



---

# ThoughtSpot Deployment Guide for SMC

*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 SMC appliance.....	2
Prerequisites.....	3
Hardware requirements .....	5
Connect the appliance.....	8
Configure nodes .....	12
Install cluster .....	17
The nodes.config file.....	25
Parameters of the nodes.config file .....	28
Using the cluster create command .....	31
Parameters of the cluster create command.....	39
Network policies.....	42
Cable reference .....	53

# 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 3]
- Step 2: Review hardware requirements [See page 5]
- Step 3: Connect the SMC appliance [See page 8]
- Step 4: Configure nodes [See page 12]
- Step 5: Install cluster [See page 17]

## 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 28]
- Using the `tscli cluster create` command [See page 31]
- Parameters of the `cluster create` command [See page 39]
- Deployment Overview [See page 0]
- Contact Support [See page 0]

# 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 \[See page 53\]](#) 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 [\[See page 42\]](#), to determine the ports you need to have open for your cluster.

## Review hardware requirements

Next, [review hardware requirements \[See page 5\]](#).

# 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 \[See page 5\]](#) 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 8]

# 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 \[See page 0\]](#). 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

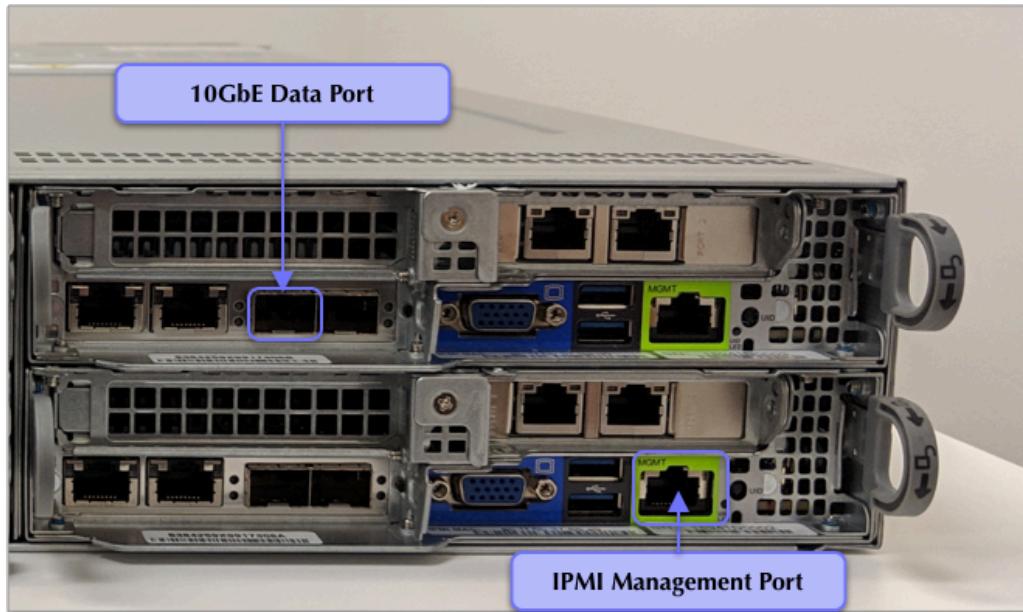
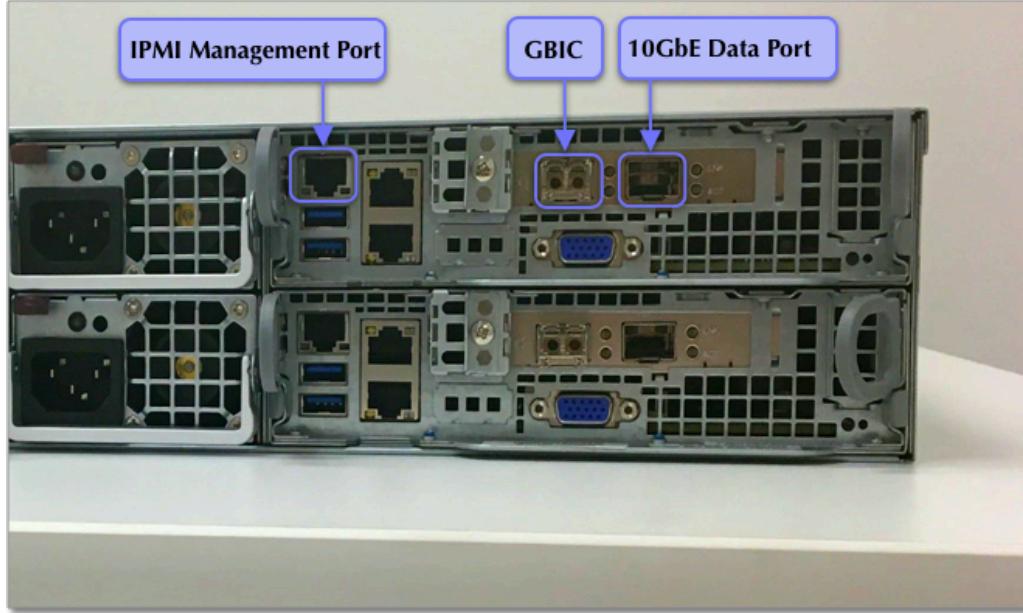
Connect the 10GbE port of each node, as illustrated in [Haswell port location \[See page 9\]](#) and [Skylake port location \[See page 9\]](#), to the 10GbE switches on your own rack, using either fiber or DAC cables.

