How to access/expose kubernetes-dashboard service outside of a cluster?

Asked 8 years, 6 months ago   Modified 1 year, 7 months ago   Viewed 155k times

▲

62

▼

🔖

🕓

I have got the following services:

```
ubuntu@master:~$ kubectl get services --all-namespaces
NAMESPACE      NAME                     CLUSTER-IP      EXTERNAL-IP   PORT(S)
AGE
default        kubernetes               100.64.0.1      <none>        443/TCP
48m
kube-system    kube-dns                 100.64.0.10     <none>
53/UDP,53/TCP    47m
kube-system    kubernetes-dashboard     100.70.83.136   <nodes>       80/TCP
47m
```

I am attempting to access kubernetes dashboard. The following response seems reasonable, taking into account curl is not a browser.

```
ubuntu@master:~$ curl 100.70.83.136
 <!doctype html> <html ng-app="kubernetesDashboard"> <head> <meta charset="utf-
8"> <title>Kubernetes Dashboard</title> <link rel="icon" type="image/png"
href="assets/images/kubernetes-logo.png"> <meta name="viewport"
content="width=device-width"> <link rel="stylesheet"
href="static/vendor.36bb79bb.css"> <link rel="stylesheet"
href="static/app.d2318302.css"> </head> <body> <!--[if lt IE 10]>
    <p class="browsehappy">You are using an <strong>outdated</strong>
browser.
    Please <a href="http://browsehappy.com/">upgrade your browser</a> to
improve your
    experience.</p>
  <![endif]--> <kd-chrome layout="column" layout-fill> </kd-chrome> <script
src="static/vendor.633c6c7a.js"></script> <script src="api/appConfig.json">
</script> <script src="static/app.9ed974b1.js"></script> </body> </html>
```

According to the documentation the right access point is https://localhost/ui. So, I am trying it and receive a bit worrying result. **Is it expected response?**

```
ubuntu@master:~$ curl https://localhost/ui
curl: (60) server certificate verification failed. CAfile: /etc/ssl/certs/ca-
certificates.crt CRLfile: none
More details here: http://curl.haxx.se/docs/sslcerts.html

curl performs SSL certificate verification by default, using a "bundle"
 of Certificate Authority (CA) public keys (CA certs). If the default
 bundle file isn't adequate, you can specify an alternate file
 using the --cacert option.
If this HTTPS server uses a certificate signed by a CA represented in
 the bundle, the certificate verification probably failed due to a
 problem with the certificate (it might be expired, or the name might
```

```
  not match the domain name in the URL).
 If you'd like to turn off curl's verification of the certificate, use
  the -k (or --insecure) option.
```

Trying the same without certificate validation. For curl it might be OK. but I have got the same in a browser, which is connecting though port forwarding via vagrant forwarded_port option.

```
ubuntu@master:~$ curl -k https://localhost/ui
Unauthorized
```

What I am doing wrong? and how to make sure I can access the UI? Currently it responds with Unauthorized.

The docs for the dashboard tell the password is in the configuration:

```
ubuntu@master:~$ kubectl config view
apiVersion: v1
clusters: []
contexts: []
current-context: ""
kind: Config
preferences: {}
users: []
```

but it seems I have got nothing... **Is it expected behavior? How can I authorize with the UI?**

kubernetes

Share Improve this question Follow

asked Oct 5, 2016 at 2:07

Andrew
2,133 ● 2 ● 21 ● 30

Have you seen stackoverflow.com/q/34306082/759019 ? – Anirudh Ramanathan Oct 5, 2016 at 6:53

3 I have seen similar. Everywhere I checked either there is no practical answer (instructions to act) or it is not compatible with 1.4.0 kubernetes or I am missing some basic things. – Andrew Oct 5, 2016 at 10:48

12 Answers

Sorted by: Highest score (default) ⇕

▲

The offical wiki is a little bit confusing so I reordered it here:

51

If you use the [recommended](#) yaml to deploy the dashboard, you should only access your dashboard by https, and you should generate your certs, refer to [guide](#). Then you can run

```
kubectl proxy --address='0.0.0.0' --accept-hosts='^*$'
```
to visit the dashboard on

"http://localhost:8001/ui". This page needs to use a token to login. To generate it, refer to this page. Also you can add `NodePort` to your yaml and access it using `<nodeip>:<port>`.

If you deploy using the http alternative method, you can **only access your dashboard by nodeip:port**. Remember to add it to yaml first!! After deployment, you should also generate your token and **add header** `Authorization: Bearer <token>` **for every request**.

I think this can help you and others who want to use kube-dashboard.

