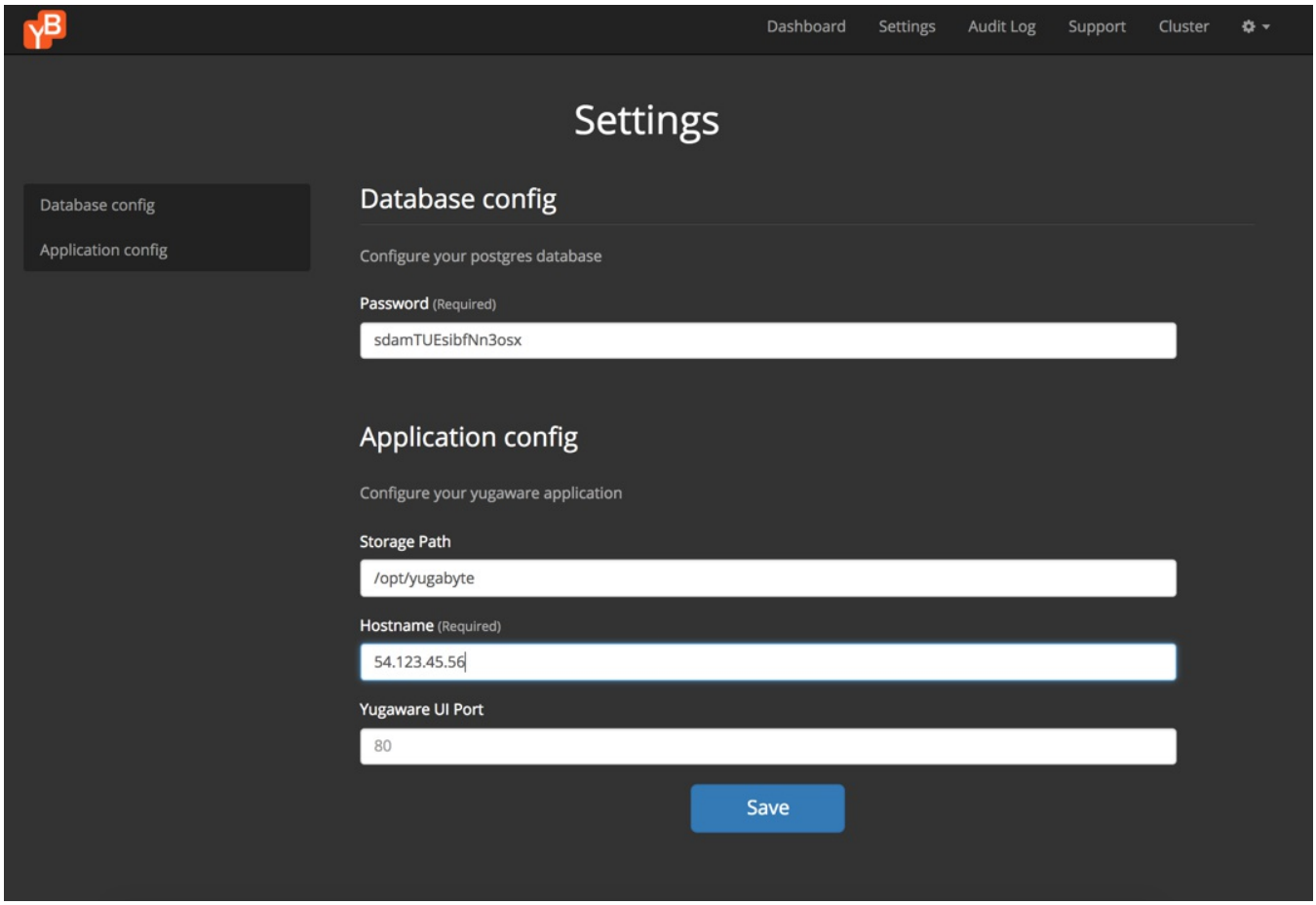
Configure Admin Console

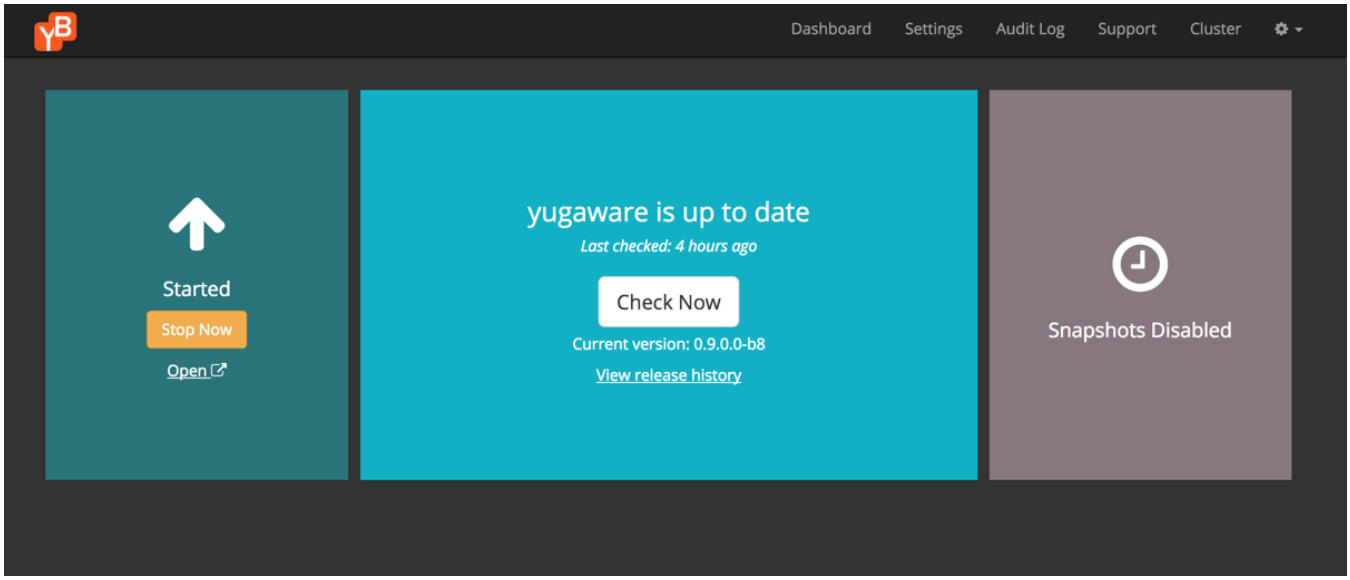
On this page

Configuring YugaWare, the YugaByte DB Admin Console, is really simple. A randomly generated password for the YugaWare config database is already pre-filled. You can make a note of it for future use or change it to a new password of your choice. Additionally, `/opt/yugabyte` is pre-filled as the location of the directory on the YugaWare host where all the YugaWare data will be stored. Clicking Save on this page will take us to the Replicated Dashboard.

