BOLT

Cluster Deployment Guide

Version 1.10.49

Contents

Copyright Notice

Document Revision History

December 10, 2018

- Initial release

June 10th, 2020

- Bolt OVA release

OVA Download

The latest Bolt OVA file is available as a secure download hosted on Amazon S3.

Your professional services representative will provide you with a secure link to download the file when it becomes available.

Deployment

Preparations

To set up Bolt, you must have:

- Bolt OVA
- Administrative access for the targeted device
- Nginx compatible SSL certificate and SSL certificate key, shared across the cluster or for specific hosts

Deployment

Network

Port Usage

Protocol	Port	Direction	Purpose
HTTPS	443	Inbound	API and file access
HTTP	80	Inbound	API and file access
TCP	4001	Inbound/Outbound	File sharing and publish-subscribe
SSH	22	Inbound	Cluster administration
RTP	32702 (UDP)	Inbound/Outbound	Multicast assist file sharing
RTCP	32703 (UDP)	Inbound/Outbound	Multicast assist file sharing (control)
RTMP	1935	Inbound	Video stream ingest
WebRTC	3478	Inbound	Traversal Using Relay NAT (TURN)

Deployment

System Requirements

Supported Platforms

VMware ESXI 5.5 and later are supported.

Virtual Machine Configuration

The minimum requirements for a Bolt node are:

CPU: 3 GHz dual core or 4 virtual processors
RAM: 8 GB
STORAGE: 80GB

Deploying the OVA

Deploy the OVA on your platform as you would any other OVA. Refer to your platform's documentation for instructions on deploying OVA files.

Cluster Setup

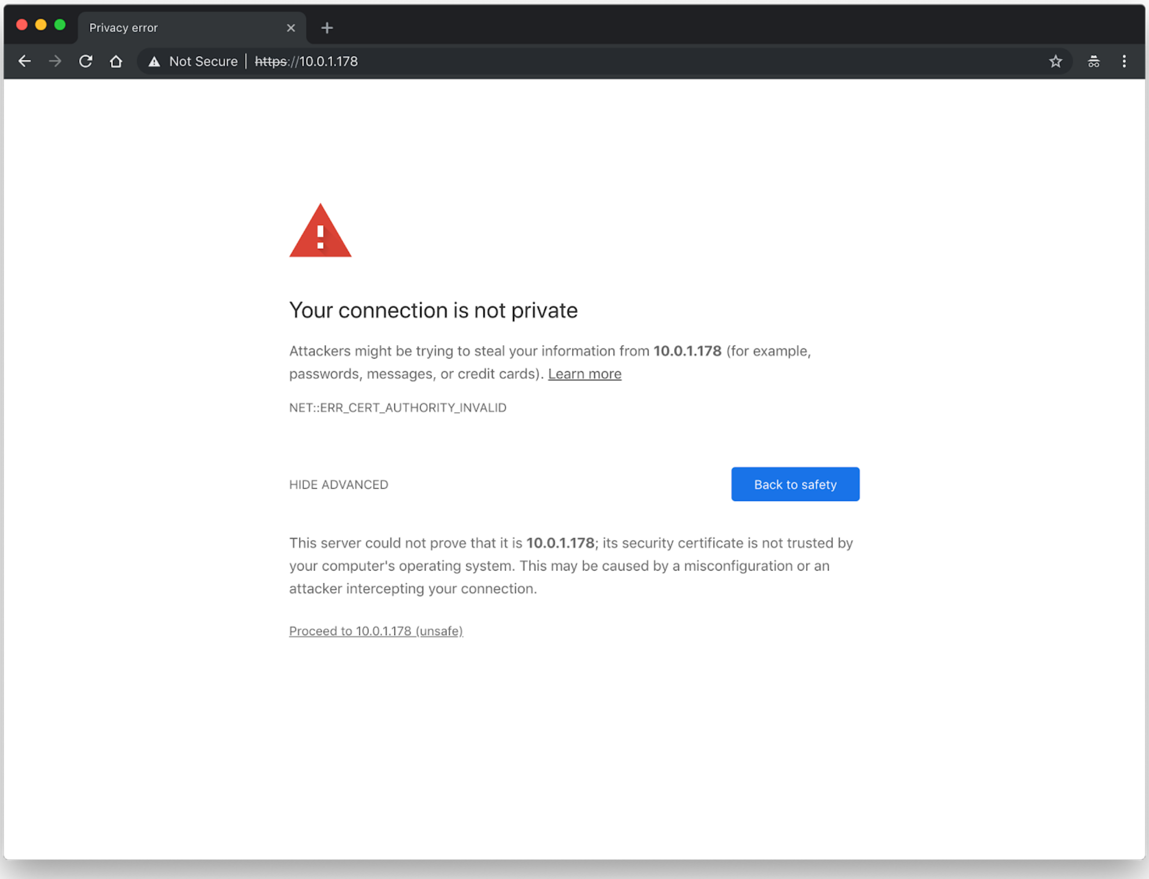
Clusters are headless and all nodes are functionally identical.

SSL Certificates

The SSL certificate and certificate key should be Nginx compatible. See - http://nginx.org/en/docs/http/configuring_https_servers.html - for more information.

Initialize Cluster

Visit the https://IP_ADDRESS path of the first node. If the node IP were **10.0.1.178,** the address would be **https://10.0.1.178/**. Proceed through the SSL certificate warnings.



Your connection is not private

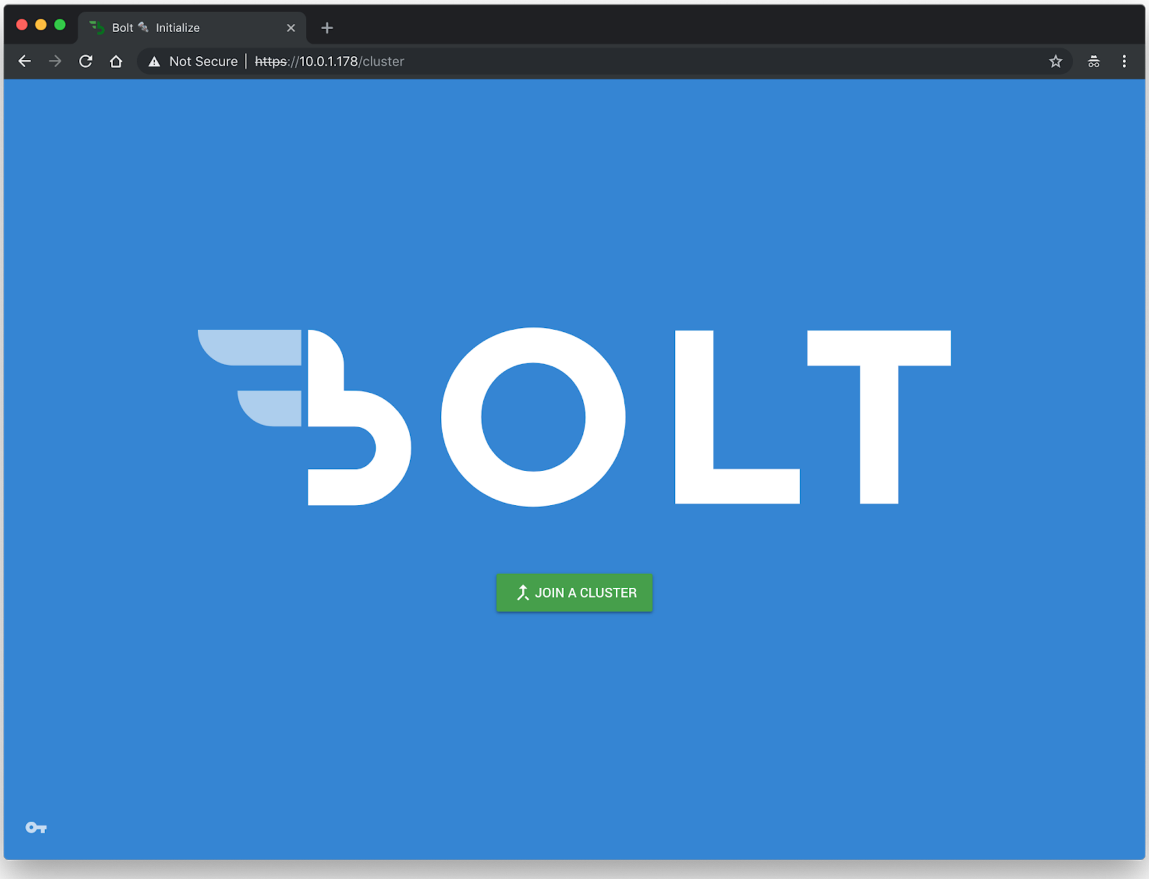
Attackers might be trying to steal your information from **10.0.1.178** (for example, passwords, messages, or credit cards). Learn more

NET::ERR_CERT_AUTHORITY_INVALID

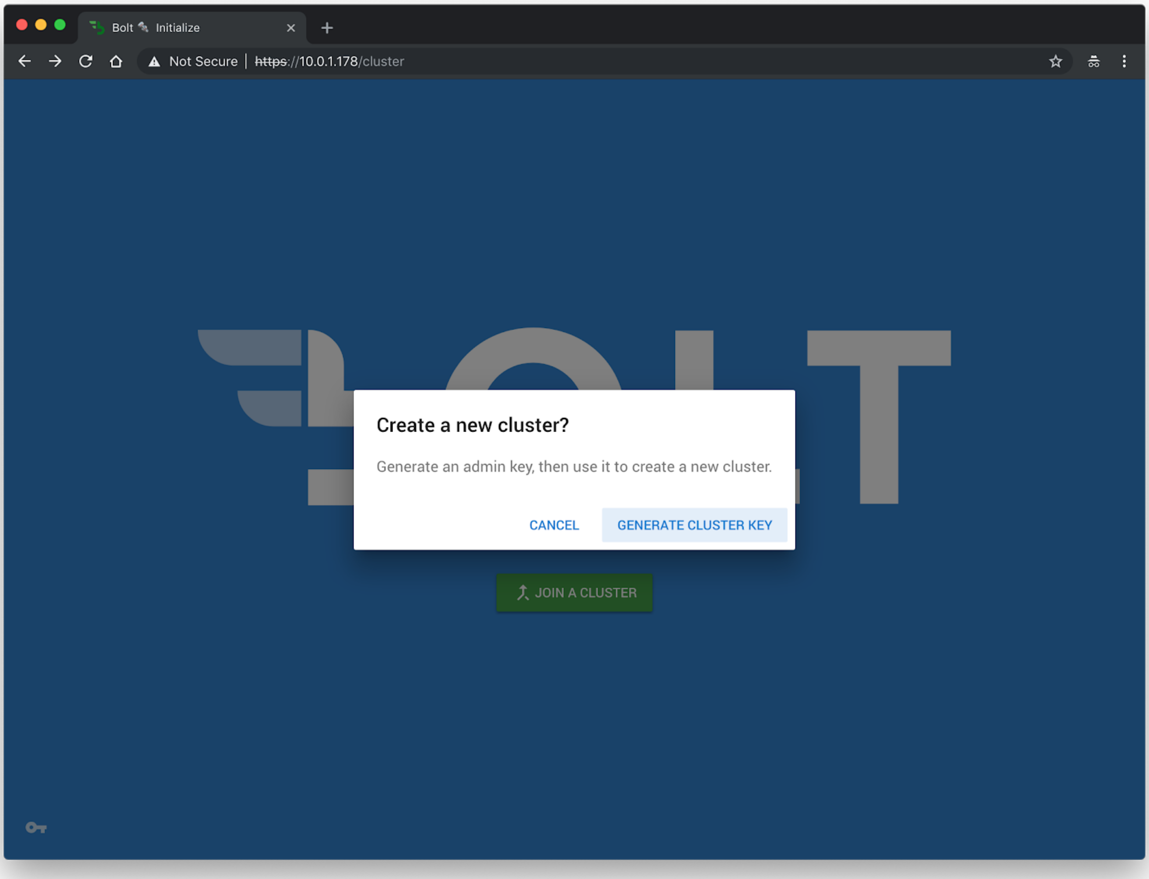
This server could not prove that it is **10.0.1.178**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

Proceed to 10.0.1.178 (unsafe)

Click on the **Key icon** on the bottom left corner of the page to open the dialog to create a new cluster.



Click on **Generate Cluster Key** button to initialize the cluster and generate the cluster administration key.



