Kupboard Console

The Kupboard Console is a web-based console that allows you to use kupboard using a web browser. Users can use CLI commands of kupboard to build a cloud native environments and deploy a variety of packages and applications. The Kupboard Console allows you to execute various tasks through the web UI without running many CLI commands directly from the terminal.

How to run Kupboard Console

The Kupboard Console can be run in a localhost environment using the commands below, and you can access the console with `localhost:8080` through a browser.

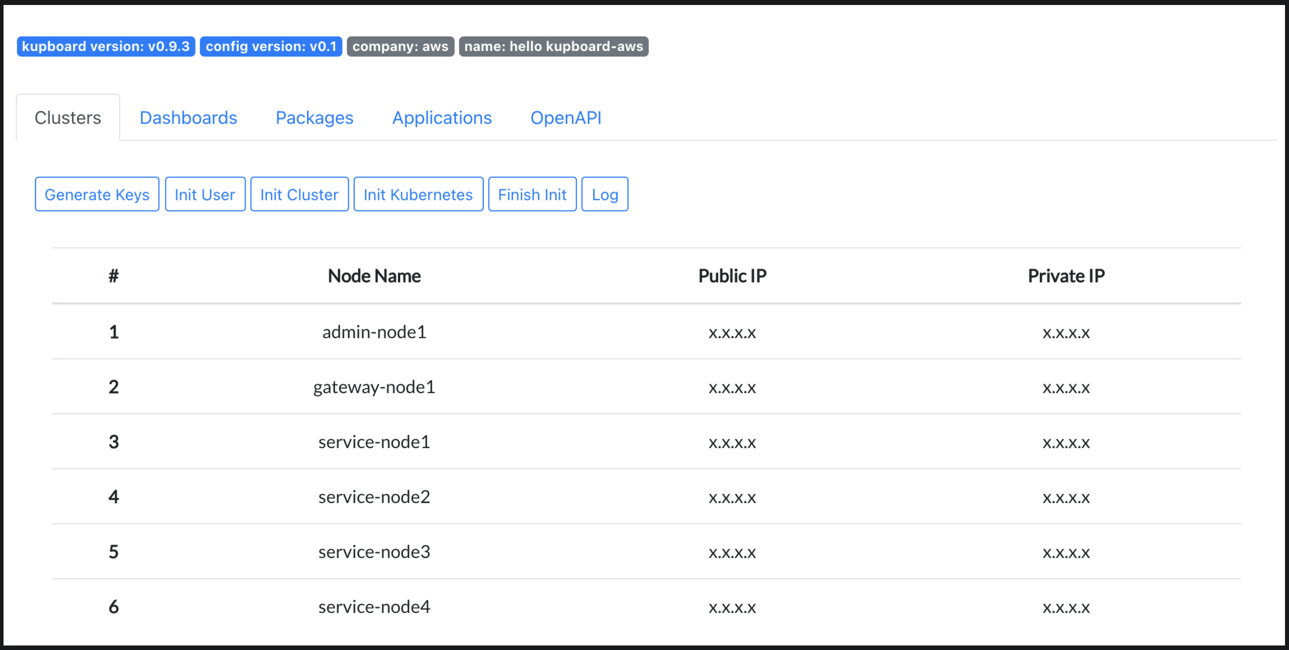
```
$ docker run -d -v $(pwd)/data:/kupboard/data -p 8080:8080
kupboard/kupboard console
```

NOTE

The Kupboard Console must be run in the same `localhost` environment as the browser.

