Installing Kubernetes Kubectl

Objective: Install kubectl, install Run two instances of a Hello World application, and create a Service object that exposes a node port. We will also use the Service object to access the running application.

Preparation: Open a single terminal window and confirm internet connection. Inform instructor if you are unable to access the web.

Outcome: Participant will complete this lab when Kubernetes CLI tool kubectl is fully installed and configured.

Data Files: Ask the instructor what files are needed for this lab.

To deploy and manage applications on Kubernetes, you'll use the Kubernetes command-line tool, `kubectl`. It lets you inspect your cluster resources, create, delete, and update components, and much more. You will use it to look at your new cluster and bring up example apps.

Step 1. Pick best method to install kubectl:

Option 1. *Install kubectl Binary Via curl*

1. Download the latest release with the command:

OS X

```
$ curl -LO https://storage.googleapis.com/kubernetes-rele
ase/release/$(curl -s https://storage.googleapis.com/kube
rnetes-release/release/stable.txt)/bin/darwin/amd64/kubec
```

```
tl
```

Linux

```
$ curl -LO https://storage.googleapis.com/kubernetes-rele
ase/release/$(curl -s https://storage.googleapis.com/kube
rnetes-release/release/stable.txt)/bin/linux/amd64/kubect
l
```

Windows

```
$ curl -LO https://storage.googleapis.com/kubernetes-rele
ase/release/$(curl -s https://storage.googleapis.com/kube
rnetes-release/release/stable.txt)/bin/windows/amd64/kube
ctl.exe
```

> *NOTE: If you want to download a specific version of `kubectl` you can replace the nested curl command from above with the version you want. (e.g. `v1.4.6, v1.5.0-beta.2`)*

2. Make the `kubectl` binary executable and move it to your PATH (e.g. `/usr/local/bin`):

```
$ chmod +x ./kubectl
$ sudo mv ./kubectl /usr/local/bin/kubectl
```

Option 2. *Extract kubectl from Release .tar.gz or Compiled Source*

1. Ask the instructor where the Compiled Source tar file is located.

2. Copy or move `kubectl` into a directory already in your PATH (e.g. `/usr/local/bin`). For example:

OS X

```
$ sudo cp platforms/darwin/amd64/kubectl /usr/local/bin/k
ubectl
```

Linux

```
$ sudo cp platforms/linux/amd64/kubectl /usr/local/bin/ku
bectl
```

3. Next make it executable with the following command:

```
$ sudo chmod +x /usr/local/bin/kubectl
```

4. The `kubectl` binary doesn't have to be installed to be executable, but the rest of the walkthrough will assume that it's in your `PATH`. If you prefer not to copy `kubectl`, you need to ensure it is in your path:

OS X

```
export PATH=<path/to/kubernetes-directory>/platforms/darw
in/amd64:$PATH
```

Linux

```
export PATH=<path/to/kubernetes-directory>/platforms/linux/amd64:$PATH
```

Option 3. *Download as part of the Google Cloud SDK*

`kubectl` can be installed as part of the Google Cloud SDK:

1. First install the Google Cloud SDK (`https://cloud.google.com/sdk/`).
2. After Google Cloud SDK installs, run the following command to install `kubectl`:

```
$ gcloud components install kubectl
```

3. Do check that the version is sufficiently up-to-date using:

```
kubectl version
```

Step 2. Configuring kubectl [Ask Instructor Before Proceeding]

In order for `kubectl` to find and access the Kubernetes cluster, it needs a kubeconfig file, which is created automatically when creating a cluster using `kube-up.sh`. By default, `kubectl` configuration lives at ~/.kube/config.

1. Make sure you're ready. Check that `kubectl` is properly

configured by getting the cluster state:

```
$ kubectl cluster-info
```

If you see a url response, you are ready to go.

Conclusion

xx