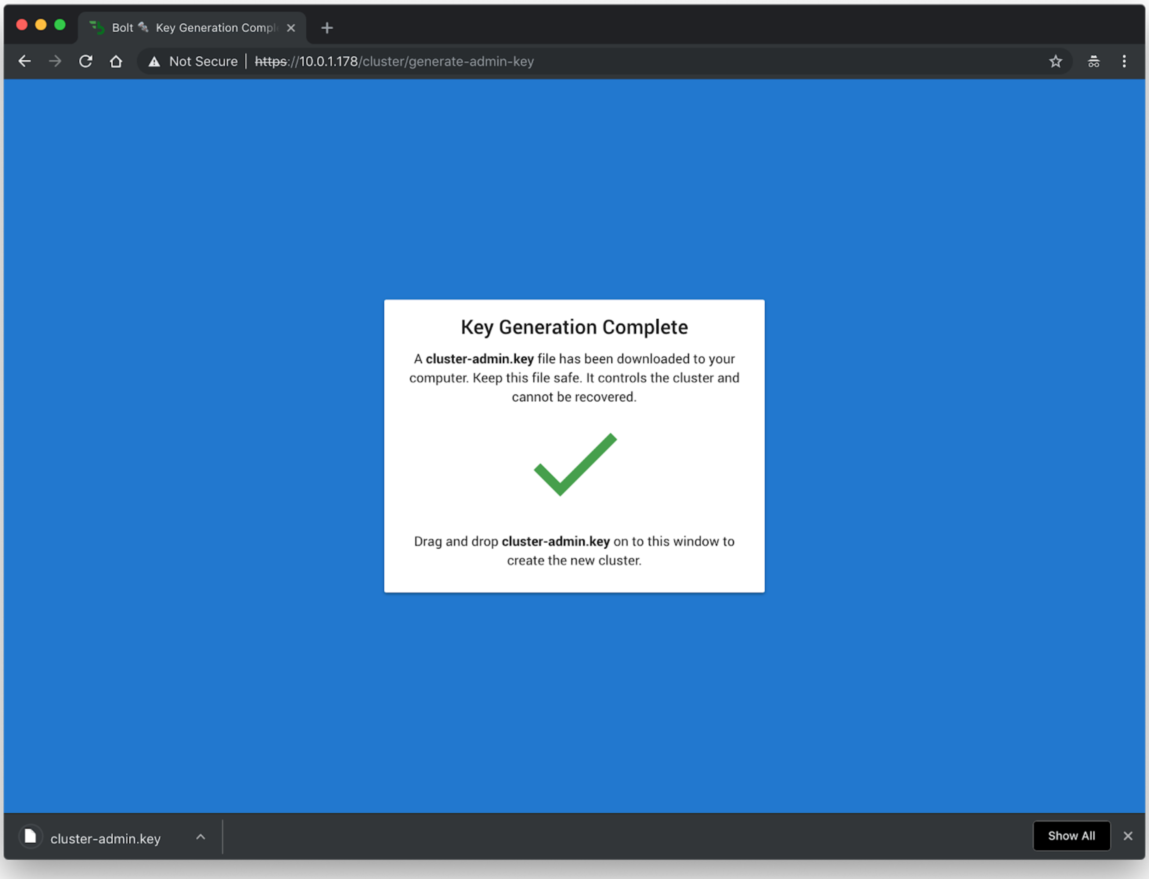
Create a new cluster?

Generate an admin key, then use it to create a new cluster.

The cluster administration key will be downloaded as **cluster-admin.key** after the key generation process is complete.

Cluster administration key acts as the authorization mechanism for cluster management. Anyone who submits the key when requested will be treated as a cluster administrator and is allowed full access to bolt cluster.

NOTE Cluster Administration key **cannot** be recovered at any point, please back it up to safe location.

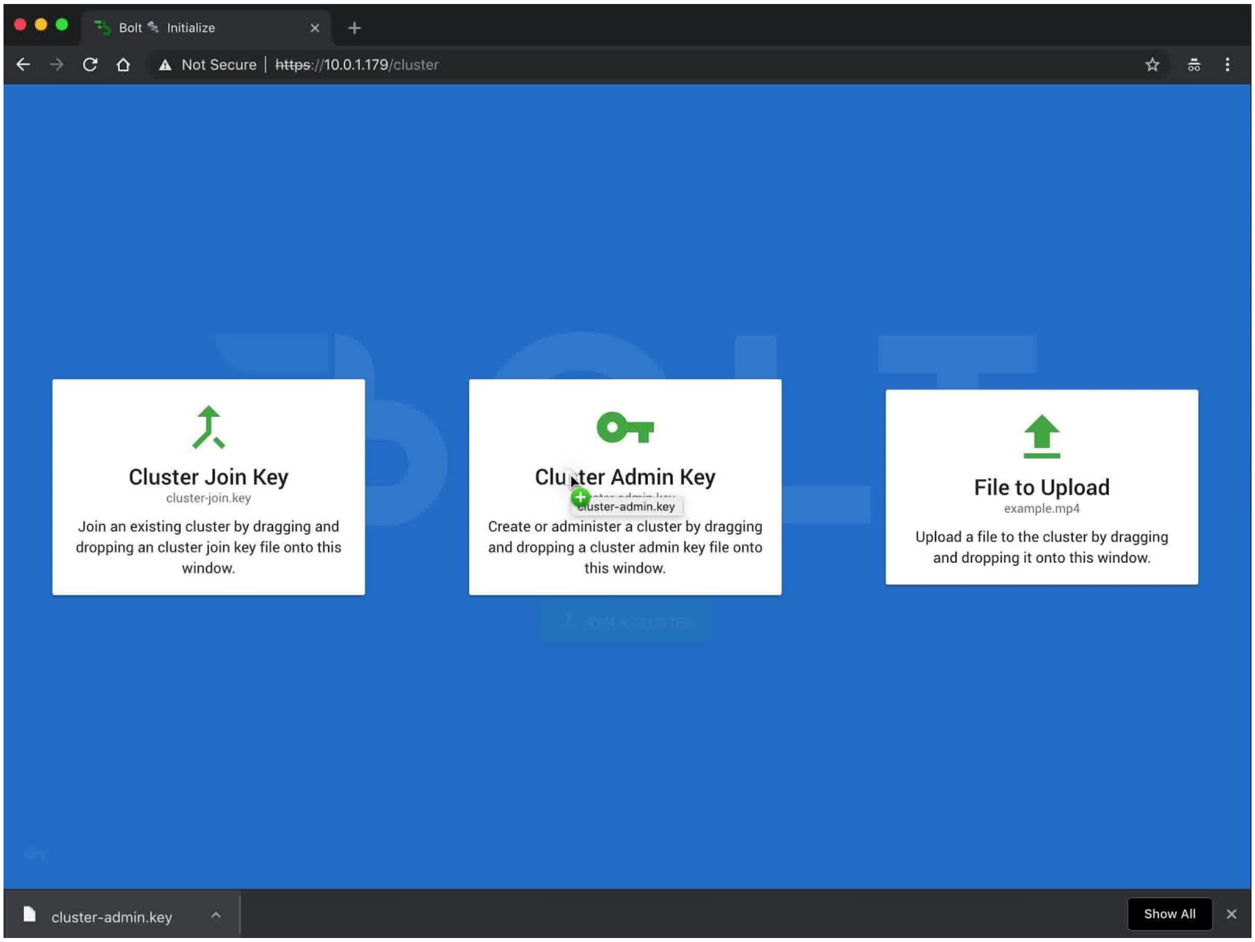

Bolt Key Generation Compl
Not Secure
~~https~~://10.0.1.178/cluster/generate-admin-key
cluster-admin.key
Show All

Key Generation Complete

A **cluster-admin.key** file has been downloaded to your computer. Keep this file safe. It controls the cluster and cannot be recovered.

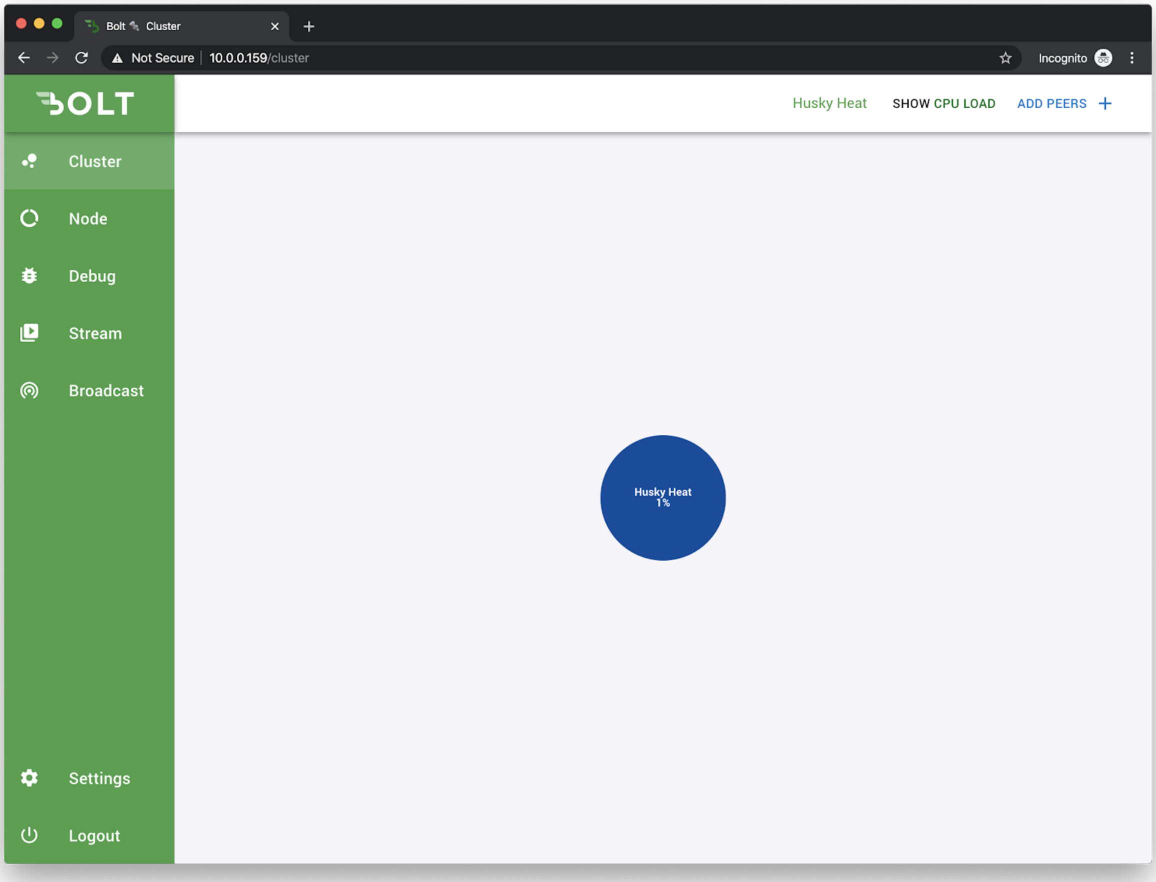
Drag and drop **cluster-admin.key** on to this window to create the new cluster.

Drag and drop the downloaded **cluster-admin.key** file on to **Cluster Admin Key** box located at the center of the page.

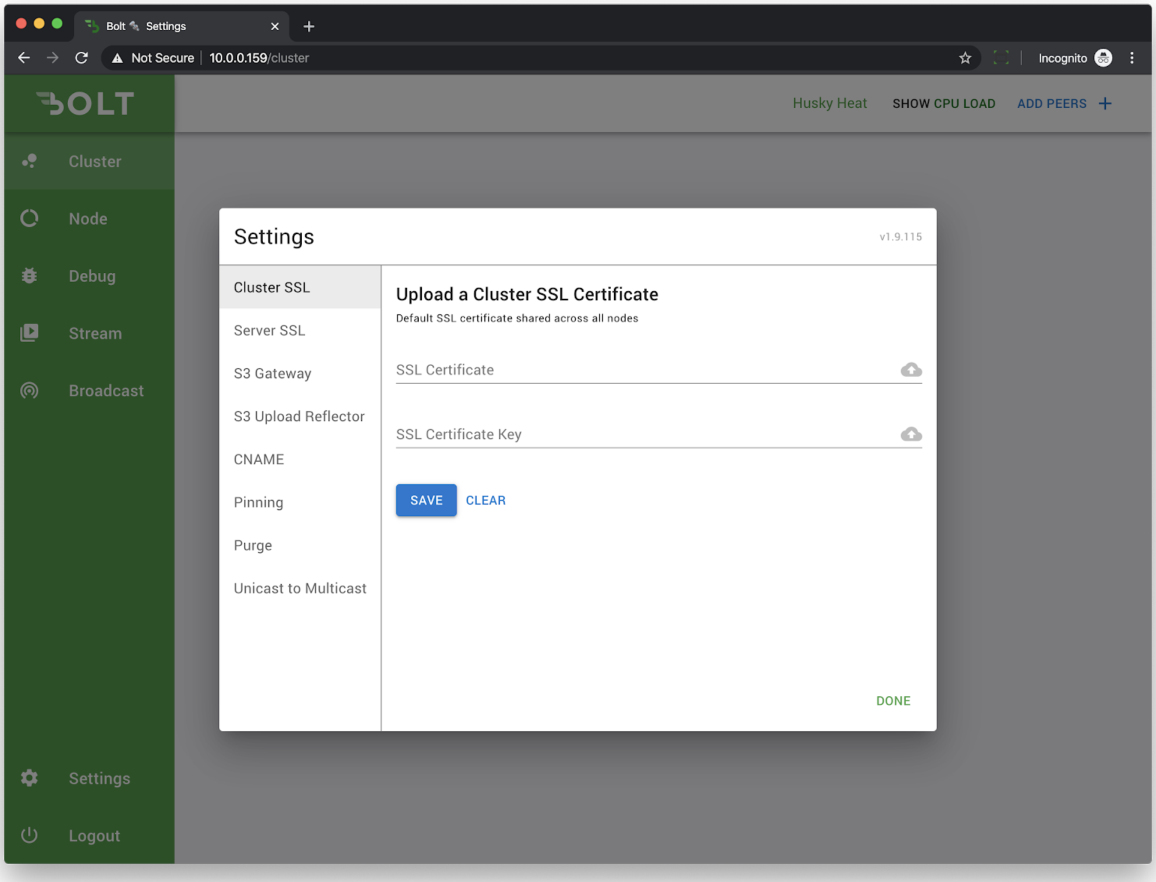


Bolt Initialize
Not Secure | ~~https~~://10.0.1.179/cluster
Cluster Join Key
cluster-join.key
Join an existing cluster by dragging and dropping an cluster join key file onto this window.
Cluster Admin Key
cluster-admin.key
Create or administer a cluster by dragging and dropping a cluster admin key file onto this window.
File to Upload
example.mp4
Upload a file to the cluster by dragging and dropping it onto this window.
cluster-admin.key
Show All

Once the key is successfully authenticated, the user is assigned admin privileges and redirected to the cluster dashboard.



