

Configuration

The configuration file of the Kupboard is created in the `yaml` format. The configuration file contains a variety of information, including versions, meta-information, global variables, and the entire cluster configuration.

NOTE

The default name of the configuration file is `kupboard.yaml`. This can be created by a project or service. If you use a name other than `kupboard.yaml`, you must specify the name of the configuration file (with the extension omitted), such as `--config <config-name>` when using the kupboard commands.

Basic Structure

Basically the configuration has 4 sections `version`, `metadata`, `vars` and `cluster`.

```
version: v0.1

metadata:
  name: myproject
  company: mycompany
  project: myproject

vars:
  ## Project Information
  ...

  ## Secrets
  ...

  ## User-defined Vars
  ...

cluster:
  admin:
    ...

  gateway:
```

```
...  
service:  
...  
storage:  
...  
elastic:  
...
```

Version

`version` can be a configuration version. It can be also a version of project or deployment.

```
version: v0.1
```

Metadata

`metadata` must have `name` and `company`. You can also add meta information that you want.

```
metadata:  
  name: hello kupboard  
  company: mycompany  
  project: myproject  
  name1: value1  
  name2: value2
```

- `name` a name of project or service
- `company` a company name
- `project` a project name

Global Variables

In the `vars` section, you can define many variables that you can refer to by `kupboard` and `kollection`.

```
## Global Variables
##
vars:
  name1: value1
  name2: value2
```

Project Information

The variables in `Project Information` are required for the service operation. To use the Harbor registry, a valid ssl certification should be prepared. If you're not using the registry, you can ignore these variables.

```
## Project Information
##
kbd_service_domain: mycompany.com
kube_enable_rook: true
kube_namespace: myproject
kube_registry_username: kupboard
kube_registry_password: Kupboard1234
kube_registry_email: username@email.com
kube_registry_domain: harbor.mycompany.com
kube_registry_url: harbor.mycompany.com/myproject
```

NOTE

`kube_registry_password` should be 8-20 characters long with at least 1 uppercase, 1 lowercase and 1 number. Check out [Harbor Package](#) for more information.

Secrets

`Secrets` variables are used to access various open source packages. If you're not using open source packages, you can ignore these variables.

```
## Secrets
##
grafana_admin_password: kupboard
harbor_admin_password: kupboard
```

```
argocd_admin_password: kupboard
mysql_root_password: kupboardroot
mysql_user_password: kupboard1234
mongodb_root_password: kupboardroot
redis_root_password: kupboardroot
influxdb_root_password: kupboardroot
minio_access_key: kupboard
minio_secret_key: kupboard1234
keycloak_admin_password: kupboard
```

Cluster

The configuration of the cluster can be customized as you want. If you want to configure it for test or demo, you can configure it with six servers as above, but you can configure three gateway servers and four or more servers in the service cluster to install a variety of solutions. If you want to build a separated Data plane, you can configure an additional storage cluster. However, the admin cluster must have 1 admin server and the cluster names cannot change.

```
## Cluster Definition
##
cluster:
  admin:
    - name: admin-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  gateway:
    - name: gateway-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  service:
    - name: service-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node2
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node3
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node4
```

```
    public_ip: x.x.x.x
    private_ip: x.x.x.x

elastic:
  - name: elastic-node1
    public_ip: x.x.x.x
    private_ip: x.x.x.x
```

Example 1

```
cluster:
  admin:
    - name: admin-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  service:
    - name: service-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node2
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node3
      public_ip: x.x.x.x
      private_ip: x.x.x.x
```

Example 2

```
cluster:
  admin:
    - name: admin-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  gateway:
    - name: gateway-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  service:
    - name: service-node1
      public_ip: x.x.x.x
```

```
    private_ip: x.x.x.x
  - name: service-node2
    public_ip: x.x.x.x
    private_ip: x.x.x.x
  - name: service-node3
    public_ip: x.x.x.x
    private_ip: x.x.x.x
  - name: service-node4
    public_ip: x.x.x.x
    private_ip: x.x.x.x
```

Example 3

```
cluster:
  admin:
    - name: admin-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  gateway:
    - name: gateway-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  service:
    - name: service-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node2
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node3
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node4
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: service-node5
      public_ip: x.x.x.x
      private_ip: x.x.x.x
```

Custom Cluster

You can also add clusters with names you want as shown in the following example.


```
## Cluster Definition
##
cluster:
  admin:
    - name: admin-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  finance:
    - name: finance-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x
    - name: finance-node2
      public_ip: x.x.x.x
      private_ip: x.x.x.x

  blockchain:
    - name: blockchain-node1
      public_ip: x.x.x.x
      private_ip: x.x.x.x
```

Hostname

The servers in the cluster section have a naming convention such as `<cluster-name>-node#`. For example, the first server in the admin cluster is `admin-node1` and the second server in the service cluster is `service-node2`.

 [Edit this page](#)