Different ways to install kubernetes

1. Play-with-k8s

2. Minikube

3. kubeadm

4. Google Kubernetes Engine (GKE)

5. Amazon EKS

6. Azure Kubernetes Engine(AKS)

**Play-with-k8s: (PWK)**

(Kubernetes URL: <https://labs.play-with-k8s.com/> )

Imagine that you want to quickly test something on your kubernetes cluster

Unfortunately, it is not ready available

What is your option now?

-----\*\*\*\*\*\*\*\*\*You can use **Play-with-k8s**\*\*\*\*\*\*\*-----------

Part-1

* Overview of **Play-With-K8s**

Part-2

* Review Demo

1. Add new Instances
2. Configure ‘Master”
3. Configure “Worker Node”
4. Test and validate

**Overview**

* Kubernetes playground – (same as like docker play) provide environment and learn kubernetes entirely through the browser{ Without Installation}
* This is provided by **docker** and created by **Tutorius**
* Released in mid-2017
* Github or docker account is required
* Created k8s cluster in seconds
* Four hours’ time limit maximum
* Maximum five instances we can add

**Master Node Configuration:**

* Add New instance
* Initialize Master Node

**kubeadm init –apiserver-advertise-address $(hostname –i)**

* Once we initilalize the above command we got some output and we have to join command safe and securely for worker node
* Configure Network plugin (help us communication between all the pods and nodes inside the cluster)
* We have multiple plugins Calico or flannel 🡪 the command will get in the initialize command output

**kubectl apply –n kube-system –f\**

[**https://cloud.weave.works/k8s/net?k8s-version=$(kubectl version | basse64 |tr -d ‘\n’)**](https://cloud.weave.works/k8s/net?k8s-version=$(kubectl%20version%20|%20basse64%20|tr%20-d%20'\n'))

* The above command install weave network for us
* Once we done initialize and network configuration, your **master node** is ready

**Worker Node Configuration:**

* We can use **kubeadm**  command to join worker node to the master
* Create New Instances
* One simple command to join the master node

**Kubeadm join –token […] --discovery-token-ca-cert-hash […] --🡪 (we will get this command in the master node**

* Maximum we can join 8 instances to master

**Testing Cluster:**

* Kubectl get no (get no. of nodes)
* Kubectl run kubernetes-bootcamp –image=gcr.io/google-samples/kubernetes-bootcamp:v1 –port=8080 (installing application from google repository)
* Kubectl get po (get no. of pods)

**Play-with-k8s Demo:**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Installation of k8s using play-with-k8s in four steps

# Step 1:

Create four instances. One for master and three for worker nodes

Required “docker” or “github’ account

<https://labs.play-with-k8s.com/>

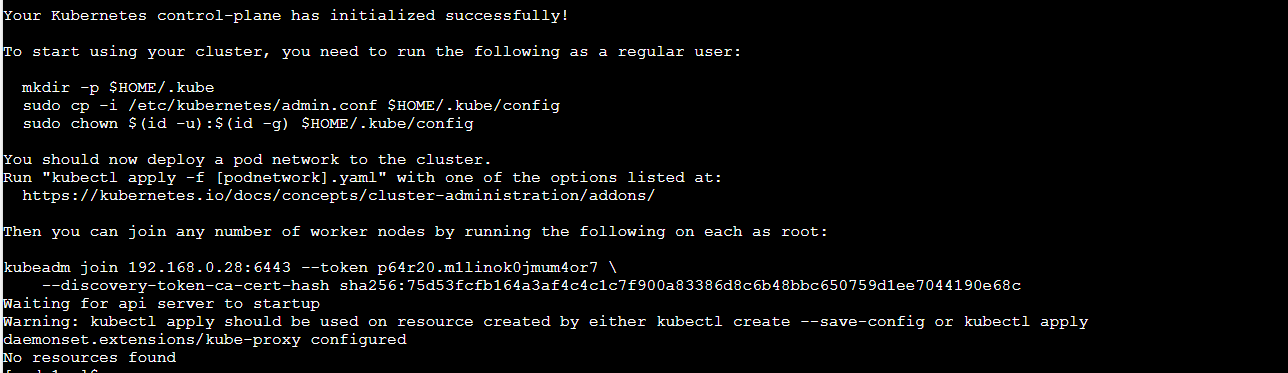
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Step 2: Initialize Master Node

kubeadm init –apiserver-advertise-address $(hostname –i)

kubectl apply –n kube-system –f\

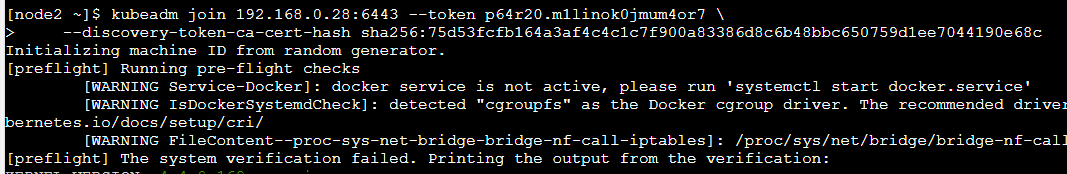
“[https://cloud.weave.works/k8s/net?k8s-version=$(kubectl version | basse64 |tr -d ‘\n’)](https://cloud.weave.works/k8s/net?k8s-version=$(kubectl%20version%20|%20basse64%20|tr%20-d%20'\n'))”