Refer to the [Cable reference \[See page 53\]](#) for information on the cable types:

- [Fiber Cables \[See page 53\]](#)
- [DAC Cables \[See page 56\]](#)

**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

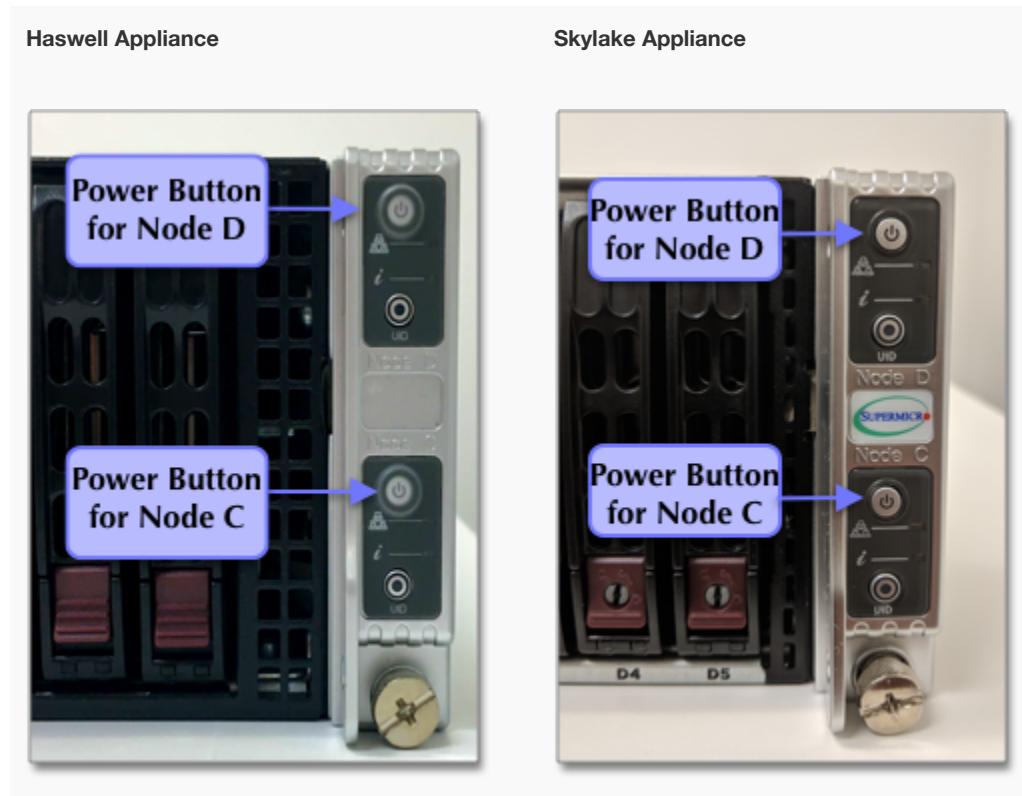
Connect the IPMI port of each node to the management switch.