Cluster

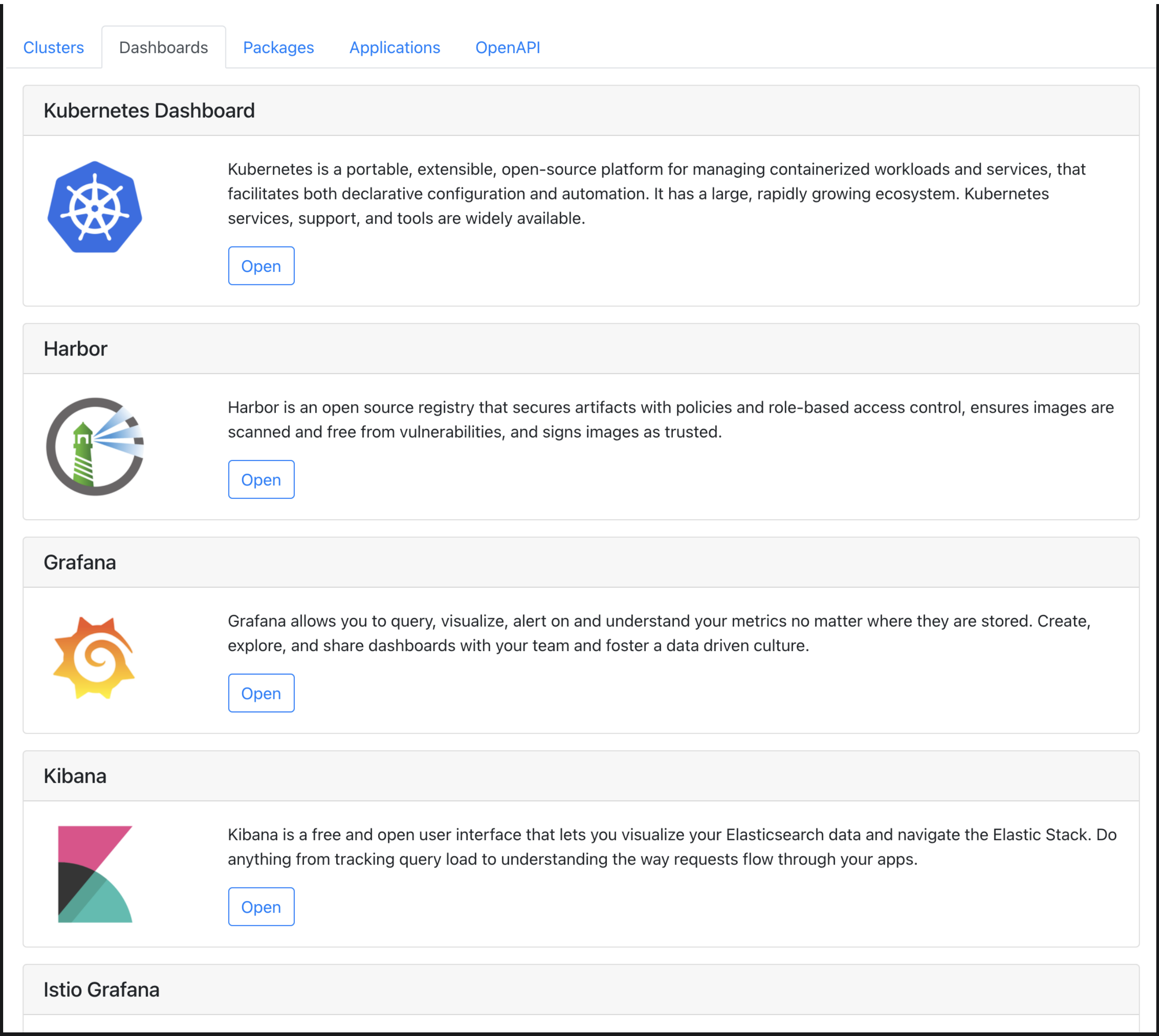
`Cluster` tab shows the cluster information defined in `kupboard.yaml`. At the top of the table are buttons for executing initialization commands. Users can perform initializations by clicking the buttons corresponding to the Setup commands without using cli commands for initialization.



#	Node Name	Public IP	Private IP
1	admin-node1	x.x.x.x	x.x.x.x
2	gateway-node1	x.x.x.x	x.x.x.x
3	service-node1	x.x.x.x	x.x.x.x
4	service-node2	x.x.x.x	x.x.x.x
5	service-node3	x.x.x.x	x.x.x.x
6	service-node4	x.x.x.x	x.x.x.x

Dashboards

`Dashboards` tab shows the dashboards of various solutions installed through default kollection, including Kubernetes Dashboard.