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**#** Step 3: Join Worker to the cluster

Kubeadm join –token […] --discovery-token-ca-cert-hash […]

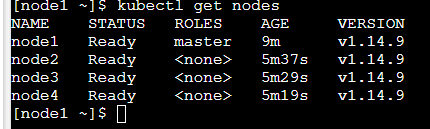


\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

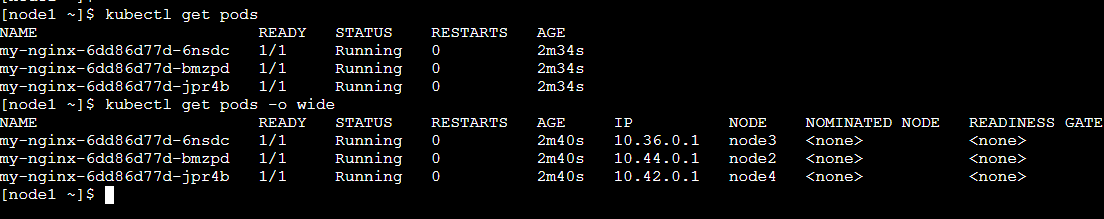
# step 4: Testing

We will deploy application on the worker nodes

* Kubectl get nodes (get no. of nodes)



* Kubectl apply –f <https://raw.githubusercontent.com/kubernetes/website/master/en/examples/application/nginx-app.yaml>
* Kubectl get pods (get no. of pods)
* Kubectl get pods –o wide



**Minikube:**

Imagine that, you want to access kubernetes from your laptop for learning or quick testing purpose

But, at same time, **you don’t want to put so much pressure** on your system resources such as CPU, Ram and Disk

So, **what is your go-to** **option** then?

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* **Minikube** \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Concept:

* Overview of Minikube

Review Demo

* Download and Install – VirtualBox, Minikube & kubectl
* Start Minikube
* Test

**Overview**:

* Tool that makes it easy to run kubernetes locally
* Runs all k8s componesrts in single-node inside a VM
* Automatically download ISO and create VM
* Require **Hypervisor**( Ex: VirtualBox)
* Require **kubectl** to manage deployments

Kubectl

Virtual Box

-------🡪 --🡪

* First we have to download **Virtualbox**, next we have to download **Minikube**  and start it
* **Minkube**’s responsibility to download and start kubernetes
* Once kubernetes is setup, then we download **kubectl** 🡪 command Line utility to interact with kubernetes

**Review Demo:**

1. Download & Install VirtualBox
2. Download “minikube-windows-amd64”

C:\>mkdir kubernetes

C:\>cd kubernetes

<https://github.com/kubernetes/minikube/releases>

**Note:**  Kubectl and minikube shall appear as shown below in

C :\> Kubernetes Location

1. Download “kubectl”

C:\Kubernetes> curl –LO <https://storage.googleapis.com/kubernetes-release/release/v1.11.0/bin/windows/amd64/kubectl.exe>

1. Start Minikube

C:\Kubernetes> minikube.exe start

1. Once we start the above command, minikube will download the ISO and having kubelet and kubeadm configuration
2. Once it download complete, minikube will create new kubernetes VM on VirtualBox, then start and configuring kubernetes and kubernetes cluster configuration
3. C:\Kubernetes>minikube status

It is required in the location where the binaries are located

1. C:\Kubernetes> minikube version

To display version of minikube

1. C:\Kubernetes> Kubectl run kubernetes-bootcamp –image=gcr.io/google-samples/kubernetes-bootcamp:v1 –port=8080

Sample deployment – we will use deploy some application

1. Kubectl get pods
2. Minikube stop

**Minikube Demo**

Demo: MiniKube

# Step 1. Download & Install “VirtualBox”

Source: <https://www.virtualbox.org/>

# Step 2: Download “minikube-windows-amd64”

C:\> mkdir Kubernetes

Source: <https://github.com/kubernetes/minikube/releases>

# Step 3:

Download “kubectl”

C:\Kubernetes> curl –LO <https://storage.googleapis.com/kubernetes-release/release/v1.11.0/bin/windows/amd64/kubectl.exe>

# Step 4: Start Minikube

C;\Kubernetes> minikube.exe start

#Step 5: Testing

C:\Kubernetes>minikube status

C:\Kubernetes>minikube version

C:\Kubernetes>kubectl get nodes

C:\Kubernetes> Kubectl run kubernetes-bootcamp –image=gcr.io/google-samples/kubernetes-bootcamp:v1 –port=8080

C:\Kubernetes>kubectl get pods

C:\Kubernetes>minikube stop