Click on the **Settings** button on the sidebar to access the cluster settings page.

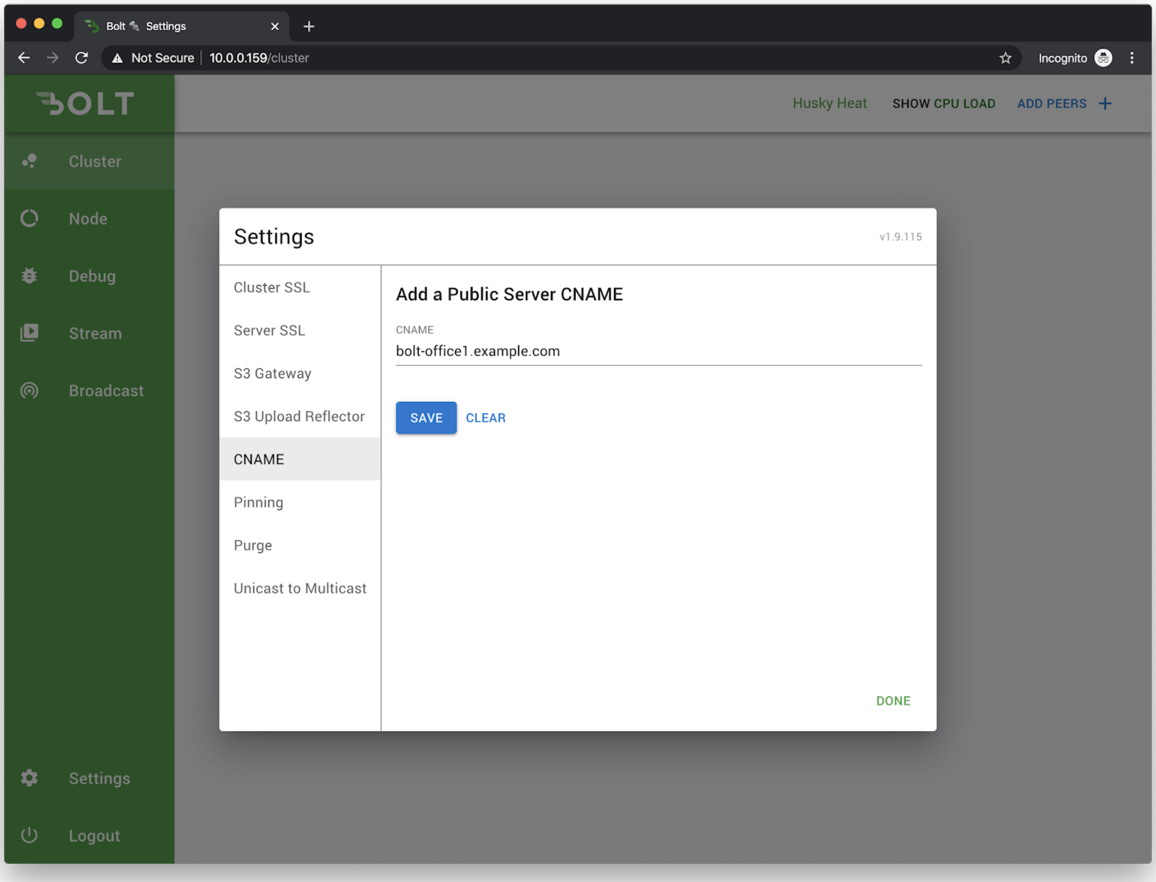


Upload a Cluster SSL Certificate

Default SSL certificate shared across all nodes

Setup Server Name

Enter a FQDN name that matched with the SSL certificates and click on the **Save** button. For example, if the SSL certificates are generated and valid for **bolt-office1.example.com** then the server name would be **bolt-office1.example.com.**



Add a Public Server CNAME

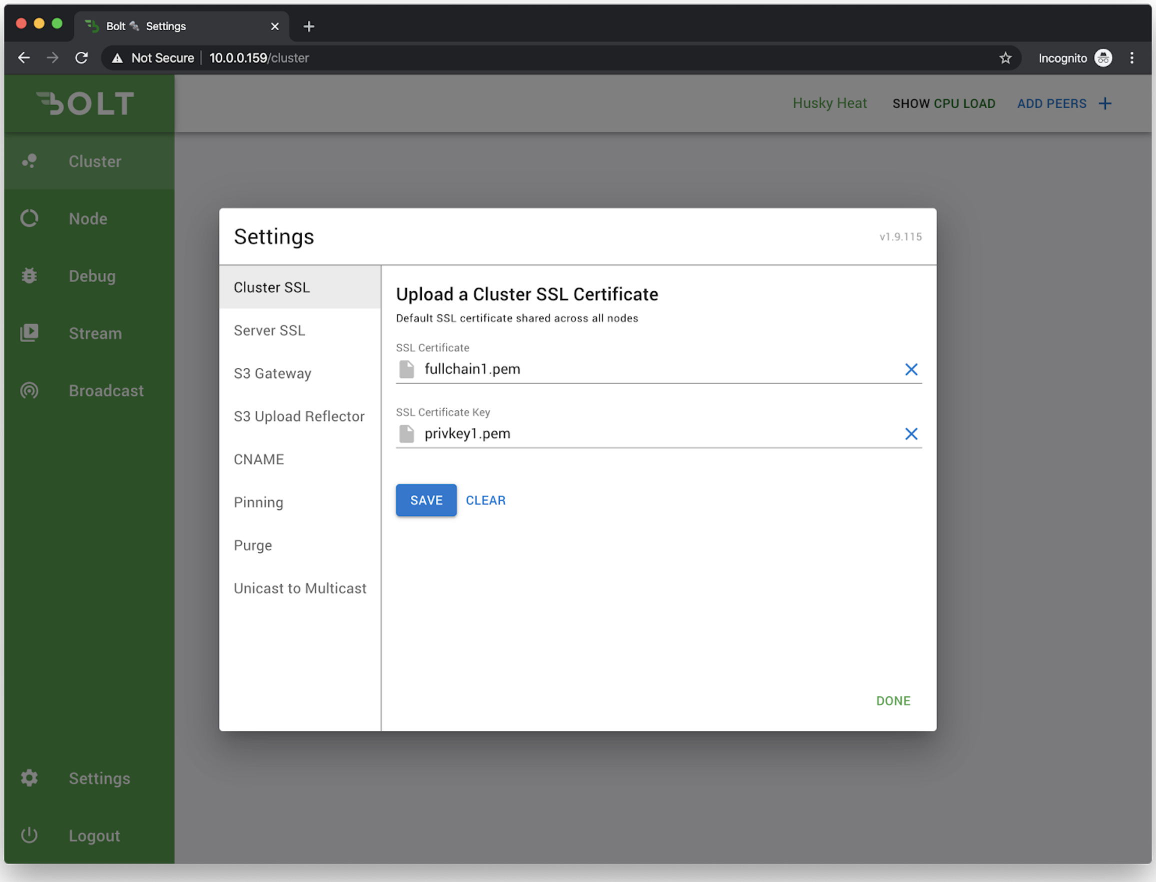
CNAME: bolt-office1.example.com

Setup SSL Certificate

Cluster SSL

Configure SSL certificate that's **shared between all the nodes** in the cluster by uploading SSL certificate and private key in PEM[1] encoded format on to the respective fields and click on **Save** button.

The Common Name of the SSL certificate becomes the hostname of the cluster.



Upload a Cluster SSL Certificate

Default SSL certificate shared across all nodes

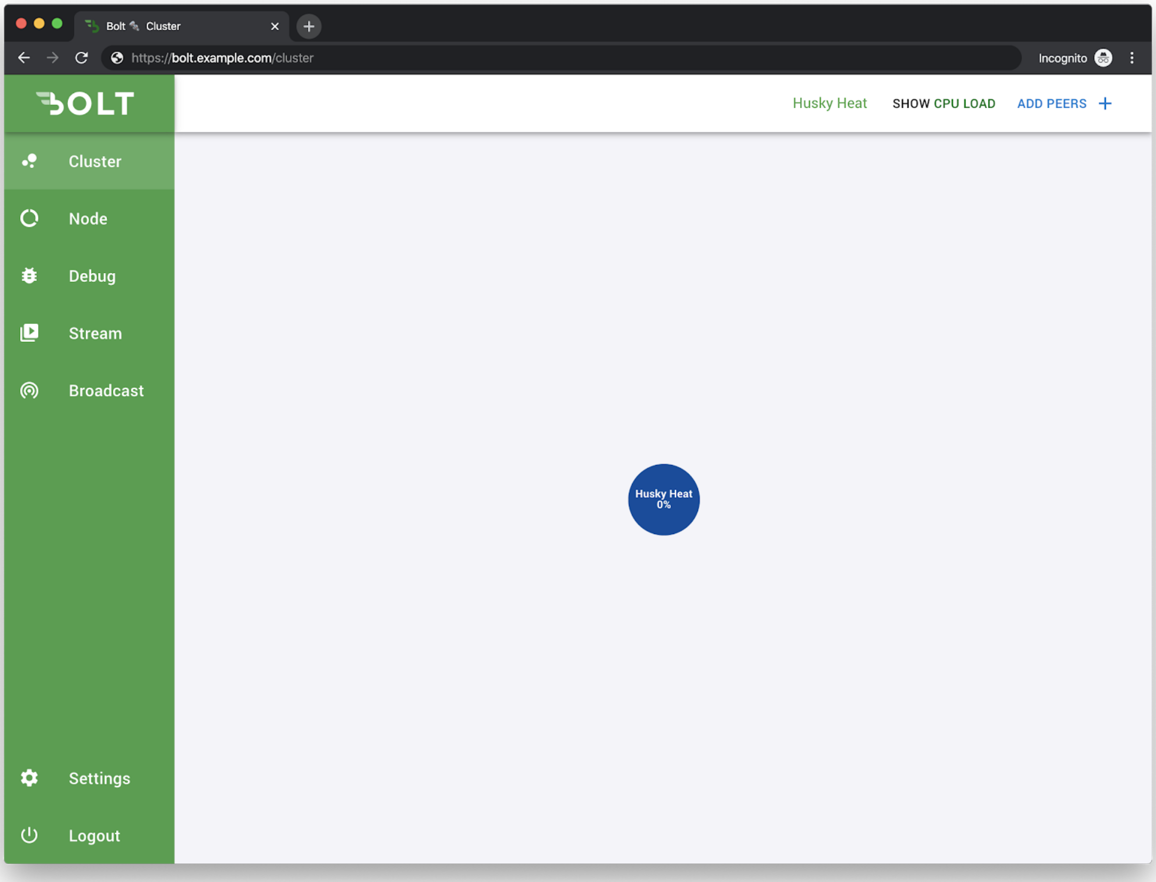
SSL Certificate: fullchain1.pem

SSL Certificate Key: privkey1.pem

[1] http://nginx.org/en/docs/http/ngx_http_ssl_module.html#ssl_certificate

After successfully updating the Cluster SSL certificates, the page will be redirected to the common name specified in the SSL certificates.