Share   Improve this answer   Follow

edited Nov 13, 2020 at 5:29   answered Sep 5, 2018 at 3:15

mario.van.zadel   ysjiang
2,949 ● 15 ● 24   **661** ● 6 ● 11

The alternative YAML URL was changed: github.com/kubernetes/dashboard/blob/master/aio/deploy/…
– Gabor Garami Aug 30, 2020 at 10:57

Thanks, I was trying to access Dashboard outside the cluster and this worked for me! – user3431976 Jul 20, 2023 at 11:11

32

You can reference the document:

https://github.com/kubernetes/dashboard/blob/master/docs/user/accessing-dashboard/README.md

The easy way is to

```
$ kubectl -n kube-system edit service kubernetes-dashboard
```

change the `.spec.type` to `NodePort`

Share   Improve this answer   Follow

edited Jun 7, 2021 at 6:04   answered Feb 22, 2018 at 3:35

Martin Bories   Yang Young
1,116 ● 1 ● 18 ● 37   **622** ● 5 ● 6

19

Have you tried this:

```
kubectl port-forward -n kubernetes-dashboard service/kubernetes-dashboard 10443:443 --address 0.0.0.0
```

In your case the namespace is different (`kube-system`), so:

```
kubectl port-forward -n kube-system service/kubernetes-dashboard 10443:443 --address 0.0.0.0
```

Now you should be able to access the dashboard at `localhost:10443` (including remotely using a `ssh` tunnel with the `-L 10433:localhost:10433` parameter).

Share   Improve this answer   Follow

edited Mar 6, 2023 at 18:57

mirekphd
6,971 ● 3 ● 56 ● 83

answered Nov 25, 2020 at 11:16

AlexW
191 ● 1 ● 2

You can reach by hitting the nodePort for the dashboard on the master

12

```
kubectl describe services kubernetes-dashboard --namespace=kube-system
NodePort:        <unset> 30042/TCP
```

http://MASTER:30042

Share   Improve this answer   Follow

answered Oct 14, 2016 at 18:17

user2363318
1,067 ● 1 ● 12 ● 31

▲

11

▼

🔖

🕘

You need to run kubectl proxy locally for accessing the dashboard outside the kubernetes cluster. This is because of the authentication mechanism. After running the below command you'll be able to view the dashboard at http://localhost/ui on your browser. The admin.conf file is the file on your kubernetes master at /etc/kubernetes/admin.conf You'll have to scp that file to the machine from which you want to access the dashboard and pass it to kubectl command.

> kubectl --kubeconfig=./admin.conf proxy -p 80

The nodePort approach as mentioned by @user2363318 will be applicable if one of the following two conditions is valid:

1. Your http client (browser or curl) is able to send auth tokens

2. Your service in the kubernetes cluster does not have auth

Share Improve this answer Follow

answered Dec 21, 2016 at 21:04

livinston
1,358 ● 2 ● 13 ● 19

---

3 I get `F0212 08:37:08.165120   20582 proxy.go:153] listen tcp 127.0.0.1:80: bind: permission denied` running your command. Also I don't have amin.conf neither in `/etc/kubernetes` nor in home folder – vladkras Feb 12, 2018 at 5:41

Coming in to this late, but you'll likely need to run that as root, since you can't bind to ports under 1024 as an unprivileged user. – Mikey T.K. Dec 17, 2019 at 22:20

▲

6

▼

🔖

🕘

Using terminal app on my laptop did the following:

```
sudo ssh  -i "MYAMOZONHOSTKEYPAIR.pem" -L 8001:127.0.0.1:8001
ubuntu@MYAMAZONHOST
```

MYAMAZONHOST is where I am running k3s.

Upon SSH did the following:

```
sudo k3s kubectl proxy
```

Then using the browser opened the url http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/

Kubernetes dashboard opened successfully.

Share Improve this answer Follow   edited Aug 4, 2020 at 6:08   answered Aug 4, 2020 at 5:24

▲

4

▼

🔖

create a new service to expose port 8001 to kubernetes-dashboard's 8443.

```
kubectl expose deployment kubernetes-dashboard -n kubernetes-dashboard --
type=LoadBalancer --name=kdash --port=8001 --target-port=8443
```

now, you can access your kubernetes-dashboard with `https://<ip>:8001` .