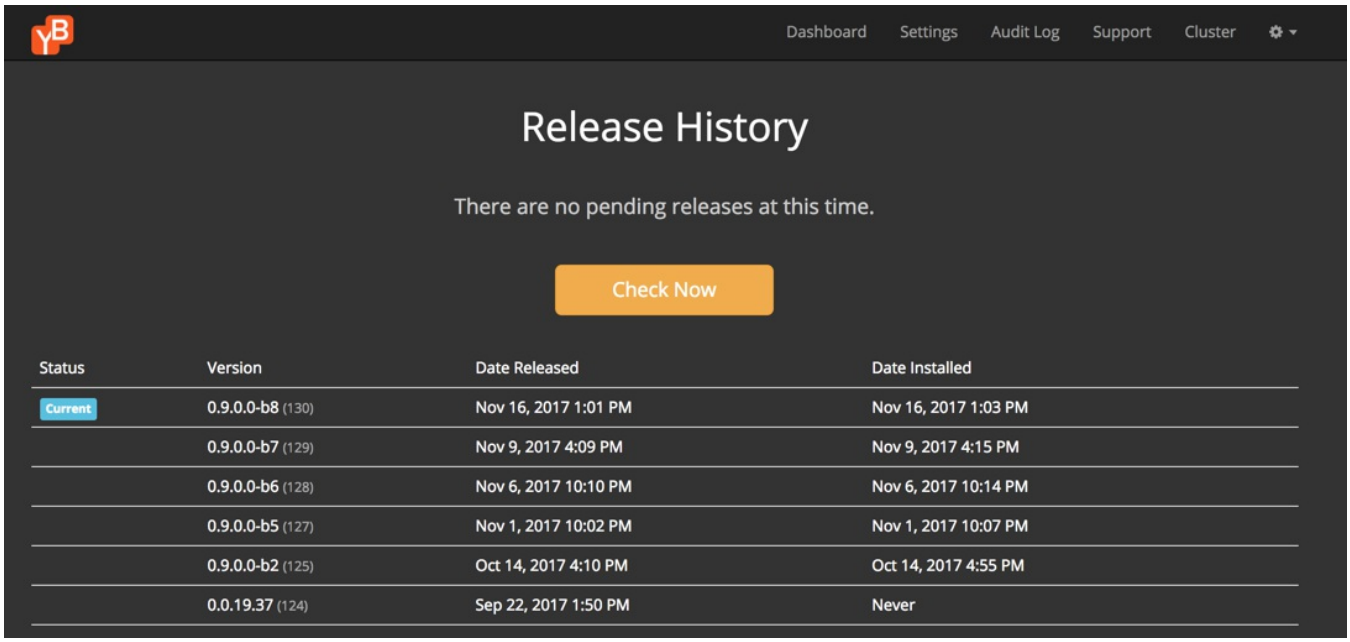


For airgapped installation , all the containers powering the YugaWare application are already available with Replicated. For non-airgapped installations, these containers will be downloaded from the Quay.io Registry when the Dashboard is first launched. Replicated will automatically start the application as soon as all the container images are

available.



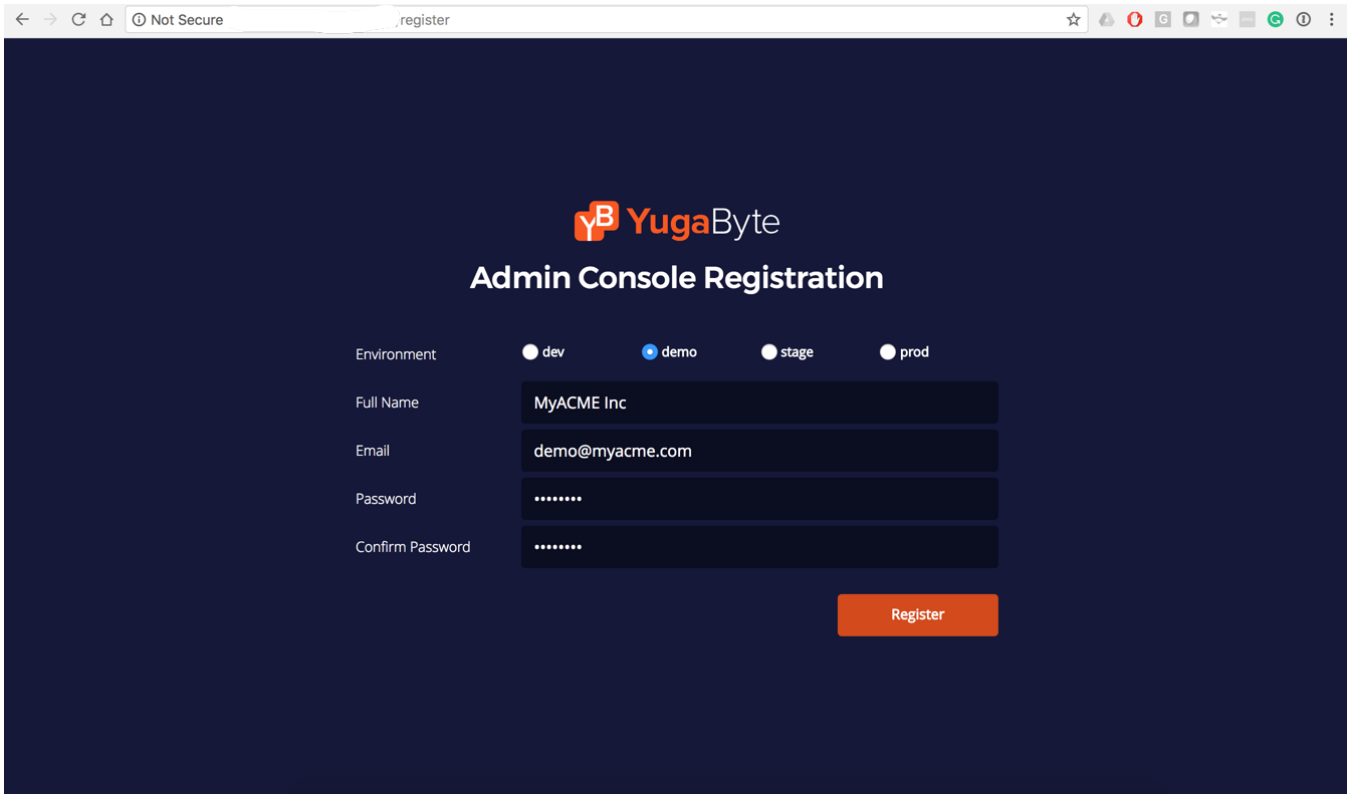
Click on “View release history” to see the release history of the YugaWare application.



