

Lab 4 - More k8s

Lab 4 - More k8s

CYBR410 - Applied Cyber Operations

~~04-29-24~~ — ~~Alexander Moomaw~~

Simple Container

Create a simple program that we can easily tell is running behind a load balancer. This can be as easy as creating a random number each time the program starts (that is returned to the user) or a counter that increments each time the page is loaded.

Create the server:

- Utilize Python's *flask* and *random* module
- Generate a global random int between 1 and 100
- Create a function bound to our web servers root directory that serves our index.html file and inserts our random integer into the home page
- Serve our content locally on port 8000

Inside index.html:

- Make a new directory inside our web server directory, named templates.
- Place our index.html file in the templates directory
- Create generic content that displays that random integer

Kubernetes

As before, push the new container to our k8s registry and create the two yaml files needed to run your container in k8s. Make sure you can reach your service and that it's running via a k8s pod (exactly the same thing as last lab).

To push our newly created web server to kubernetes:

- `docker tag flask localhost:32000/flask:k8s`
- `docker push localhost:32000/flask:k8s`

What needs to change in our YAML file:

- Inside test.yaml:
 - image: localhost:32000/flask:k8s

Spin up our cluster:

- microk8s.kubectl apply -f test.yaml
- microk8s.kubectl apply -f test-svc.yaml

More Kubernetes

Delete a process running in the pod/container you created (kill -9 PID). Run the command that shows the pods (microk8s.kubectl get pods -all-namespaces), has anything changed? Elaborate.

- When you kill a process within a pod, it will recognize a crashed process and restart the pod. This can be indicated by viewing the affected pod: microk8s.kubectl get pods -all-namespaces

```
default      webserver2-8dbb9bf65-qpkjz      1/1      Running
- 1 (6s ago)      87s
```

Try and reach the service you created many times (curl or browser). Can you observe the load balancer at work? Do you always get the same page? Do you always get a different page?

- You don't always get the same page, and you don't always get a different page. Load balancing at this scale is seemingly done at random just due to the fact that there really isn't too much load on any of the instances. So sometimes I resolved to the same instance, sometimes I did not.

```
user@410-CyberDefense:~/Documents/dock$ curl http://10.152.183.226:8000/
Hosting from container: <td>36</td>user@410-CyberDefense:~/Documents/dock$ curl
http://10.152.183.226:8000/
Hosting from container: <td>51</td>user@410-CyberDefense:~/Documents/dock$ curl
http://10.152.183.226:8000/
Hosting from container: <td>51</td>user@410-CyberDefense:~/Documents/dock$ curl
http://10.152.183.226:8000/
Hosting from container: <td>51</td>user@410-CyberDefense:~/Documents/dock$ curl
http://10.152.183.226:8000/
Hosting from container: <td>51</td>user@410-CyberDefense:~/Documents/dock$ curl
http://10.152.183.226:8000/
Hosting from container: <td>36</td>user@410-CyberDefense:~/Documents/dock$ curl
http://10.152.183.226:8000/
Hosting from container: <td>18</td>user@410-CyberDefense:~/Documents/dock$
```

Describe a service (microk8s.kubectl describe service SERVICE NAME). Can I reach each individual pod being load balanced? Where do you see this information? Explain how this could be useful

- Each pod within that load balancer has it's own IP address and is able to be individually accessed:

```
user@410-CyberDefense:~/Documents/dock$ microk8s.kubectl get pods -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP
NODE                                NOMINATED NODE   READINESS GATES
webserver2-8dbb9bf65-fht6p          1/1     Running   1 (8m49s ago)    9m25s   10.1.56.144
410-cyberdefense                    <none>
webserver2-8dbb9bf65-qpkjz          1/1     Running   1 (8m4s ago)     9m25s   10.1.56.141
410-cyberdefense                    <none>
webserver2-8dbb9bf65-xqd4g          1/1     Running   0              9m25s   10.1.56.145
410-cyberdefense                    <none>
```

- This can be useful in instances where you want to individually troubleshoot a specific pod. Or if you want a specific action to always be sent to a specific pod.

Sources Cited:

- <https://kubernetes.io/docs/reference/kubectl/quick-reference/>