



Installing Flow Enterprise Server 2023.2.1

Tags: [Flow Enterprise Server](#)

Before you begin your Flow Enterprise Server 2023.2.1 installation, carefully read the [system requirements](#).

This article covers installing Flow Enterprise for the first time. Read more about upgrading Flow if you want to [upgrade to 2023.2.1](#) from a previous version.

The Flow application is based on two components: Replicated's KOTS platform and the Pluralsight Flow application stack which runs on top of it. During installation, KOTS is installed first, then Flow is added as an application on this platform.

In this article

[Preinstallation instructions](#)

[Install Flow Enterprise](#)

[Configure the KOTS app](#)

Preinstallation instructions

- Read and follow the [Flow Enterprise Server 2023.2.1 system requirements](#).
- If you're using an external database, have your database administrator complete the necessary steps to [prepare the database](#).
- Download the latest copy of the `flow-enterprise-tools` package. Version 3.0.0.0 or greater is required. This is provided by your Pluralsight support representative. Install this package on each node of the Flow cluster.
- For the host server, copy `flow-enterprise-tools-<channel>[-airgap]-<version>.tar.gz` to the home directory of the user account used for the installation on the host server.
- Extract the tools file using `tar xvf flow-enterprise-tools-<channel>[-airgap]-<version>.tar.gz`.

Note: Run any tool from the bin directory by running `cd /path/to/flow-enterprise-tools/bin ./[tool name]`. Install the tools package with the `install-enterprise-tools.sh` script `cd /path/to/flow-enterprise-tools ./install-enterprise-tools.sh`. The script will ask where to install the components. The default is `/usr/local/share/flow-enterprise-tools`.

- Install the `flow-enterprise-tools` package using the script `install-enterprise-tools.sh`. This allows installation of the scripts and binaries in the system where users can access them from standard `PATH`

settings. You can also run this package's scripts in-place.

Important: For the root user, `/usr/local/bin` must be in the `PATH` environment variable. Set up the root user as a Flow user after installing Flow. This is critical if your OS is hardened.

- Create the following directories. These can all be bundled in a single volume:

app directory (default: `/opt/flow`)

```
mkdir -p [app_directory]
```

```
chmod 755 [app_directory]
```

Once the app directory is created, change the ownership of the directory to:

```
chown -R 37355:37355 [app_directory]
```

- If you're using an embedded database, make a database directory (default: `/opt/flow/database`) on the primary host server only:

```
mkdir -p [database_directory]
```

```
chmod 725 [database_directory]
```

Once the database directory is created, change the ownership of the directory to:

```
chown -R 1000:1000 [database_directory]
```

If your version of Flow Enterprise Server is airgapped, download the airgap bundle from [Replicated \(external site, opens in new tab\)](#). A password is required. If you can't access Replicated, contact Support for assistance.

Tip: Depending on how you install Flow, you need to download a few different packages.

- `flow-enterprise-tools` contains the tools for installation and maintenance of the Kubernetes framework.
- Download the airgap version for airgapped installations. The application airgap bundle, only for airgap installations, is downloaded from Replicated. It contains the Flow application files, which are installed into the Kubernetes framework provided by `flow-enterprise-tools`.

[back to top](#)

Install Flow Enterprise

Now you're ready to install Flow Enterprise 2023.2.1. For an online installation, run `sudo flow-tools install`. For an airgapped installation, run `sudo flow-tools install -a`. This only needs to be done on the primary node.

Important: A raw block device for each node is required for all Flow installations.

The device filter is in GoLang regular expression format. Device names vary between distributions. Add multiple filters if the devices are named differently in worker nodes. An example of such a case may look like `sd[b-`

