-cgroups
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
[Wed Jan  8 23:16:12 2020] START Diagnosing callosum
/usr/lib/python2.7/site-packages/urllib3/connectionpool.py:85
2: InsecureRequestWarning: Unverified HTTPS request is being ma
de. Adding certificate verification is strongly advised. See: h
ttps://urllib3.readthedocs.io/en/latest/advanced-usage.html#ss
l-warnings
    InsecureRequestWarning)
[Wed Jan  8 23:16:12 2020] SUCCESS
#####
#####
```

Your output may look something like the above. Ensure that all diagnostics show `SUCCESS`.

**● Warning:** If `tscli cluster check` returns an error, it may suggest you run `tscli storage gc` to resolve the issue. If you run `tscli storage gc`, note that it restarts your cluster.

### Step 3: Finalize installation

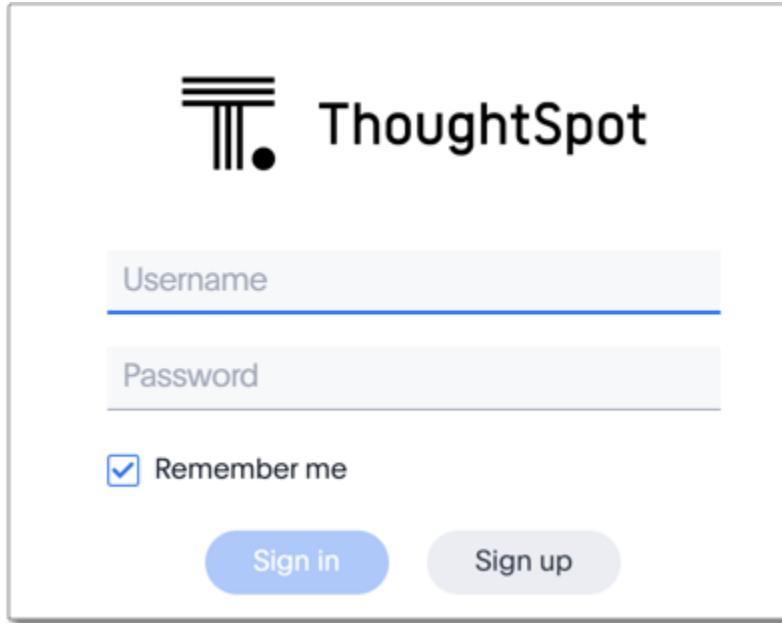
After the cluster status changes to `READY`, sign in to ThoughtSpot on your browser. Follow these steps:

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:
  - Click **Advanced**
  - Click **Proceed**
4. The ThoughtSpot sign-in page appears.
5. In the [ThoughtSpot sign-in window \[See page 277\]](#), enter admin credentials, and click **Sign in**.  
ThoughtSpot recommends changing the default admin password.



## Lean configuration

**(For use with thin provisioning only)** If you have a [small or medium instance type \[See page 167\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 290\]](#) for assistance with this configuration.

## Related information

Use these references for successful installation and administration of ThoughtSpot:

- [The nodes.config file \[See page 0\]](#)
- [Parameters of the nodes.config file \[See page 30\]](#)
- [Using the tscli cluster create command \[See page 33\]](#)
- [Parameters of the `cluster create` command \[See page 41\]](#)
- [Deployment Overview \[See page 116\]](#)
- [Contact Support \[See page 290\]](#)

# Network policies

**Summary:** Lists the required and optional ports for an installation.

For regular operations and for debugging, there are some ports you must keep open to network traffic from end users. Another, larger list of ports must be kept open for network traffic between the nodes in the cluster.

## Required ports for operations and debugging

The following ports must be open for requests from your user population.