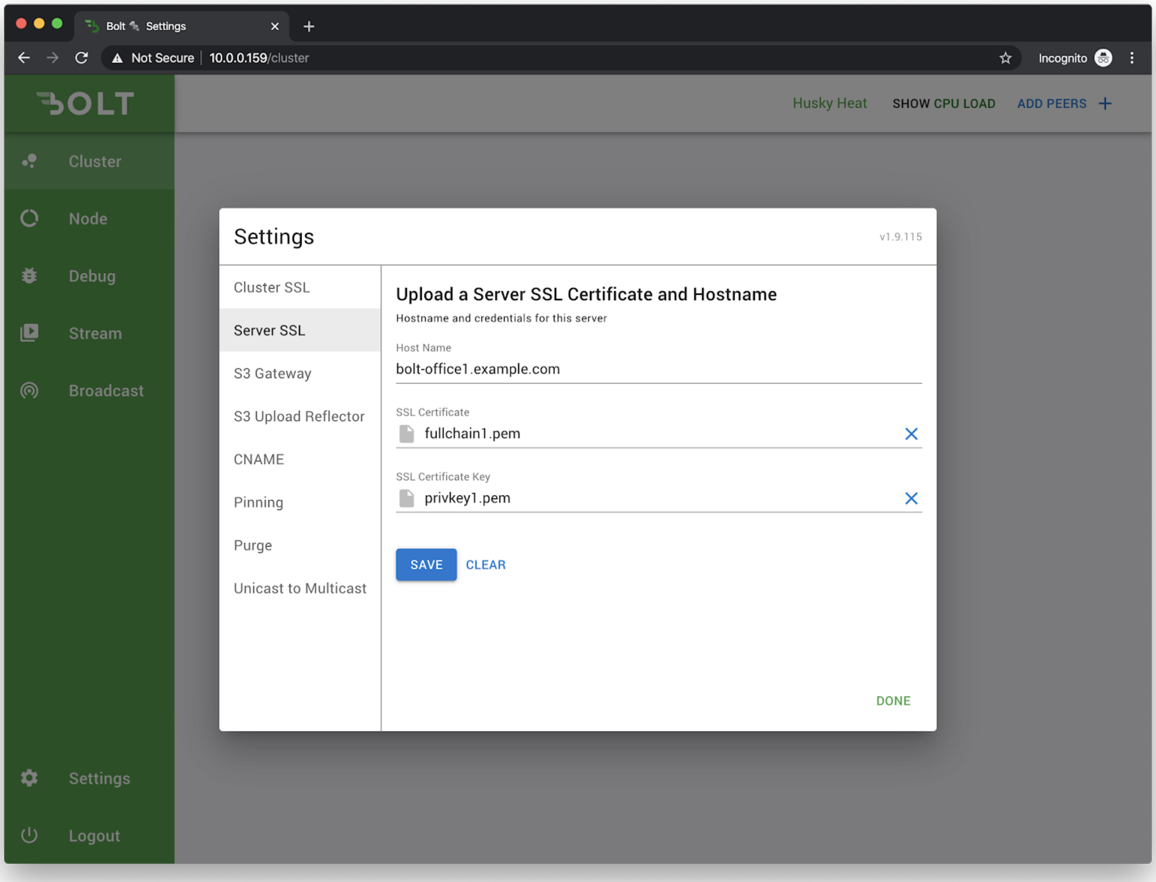
For example, If the SSL certificates are generated for **bolt.example.com**, then the cluster admin would be redirected to **https://bolt.example.com/cluster**.



Server SSL

Configure the hostname and SSL certificate for a **specific node** in the cluster by entering the hostname and uploading SSL certificate and private key in PEM[2] format to the respective fields and click on the Save button.

For example, If the SSL certificates are generated for the hostname **bolt-office1.bunchdev.com,**



Settings

v1.9.115

Upload a Server SSL Certificate and Hostname

Hostname and credentials for this server

Host Name: bolt-office1.example.com

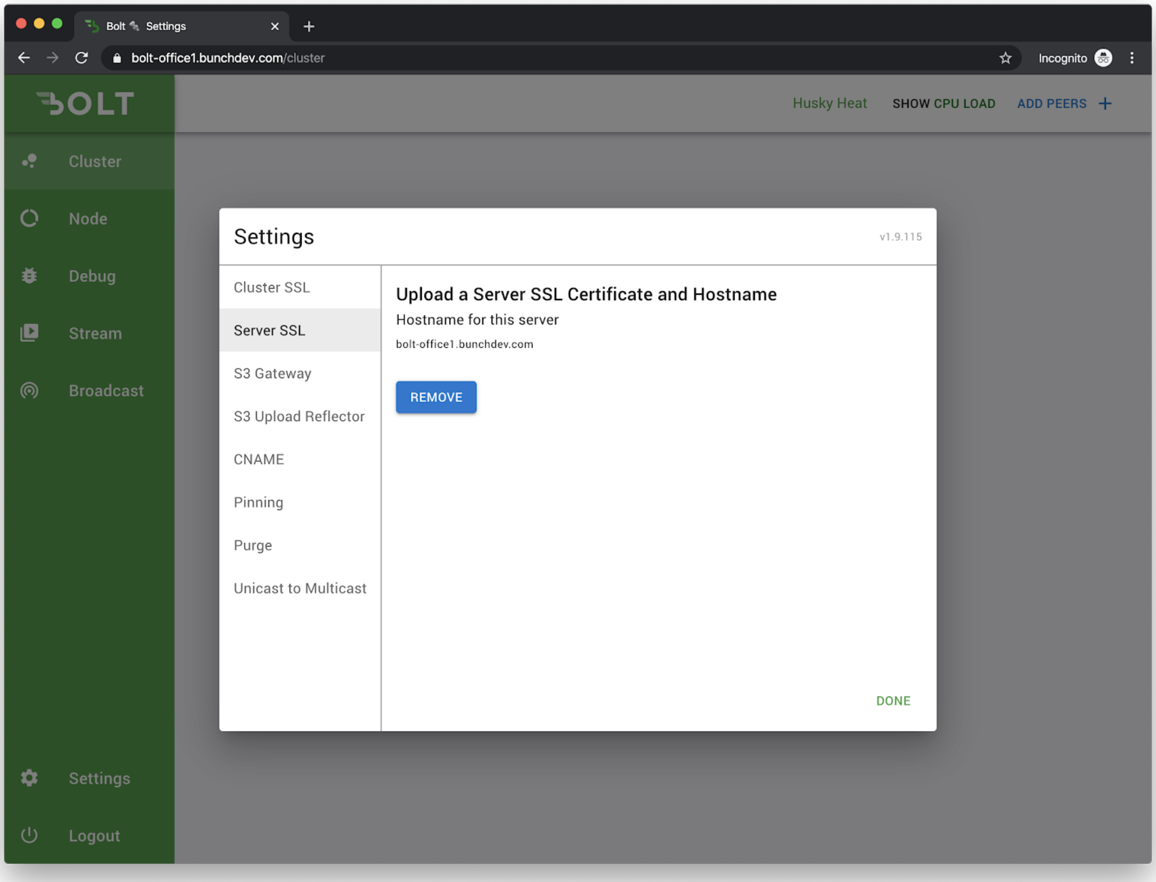
SSL Certificate: fullchain1.pem

SSL Certificate Key: privkey1.pem

[2] http://nginx.org/en/docs/http/ngx_http_ssl_module.html#ssl_certificate

After successfully updating the Server SSL certificates, the Server SSL certificates would take priority over Cluster SSL certificates for the individual Bolt server.

Removing the individual Server SSL certificates would allow the Bolt server to use the Cluster SSL Certificates.



Upload a Server SSL Certificate and Hostname

Hostname for this server

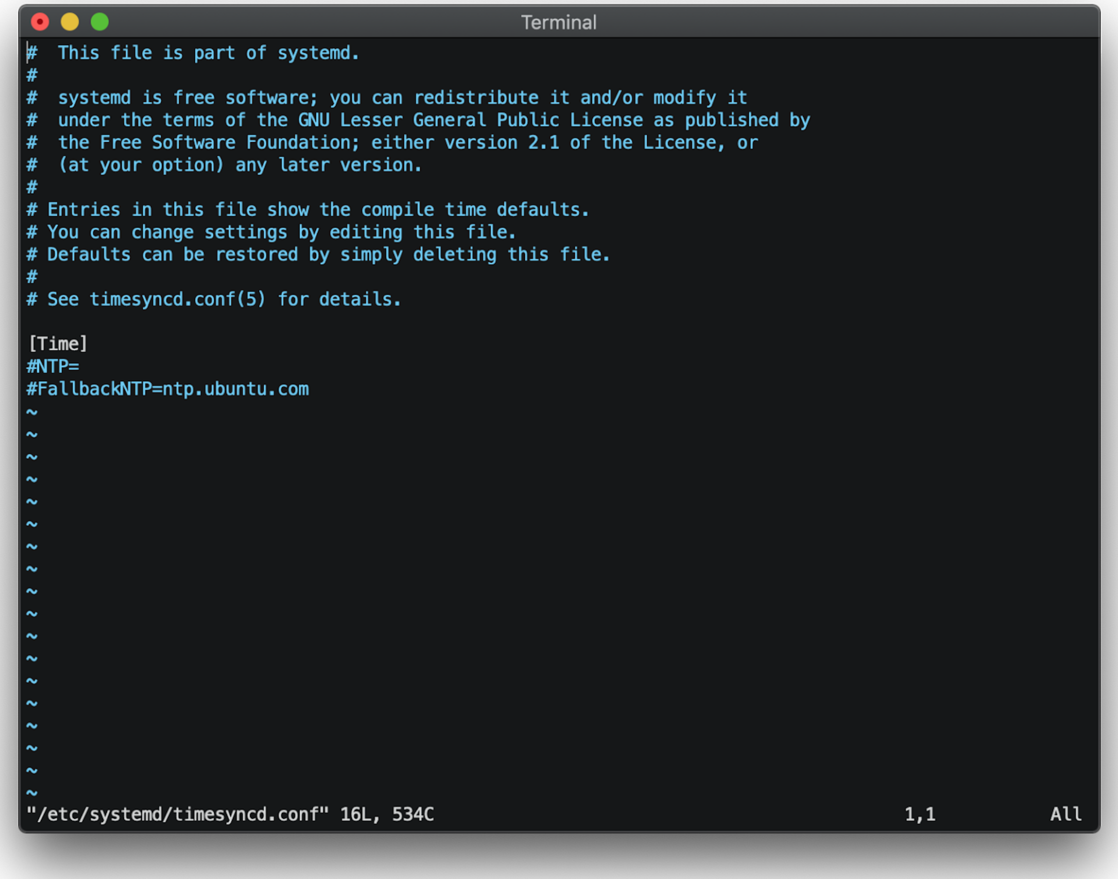
bolt-office1.bunchdev.com

Set up Custom Timeserver (Optional)

Ubuntu 16.04 by default uses the time server at **ntp.ubuntu.com.**

To change the default server, paste the following command in the terminal to edit the configuration file in the VIM editor:

```
sudo vim /etc/systemd/timesyncd.conf
```



```
#  This file is part of systemd.
#
#  systemd is free software; you can redistribute it and/or modify it
#  under the terms of the GNU Lesser General Public License as published by
#  the Free Software Foundation; either version 2.1 of the License, or
#  (at your option) any later version.
#
# Entries in this file show the compile time defaults.
# You can change settings by editing this file.
# Defaults can be restored by simply deleting this file.
#
# See timesyncd.conf(5) for details.

[Time]
#NTP=
#FallbackNTP=ntp.ubuntu.com
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"/etc/systemd/timesyncd.conf" 16L, 534C                    1,1           All
```

