

# Authorizing Requests and Creating a Cluster

In this lesson, we will look into various authorization methods and choose one for our use. We'll also create a cluster to start working with the authorization techniques.

## WE'LL COVER THE FOLLOWING ^

- Authorization Methods
- We will Go with RBAC
- Creating A Cluster

## Authorization Methods #

Just like almost everything else in Kubernetes, authorization is modular. We can choose to use *Node*, *ABAC*, *Webhook*, or *RBAC* authorization.

**Node:** Node authorization grants permissions to kubelets based on the Pods they are scheduled to run.

**ABAC:** Attribute-based access control (ABAC) is based on attributes combined with policies and is considered deprecated in favor of RBAC.

**Webhooks:** Webhooks are used for event notifications through HTTP POST requests.

**RBAC:** Role-based access control (RBAC) grants (or denies) access to resources based on roles of individual users or groups.

## We will Go with RBAC #

Among the four authorization methods, RBAC is the right choice for user-based authorization. Since we'll focus this chapter on the exploration of the means to authorize humans, RBAC will be our primary focus.

*What can we do with RBAC?*

- We can use it to secure the cluster by allowing access only to authorized users.
- We can define roles that would grant different levels of access to users and groups. Some could have god-like permissions that would allow them to do almost anything, while others could be limited only to basic non-destructive operations. There can be many other roles in between.
- We can combine RBAC with Namespaces and allow users to operate only within specific segments of a cluster.
- There are many other combinations we could apply depending on particular use-cases.

We'll leave the rest for later and explore details through a few examples. As you might already suspect, we'll kick it off with a new Minikube cluster.

## Creating A Cluster #

The commands that will create a Minikube cluster are as follows.

```
cd k8s-specs
git pull
minikube start --vm-driver virtualbox
kubectl config current-context
```



! RBAC is installed by default starting from minikube v0.26. If your version is older than that, you'll need to add `--extra-config apiserver.Authorization.Mode=RBAC` argument. Or, better yet, upgrade your minikube binary.

It might come in handy to have a few objects in the cluster so we'll deploy the `go-demo-2` application. We'll use it to test different permutations of the authorization strategies we'll use soon.

The definition of the `go-demo-2` application is the same as the one we created in the previous chapters so we'll skip the explanation and just execute `kubectl create`.

```
kubectl create \
-f auth/go-demo-2.yml \
```



```
--record --save-config
```

---

In the next lesson, we will explore how to create users who want to access our cluster.