Port	Protocol	Service Name	Direction	Source	Destination	Description
22	SSH	SSH	bidirectional	Administrators IP addresses	All nodes	Secure shell access. Also used for scp (secure copy).
443	HTTPS	HTTPS	bidirectional	All users IP addresses	All nodes	Secure HTTP.
12345	TCP	Simba	bidirectional	Administrators IP addresses	All nodes	Port used by ODBC and JDBC drivers when connecting to ThoughtSpot.

## Network Ports

This reference lists the potential ports to open when setting up your security group.

### Required ports for intracluster operation

Static ports are used for communication between services within the cluster. ThoughtSpot recommends that you open all ports within a cluster. This is not required, but it will ensure that cluster communication works properly if additional ports are used in a future software release.

If your organization does not allow you to open all ports, make sure you open the required intrACLuster ports listed in the following table. In addition, a number of ports are dynamically assigned to services, which change between runs. The dynamic ports come from the range of ports that are dynamically allocated by Linux (20K+).

Port	Protocol	Service Name	Direction	Source	Dest.	Description
80	TCP	nginx	inbound	All nodes	All nodes	Primary app HTTP port (nginx)
443	TCP	Secure nginx	inbound	All nodes	All nodes	Primary app HTTPS port (nginx)
2100	RPC	Oreo RPC port	bidirectional	All nodes	All nodes	Node daemon RPC
2101	HTTP	Oreo HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Node daemon HTTP
2181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
3181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
4181	TCP	Zookeeper servers listen on this RPC port for client connections	bidirectional	All nodes	All nodes	Zookeeper servers listen on this RPC port for client connections
2200	RPC	Orion master RPC port	bidirectional	All nodes	All nodes	Internal communication with the cluster manager
2201	HTTP	Orion master HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the cluster manager

Port	Protocol	Service Name	Direction	Source	Dest.	Description
2205	TCP	Cluster update service TCP port	bidirectional	All nodes	All nodes	Internal communication with the cluster manager
2210	RPC	Cluster stats service RPC port	bidirectional	All nodes	All nodes	Internal communication with the stats collector
2211	HTTP	Cluster stats service HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the stats collector
2230	RPC	Callosum stats collector RPC port	bidirectional	All nodes	All nodes	Internal communication with the BI stats collector
2231	HTTP	Callosum stats collector HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the BI stats collector
2240	RPC	Alert manager	bidirectional	All nodes	All nodes	Port where alerting service receives alert events
2241	RPC	Alert manager	bidirectional	All nodes	All nodes	Port where alerting service receives alert events
2888	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
3181	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Dest.</b>	<b>Description</b>
3888	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
4000	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4001	HTTP	Falcon worker HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the data cache
4002	HTTP	Falcon worker HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the data cache
4003	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4004	RPC	Falcon worker RPC port	bidirectional	All nodes	All nodes	Port used by data cache for communication between themselves
4021	RPC	Sage metadata service port (exported by Tomcat), Callosum services like meta-data services, medata-dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata

Port	Protocol	Service Name	Direction	Source	Dest.	Description
4181	RPC	Ports used by Zookeeper servers for communication between themselves	bidirectional	All nodes	All nodes	Ports used by Zookeeper servers for communication between themselves
4201	HTTP	Sage auto complete server HTTP interface port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4231	HTTP	Sage index server HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4232	RPC	Sage index server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4233	RPC	Sage index server RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4241	HTTP	Sage auto complete server HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Port used to debug the search service
4242	RPC	Sage auto complete server RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4243	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4244	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4245	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication

Port	Protocol	Service Name	Direction	Source	Dest.	Description
4243	RPC	Sage auto complete server metadata subscriber port	bidirectional	All nodes	All nodes	Port used for search internal communication
4251	RPC	Sage master RPC port	bidirectional	All nodes	All nodes	Port used for search service internal communication
4405	RPC	Diamond (graphite) port	bidirectional	All nodes	All nodes	Port used for communication with monitoring service
4406	RPC	Diamond (graphite) port	bidirectional	All nodes	All nodes	Port used for communication with monitoring service
4500	RPC	Trace vault service RPC port	bidirectional	All nodes	All nodes	Trace collection for ThoughtSpot services
4501	HTTP	Trace vault service HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug trace collection
4851	RPC	Graphite manager RPC port	bidirectional	All nodes	All nodes	Communication with graphite manager
4852	HTTP	Graphite manager HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug graphite manager
4853	RPC	Elastic search stack (ELK) manager RPC port	bidirectional	All nodes	All nodes	Communication with log search service
4853	HTTP	Elastic search stack (ELK) manager HTTP port	bidirectional	Admin IP addresses and all nodes	All nodes	Debug log search service
9200	RPC	Elastic search (ELK)	bidirectional	All nodes	All nodes	Communication with log search service

Port	Protocol	Service Name	Direction	Source	Dest.	Description
5021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
5432	Postgres	Postgres database server port	bidirectional	All nodes	All nodes	Communication with Postgres database
6021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
7021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
8020	RPC	HDFS namenode server RPC port	bidirectional	All nodes	All nodes	Distributed file system (DFS) communication with clients
8021	RPC	Callosum services like meta-data services, metadata dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts metadata service for metadata
8080	HTTP	Tomcat	bidirectional	All nodes	All nodes	BI engine communication with clients
8081	HTTP	Callosum/Tomcat status	bidirectional	All nodes	All nodes	BI engine communication with clients
8787	HTTP	Periscope (UI) service HTTP port	bidirectional	All nodes	All nodes	Administration UI back end

Port	Protocol	Service Name	Direction	Source	Dest.	Description
8888	HTTP	HTTP proxy server (tinyproxy)	bidirectional	All nodes	All nodes	Reverse SSH tunnel
11211	Memcached	Memcached server port	bidirectional	All nodes	All nodes	BI engine cache
12345	ODBC	Simba server port	bidirectional	All nodes	All nodes	Port used for ETL (extract, transform, load)
8480	HTTP	HDFS journalnode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
8485	HTTP	HDFS journalnode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50070	HTTP	HDFS namenode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50090	HTTP	HDFS secondary namenode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS metadata
50075	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
50010	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
50020	HTTP	HDFS datanode server HTTP port	bidirectional	All nodes	All nodes	Debug DFS data
7000	TCP	Cassandra KV store database	bidirectional	All nodes	All nodes	Debug DFS data
7001	TCP	Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
9042	HTTP	Munshi server impression service, Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
9160	TCP	Cassandra	bidirectional	All nodes	All nodes	Debug DFS data
4010	HTTP	Falcon moderator	bidirectional	All nodes	All nodes	Debug DFS data
4011	HTTP	Falcon moderator	bidirectional	All nodes	All nodes	Debug DFS data

Port	Protocol	Service Name	Direction	Source	Dest.	Description
20123 - 32768	TCP (dynamic)	Dynamic port in this range used for various services and ancillary services like atlas, caffeine, call-home, callosum, falcon, monitoring, munshi server, nlp, object_search, postgres, sage UBR, spo-tiq snapshot, timely	All nodes	Services		
5270	TCP	Cluster monitoring service (ELK)	bidirectional	All nodes	All nodes	Services
5271	TCP	Cluster monitoring service (ELK)	bidirectional	All nodes	All nodes	Services
5601	TCP	Kibana UI (ELK)	bidirectional	All nodes	All nodes	Services
6311	TCP	R service	bidirectional	All nodes	All nodes	Services
8008	TCP	Video recorder	bidirectional	All nodes	All nodes	Services
9090	TCP	Timely	bidirectional	All nodes	All nodes	Services
ICMPv4		Used for health check of cluster nodes	bidirectional	All nodes	All nodes	Services

### Required ports for inbound and outbound cluster access

ThoughtSpot uses static ports for inbound and outbound access to the cluster.

