Getting Started with Secrets

This is an introductory lesson about Kubernetes Secrets.

WE'LL COVER THE FOLLOWING ∧

- Understanding Secrets
- Creating A Cluster

Understanding Secrets

> We cannot treat all information equally. Sensitive data needs to be handled with additional care. Kubernetes provides an additional level of protection through Secrets.

A Secret is a relatively small amount of sensitive data. Some of the typical candidates for Secrets would be passwords, tokens, and SSH keys.

Kubernetes Secrets are very similar to ConfigMaps. If you compare the differences in the syntax, you'll notice that there are only a few (if any). Conceptually, both ConfigMaps and Secrets are, more or less, the same. If you are familiar with ConfigMaps, you should have no trouble applying that knowledge to Secrets.

> We already used Secrets without even knowing. Every Pod we created so far had a Secret mounted automatically by the system.

We'll start by exploring auto-generated Secrets and proceed to produce some ourselves.

> ℹ All the commands from this chapter are available in the 10-secret.sh

Creating A Cluster

We'll continue using Minikube, so the instructions for creating a cluster are still the same. They should be engraved in the back of your brain so we'll just execute them without any explanation.

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

We'll start by deploying an application without creating any user-defined Secret.

In the next lesson, we will explore Kubernetes built-in Secrets.