Kubernetes Dashboard

Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation. It has a large, rapidly growing ecosystem. Kubernetes services, support, and tools are widely available.

Harbor

Harbor is an open source registry that secures artifacts with policies and role-based access control, ensures images are scanned and free from vulnerabilities, and signs images as trusted.

Grafana

Grafana allows you to query, visualize, alert on and understand your metrics no matter where they are stored. Create, explore, and share dashboards with your team and foster a data driven culture.

Kibana

Kibana is a free and open user interface that lets you visualize your Elasticsearch data and navigate the Elastic Stack. Do anything from tracking query load to understanding the way requests flow through your apps.

Istio Grafana

To access each dashboard, you must first bind two ports `443` and `8091` when you run the Kupboard Console.

```
$ docker run -d -v $(pwd)/data:/kupboard/data -p 8080:8080 -p 443:443
-p 8091:8091 kupboard/kupboard console
```

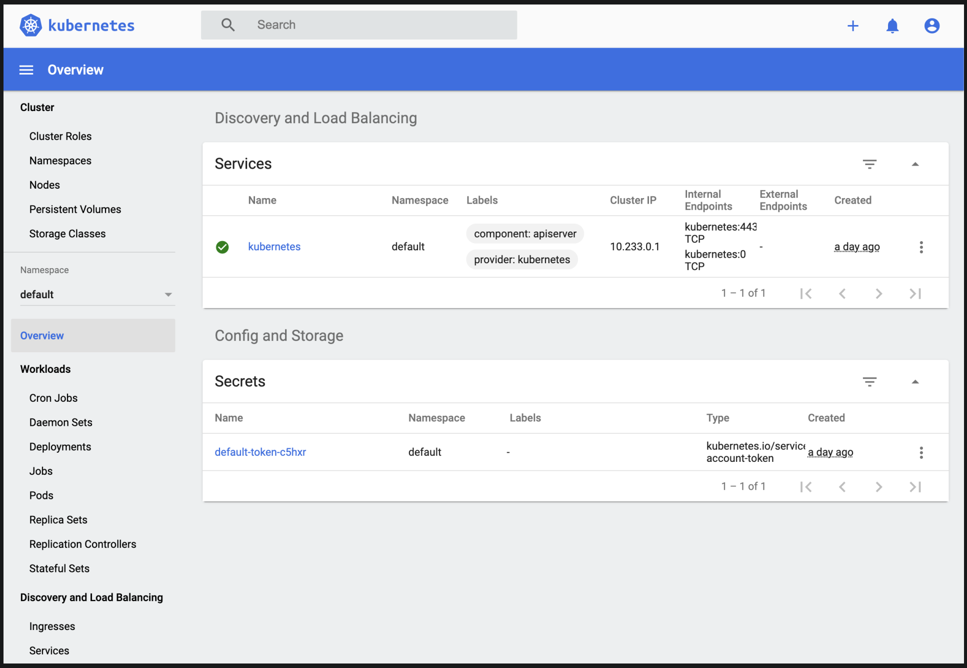
Then, you need to add the following domains on DNS system or edit `/etc/hosts` to add the following items. This is because each dashboard cannot be accessed directly over the external Internet and it's only allowed from the `443` and `8091` ports of the Kupboard Console.

```
127.0.0.1 grafana.mydomain.com
127.0.0.1 kibana.mydomain.com
127.0.0.1 istio-grafana.mydomain.com
127.0.0.1 istio-kiali.mydomain.com
127.0.0.1 istio-jaeger.mydomain.com
```

```
127.0.0.1 minio.mydomain.com
127.0.0.1 argocd.mydomain.com
127.0.0.1 kafka.mydomain.com
127.0.0.1 keycloak.mydomain.com
```

To access the dashboard, you must first install the `nginx-console` of the default kollection, and if you have trouble in accessing despite installing `nginx-console`, run the `connection` action of `dashboard proxy` and `console proxy` and then try it again.