Port	Protocol	Service Name	Direction	Source	Dest.	Description
22	SCP	SSH	bidirectional	ThoughtSpot Support	All nodes	Secure shell access.
80	HTTP	HTTP	bidirectional	ThoughtSpot Support	All nodes	Hypertext Transfer Protocol for website traffic.

Port	Protocol	Service Name	Direction	Source	Dest.	Description
443	HTTPS	HTTPS	bidirectional	ThoughtSpot Support	All nodes	Secure HTTP.
12345	TCP	Simba	bidirectional	ThoughtSpot Support	All nodes	Port used by ODBC and JDBC drivers when connecting to ThoughtSpot.
2049	TCP	NFS: In case one needs to mount NFS share on TS node.	bidirectional	ThoughtSpot Support	All nodes	Port used by NFS.
123	UDP	NTP service	bidirectional	ThoughtSpot Support	All nodes	Port used by NTP service.

Port	Protocol	Service Name	Direction	Source	Destination	Description
443	TCP	HTTPS	outbound	All nodes	208.83.110.20	For transferring files to thoughtspot.egnyte.com.
443	TCP	HTTPS	outbound	All nodes	For transferring product usage data to mixpanel cloud.	outbound
443	TCP	HTTPS	outbound	All nodes	je8b47jfif.execute-api.us-east-2.amazonaws.com s3.us-west-1.amazonaws.com s3-us-west-1.amazonaws.com s3.dualstack.us-west-1.amazonaws.com	For transferring monitoring data to InfluxCloud. (Given address will resolve to point to AWS instances).
25 or 587	SMTP	SMTP or Secure SMTP	outbound	All nodes and SMTP relay (provided by customer)	All nodes	Allow outbound access for the IP address of whichever email relay server is in use. This is for sending alerts to ThoughtSpot Support.

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Destination</b>	<b>Description</b>
389 or 636	TCP	LDAP or LDAPS	outbound	All nodes and LDAP server (provided by cus- tomer)	All nodes	Allow outbound access for the IP address of the LDAP server in use.

### Required ports for IPMI (Intelligent Platform Management Interface)

ThoughtSpot uses static ports for out-of-band IPMI communications between the cluster and ThoughtSpot support.

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Dest.</b>	<b>Description</b>
80	HTTP	HTTP	bidirectional	ThoughtSpot Support	All nodes	Hypertext Transfer Protocol for website traffic.
443	TCP	S-HTTP	bidirectional	ThoughtSpot Support	All nodes	IPMI GUI and for HTML5-based IPMI console access.
623	UDP	Serial-over-LAN	bidirectional	ThoughtSpot Support	All nodes	IPMI GUI and for HTML5-based IPMI console access.

# Contact ThoughtSpot support

**Summary:** There are several ways to contact ThoughtSpot support.

You can contact ThoughtSpot by [phone \[See page 290\]](#), [mail \[See page 0\]](#), [email \[See page 291\]](#), or by [filing a support ticket \[See page 290\]](#).

## File a support ticket

If you encounter a technical issue, file a support ticket using the Support Portal ticket filing system at:

[http://support.thoughtspot.com/ \[See page 0\]](http://support.thoughtspot.com/)

Please provide as much detail as possible about your issue, to help us resolve it quickly.

You need a Support Portal login to file a ticket. If you do not have one, contact your ThoughtSpot liaison or email ThoughtSpot support using the details provided [below \[See page 291\]](#).

## Address

ThoughtSpot, Inc. 910 Hermosa Ct Sunnyvale, CA 94085

## Phone numbers

Phone Number	Description
1-800-508-7008 ext 1	ThoughtSpot Support
1-800-508-7008	Toll-free number for ThoughtSpot headquarters.

## Email

Reason for contacting us	Email
Sales inquiries	<a href="mailto:sales@thoughtspot.com">sales@thoughtspot.com</a> [See page 0]
Customer support and software update inquiries	<a href="mailto:support@thoughtspot.com">support@thoughtspot.com</a> [See page 0]
Other inquiries	<a href="mailto:hello@thoughtspot.com">hello@thoughtspot.com</a> [See page 0]