LAB 4

Kubernetes Namespaces

Lab Objectives

Describe how to get, create, and show the attributes a namespace.

Lab Structure - Overview

1. Display the existing namespaces installed by default
2. Create a new barebones namespace

Lab Overview

Conventions

Lab Guide Conventions

`reboot`	Any text a student needs to enter is printed like this.
<your.ip>	Any time a student needs to insert their own value, the text has brackets.
	Focuses the student's attention to a particular part of an image.
File	User Interface (UI) buttons and objects are bold.
Special Font	Unusual or important words or phrases are marked with italics.

Code Blocks

Blocks of sample code are set apart from the body and marked accordingly. It is recommended that students do not copy/paste text from the lab into their files. Extra formatting is often transferred in this process and can result in failed operations.

```
# ls -l /var/www/html/index.html
-rw-rw-r--  1 root root 1872 Jun 21 09:33 /var/www/html/index.html
# date
Wed Jun 21 09:33:42 EDT 200
```

1. Using kubectl to output configured namespaces

Step by Step Guide

This process will take approximately 10 minutes.

Step	Action
1.	Open a terminal console (iTerm, Terminal, PowerShell, Ubuntu Bash, Git Bash, etc).

2.	Run the `kubectl get namespace` command. This command will output the currently configured namespaces.
	
	`$ kubectl get namespace`
	`NAME STATUS AGE`
	`default Active 2h kube-`
	`public Active 2h kube-`
	`system Active 2h`

3.	Run the `kubectl get namespace default -o json` command. This will output more details regarding the specified namespace. The -o changes the output, in this case, a json formatted response. Rerun the command, but change json to yaml.
	
	`$ kubectl get namespace default -o json`
	`{`
	`"apiVersion": "v1",`
	`"kind": "Namespace",`
	`"metadata": {`
	`"creationTimestamp": "2017-04-23T19:29:56Z",`
	`"name": "default",`
	`"resourceVersion": "7",`
	`"selfLink": "/api/v1/namespacesdefault",`
	`"uid": "3b9c03f3-285b-11e7-abf6-7e9505c9bb2f"`
	`},`
	`"spec": {`
	`"finalizers": [`
	`"kubernetes"`
	`]`
	`},`
	`"status": {`
	`"phase": "Active"`
	
	`$ kubectl get namespace default`
	`NAME STATUS AGE`
	`default Active 2h`

2. Create a namespace

Step by Step Guide

This process will take approximately 5 minutes.

Step	Action
1.	Open a terminal console (iTerm, Terminal, PowerShell, Ubuntu Bash, Git Bash, etc).

2.	Run the **`kubectl create namespace dev`** command. This command will create a new namespace called dev. Run the **`kubectl get namespace`** to output the current namespaces.
	
	`$ kubectl create namespace dev`
	`namespace "dev" created`
	
	`$ kubectl get namespace`
	`NAME  STATUS  AGE`
	`default  Active  2h`
	`dev  Active  1m`
	`kube-public  Active  2h`
	`kube-system  Active  2h`

3.	Run the **`kubectl describe namespace dev`** command. This command will output more details of the namespace dev. Please note, the -o switch does not work with the `describe` sub-command.
	
	`$ kubectl describe namespace dev`
	`Name:  dev`
	`Labels:  <none>`
	`Annotations: <none>`
	`Status:  Active`
	
	`No resource quota.`
	
	`No resource limits.`

4.	In the follow up labs; we'll create some objects in the namespace.

Lab Complete!