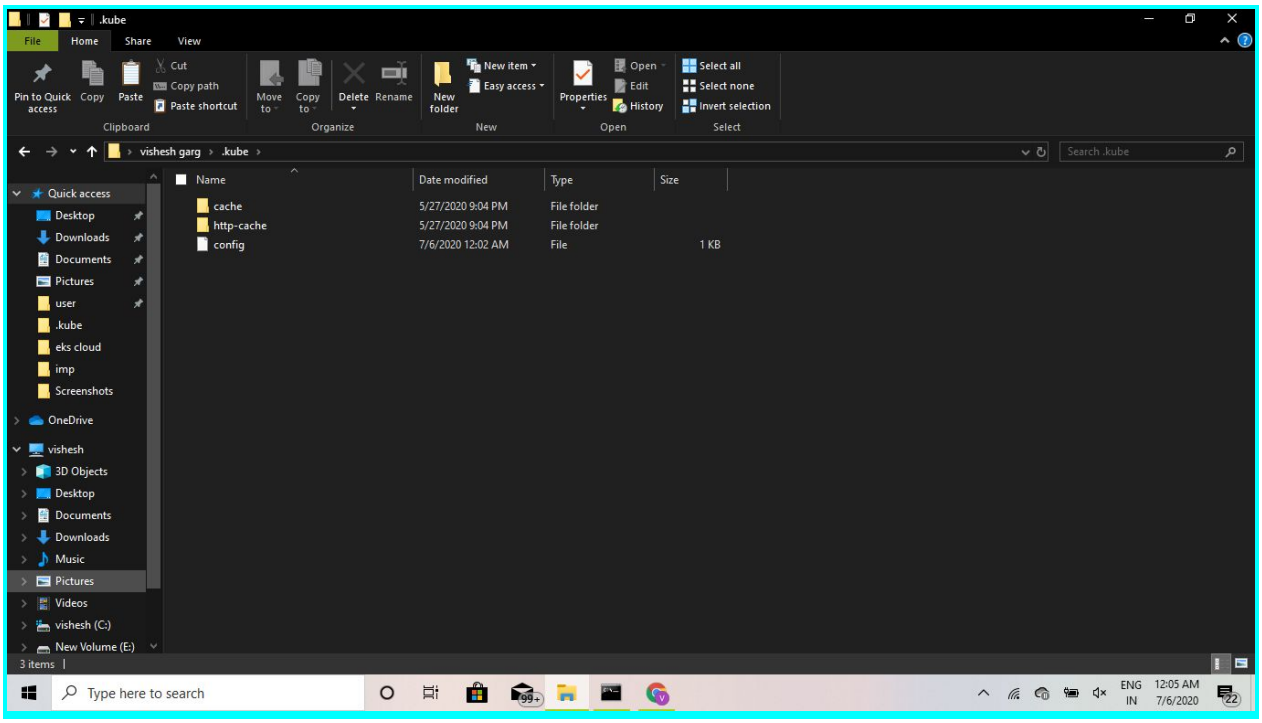
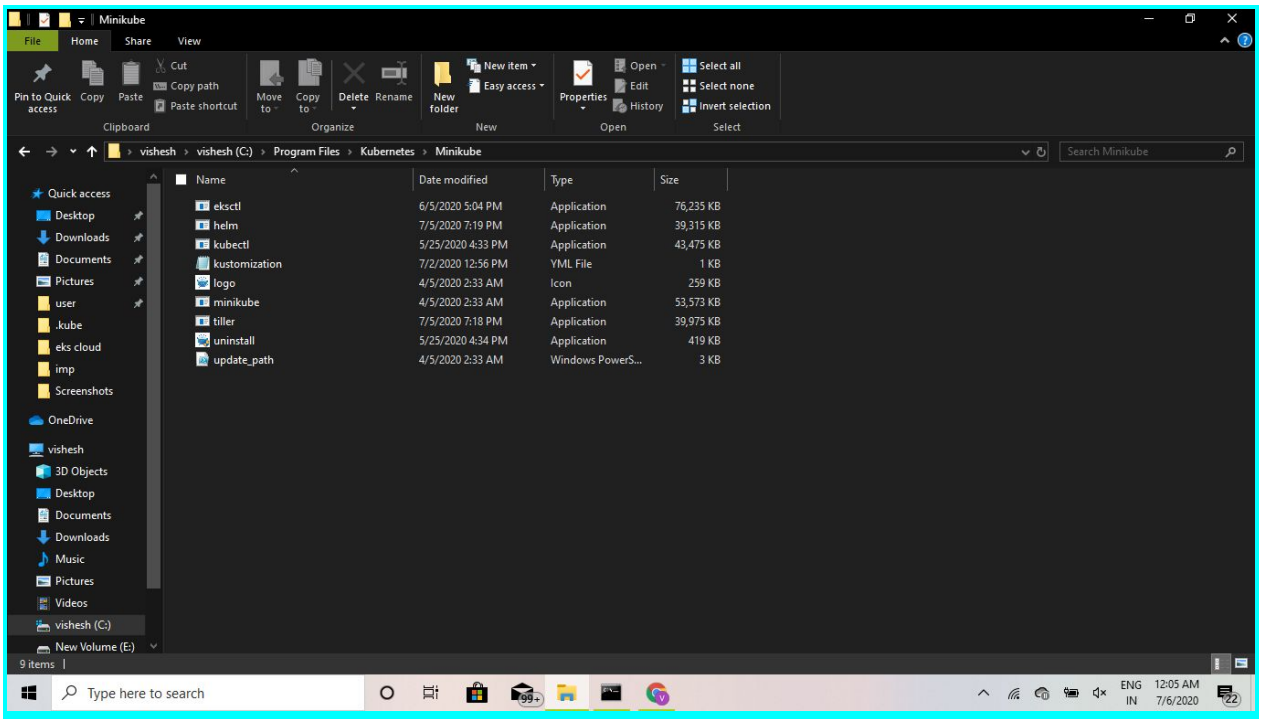
