

K9s installation.

What is K9s in Kubernetes?

K9s is a terminal-based UI for managing Kubernetes clusters efficiently. It provides a real-time view of your cluster resources and allows you to interact with Kubernetes workloads using simple keyboard shortcuts.

Why Use K9s?

1. **Faster Cluster Management** – No need to type long kubectl commands.
2. **Real-time Monitoring** – Watch pods, deployments, logs, and events in real time.
3. **Better Navigation** – Quickly switch between different Kubernetes resources.
4. **Efficient Debugging** – Stream logs, exec into containers, and manage pods easily.
5. **Lightweight & CLI-based** – No need for a heavy GUI like Lens.

Steps to install:

```
curl -sS https://webinstall.dev/k9s | bash
```

```
controlplane $ curl -sS https://webinstall.dev/k9s | bash

>>> Welcome to Webi! - modern tools, instant installs. <<<
We expect your experience to be absolutely perfect!

Success? Star it!   https://github.com/webinstall/webi-installers
Problem? Report it: https://github.com/webinstall/webi-installers/issues
                    (your system is GNU/Linux/x86_64 with libc & curl+wget)

Bootstrapping Webi
  Downloading https://webinstall.dev/packages/webi/webi.sh
    to ~/.local/bin/webi
  Running ~/.local/bin/webi k9s@stable

Installing k9s ...
  Found ~/.local/bin
  Initializing ~/.config/envman/
  Edit ~/.profile to source ~/.config/envman/load.sh
  Edit ~/.bashrc to source ~/.config/envman/load.sh
  Downloading k9s from
    https://github.com/derailed/k9s/releases/download/v0.32.7/k9s_linux_amd64.tar.gz
  Saved as ~/Downloads/webi/k9s/0.32.7/k9s_linux_amd64.tar.gz
  Extracting ~/Downloads/webi/k9s/0.32.7/k9s_linux_amd64.tar.gz
  Installing to ~/.local/opt/k9s-v0.32.7/bin/k9s

  Installed 'k9s v0.32.7' as ~/.local/bin/k9s

  Edit ~/.config/envman/PATH.env to add:
    ~/.local/bin

>>> ACTION REQUIRED <<<
Copy, paste & run the following command:
source ~/.config/envman/PATH.env
(newly opened terminal windows will update automatically)
```

```
controlplane $ echo 'export PATH=$HOME/.local/bin:$PATH' >> ~/.bashrc
```

```
controlplane $ source ~/.bashrc
```

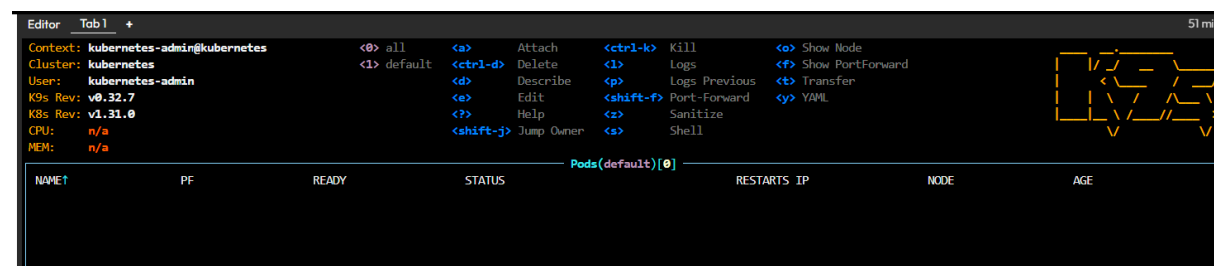
k9s version

```
controlplane $ echo 'export PATH=$HOME/.local/bin:$PATH' >> ~/.bashrc
controlplane $ source ~/.bashrc
controlplane $ k9s version

  ____  __  _  _ 
 / ___||  \| \| | |
| |___| |__| | | |
 \___ \|  __| | | |
  ___/ | |__| | | |
 |___|_|_|_|_|_|_|

Version:      v0.32.7
Commit:      6b5d24f5741a1789fb97ba3e11f0ee868d93459d
Date:        2024-11-16T20:22:28Z
controlplane $
```