After starting the YugaWare application, you must register a new tenant in YugaWare by following the instructions in the section below

Register tenant

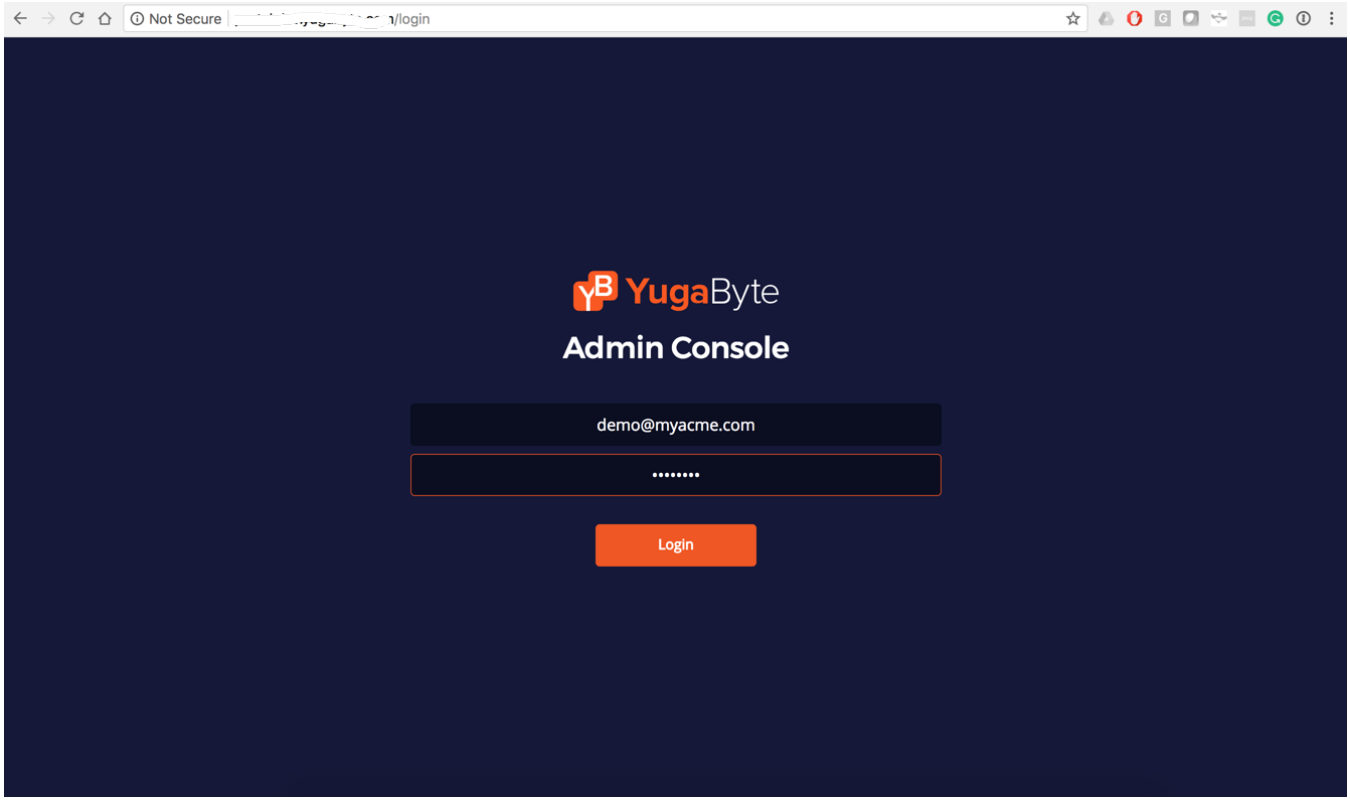
Go to http://yugaware-host-public-ip/register to register a tenant account. Note that by default YugaWare runs as a single-tenant application.