See [Haswell port location \[See page 9\]](#) and [Skylake port location \[See page 9\]](#).

## Step 3: Turn on nodes

Turn on the power to the nodes by pressing the power button; see [Appliance Power Button \[See page 10\]](#).

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. \[See page 12\]](#)

# 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 \[See page 0\]](#) 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 \[See page 19\]](#) in the next article, your appliance is configured.
4. Skip to [Finalize installation \[See page 21\]](#).

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 \[See page 25\]](#).

```
$ 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 \[See page 27\]](#). 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 \[See page 28\]](#) 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 \[See page 28\]](#). 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 \[See page 23\]](#).

```
$ 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. \[See page 17\]](#)

# 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 \[See page 31\]](#) and [Parameters of the cluster create command \[See page 39\]](#).

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.

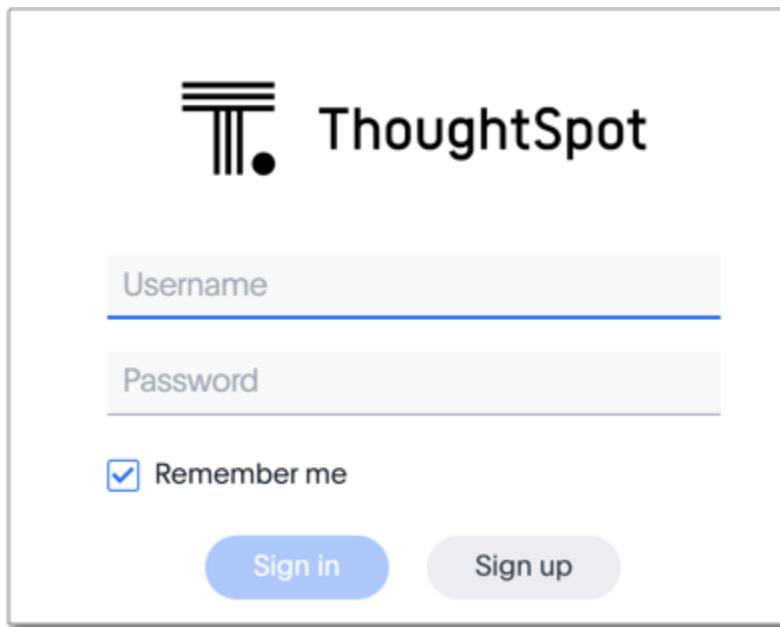
`https://<IP-address>`

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 \[See page 22\]](#), 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 \[See page 0\]](#), with less than 100GB of data, advanced lean configuration is required before loading any data into ThoughtSpot. After installing the cluster, contact [ThoughtSpot Support \[See page 0\]](#) 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 \[See page 28\]](#) 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 \[See page 28\]](#). If you delete quotation marks, commas, or other parts of the code, setup may fail.

See [Parameters of the nodes.config file \[See page 28\]](#) 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 \[See page 2\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 0\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 0\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 0\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 0\]](#)
- [Deploying on the Dell Appliance \[See page 0\]](#)
- [Parameters of the nodes.config file \[See page 28\]](#)

# 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 \[See page 25\]](#) example.

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

```
$ vim nodes.config
```

Fill in the areas specified in [Parameters of nodes.config \[See page 29\]](#) 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 \[See page 29\]](#). If you delete quotation marks, commas, or other parts of the code, setup may fail.

See Parameters of `nodes.config` [See page 29] 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 27] 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 \[See page 2\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 0\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 0\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 0\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 0\]](#)
- [Deploying on the Dell Appliance \[See page 0\]](#)
- [The nodes.config file \[See page 25\]](#)

# 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 \[See page 39\]](#).
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 \[See page 39\]](#). if you delete colons, commas, or other parts of the code, setup may fail.