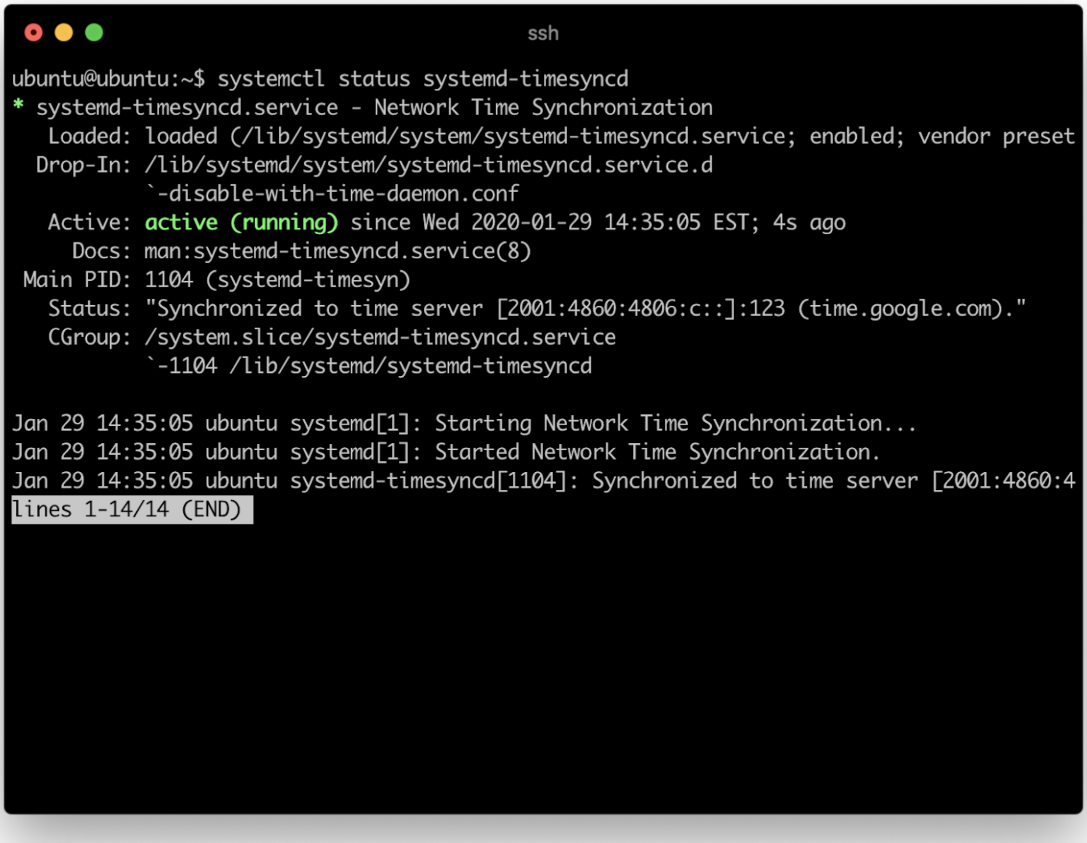
Uncomment the NTP line by removing the hash and enter the desired time server address. For example, if the NTP time server was **time.google.com**, the entry would be **NTP=time.google.com**

Quit the editor by hitting **ESC**, and type **:wq** to save and exit. Hit **Enter**. Restart the time server by running the following command in the terminal:

```
systemctl restart systemd-timesyncd
```

To check the status of the timeserver, run the following command in the terminal:

```
systemctl status systemd-timesyncd
```

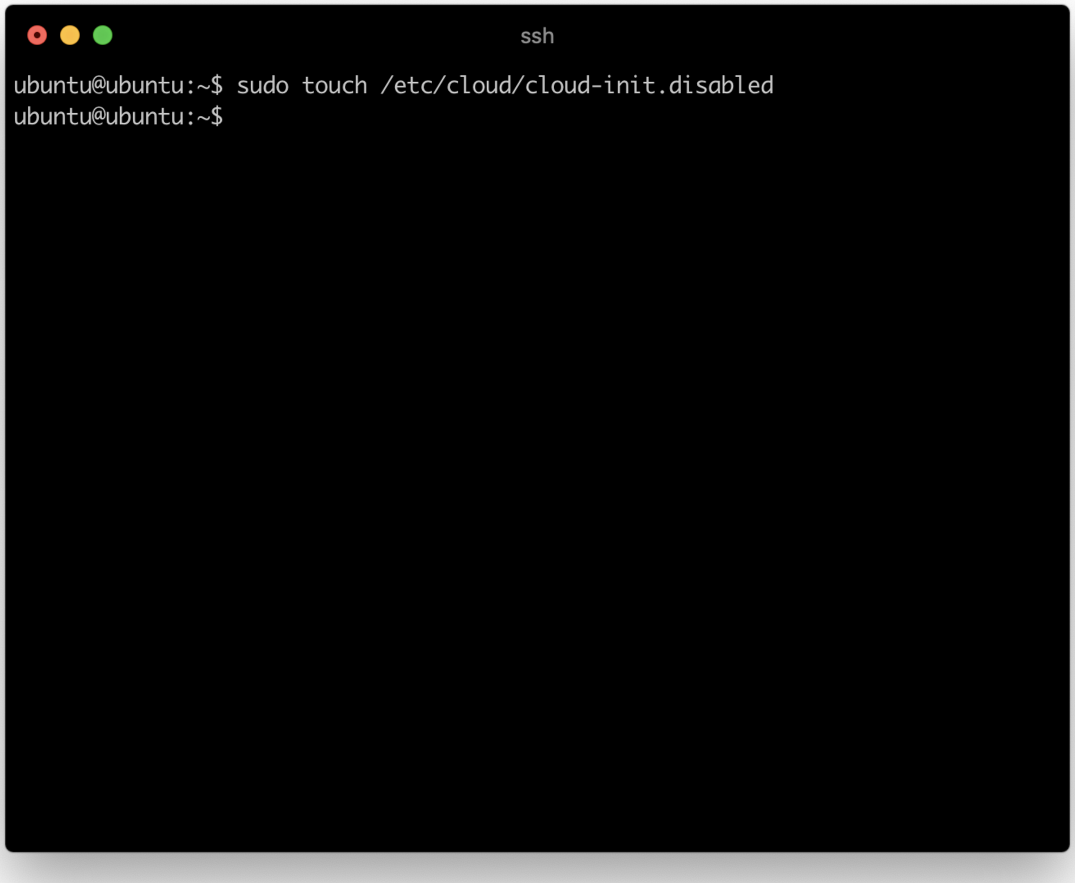


```
ubuntu@ubuntu:~$ systemctl status systemd-timesyncd
* systemd-timesyncd.service - Network Time Synchronization
   Loaded: loaded (/lib/systemd/system/systemd-timesyncd.service; enabled; vendor preset
  Drop-In: /lib/systemd/system/systemd-timesyncd.service.d
           `-disable-with-time-daemon.conf
   Active: active (running) since Wed 2020-01-29 14:35:05 EST; 4s ago
     Docs: man:systemd-timesyncd.service(8)
 Main PID: 1104 (systemd-timesyn)
   Status: "Synchronized to time server [2001:4860:4806:c::]:123 (time.google.com)."
   CGroup: /system.slice/systemd-timesyncd.service
           `-1104 /lib/systemd/systemd-timesyncd

Jan 29 14:35:05 ubuntu systemd[1]: Starting Network Time Synchronization...
Jan 29 14:35:05 ubuntu systemd[1]: Started Network Time Synchronization.
Jan 29 14:35:05 ubuntu systemd-timesyncd[1104]: Synchronized to time server [2001:4860:4
lines 1-14/14 (END)
```

Disable Cloud-Init (Optional)

Cloud-Init is a service that initializes the servers running in cloud platforms. However it's not required when running Bolt server on-premise. Disable the cloud-init service by creating an empty file at `/etc/cloud/cloud-init.disabled.`

```
sudo touch /etc/cloud/cloud-init.disabled
```



```
ubuntu@ubuntu:~$ sudo touch /etc/cloud/cloud-init.disabled
ubuntu@ubuntu:~$
```

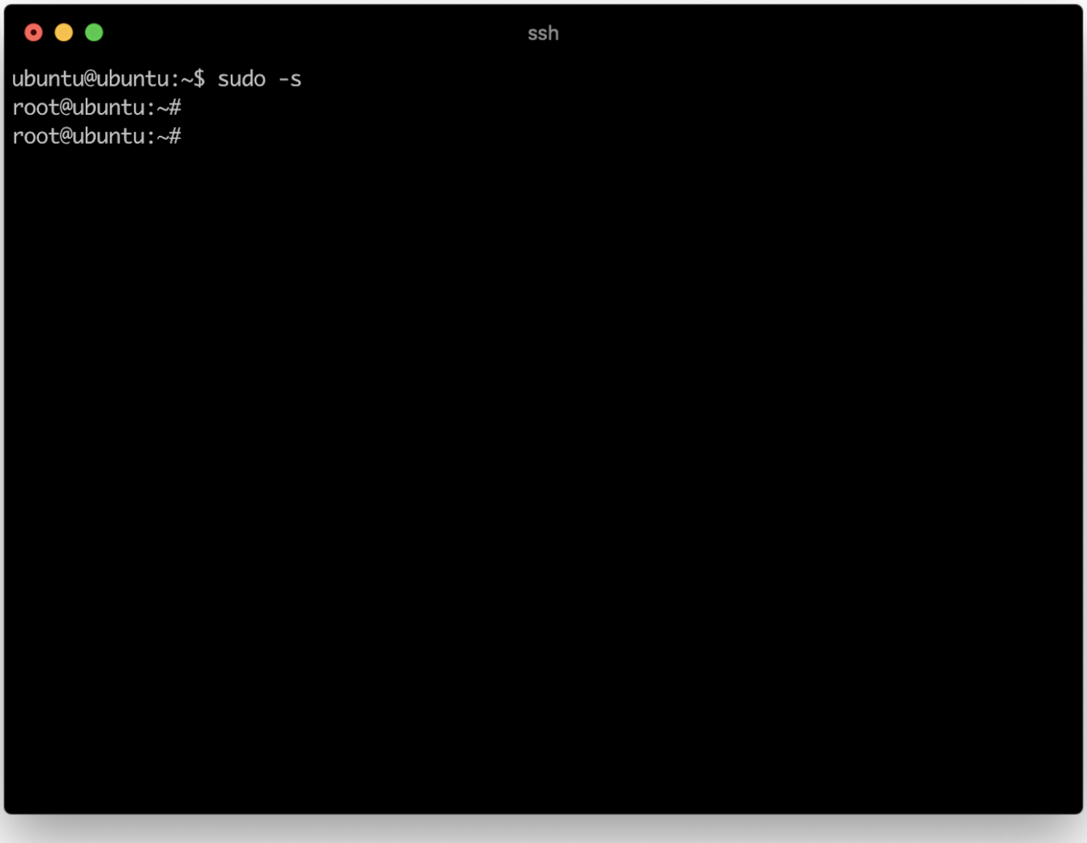
Set up Static IP Address (Optional)

Bolt uses DHCP to acquire an IP Address from the network. Run through the following steps to set up a Static IP address on the Bolt server.

Access the Bolt virtual terminal and login with the following credentials
Username: ubuntu
Password: ubuntu

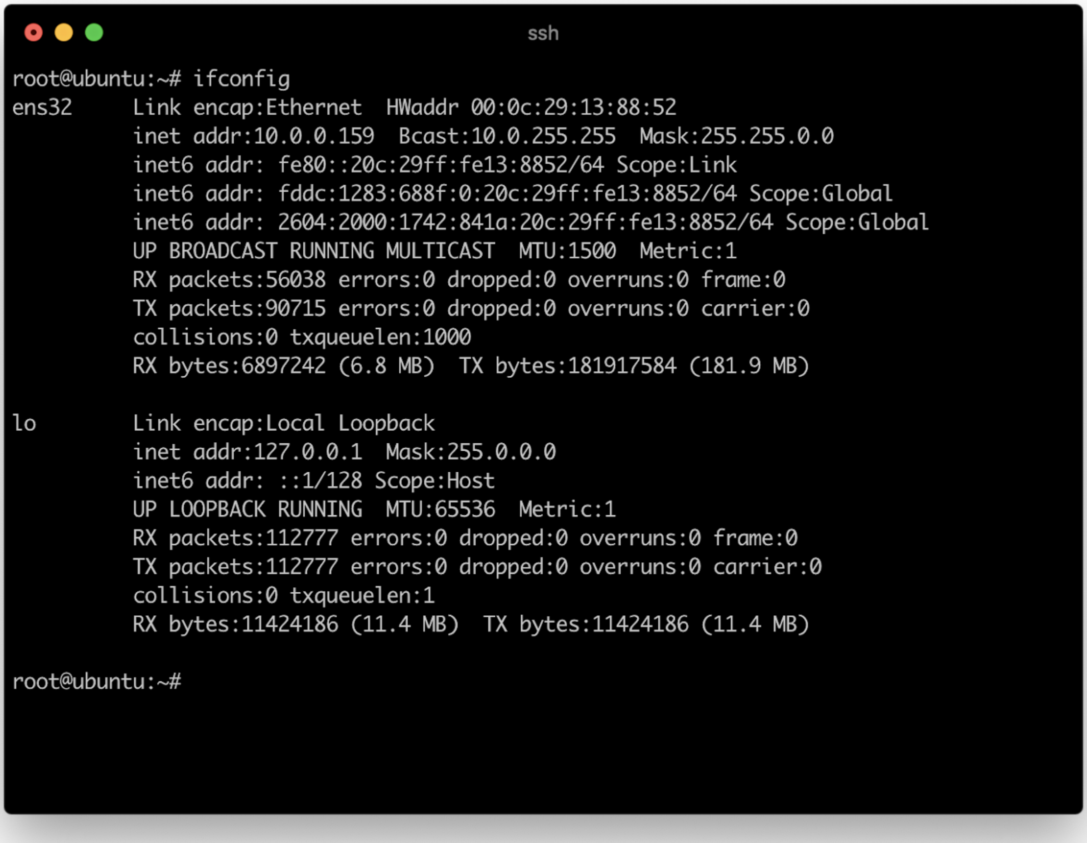
Switch to root user by executing

```
sudo -s
```



```
ubuntu@ubuntu:~$ sudo -s
root@ubuntu:~#
root@ubuntu:~#
```