Example: Kubernetes Dashboard



Discovery and Load Balancing

Services

	Name	Namespace	Labels	Cluster IP	Internal Endpoints	External Endpoints	Created
✓	kubernetes	default	component: apiserver provider: kubernetes	10.233.0.1	kubernetes:443 TCP kubernetes:0 TCP	-	a day ago

1 – 1 of 1

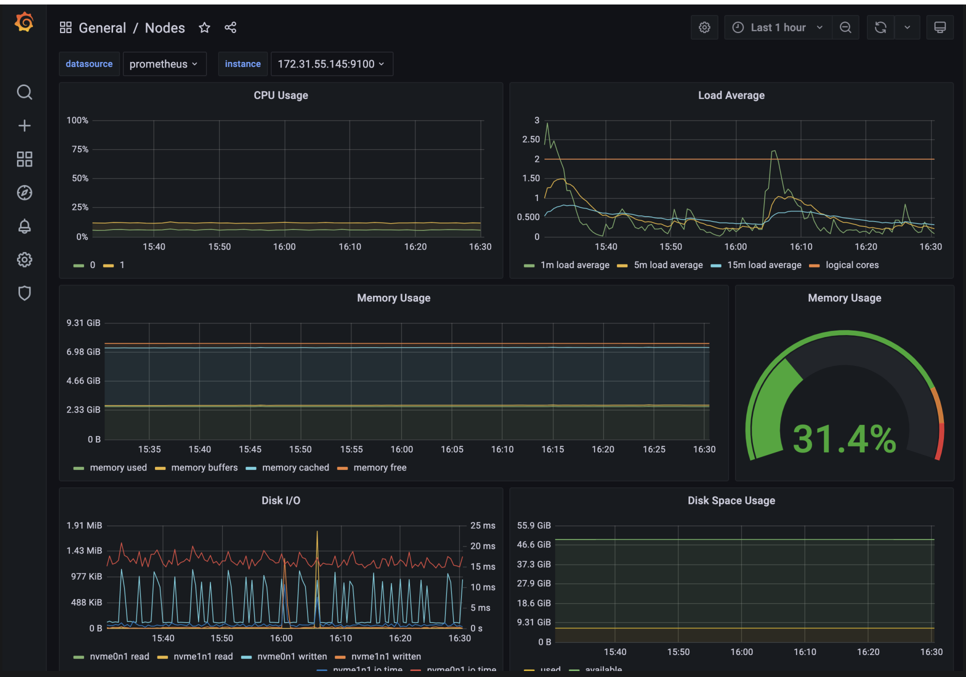
Config and Storage

Secrets

Name	Namespace	Labels	Type	Created
default-token-c5hxr	default	-	kubernetes.io/service-account-token	a day ago