z)|nvme[1-3]|xvd[b-c].

```
[admin_user@primary-node ~]$ sudo flow-tools install
```

```
[INFO] Running flow-tools with args : install
```

```
[INFO] Verifying installation environment...
```

```
[INFO] HTTP command (curl): OK
```

```
[INFO] Archive command (tar): OK
```

```
[INFO] Swarm does not exist: OK
```

```
[INFO] Verifying system requirements...
```

```
[INFO] Checking networking...
```

```
[INFO] sysctl command : OK
```

```
[INFO] IPV6 Kernel module: LOADED
```

```
[INFO] IPV6 Check : OK
```

```
[INFO] IPv4 Forwarding: ENABLED
```

```
[INFO] Check IPTable Rules: OK
```

```
[INFO] Detecting proxy: NOT DETECTED
```

```
[INFO] https://replicated.app site check : OK
```

```
[INFO] Checking hardware...
```

```
[INFO] CPU: OK
```

```
[INFO] Memory: OK
```

```
[INFO] Space check in /var/lib/containerd: OK
```

```
[INFO] Space check in /var/lib/kubelet: OK
```

```
[INFO] Space check in /opt/replicated: OK
```

```
[INFO] Space check in /var/lib/kurl: OK
```

```
[INFO] Space check in /tmp: OK
```

```
[INFO] Space for Repo cache in /opt/flow: 199 GB
```

```
[INFO] Disk Space Check: OK
```

[INFO] Non SSD Disks: NOT DETECTED

[INFO] Checking filesystem and permissions...

[INFO] Login restrictions check: OK

[INFO] Selinux Status: enabled

[INFO] Selinux Current mode: permissive

[INFO] bash Umask setting: OK

[INFO] /etc/profile Umask setting: OK

[INFO] Checking PATH for /usr/local/bin: OK

[INFO] Checking distro...

[INFO] No existing ceph raw disks detected

[INFO] Installation type is: NEW

=== Discovered Block Devices ===

/dev/nvme1n1

Above is the list of block devices found during valid device discovery

Please provide pattern to match devices that should be used for K8s volume storage: nvme1n1

[INFO] Validating block storage device filter...

Device match: /dev/nvme1n1

Device size: 150G

Device status: valid

[INFO] Total valid block storage: 150G

[INFO] Block storage: OK

[INFO] Adding patch to use raw ceph block devices for installation

[INFO] Installing KOTS application

[INFO] Saving environment

[INFO] Fetching kurl.sh installation script from: <https://k8s.kurl.sh/version/v2023.04.13-0/flow-enterprise>

[INFO] Fetching join script from: <https://k8s.kurl.sh/version/v2023.04.13-0/flow-enterprise/join.sh>

⚙ Running install with the argument(s): installer-spec-file=/tmp/flow-tools49o/installer-patch.yaml

Downloading package kurl-bin-utils-v2023.04.13-0.tar.gz

The installer will use network interface 'eth0' (with IP address '192.168.0.80')

Downloading package weave-2.8.1-20230324.tar.gz

Downloading package rook-1.11.2.tar.gz

Downloading package ekco-0.26.5.tar.gz

Downloading package contour-1.24.2.tar.gz

Downloading package registry-2.8.1.tar.gz

Downloading package prometheus-0.63.0-45.7.1.tar.gz

Downloading package kotsadm-1.97.0.tar.gz

Downloading package kubernetes-1.25.8.tar.gz

⚙ Running host preflights

[TCP Port Status] Running collector...

[CPU Info] Running collector...

[Amount of Memory] Running collector...

[Block Devices] Running collector...

[Host OS Info] Running collector...

[Ephemeral Disk Usage /var/lib/kubelet] Running collector...

[Ephemeral Disk Usage /var/lib/containerd] Running collector...

[Ephemeral Disk Usage /var/lib/rook] Running collector...

[Kubernetes API TCP Port Status] Running collector...

[ETCD Client API TCP Port Status] Running collector...

[ETCD Server API TCP Port Status] Running collector...

[ETCD Health Server TCP Port Status] Running collector...

[Kubelet Health Server TCP Port Status] Running collector...

[Kubelet API TCP Port Status] Running collector...

[Kube Controller Manager Health Server TCP Port Status] Running collector...

[Kube Scheduler Health Server TCP Port Status] Running collector...

[Filesystem Latency Two Minute Benchmark] Running collector...

[Host OS Info] Running collector...

[Weave Network Policy Controller Metrics Server TCP Port Status] Running collector...

[Weave Net Metrics Server TCP Port Status] Running collector...

[Weave Net Control TCP Port Status] Running collector...

[Block Devices] Running collector...

[Pod csi-rbdplugin Host Port] Running collector...

[Node Exporter Metrics Server TCP Port Status] Running collector...

[Can Access Replicated API] Running collector...

[Host OS Info] Running collector...