Share   Improve this answer   Follow

answered Sep 16, 2021 at 5:39

song940
41 ● 3

▲

2

▼

🔖

🕑

incase anyone is facing this issue in 2023

my setup and goal is to simply access remotely via my workstation browser since my cluster and dashboard is in another server

in control-plane node :

```
kubectl port-forward -n kubernetes-dashboard service/kubernetes-dashboard
<TARGET_PORT>:443 --address 0.0.0.0
```

in your workstation :

```
ssh -v -N -L <TARGET_PORT>:127.0.0.1:8443 <USERNAME>@<ADDRESS>
```

where `<USERNAME>` is your remote server username and `<ADDRESS>` is your remote server ip address

then open browser in your workstation and type in `https://127.0.0.1:<TARGET_PORT>`

Share   Improve this answer   Follow

answered Sep 14, 2023 at 5:14

zer0
93 ● 1 ● 12

Just add a little clarity to the above response,

1

> Change type: ClusterIP to type: NodePort and save file.
>
> Dashboard has been exposed on port 31707 (HTTPS). Now you can access it from your browser at: https://master-ip:31707.

Note: The port mentioned could be difference in your case, just run the below kubectl command to get the port from master.

> kubectl -n kube-system get service kubernetes-dashboard

Share Improve this answer Follow

answered Sep 15, 2019 at 0:12

anrajme
905 ● 9 ● 7

1   "Just add a little clarity to the **above response**" is more deceiving than clear because the answers are
     ordered differently for each SO user! – dvdblk Feb 6, 2020 at 20:42

Apply those two yaml files, and you should be able to access it. This is assumed that you already
have a valid domain:

1

```
kubectl apply -f
```
```
https://raw.githubusercontent.com/kubernetes/dashboard/v2.4.0/aio/deploy/recommended.yaml
```

```
clusterissuer.yml
```

```
apiVersion: cert-manager.io/v1
kind: ClusterIssuer
metadata:
  name: letsencrypt  # cluster issuer name
spec:
  acme:
    server: https://acme-v02.api.letsencrypt.org/directory # production url
    email: abc@abc.com
    privateKeySecretRef:
      name: k8-dashboard-tls-production # secret name
    solvers:
    - http01:
        ingress:
          class: nginx
```

`ingress.yml`

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  annotations:
    cert-manager.io/cluster-issuer: letsencrypt
    nginx.ingress.kubernetes.io/backend-protocol: "HTTPS"
  name: dashboard-api-ingress
  namespace: kubernetes-dashboard
spec:
  ingressClassName: nginx
  rules:
    - host: example.com
      http:
        paths:
        - path: /
          pathType: Prefix
          backend:
            service:
              name: kubernetes-dashboard
              port:
                number: 443
    tls:
      - hosts:
          - example.com
        secretName: k8-dashboard-tls-production
```

Share Improve this answer Follow

answered Feb 15, 2022 at 18:18

Hoang Minh
1,262 ● 2 ● 24 ● 49

For exposing kube dashboard through nginx ingress

▲

1

▼

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: kubernetes-dashboard-ingress
  namespace: kubernetes-dashboard
  annotations:
    nginx.ingress.kubernetes.io/backend-protocol: HTTPS
    nginx.ingress.kubernetes.io/configuration-snippet: |-
      proxy_ssl_server_name on;
      proxy_ssl_name $host;
    nginx.ingress.kubernetes.io/ssl-redirect: "true"
    nginx.ingress.kubernetes.io/force-ssl-redirect: "true"
    nginx.ingress.kubernetes.io/add-base-url: "true"
    nginx.ingress.kubernetes.io/rewrite-target: /$1

  spec:
    ingressClassName: nginx
    rules:
      - host: your.domain.it
        http:
```

```
      paths:
      - path: /(.*)
        pathType: Prefix
        backend:
          service:
            name: kubernetes-dashboard
            port:
              number: 443
```

▶ Run code snippet   ⧉ Expand snippet

assuming the service kubernetes-dashboard on namespace kubernetes-dashboard has been configured on the port 443

Share   Improve this answer   Follow

answered Oct 14, 2022 at 18:35

Salvatore Napoli
444 ● 3 ● 9

The easiest way is to

▲

1

   kubectl proxy --port=8080 --address=0.0.0.0 --disable-filter=true &

▼

Then you can access it on **IP:8080** or you can access it using ALB/TG combination in case of AWS.

🔖

🕑

You can take the reference from document:

https://www.eksworkshop.com/beginner/040_dashboard/dashboard/

Share   Improve this answer   Follow

answered Feb 7, 2023 at 14:51

Suraj Singh Rathore
1,202 ● 10 ● 10

1   This worked for me with minikube version: v1.30.1 – jftuga Apr 14, 2023 at 20:57

Start asking to get answers

Find the answer to your question by asking.

Ask question

Explore related questions

kubernetes

See similar questions with these tags.