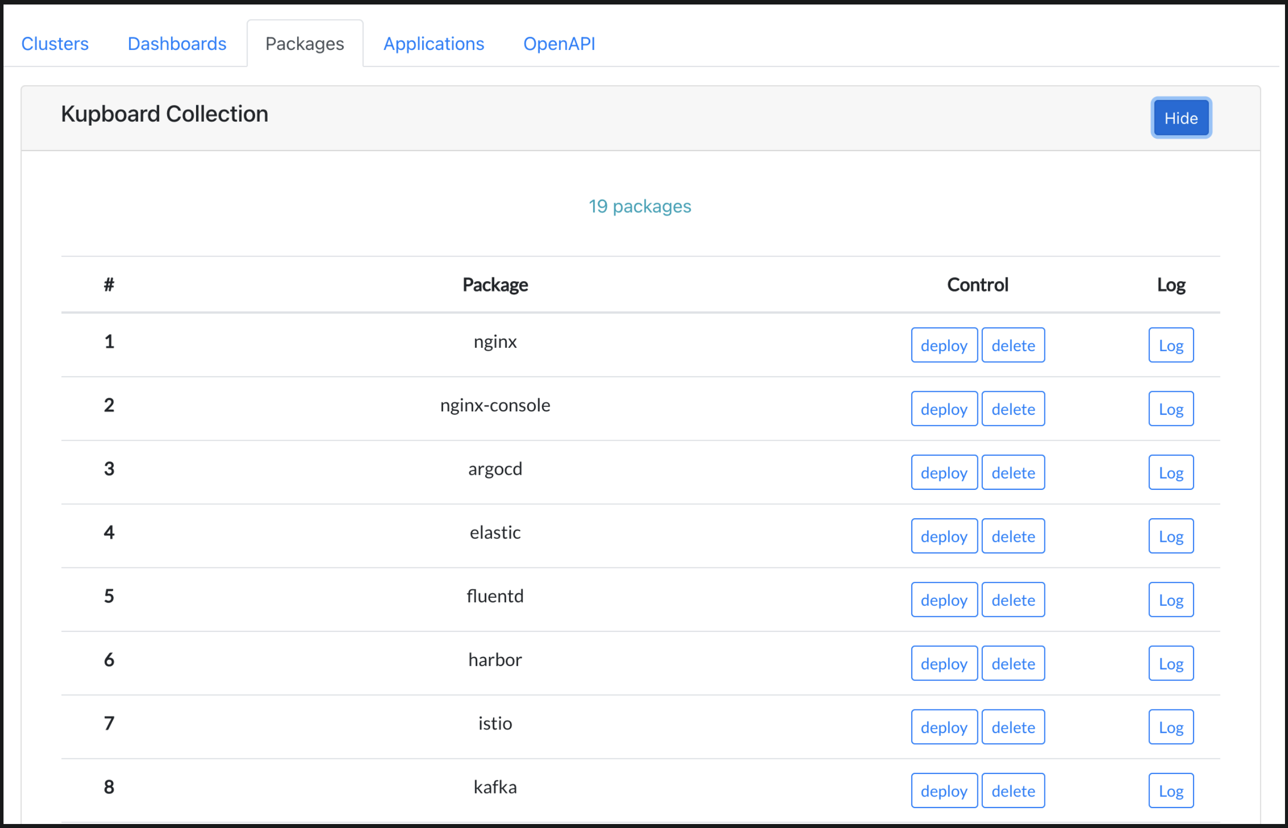
1 – 1 of 1

Example: Grafana



Packages

`Packages` tab lists the package of kollections added to the `data/kollection` as well as the package of the default kollection, and can execute the actions provided by each package.