10426 19:53:35.313332 3925 analyzer.go:76] excluding "Certificate Key Pair" analyzer

10426 19:53:35.313363 3925 analyzer.go:76] excluding "Certificate Key Pair" analyzer

10426 19:53:35.313370 3925 analyzer.go:76] excluding "Kubernetes API Health" analyzer

10426 19:53:35.313415 3925 analyzer.go:76] excluding "Ephemeral Disk Usage /var/lib/docker" analyzer

10426 19:53:35.313457 3925 analyzer.go:76] excluding "Ephemeral Disk Usage /var/openlibs" analyzer

10426 19:53:35.313469 3925 analyzer.go:76] excluding "Kubernetes API Server Load Balancer" analyzer

10426 19:53:35.313476 3925 analyzer.go:76] excluding "Kubernetes API Server Load Balancer Upgrade" analyzer

10426 19:53:35.313545 3925 analyzer.go:76] excluding "Kubernetes API TCP Connection Status" analyzer

10426 19:53:35.313692 3925 analyzer.go:76] excluding "Collectd Support" analyzer

10426 19:53:35.313714 3925 analyzer.go:76] excluding "Docker Support" analyzer

10426 19:53:35.313719 3925 analyzer.go:76] excluding "Containerd and Weave Compatibility" analyzer

[PASS] Number of CPUs: This server has at least 4 CPU cores

[PASS] Amount of Memory: The system has at least 8G of memory

[PASS] Ephemeral Disk Usage /var/lib/kubelet: The disk containing directory /var/lib/kubelet has at least 30Gi of total space, has at least 10Gi of disk space available, and is less than 60% full

[PASS] Ephemeral Disk Usage /var/lib/containerd: The disk containing directory /var/lib/containerd has at least 30Gi of total space, has at least 10Gi of disk space available, and is less than 60% full.

[PASS] Ephemeral Disk Usage /var/lib/rook: The disk containing directory /var/lib/rook has sufficient space

[PASS] Kubernetes API TCP Port Status: Port 6443 is open

[PASS] ETCD Client API TCP Port Status: Port 2379 is open

[PASS] ETCD Server API TCP Port Status: Port 2380 is open

[PASS] ETCD Health Server TCP Port Status: Port 2381 is available

[PASS] Kubelet Health Server TCP Port Status: Port 10248 is available

[PASS] Kubelet API TCP Port Status: Port 10250 is open

[PASS] Kube Controller Manager Health Server TCP Port Status: Port 10257 is available

[PASS] Kube Scheduler Health Server TCP Port Status: Port 10259 is available

[PASS] Filesystem Performance: Write latency is ok (p99 target < 10ms, actual: 2.236242ms)

[PASS] NTP Status: System clock is synchronized

[PASS] NTP Status: Timezone is set to UTC

[PASS] Weave Network Policy Controller Metrics Server TCP Port Status: Port 6781 is available

[PASS] Weave Net Metrics Server TCP Port Status: Port 6782 is available

[PASS] Weave Net Control TCP Port Status: Port 6783 is open

[PASS] Block Devices: One available block device

[PASS] Pod csi-rbdplugin Host Port Status: Port 9090 is available for use.

[PASS] Node Exporter Metrics Server TCP Port Status: Port 9100 is available

[PASS] Can Access Replicated API: Connected to https://replicated.app

⚙ Host preflights success

Found pod network: 10.32.0.0/20

Found service network: 10.96.0.0/22

⚙ Addon containerd 1.6.19

⚙ Installing host packages containerd.io

✓ Host packages containerd.io installed

Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service →
/usr/lib/systemd/system/containerd.service.

Restarting containerd...

Service containerd restarted.

unpacking registry.k8s.io/pause:3.6

(sha256:79b611631c0d19e9a975fb0a8511e5153789b4c26610d1842e9f735c57cc8b13)...done

unpacking docker.io/replicated/kurl-util:v2023.04.13-0

(sha256:ddd6aa44489f719961c14ddc4e961e97c8e45303b61e3b21b63cda471943f70c)...done

* Applying /usr/lib/sysctl.d/10-default-yama-scope.conf ...

kernel.yama.ptrace_scope = 0

* Applying /usr/lib/sysctl.d/50-coredump.conf ...

kernel.core_pattern = |/usr/lib/systemd/systemd-coredump %P %u %g %s %t %c %h %e

kernel.core_pipe_limit = 16

* Applying /usr/lib/sysctl.d/50-default.conf ...