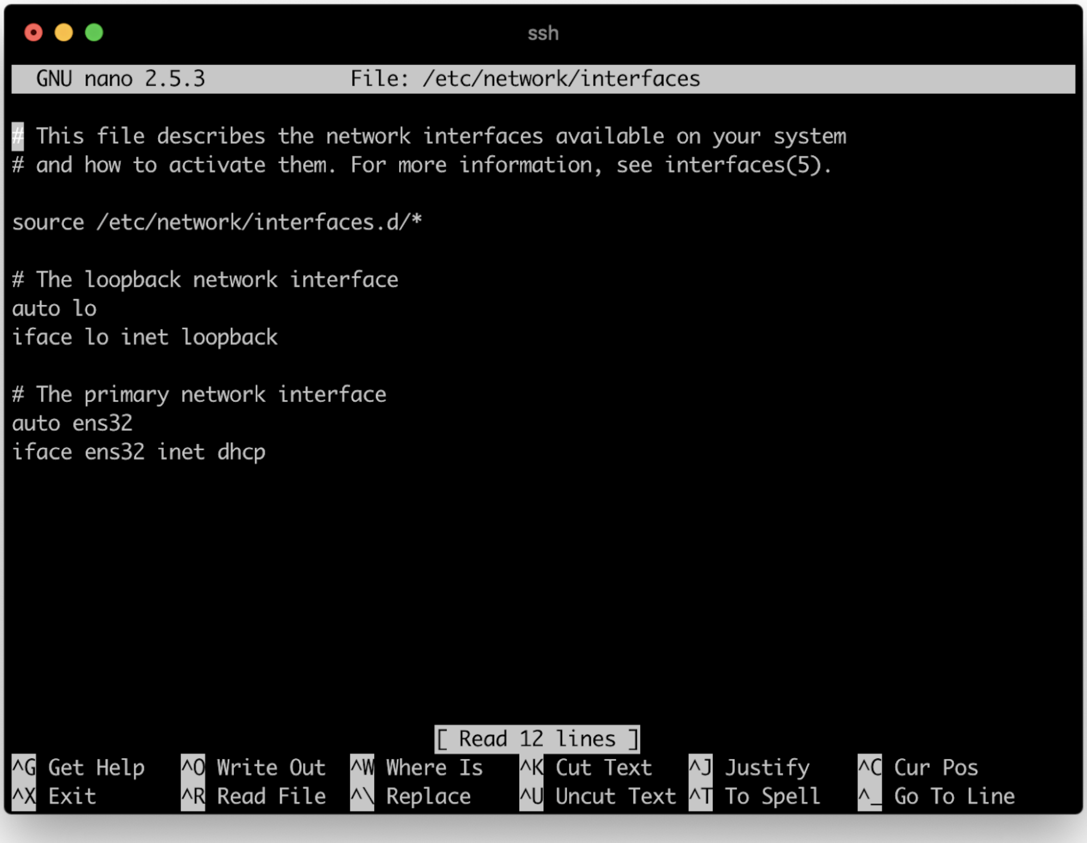
Identify the primary network interface by executing **`ifconfig`**. In this case, the primary network interface is **ens32**.



```
root@ubuntu:~# ifconfig
ens32     Link encap:Ethernet  HWaddr 00:0c:29:13:88:52
          inet addr:10.0.0.159  Bcast:10.0.255.255  Mask:255.255.0.0
          inet6 addr: fe80::20c:29ff:fe13:8852/64 Scope:Link
          inet6 addr: fddc:1283:688f:0:20c:29ff:fe13:8852/64 Scope:Global
          inet6 addr: 2604:2000:1742:841a:20c:29ff:fe13:8852/64 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:56038 errors:0 dropped:0 overruns:0 frame:0
          TX packets:90715 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:6897242 (6.8 MB)  TX bytes:181917584 (181.9 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:112777 errors:0 dropped:0 overruns:0 frame:0
          TX packets:112777 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:11424186 (11.4 MB)  TX bytes:11424186 (11.4 MB)

root@ubuntu:~#
```

Open the network configuration file as a root user to setup a static network entry for the primary network interface.

```
sudo nano /etc/network/interfaces
```



```
  GNU nano 2.5.3              File: /etc/network/interfaces

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens32
iface ens32 inet dhcp









                                   [ Read 12 lines ]
^G Get Help     ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos
^X Exit         ^R Read File    ^\ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line
```

If the primary network interface has a different name, the interfaces file might look different. For example, If the primary network interface is **ens32**, the **/etc/network/interfaces** file looks like,

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens32
iface ens32 inet dhcp
```

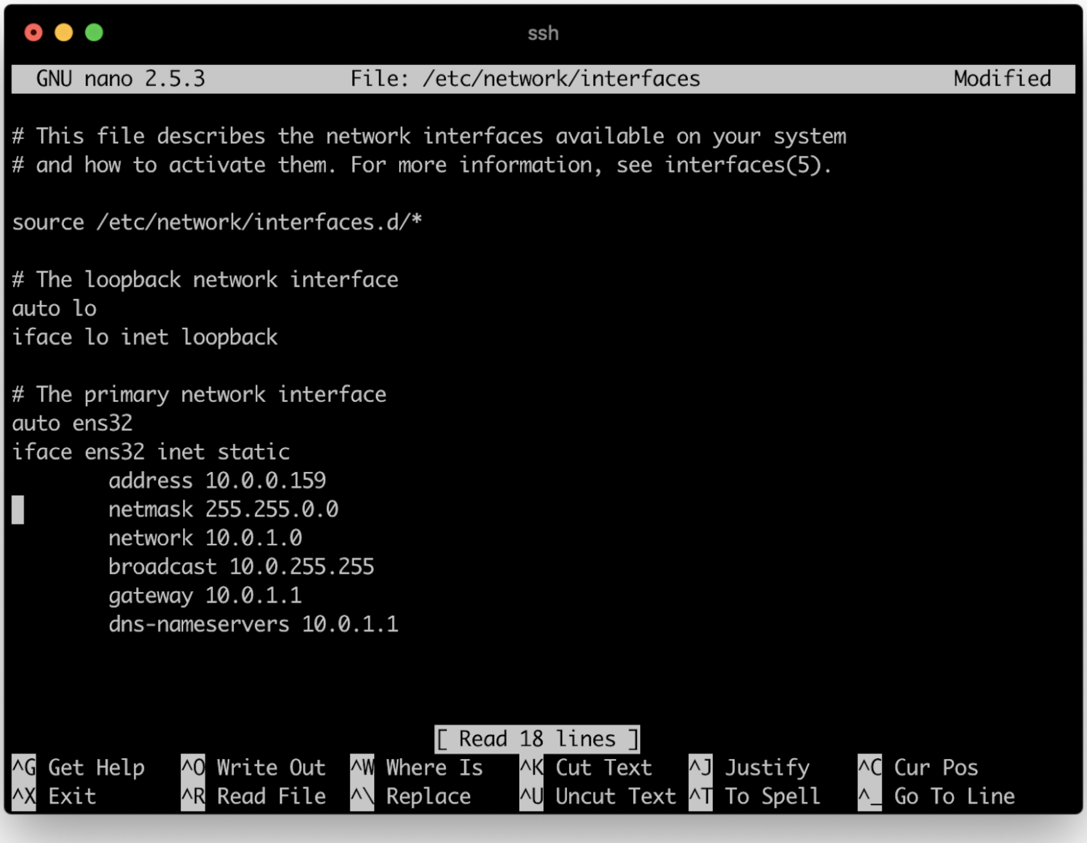
Update the network configuration of primary network interface by replacing **iface ens32 inet dhcp** with the following,

```
iface ens32 inet static
      address 10.0.0.159
      netmask 255.255.0.0
      network 10.0.1.0
      broadcast 10.0.255.255
      gateway 10.0.1.1
      dns-nameservers 10.0.1.1
```

The updated network interfaces file will most likely look similar to,

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens32
iface ens32 inet static
            address 10.0.0.159
            netmask 255.255.0.0
            network 10.0.1.0
            broadcast 10.0.255.255
            gateway 10.0.1.1
            dns-nameservers 10.0.1.1
```

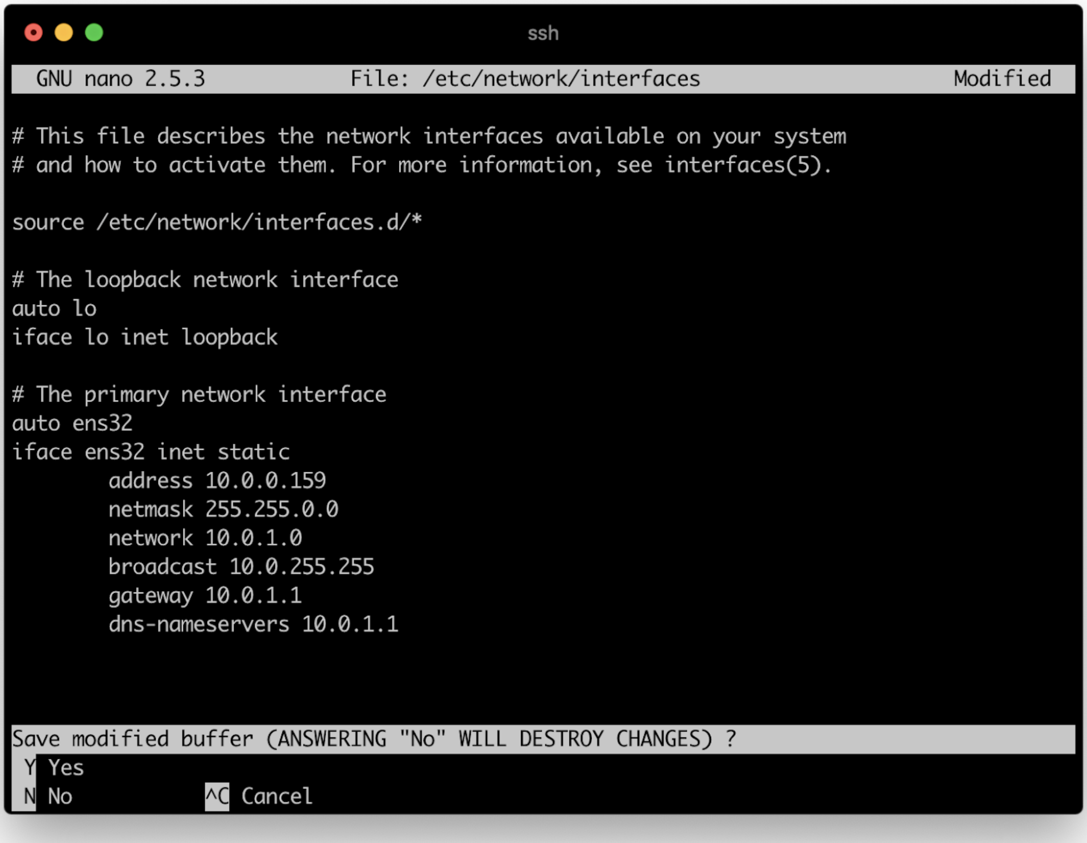
When modifications are complete, press **CTRL** + **X** to exit out of the editor.



ssh

```
  GNU nano 2.5.3              File: /etc/network/interfaces                    Modified

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens32
iface ens32 inet static
        address 10.0.0.159
        netmask 255.255.0.0
        network 10.0.1.0
        broadcast 10.0.255.255
        gateway 10.0.1.1
        dns-nameservers 10.0.1.1



                                   [ Read 18 lines ]
^G Get Help     ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos
^X Exit         ^R Read File    ^\ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line
```

Press **Y** when asked to save changes and press **ENTER** to exit out of the editor.