K9s



Now I create an deployment and svc for sample.

```
apiVersion: v1
kind: Service
metadata:
  name: nginx-service
spec:
  selector:
    app: nginx
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
  type: LoadBalancer
```

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:latest
          ports:
            - containerPort: 80

```

```

controlplane $ vi deploy.yaml
controlplane $ vi svc.yaml
controlplane $ k create -f deploy.yaml
deployment.apps/nginx-deployment created
controlplane $ k create -f svc.yaml
service/nginx-service created
controlplane $

```

Now I view in manually

```

controlplane $ k get pod,svc,deployment,node -o wide

```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED	NODE	READINESS	GATES
pod/nginx-deployment-54b9c68f67-2dfw5	1/1	Running	0	5m50s	192.168.0.4	controlplane	<none>		<none>	
pod/nginx-deployment-54b9c68f67-jsb9b	1/1	Running	0	5m50s	192.168.1.4	node01	<none>		<none>	
pod/nginx-deployment-54b9c68f67-s4mzf	1/1	Running	0	5m51s	192.168.1.5	node01	<none>		<none>	

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE	SELECTOR
service/kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	42h	<none>
service/nginx-service	LoadBalancer	10.109.74.0	<pending>	80:31268/TCP	5m43s	app=nginx

NAME	READY	UP-TO-DATE	AVAILABLE	AGE	CONTAINERS	IMAGES	SELECTOR
deployment.apps/nginx-deployment	3/3	3	3	5m51s	nginx	nginx:latest	app=nginx

NAME	STATUS	ROLES	AGE	VERSION	INTERNAL-IP	EXTERNAL-IP	OS-IMAGE	KERNEL-VERSION	CONTAINER-RUNTIME
node/controlplane	Ready	control-plane	42h	v1.31.0	172.30.1.2	<none>	Ubuntu 20.04.5 LTS	5.4.0-131-generic	containerd://1.7.22
node/node01	Ready	<none>	41h	v1.31.0	172.30.2.2	<none>	Ubuntu 20.04.5 LTS	5.4.0-131-generic	containerd://1.7.22

```

controlplane $

```

Now I view in k9s

```

Editor  Tab 1  +
Context: kubernetes-admin@kubernetes
Cluster: kubernetes
User: kubernetes-admin
K9s Rev: v0.32.7
K8s Rev: v1.31.0
CPU: n/a
MEM: n/a

<0> all      <a> Attach      <ctrl-k> Kill      <o> Show Node
<1> default  <ctrl-d> Delete   <l> Logs       <f> Show PortForward
               <d> Describe  <p> Logs Previous <t> Transfer
               <e> Edit     <shift-f> Port-Forward <y> YAML
               <?> Help    <z> Sanitize
               <shift-j> Jump Owner <s> Shell

Pod(s)(default)[3]
NAME      PF  READY  STATUS    RESTARTS  IP          NODE        AGE
nginx-deployment-54b9c68f67-2dfu5  •    1/1    Running    0          192.168.0.4  controlplane 6m55s
nginx-deployment-54b9c68f67-jsb9b  •    1/1    Running    0          192.168.1.4  node01        6m55s
nginx-deployment-54b9c68f67-s4mzf  •    1/1    Running    0          192.168.1.5  node01        6m56s

```

```

Context: kubernetes-admin@kubernetes
Cluster: kubernetes
User: kubernetes-admin
K9s Rev: v0.32.7
K8s Rev: v1.31.0
CPU: n/a
MEM: n/a

<0> all      <b> Bench Run/Stop <l> Logs
<1> default  <ctrl-d> Delete   <p> Logs Previous
               <d> Describe  <shift-f> Port-Forward
               <e> Edit     <f> Show PortForward
               <?> Help    <y> YAML
               <shift-j> Jump Owner

Service(s)(default)[2]
NAME      TYPE        CLUSTER-IP    EXTERNAL-IP    PORTS    AGE
kubernetes ClusterIP    10.96.0.1      <pending>      https:443->0 42h
nginx-service LoadBalancer 10.109.74.0    <pending>      80->31268    8m17s

```

Summary of K9s Key Shortcuts

Key Action

- p** Show pods
- d** Show deployments
- e** Edit YAML
- r** Restart pod/deployment
- x** Delete pod/deployment
- l** View logs
- s** Open shell inside pod