Refer to [Parameters of the tscli cluster create command \[See page 39\]](#) 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 \[See page 2\]](#)
- [Configure ThoughtSpot nodes in AWS \[See page 0\]](#)
- [Configure ThoughtSpot nodes in GCP \[See page 0\]](#)
- [Configure ThoughtSpot nodes in VMware \[See page 0\]](#)
- [Configure ThoughtSpot nodes in Azure \[See page 0\]](#)
- [Deploying on the Dell Appliance \[See page 0\]](#)
- [Parameters of the tscli cluster create command \[See page 39\]](#)

# 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 \[See page 40\]](#) below.

Refer to [Parameters of ThoughtSpot Installer \[See page 40\]](#) for further information. Refer to [Using the tscli cluster create command \[See page 31\]](#) 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 \[See page 40\]](#). 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 \[See page 0\]](#) 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 2]
- Configure ThoughtSpot nodes in AWS [See page 0]
- Configure ThoughtSpot nodes in GCP [See page 0]
- Configure ThoughtSpot nodes in VMware [See page 0]
- Configure ThoughtSpot nodes in Azure [See page 0]
- Deploying on the Dell Appliance [See page 0]
- Using the tscli cluster create command [See page 31]

# 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, medata-dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts meta-data 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, medata-dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts meta-data service for metadata
7021	RPC	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 meta-data 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, medata-dependency service, scheduling service, session-less service, spotiq service	bidirectional	All nodes	All nodes	Port where search service contacts meta-data 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.

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Dest.</b>	<b>Description</b>
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.

<b>Port</b>	<b>Protocol</b>	<b>Service Name</b>	<b>Direction</b>	<b>Source</b>	<b>Destination</b>	<b>Description</b>
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

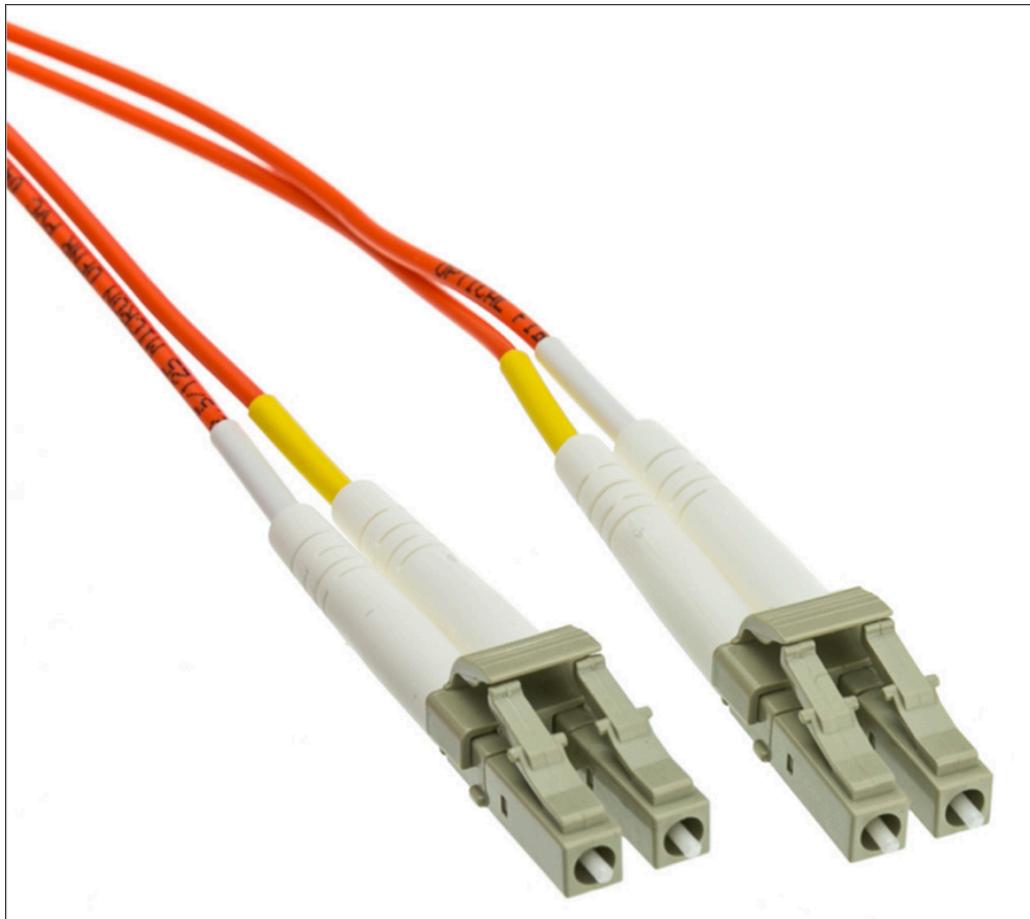
**Summary:** This section reviews the types of cables needed for 10GbE networking and how to plug them in.