kernel.sysrq = 16

kernel.core_uses_pid = 1

kernel.kptr_restrict = 1

net.ipv4.conf.all.rp_filter = 1

net.ipv4.conf.all.accept_source_route = 0

net.ipv4.conf.all.promote_secondaries = 1

net.core.default_qdisc = fq_codel

fs.protected_hardlinks = 1

fs.protected_symlinks = 1

* Applying /usr/lib/sysctl.d/50-libkcapioptmem_max.conf ...

net.core.optmem_max = 81920

* Applying /usr/lib/sysctl.d/50-pid-max.conf ...

kernel.pid_max = 4194304

* Applying /etc/sysctl.d/99-replicated-ipv4.conf ...

net.bridge.bridge-nf-call-iptables = 1

net.ipv4.conf.all.forwarding = 1

* Applying /etc/sysctl.d/99-sysctl.conf ...

net.bridge.bridge-nf-call-iptables = 1

net.ipv4.conf.all.forwarding = 1

* Applying /etc/sysctl.conf ...

net.bridge.bridge-nf-call-iptables = 1

net.ipv4.conf.all.forwarding = 1

⚙️ Install kubelet, kubectl and cni host packages

kubelet command missing - will install host components

⚙️ Installing host packages kubelet-1.25.8 kubectl-1.25.8 git

✓ Host packages kubelet-1.25.8 kubectl-1.25.8 git installed

Restarting Kubelet

Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service →
/usr/lib/systemd/system/kubelet.service.

✓ Kubernetes host packages installed

unpacking registry.k8s.io/kube-apiserver:v1.25.8

(sha256:4f6f4b1ccacab5fe9e1c0f2c882193675a675ab1536b309a4365cf93b976672bf)...done

unpacking registry.k8s.io/kube-proxy:v1.25.8

(sha256:ff46f3cfe059ea5625048066d3c0354ec77e73291c5df59aedd225b394ab3689)...done

unpacking registry.k8s.io/etcd:3.5.6-0

(sha256:00e797072c1d464279130edbd58cbe862ff94972b82aeac5c0786b6278e21455)...done

unpacking registry.k8s.io/kube-scheduler:v1.25.8

(sha256:95861c02f28cc2e0feaf9dfcd7748d6d746a8c2b104c2e85a4e8a682b1d9b8a3)...done

unpacking registry.k8s.io/coredns/coredns:v1.9.3

(sha256:df9ab8f5cf54a9ec2ad6c14dadb4bd6c37e4bc80d54f0110534c4237607d2ea2)...done

unpacking registry.k8s.io/kube-controller-manager:v1.25.8

(sha256:bd29659cc2115f1a6f5cca080ebbf973471189ed0936e0e33ebc5fbc6bbf4707)...done

unpacking registry.k8s.io/pause:3.8

(sha256:bc7a375f431244f9649999cd506fe6dc4b7f071ddc385c6bcb372288d4b773d4)...done

/var/lib/kurl/krew /var/lib/kurl /home/admin_user

/var/lib/kurl /home/admin_user

/var/lib/kurl/packages/kubernetes/1.25.8/assets /var/lib/kurl /home/admin_user

/var/lib/kurl /home/admin_user

...

⚙ Initialize Kubernetes

⚙ generate kubernetes bootstrap token

Kubernetes bootstrap token: unyplr.9ah9q2j8dgruk2a8

This token will expire in 24 hours

W0426 19:59:53.163991 6079 common.go:84] your configuration file uses a deprecated API spec: "kubeadm.k8s.io/v1beta2". Please use 'kubeadm config migrate --old-config old.yaml --new-config new.yaml', which will write the new, similar spec using a newer API version.

W0426 19:59:53.165399 6079 common.go:84] your configuration file uses a deprecated API spec: "kubeadm.k8s.io/v1beta2". Please use 'kubeadm config migrate --old-config old.yaml --new-config new.yaml', which will write the new, similar spec using a newer API version.

[init] Using Kubernetes version: v1.25.8

[preflight] Running pre-flight checks

[WARNING FileExisting-tc]: tc not found in system path

[preflight] Pulling images required for setting up a Kubernetes cluster

[preflight] This might take a minute or two, depending on the speed of your internet connection

[preflight] You can also perform this action in beforehand using 'kubeadm config images pull'

...

