

install

```
ubuntu@k0s-ctrl:~$ curl --proto '=https' --tlsv1.2 -sSf https://get.k0s.sh |
sudo sh
ubuntu@k0s-ctrl:~$ k0s version
v1.31.1+k0s.1
```

Crea il servizio systemd:

```
ubuntu@k0s-ctrl:~$ sudo k0s install
```

Bootstrap controller

Bootstrap del nodo controller (non single mode)

```
ubuntu@k0s-ctrl:~$ sudo mkdir -p /etc/k0s
ubuntu@k0s-ctrl:~$ k0s config create | sudo tee /etc/k0s/k0s.yaml
ubuntu@k0s-ctrl:~$ sudo k0s install controller -c /etc/k0s/k0s.yaml
ubuntu@k0s-ctrl:~$ sudo systemctl start k0scontroller.service
```

Verifica:

```
ubuntu@k0s-ctrl:~$ sudo k0s status
Version: v1.31.1+k0s.1
Process ID: 7885
Role: controller
Workloads: false
SingleNode: false
```

Ambiente di lavoro

Per rendere più comoda la gestione:

```
ubuntu@k0s-ctrl:~$ mkdir -p .config/k0s
ubuntu@k0s-ctrl:~$ sudo k0s kubeconfig admin | tee ~/.config/k0s/admin.conf
ubuntu@k0s-ctrl:~$ echo "alias kubectl='k0s kubectl --
kubeconfig=\"$HOME\"/.config/k0s/admin.conf'" >> ~/.bash_aliases
```

creazione worker

Sul controller creare un token (si può redirigere su file)

```
ubuntu@k0s-ctrl:~$ sudo k0s token create --role=worker --expiry=100h
```

Sul candidato nodo installare k0s con curl (vedi sopra) e copiare il token generato nel controller in un file (es `k8sworkertoken`)

```
ubuntu@a6:~$ sudo k0s install worker --token-file ./k8s_worker_token
ubuntu@a6:~$ sudo k0s start
</code>
```

Verificare il successo dell'operazione sul controller

```
<code>
ubuntu@k0s-ctrl:~$ kubectl get node
NAME     STATUS    ROLES    AGE     VERSION
a6       Ready    <none>   4m6s    v1.31.1+k0s
```

k0s incorpora kubectl, che fa prefissato da k0s. Esempio:

```
ubuntu@k0s-ctrl:~$ sudo k0s kubectl cluster-info
```

From:
<https://wiki.csgalileo.org/> - **Galileo Labs**

Permanent link:
<https://wiki.csgalileo.org/tips/k0s>

Last update: **2024/11/06 10:24**