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.

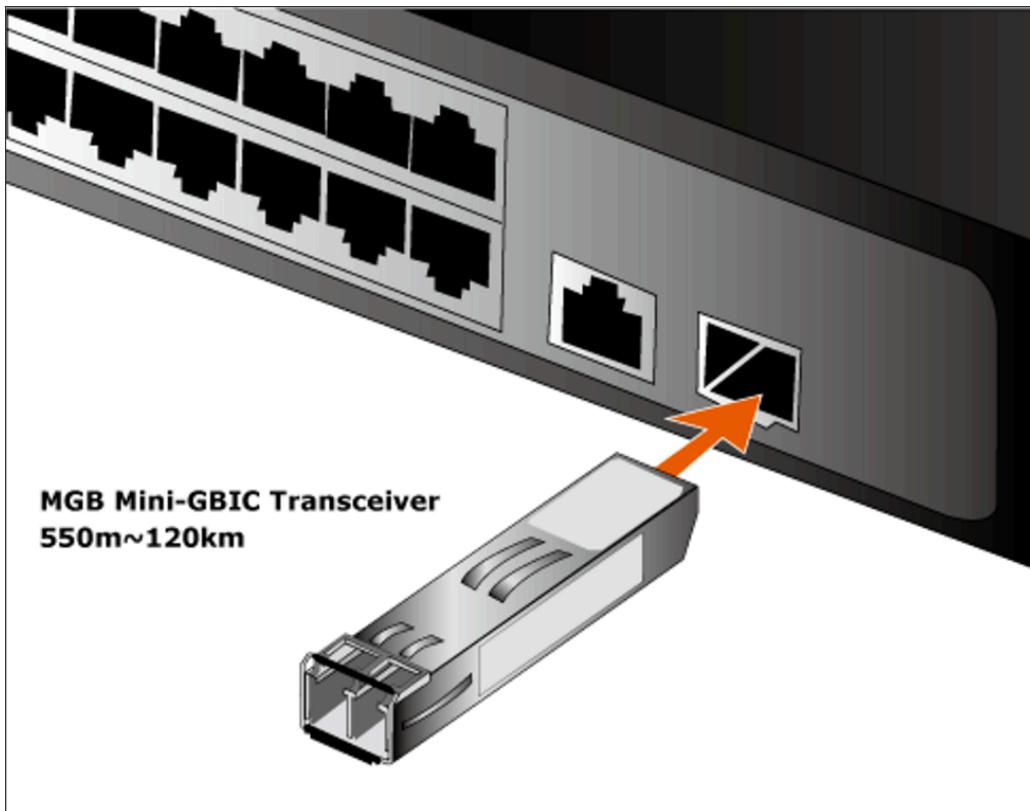


These cables require gigabit interface converters (GBICs), SFP+ form factor.