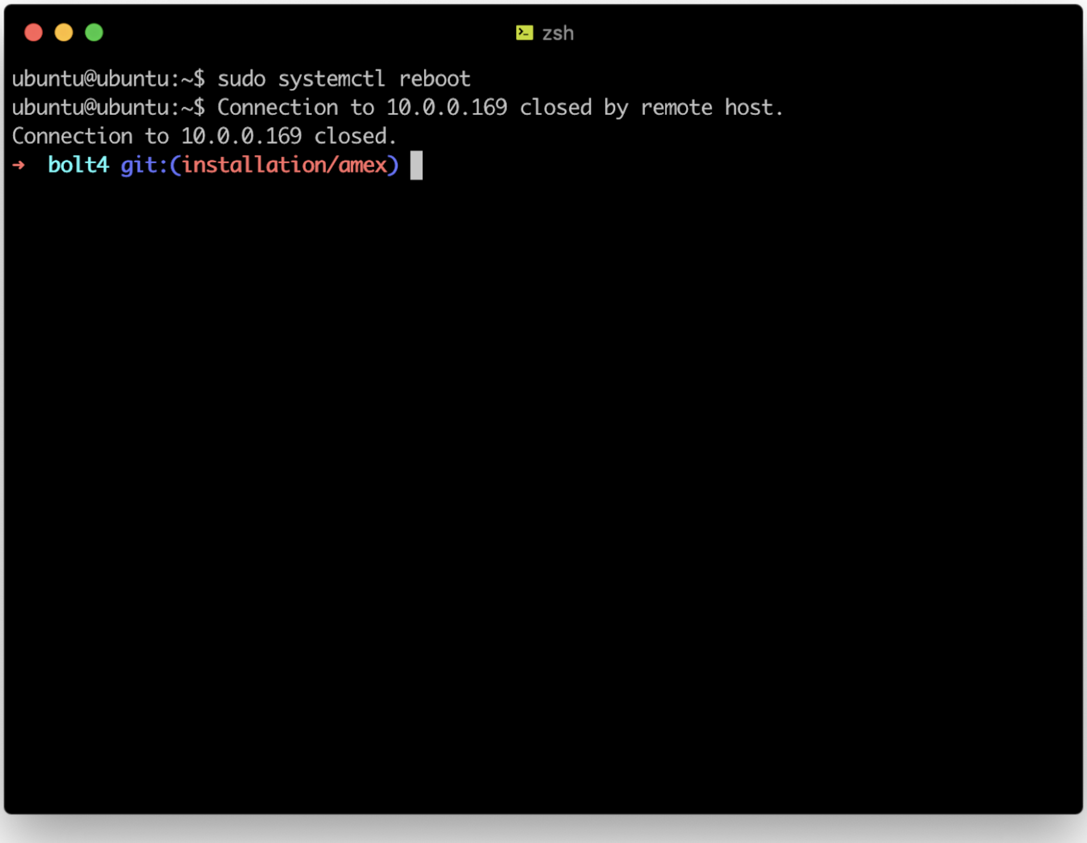
```
  GNU nano 2.5.3               File: /etc/network/interfaces                  Modified

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto ens32
iface ens32 inet static
        address 10.0.0.159
        netmask 255.255.0.0
        network 10.0.1.0
        broadcast 10.0.255.255
        gateway 10.0.1.1
        dns-nameservers 10.0.1.1



Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
 Y Yes
 N No           ^C Cancel
```

Remove the **cloud-init** network configuration file located at **/etc/network/interfaces.d/50-cloud-init.cfg.**

```
sudo rm /etc/network/interfaces.d/50-cloud-init.cfg
```

Restart the bolt server to apply the updated network configuration,

```
sudo systemctl reboot
```

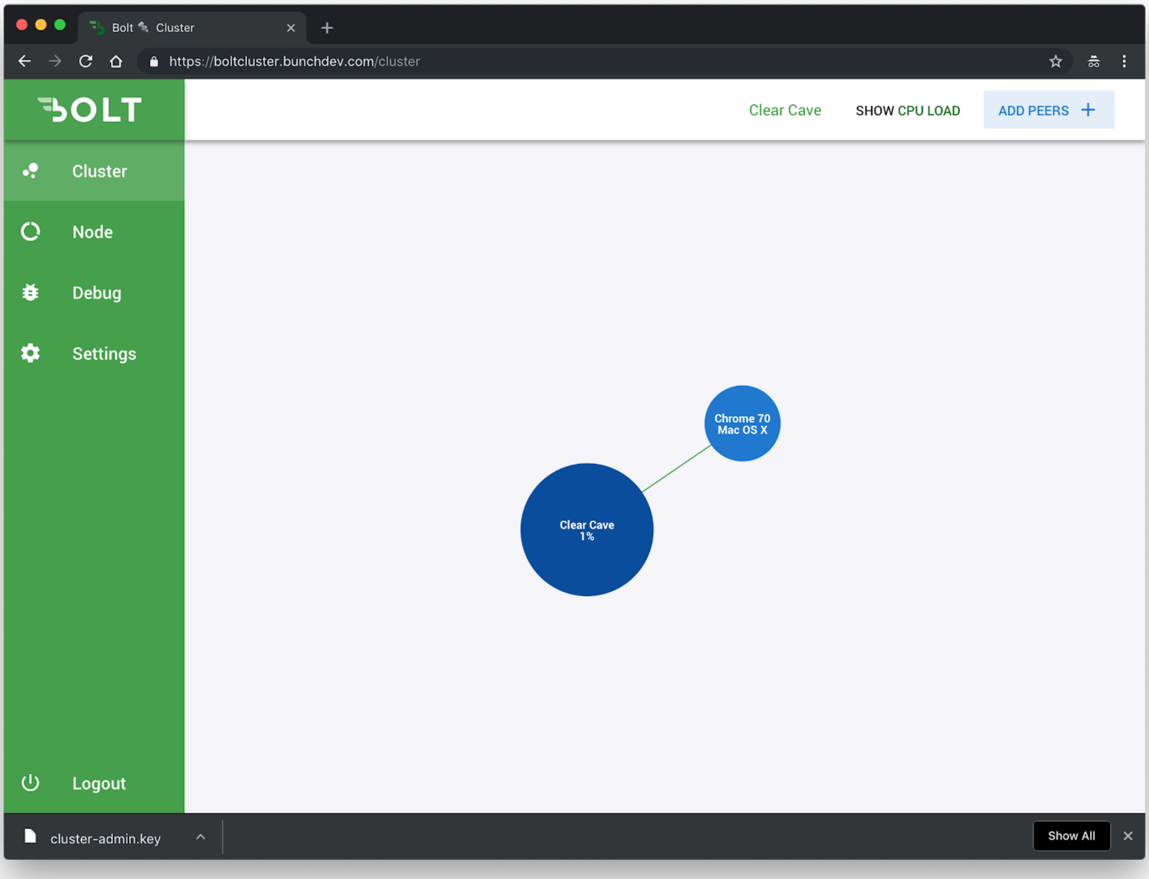


```
ubuntu@ubuntu:~$ sudo systemctl reboot
ubuntu@ubuntu:~$ Connection to 10.0.0.169 closed by remote host.
Connection to 10.0.0.169 closed.
➜  bolt4 git:(installation/amex)
```

Join Cluster

Add Peers

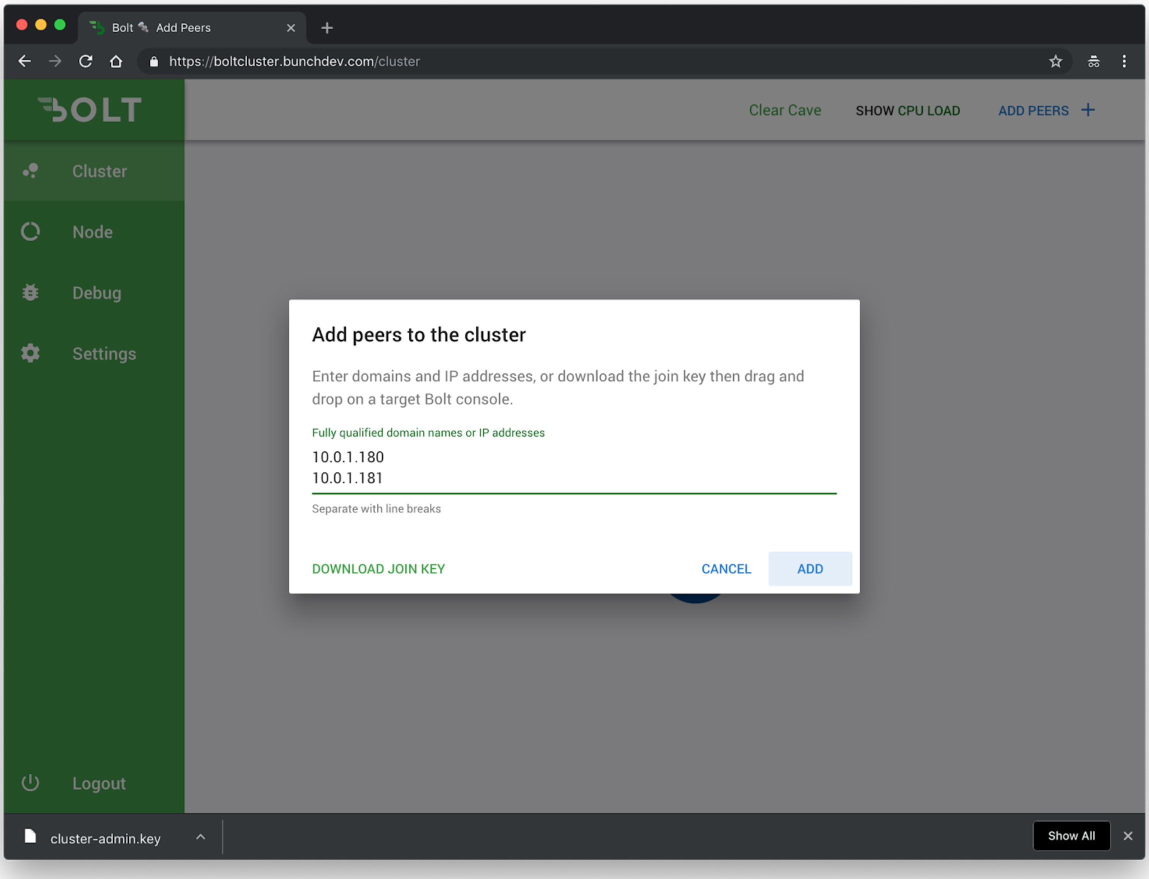
Click on the Cluster button on the left sidebar to navigate to the cluster page. Click on **Add Peers +** button at the top right corner of the page to add peers to the cluster.



Enter the IP address or domain names of the peers in the form “Fully qualified domain names or IP addresses” and click on the **Add** button to add them to the cluster.

For example, if **10.0.1.180** and **10.0.1.181** are the IP address of the peers then add them to the form one by one on a new line.

10.0.1.180
Press <**ENTER**>
10.0.1.181



Add peers to the cluster

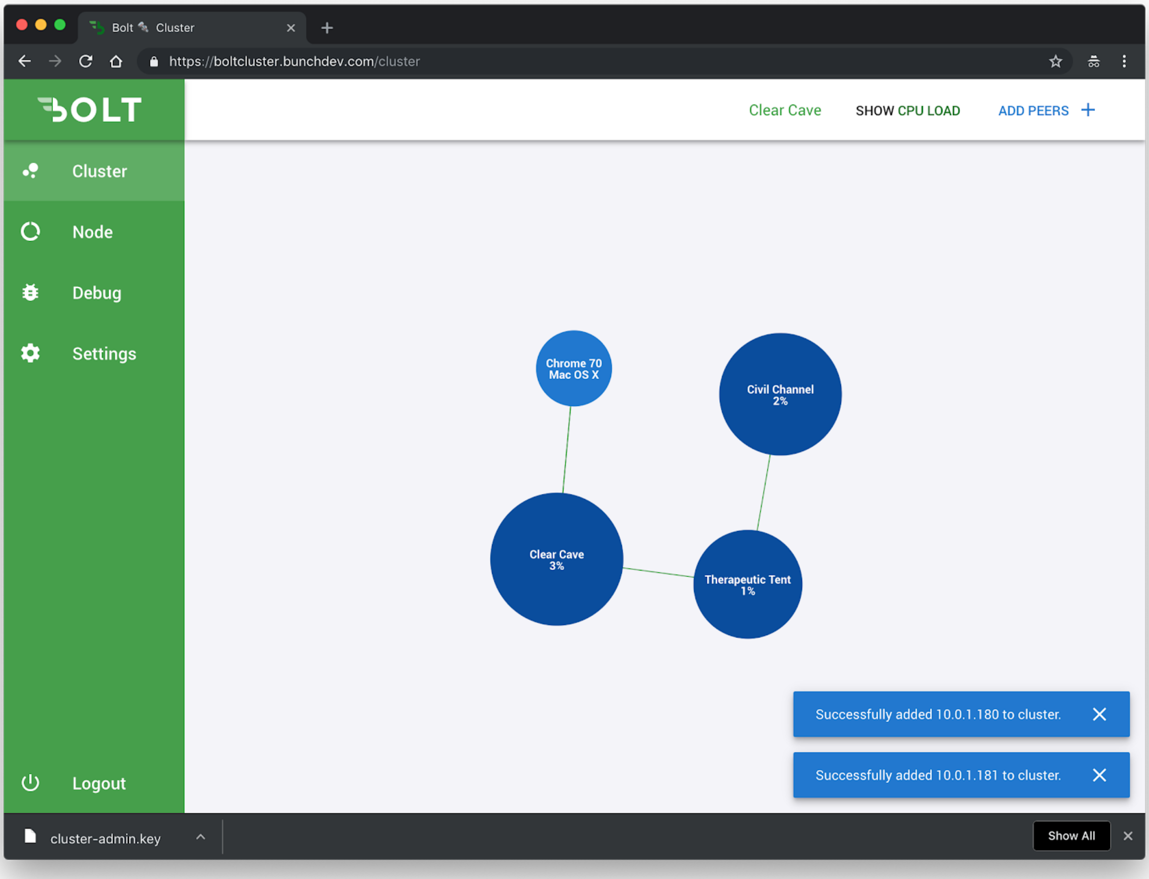
Enter domains and IP addresses, or download the join key then drag and drop on a target Bolt console.

