Dealing with Configuration Data

Use Config maps to send dynamic values into deployment

Applications require a way for us to pass data to them that can be changed at deploy time. Examples of this might be log levels or URLs of external systems that the application might need at start-up time. Instead of hard-coding these values, we use configmaps in Kubernetes to pass these values as environment variables to the container.

I have a deployment where I have a container called karthequian reader and, as an environment variable, I want to pass log level of error. Let's take a look at the logs We'll notice that this says that the log level passed via environment variable was error as per the application. What we'd like to do is we'd like to change this at deployment time.

```
RESTARTS
                        READY
logreader-7b746fc87-z4dzj
                               Running
                                                  117s
_og level passed via env variables was:
                                    'error
Log level passed via env variables was:
og level passed via env variables was:
                                    'error
og level passed via env variables was:
   level passed via env variables was:
   level passed via env variables was: 'error
   level passed via env variables was: 'error'
PS C:\Users\msuban01\Desktop\kube\Exercise\05_03_Configmaps> <mark>kubectl</mark> create configmap logger --from-literal=log_level=de
configmap/logger created
```

For this, we can use something called configmaps. We're going to call the configmap logger and we're going to take this from literal and we're going to call the actual value log_level=debug. This means that we are going to pass something called log_level=debug to the actual container as an environment variable.

We get configmap created and let's take a lot at the yaml for the deployment and how that's going to change when we use configmap. Open this file up and we can see that this is similar to the last example that we opened up, except you'll notice that we have an environment variable named log level, but the values are values from configmapRef, and the name of the configmap, and the key for this is the log_level. So this structure basically takes configmap, which is called logger, and reads the key, called log level, out of the configmap.

After creating the configmap we can take a look at the actual configmaps that we have created. As we notice, we have the logger that we've created from before and we can actually introspect into this by doing a kubectl, get configmap/logger -o yaml, similar to the way we look at deployments. And this returns the entire configmap to us (you'll notice that in the data section, as a log_level=debug, which is what we had initially set it to).

The log level passed via environment variables was debugged. Successfully we have used configmaps to send dynamic values into out deployment.

```
PS C:\Users\msuban01\Desktop\kube\Exercise\05_03_Configmaps> <mark>kubectl</mark> create -f reader-configmap-deployment.yaml
deployment.apps/logreader-dynamic created
PS C:\Users\msuban01\Desktop\kube\Exercise\05_03_Configmaps>                 <mark>kubectl</mark> get configmap
NAME
                   DATA AGE
                           65m
                            7m7s
logger
PSC:\Users\msuban01\Desktop\kube\Exercise\05_03_Configmaps> <mark>kubectl</mark> get configmap/logger -o yaml
apiVersion: v1
data:
 log_level: debug
kind: ConfigMap
metadata:
 creationTimestamp: "2022-02-22T22:19:10Z"
  name: logger
 namespace: default
resourceVersion: "3076"
uid: 837a6f6d-08fb-42e8-80a4-0df876070f52
PS C:\Users\msuban01\Desktop\kube\Exercise\05_03_Configmaps>                 <mark>kubectl</mark> get deployments
                    READY UP-TO-DATE AVAILABLE AGE
NAME
                                                         16m
logreader
logreader-dynamic
                     1/1
                                                         6m22s
PS C:\Users\msuban01\Desktop\kube\Exercise\05_03_Configmaps> kubectl get pods
JAME
                                    READY STATUS
                                                          RESTARTS AGE
logreader-7b746fc87-z4dzj
                                               Running
                                                                      16m
logreader-dynamic-86999f4cc-9b89x 1/1
                                               Running
                                                          0
                                                                      6m31s
PS C:\Users\msuban01\Desktop\kube\Exercise\05_03_Configmaps> kubectl logs logreader-dynamic-86999f4cc-9b89x
Log level passed via env variables was: 'debug'
Log level passed via env variables was: 'debug'
Log level passed via env variables was: 'debug
og level passed via env variables was: 'debug
    level passed via env variables was: 'debug
   level passed via env variables was: 'debug
    level passed via env variables was: 'debug
Log
Log level passed via env variables was: 'debug
    level passed via env variables was: 'debug
Log
    level passed via env variables was: 'debug
    level passed via env variables was: 'debug
   level passed via env variables was: 'debug
Log
Log level passed via env variables was: 'debug
    level passed via env variables was: 'debug
Log
Log level passed via env variables was: 'debug
    level passed via env variables was: 'debug
Log
    level passed via env variables was: 'debug
    level passed via env variables was: 'debug
    level passed via env variables was: 'debug
    level passed via env variables was: 'debug
Log
    level
          passed via env variables was:
```