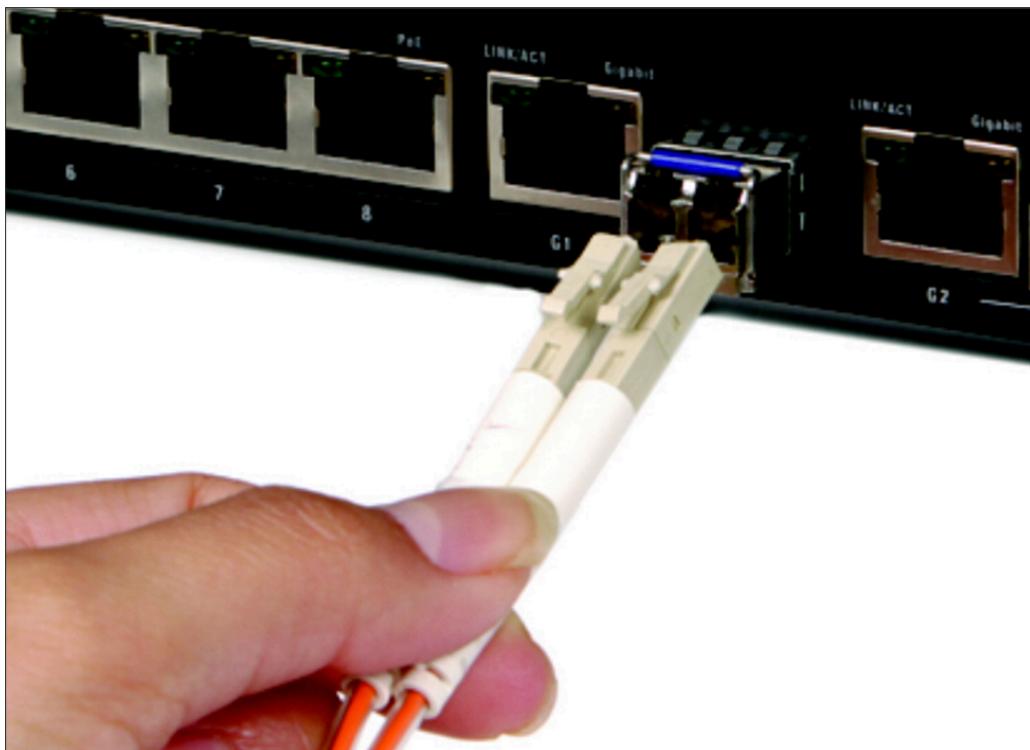
**Remember:** ThoughtSpot does not supply cables or GBICs



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.

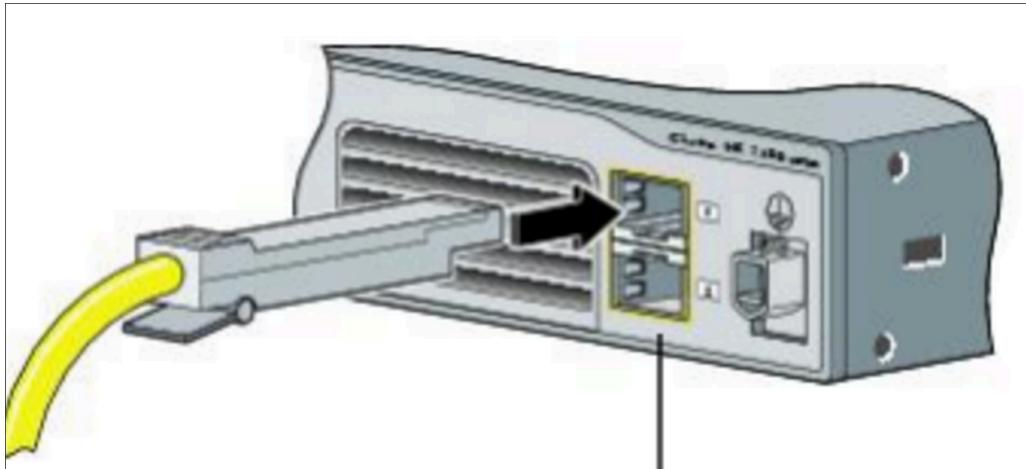


## Option 2 - DAC/Twinax cables

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.