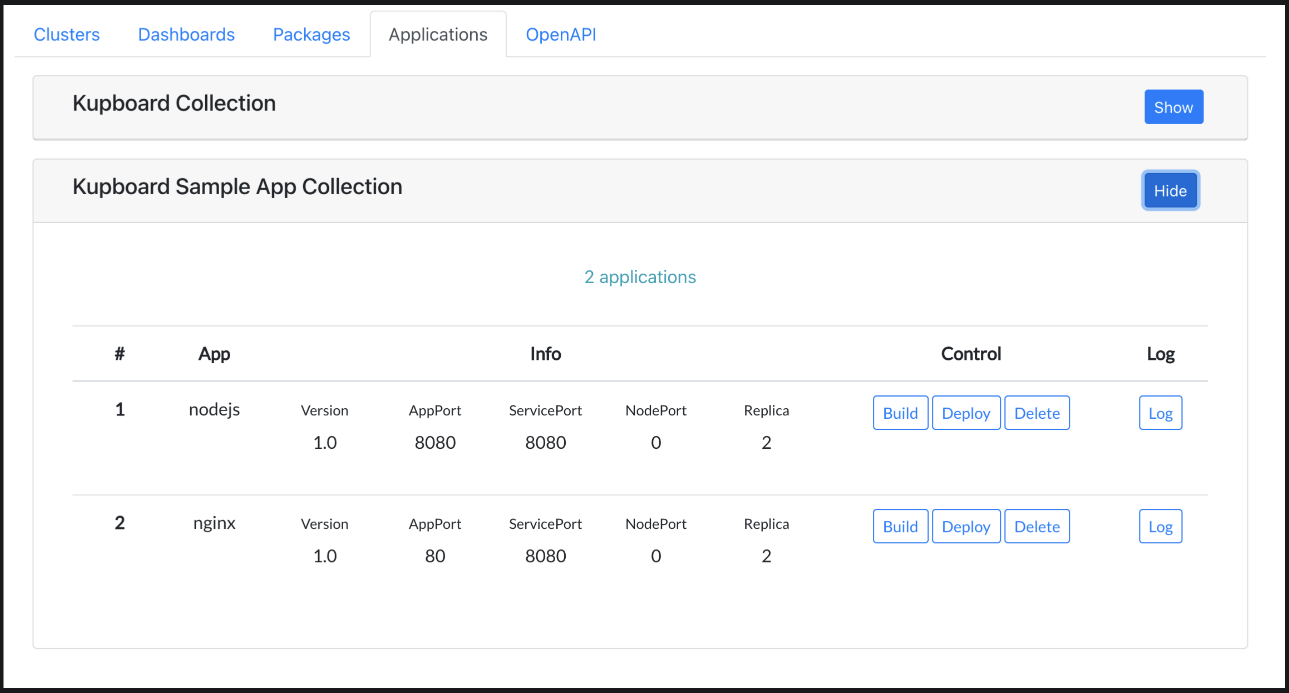
Kupboard Collection

19 packages

#	Package	Control	Log
1	nginx	deploy delete	Log
2	nginx-console	deploy delete	Log
3	argocd	deploy delete	Log
4	elastic	deploy delete	Log
5	fluentd	deploy delete	Log
6	harbor	deploy delete	Log
7	istio	deploy delete	Log
8	kafka	deploy delete	Log

Applications

`Applications` tab lists the applications that are included in the kollection added to the `data/kollection`, and allow users execute commands to build/apply/delete application.



Kupboard Collection

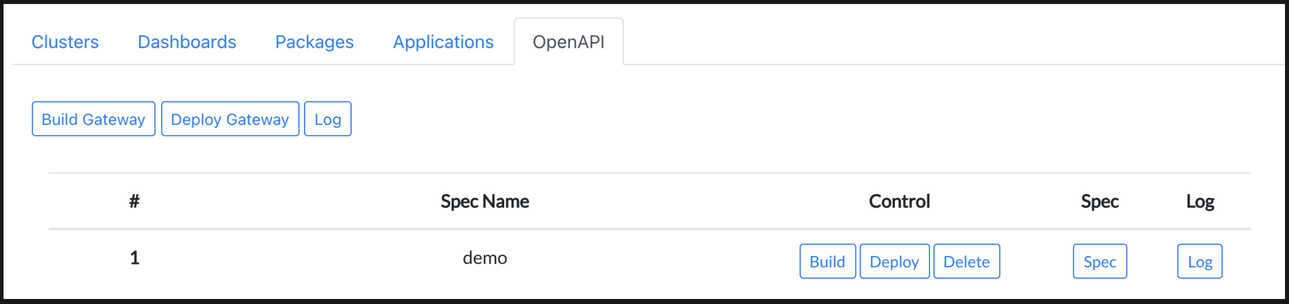
Kupboard Sample App Collection

2 applications

#	App	Info					Control	Log
1	nodejs	Version 1.0	AppPort 8080	ServicePort 8080	NodePort 0	Replica 2	Build Deploy Delete	Log
2	nginx	Version 1.0	AppPort 80	ServicePort 8080	NodePort 0	Replica 2	Build Deploy Delete	Log

OpenAPI

`OpenAPI` tab lists the openapi specifications included in the `data/api` and allows build/deploy/delete just like the application.



#	Spec Name	Control	Spec	Log
1	demo	Build Deploy Delete	Spec	Log

Edit this page