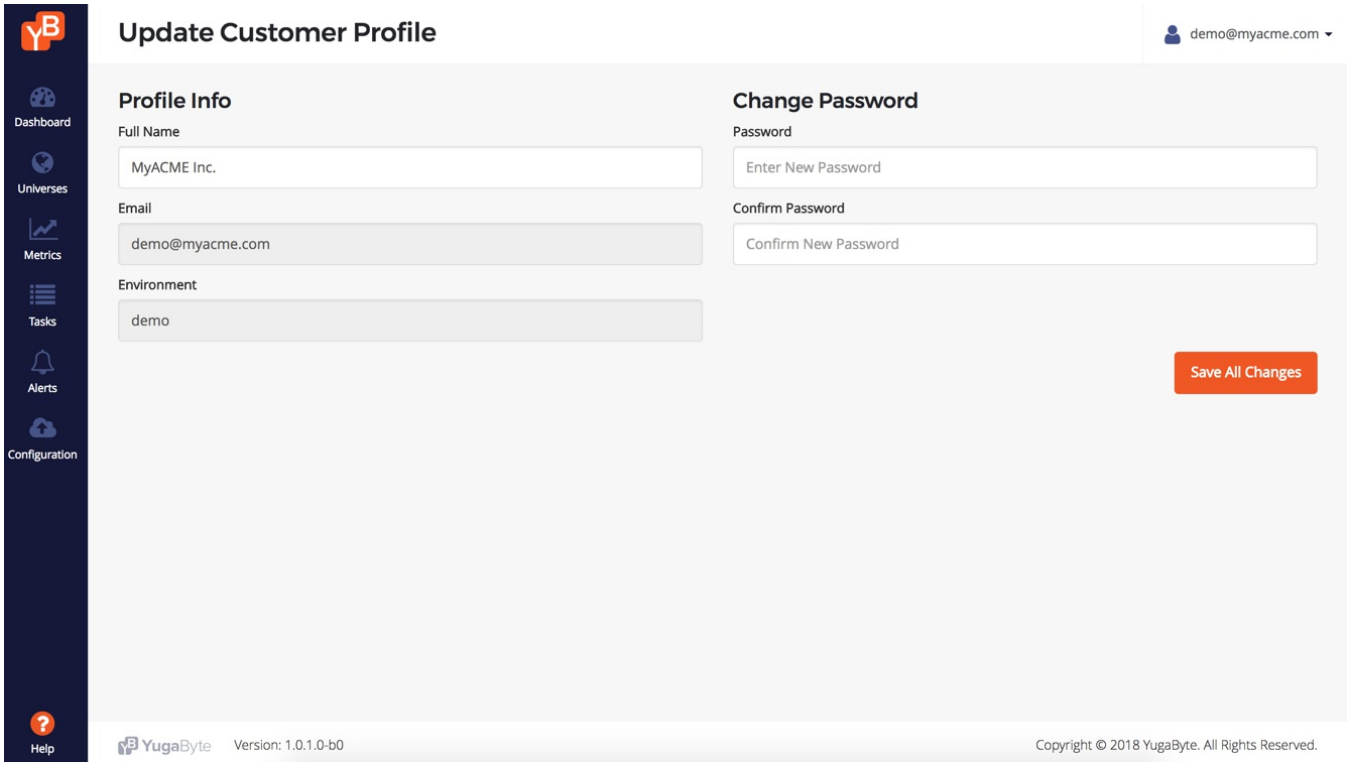
After clicking Submit, you will be automatically logged into YugaWare. You can now proceed to configuring cloud providers in YugaWare.

Logging in

By default, http://yugaware-host-public-ip redirects to http://yugaware-host-public-ip/login. Login to the application using the credentials you had provided during the Register customer step.



By clicking on the top right dropdown or going directly tohttp://yugaware-host-public-ip/profile, you can change the profile of the customer provided during the Register customer step.



Next step is to configure one or more cloud providers in YugaWare as documented here.

Backup data

We recommend a weekly machine snapshot and weekly backups of `/opt/yugabyte`.

Doing a machine snapshot and backing up the above directory before performing an update is recommended as well.

Upgrade

Upgrades to YugaWare are managed seamlessly in the Replicated UI. Whenever a new YugaWare version is available for upgrade, the Replicated UI will show the same. You can apply the upgrade anytime you wish.