Fully qualified domain names or IP addresses

10.0.1.180
10.0.1.181

Separate with line breaks

After the peers are successfully registered on the cluster, they pop up on the cluster page as dark blue circles directly connected to the first node.



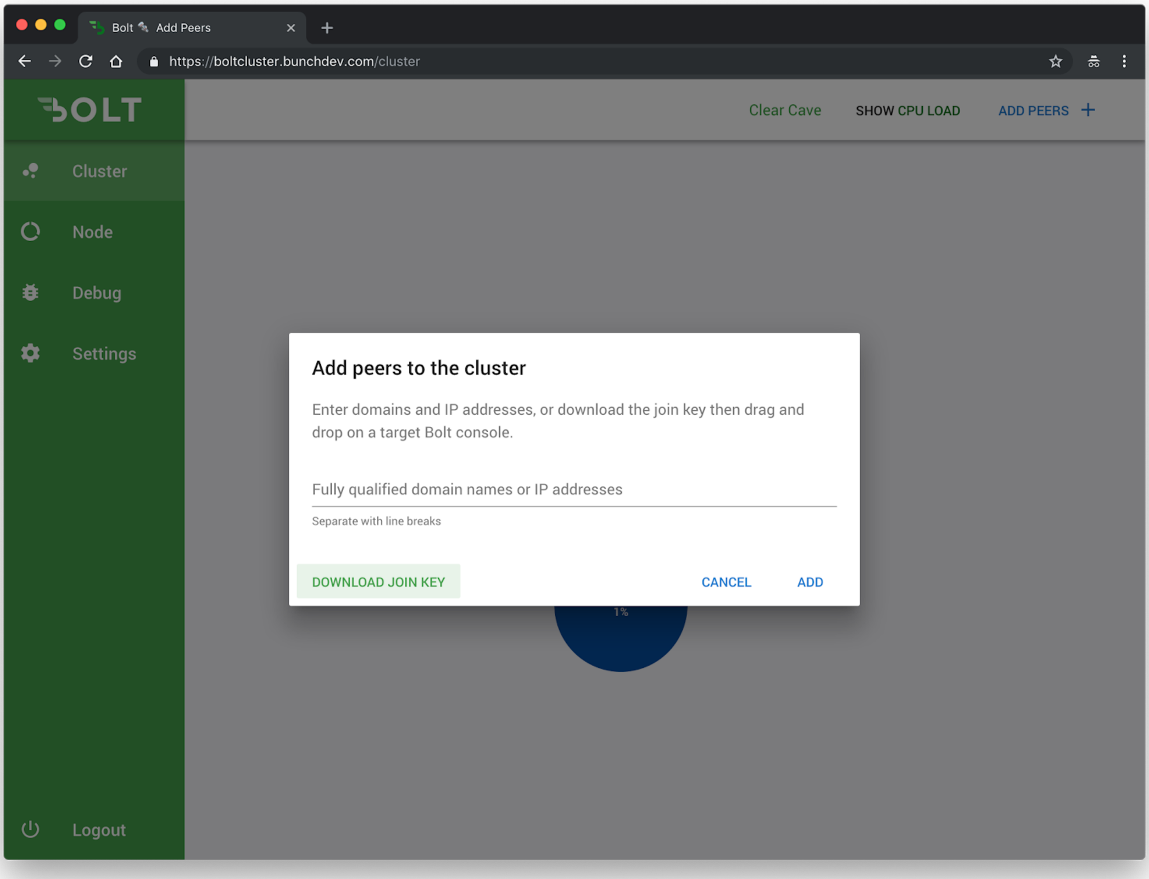
Successfully added 10.0.1.180 to cluster.

Successfully added 10.0.1.181 to cluster.

Cluster Join Key

Peers can be added to the cluster via cluster join key. Click on **Add Peers +** button to open the **Add peers to the cluster** dialog.

Click on **Download Join Key** button to download the cluster join key.



Add peers to the cluster

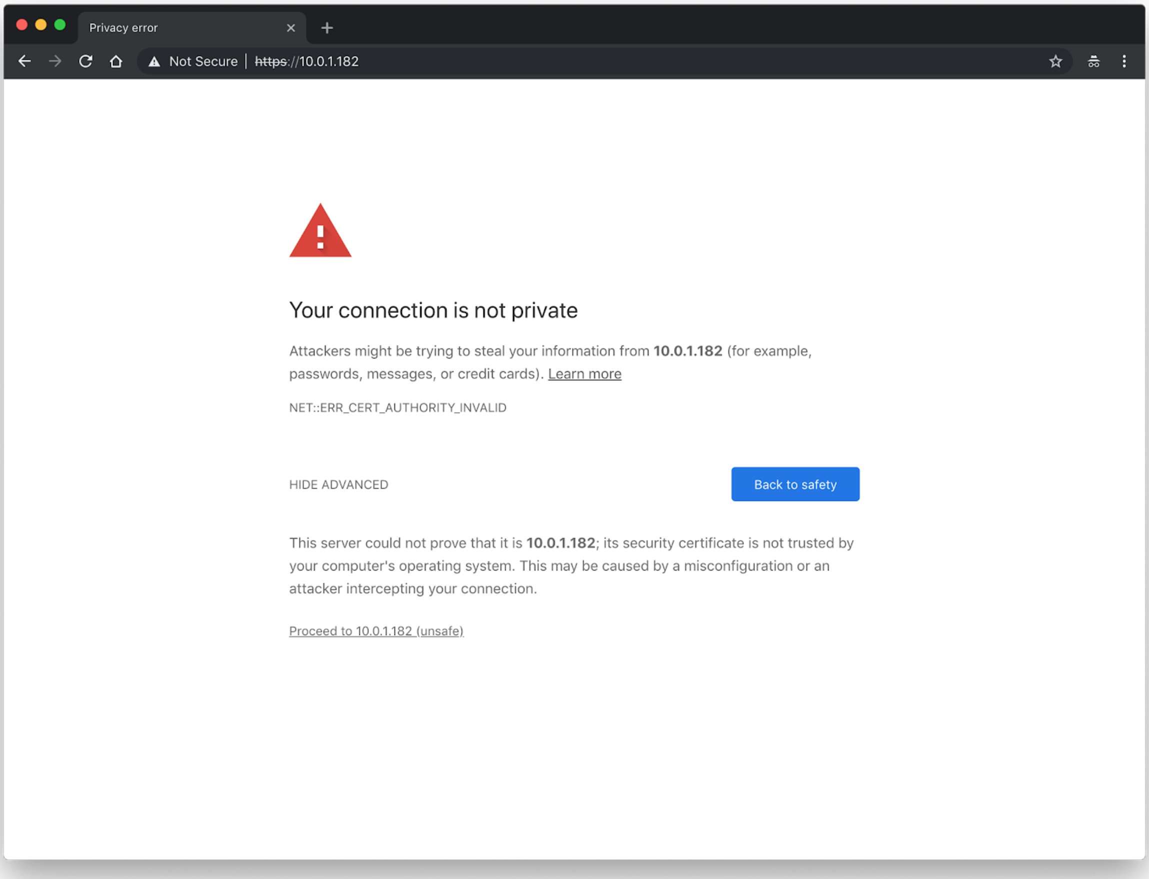
Enter domains and IP addresses, or download the join key then drag and drop on a target Bolt console.

Fully qualified domain names or IP addresses

Separate with line breaks

Open the **https://PEER_IP_ADDRESS** page on the browser. For example, if the peer IP address is **10.0.1.182** then the URL would be **https://10.0.1.182**.

Proceed through the SSL certificate warnings.



Your connection is not private

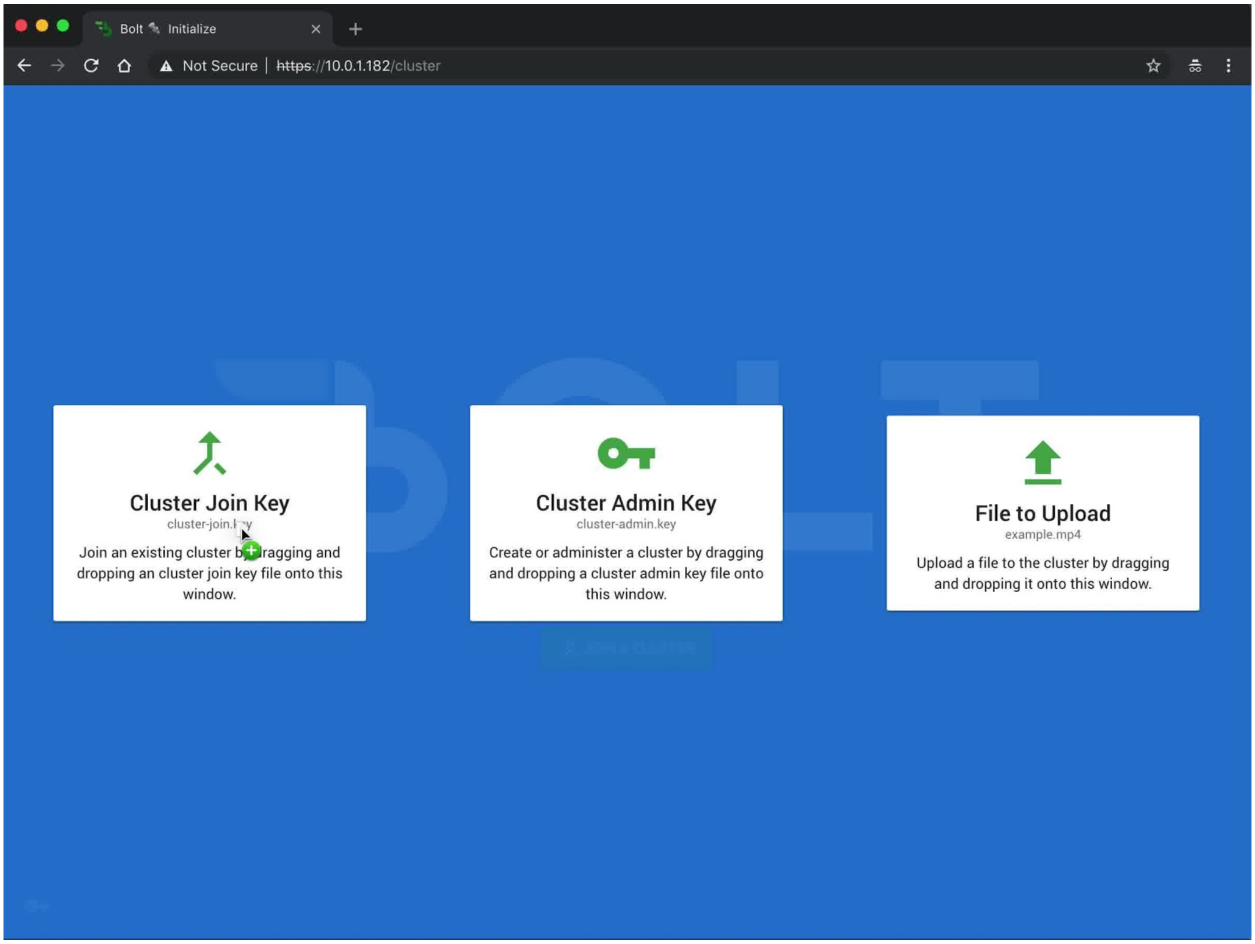
Attackers might be trying to steal your information from **10.0.1.182** (for example, passwords, messages, or credit cards). Learn more

NET::ERR_CERT_AUTHORITY_INVALID

This server could not prove that it is **10.0.1.182**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

Proceed to 10.0.1.182 (unsafe)

Drag and drop the **cluster-join.key** from the previous step on the **Cluster Join Key** box located at the center-left side of the page.



Cluster Join Key
cluster-join.key
Join an existing cluster by dragging and dropping an cluster join key file onto this window.

Cluster Admin Key
cluster-admin.key
Create or administer a cluster by dragging and dropping a cluster admin key file onto this window.

File to Upload
example.mp4
Upload a file to the cluster by dragging and dropping it onto this window.