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

```
mkdir -p $HOME/.kube
```

```
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
```

```
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

Alternatively, if you are the root user, you can run:

```
export KUBECONFIG=/etc/kubernetes/admin.conf
```

You should now deploy a pod network to the cluster.

Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:

<https://kubernetes.io/docs/concepts/cluster-administration/addons/>

```
--discovery-token-ca-cert-hash
```

```
sha256:490bf2f2a73ef826e509837433dcc1b3243ed14a15f65d5c8b9b58af4a263ba4
```

```
node/primary-node already uncordoned
```

Waiting for kubernetes api health to report ok

node/primary-node labeled

✓ Kubernetes Master Initialized

Kubernetes control plane is running at https://192.168.0.80:6443

CoreDNS is running at https://192.168.0.80:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.

namespace/kurl created

✓ Cluster Initialized

service/registry created

customresourcedefinition.apiextensions.k8s.io/installers.cluster.kurl.sh created

installer.cluster.kurl.sh/flow-enterprise created

installer.cluster.kurl.sh/merged created

installer.cluster.kurl.sh/flow-enterprise labeled

configmap/kurl-last-config created

configmap/kurl-current-config created

configmap/kurl-current-config patched

configmap/kurl-cluster-uuid created

dir="ltr">🌀 Addon weave 2.8.1-20230324

serviceaccount/weave-net created

role.rbac.authorization.k8s.io/weave-net created

clusterrole.rbac.authorization.k8s.io/weave-net created

rolebinding.rbac.authorization.k8s.io/weave-net created

clusterrolebinding.rbac.authorization.k8s.io/weave-net created

secret/weave-passwd created

daemonset.apps/weave-net created

⚙️ Checking cluster networking

Checking if weave-net binary can be found in the path /opt/cni/bin/

lrwxrwxrwx. 1 root root 18 Apr 26 20:00 weave-net -> weave-plugin-2.8.1

Waiting up to 10 minutes for node to report Ready

service/kurlnet created

pod/kurlnet-server created

pod/kurlnet-client created

Waiting for kurlnet-client pod to start

Warning: Immediate deletion does not wait for confirmation that the running resource has been terminated.
The resource may continue to run on the cluster indefinitely.

pod "kurlnet-client" force deleted

pod "kurlnet-server" force deleted

service "kurlnet" deleted

configmap/kurl-current-config patched

⚙️ Addon rook 1.11.2

...

Awaiting rook-ceph pods

Flag --short has been deprecated, and will be removed in the future. The --short output will become the default.

storageclass.storage.k8s.io/default created

storageclass.storage.k8s.io/rook-cephfs created

cephblockpool.ceph.rook.io/replicapool created

cephcluster.ceph.rook.io/rook-ceph created

cephfilesystem.ceph.rook.io/rook-shared-fs created

cephobjectstore.ceph.rook.io/rook-ceph-store created

Awaiting rook-ceph dashboard password

⌵

Rook Ceph 1.4+ requires a secondary, unformatted block device attached to the host.

If you are stuck waiting at this step for more than two minutes, you are either missing the device or it is already formatted.

* If it is missing, attach it now and it will be picked up; or CTRL+C, attach, and re-start the installer

* If the disk is attached, try wiping it using the recommended zap procedure:

<https://rook.io/docs/rook/v1.10/Storage-Configuration/ceph-teardown/?h=zap#zapping-devices>

...

⚙ Addon ekco 0.26.5

...

⚙ Addon contour 1.24.2

...

⚙ Addon registry 2.8.1

object store bucket docker-registry created

Command "format-address" is deprecated, use 'kurl netutil format-ip-address' instead

configmap/registry-config created

configmap/registry-velero-config created

secret/registry-s3-secret created

secret/registry-session-secret created

service/registry configured

deployment.apps/registry created

secret/registry-htpasswd created

secret/registry-htpasswd patched

secret/registry-creds created

secret/registry-creds patched

/var/lib/kurl/addons/registry/2.8.1/tmp /var/lib/kurl /home/admin_user

Generating a RSA private key

.....+++++

.....+++++

writing new private key to 'registry.key'

Signature ok

subject=CN = registry.kurl.svc.cluster.local

Getting CA Private Key

secret/registry-pki created

/var/lib/kurl /home/admin_user

waiting for the registry to start

configmap/kurl-current-config patched

⚙ Addon prometheus 0.63.0-45.7.1

...

⚙ Addon kotsadm 1.97.0

secret/kubelet-client-cert created

Retrieving app metadata: url=https://replicated.app, slug=flow-enterprise

...

⚙ Persisting the kurl installer spec

configmap/kurl-config created

Rook Post-init: Installing Prometheus ServiceMonitor and Ceph Grafana Dashboard

configmap/ceph-cluster-dashboard created

servicemonitor.monitoring.coreos.com/rook-ceph-servicemonitor created

Scaling up EKCO deployment to 1 replica

deployment.apps/ekc-operator scaled

No resources found

Installation

Complete ✓

The UIs of Prometheus, Grafana and Alertmanager have been exposed on NodePorts 30900, 30902 and 30903 respectively.

To access Grafana use the generated user:password of admin:xxxxxxxx .

Kotsadm: `http://192.168.0.80:8800`

Login with password (will not be shown again): xxxxxxxx

This password has been set for you by default. It is recommended that you change this password; this can be done with the following command: `kubectl kots reset-password default`

To access the cluster with kubectl:

```
bash -l
```

Kurl uses `/etc/kubernetes/admin.conf`, you might want to copy kubeconfig to your home directory:

```
cp /etc/kubernetes/admin.conf ~/.kube/config
```

```
chown -R 150028 ~/.kube
```



```
echo unset KUBECONFIG >> ~/.bash_profile
```

```
bash -l
```

You will likely need to use sudo to copy and chown /etc/kubernetes/admin.conf.

[INFO] Loading environment

Flag --short has been deprecated, and will be removed in the future. The --short output will become the default.

[INFO] No existing ceph raw disks detected

Kubernetes connection credentials for worker node. Expires in 24 hours

```
Kubernetes Connection String : kubernetes-master-address=192.168.0.80:6443 kubeadm-  
token=l9rwpr.pu0ptqhzroko2u2g kubeadm-token-ca-  
hash=sha256:490bf2f2a73ef826e509837433dcc1b3243ed14a15f65d5c8b9b58af4a263ba4 kubernetes-  
version=1.25.8 docker-registry-ip=10.96.2.129
```

You may add additional command line options to the flow-tools join command.

Run ./flow-tools join --help for all available flags and options like [-a|-f|-k|-n|--proxy] etc.

Node join command for this cluster is below:

```
sudo ./flow-tools join kubernetes-master-address=192.168.0.80:6443 kubeadm-  
token=l9rwpr.pu0ptqhzroko2u2g kubeadm-token-ca-  
hash=sha256:490bf2f2a73ef826e509837433dcc1b3243ed14a15f65d5c8b9b58af4a263ba4 kubernetes-  
version=1.25.8 docker-registry-ip=10.96.2.129
```

node/primary-node labeled

[INFO] Primary node has been labelled with

```
gui=true
```

```
worker=true
```

If adding an additional node, please run the following,

after adding a worker node:

```
kubectl label nodes worker- --selector='node-role.kubernetes.io/master'
```

```
kubectl label nodes worker= --selector='!node-role.kubernetes.io/master'
```

```
[]
```

```
· Reset the admin console password for default
```

```
Enter a new password for the admin console (6+ characters): .....
```

```
· The admin console password has been reset
```

```
[INFO] Setting up kubectl command for current user
```

```
[INFO] Processing home directory: /home/admin_user
```

```
[INFO] Setting up kube-config for user: admin_user
```

On a single-node cluster, the installation labels the primary node appropriately by default. However, in a multi-node installation, label the worker nodes as directed from the installation command's output as seen above.

Reset the console password by running `kubectl kots reset-password -n default` if desired.

Next, install Flow Enterprise on each of the worker nodes using the join command output at the end of the installation of the primary node. Read more about [joining a node to the cluster](#).

[back to top](#)

Configure the KOTS app

Configure TLS

1. Open your Chrome browser and go to the URL provided at the end of the installation on the primary node. It will look like `http://<ip address of server>:8800`.

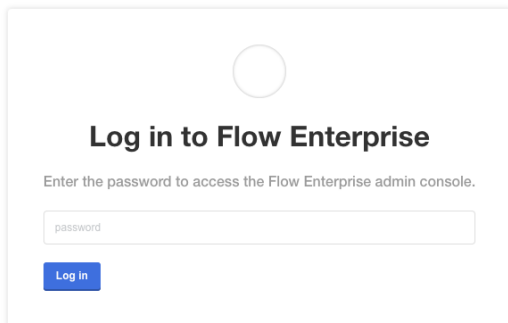
Note: You'll receive an error saying your connection is not private. Flow uses a self-signed SSL/TLS Certificate to secure the communication between your local machine and the admin console during setup. This is secure.

2. Click **Advanced**, then click **Proceed** to continue to the admin console.
3. On the next screen, enter the following information from the [Before you begin](#) steps:
 1. Enter your hostname
 2. Upload your key and certificate
4. Click **Upload & Continue**

Log in to the KOTS admin console

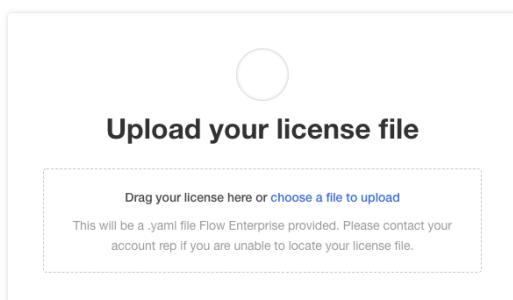
Note: The browser should redirect to the main KOTS admin console and prompt you to login. If not, navigate to `https://[hostname]:8800`.

1. Enter the password provided during installation at the end of the installation script output.



The screenshot shows a login page with a circular logo at the top. Below the logo is the heading "Log in to Flow Enterprise". Underneath, it says "Enter the password to access the Flow Enterprise admin console." There is a text input field with the placeholder text "password". At the bottom left of the form is a blue "Log in" button.

2. Upload the license file provided by your Flow representative.



The screenshot shows a page with a circular logo at the top. Below the logo is the heading "Upload your license file". Underneath, there is a dashed box containing the text: "Drag your license here or [choose a file to upload](#)". Below this is a smaller line of text: "This will be a .yaml file Flow Enterprise provided. Please contact your account rep if you are unable to locate your license file."

3. On the following screen, there are two options for installing Flow. If your server doesn't have internet access, choose the airgapped environment option and upload the pre-downloaded airgapped binary zip file. Otherwise, click the **download from the internet** link to continue.

Set up Flow configurations

Once the license is validated, the configuration screen appears.

The following configuration changes are recommended:

- Match the Flow URL to the fully qualified domain name registered earlier.
- Provide a valid email address to receive email alerts.
- Click **Email settings** and provide the details of the SMTP/Email server port, credentials, protocol, and other details.
- If there's a proxy involved, check **Use an outbound HTTP/HTTPS Proxy** and add the details.
- Enter the details for your external Postgres database.

Once this page is configured, scroll to the bottom of the page and click **Save config**.

Deploy Flow

To deploy Flow, on the next screen click **Deploy**.

Confirm the pop-up window by clicking **Yes, deploy**.

To monitor the deployment process, run `kubectl get pods` on the primary node to see each pod's status. Wait until all pods are in a running state.

Once all pods are running, start Flow. Then proceed to log in to the Flow Enterprise Application using the original URL for the application.

Additional configurations after installing Flow

- To prevent your Flow Enterprise Server cluster from running into a forced eviction due to a lack of disk space, go to Disk Pressure Check Settings and click the checkbox next to Enable Disk Pressure Check and Scale Down.

Disk Pressure Check Settings

This section contains settings that are used to control cluster disk space and automatic scale down.

Enable Disk Pressure Check and Scale Down
Optional feature to enable automatic scaledown on disk pressure to prevent volume corruption

Minimum Disk Free Percentage to avoid auto scale down Required
Minimum amount of total disk space on cluster available before a shutdown is initiated. Integer value between 10-90.

20
Default value: 20

Cronjob schedule to check for disk pressure on cluster Required
Schedule to run an utility pod to run the disk pressure scaledown check

*/*30 * * * *
Default value: */30 * * * *

Beta Features

Enable Beta Features

Save config

- [Enable monitoring services.](#)
- [Automatic certificate renewal.](#)

Log in to Flow

Navigate to your Flow URL. Enter your organization name and primary email. Choose a password to log in.

Connect with your Pluralsight Flow representative to talk about next steps for your onboarding experience. For steps to get started with Flow, check out our [Getting Started guide](#).

If you need help, please contact [Pluralsight Support](#).