Upgrades to Replicated are as simple as rerunning the Replicated install command. This will upgrade Replicated components with the latest build.

Uninstall

Stop and remove the YugaWare application on Replicated first.

```
# stop the yugaware application on replicated
$ /usr/local/bin/replicated apps
```

```
# replace <appid> with the application id of yugaware from the command above
$ /usr/local/bin/replicated app <appid> stop
```

```
# remove yugaware app
$ /usr/local/bin/replicated app <appid> rm
```

```
# remove all yugaware containers
$ docker images | grep "yuga" | awk '{print $3}' | xargs docker rmi -f
```

```
# delete the mapped directory
$ rm -rf /opt/yugabyte
```

And then uninstall Replicated itself by following instructions documented here.

Troubleshoot

SELinux turned on on YugaWare host

If your host has SELinux turned on, then docker-engine may not be able to connect with the host. Run the following commands to open the ports using firewall exceptions.

```
sudo firewall-cmd --zone=trusted --add-interface=docker0
sudo firewall-cmd --zone=public --add-port=80/tcp
sudo firewall-cmd --zone=public --add-port=443/tcp
sudo firewall-cmd --zone=public --add-port=8800/tcp
sudo firewall-cmd --zone=public --add-port=5432/tcp
sudo firewall-cmd --zone=public --add-port=9000/tcp
sudo firewall-cmd --zone=public --add-port=9090/tcp
sudo firewall-cmd --zone=public --add-port=32769/tcp
sudo firewall-cmd --zone=public --add-port=32770/tcp
sudo firewall-cmd --zone=public --add-port=9880/tcp
sudo firewall-cmd --zone=public --add-port=9874-9879/tcp
```

Unable to perform passwordless ssh into the data nodes

If your YugaWare host is not able to do passwordless ssh to the data nodes, follow the steps below.

```
# Generate key pair
$ ssh-keygen -t rsa
```

```
# Setup passwordless ssh to the data nodes with private IPs 10.1.13.150, 10.1.13.151, 10.1.13.152
$ for IP in 10.1.13.150 10.1.13.151 10.1.13.152; do
  ssh $IP mkdir -p .ssh;
  cat ~/.ssh/id_rsa.pub | ssh $IP 'cat >> .ssh/authorized_keys';
done
```

Check host resources on the data nodes

heck resources on the data nodes with private IPs 10.1.13.150, 10.1.13.151, 10.1.13.152

```
for IP in 10.1.13.150 10.1.13.151 10.1.13.152; do echo $IP; ssh $IP 'echo -n "CPUs: ";cat /proc/cpuinfo | g
```

```
10.1.12.103
CPUs: 72
Mem: 251G
Disk: /dev/sda2       160G   13G  148G   8% /
10.1.12.104
CPUs: 88
Mem: 251G
Disk: /dev/sda2       208G   22G  187G  11% /
10.1.12.105
CPUs: 88
Mem: 251G
Disk: /dev/sda2       208G  5.1G  203G   3% /
```

Create mount paths on the data nodes

Create mount paths on the data nodes with private IPs 10.1.13.150, 10.1.13.151, 10.1.13.152.

```
for IP in 10.1.12.103 10.1.12.104 10.1.12.105; do ssh $IP mkdir -p /mnt/data0; done
```

SELinux turned on for data nodes

Add firewall exceptions on the data nodes with private IPs 10.1.13.150, 10.1.13.151, 10.1.13.152.

```
for IP in 10.1.12.103 10.1.12.104 10.1.12.105
do
  ssh $IP firewall-cmd --zone=public --add-port=7000/tcp;
  ssh $IP firewall-cmd --zone=public --add-port=7100/tcp;
  ssh $IP firewall-cmd --zone=public --add-port=9000/tcp;
  ssh $IP firewall-cmd --zone=public --add-port=9100/tcp;
  ssh $IP firewall-cmd --zone=public --add-port=11000/tcp;
  ssh $IP firewall-cmd --zone=public --add-port=12000/tcp;
  ssh $IP firewall-cmd --zone=public --add-port=9300/tcp;
  ssh $IP firewall-cmd --zone=public --add-port=9042/tcp;
  ssh $IP firewall-cmd --zone=public --add-port=6379/tcp;
done
```


Deploy
Install Admin Console

Deploy
Configure Cloud Providers

Give Feedback

Was this page helpful?

YES NO