HELM AND TILLER :



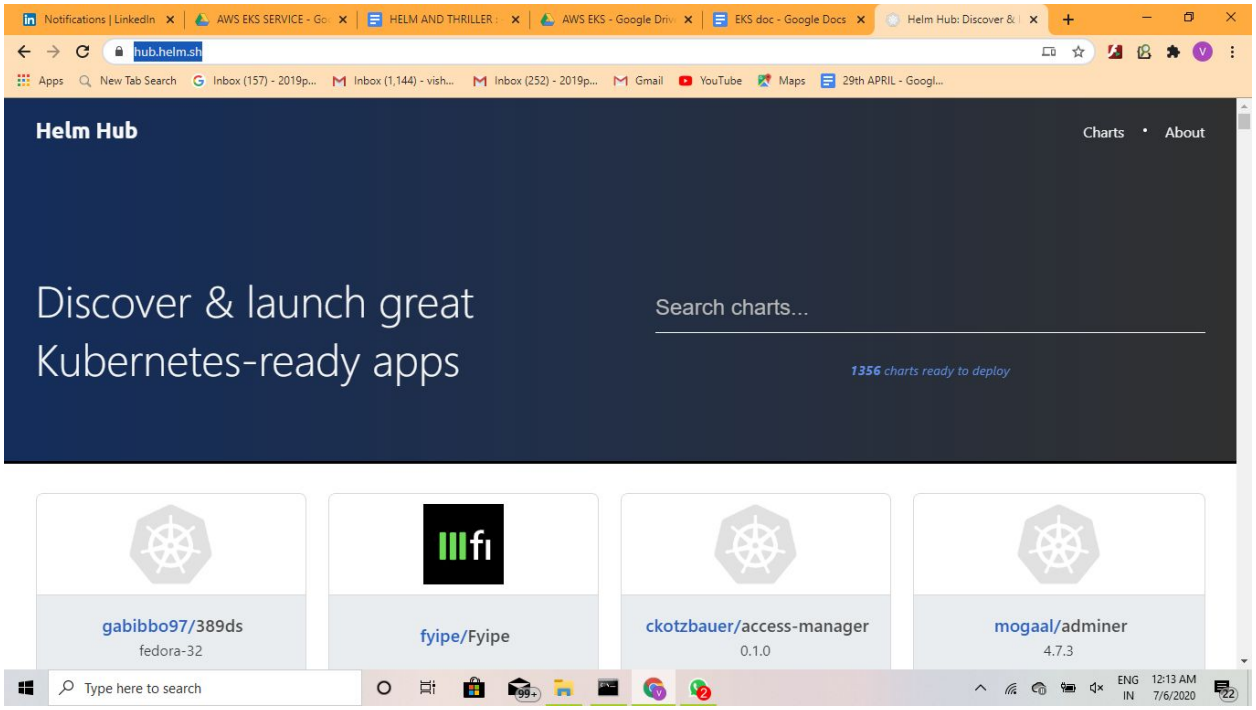


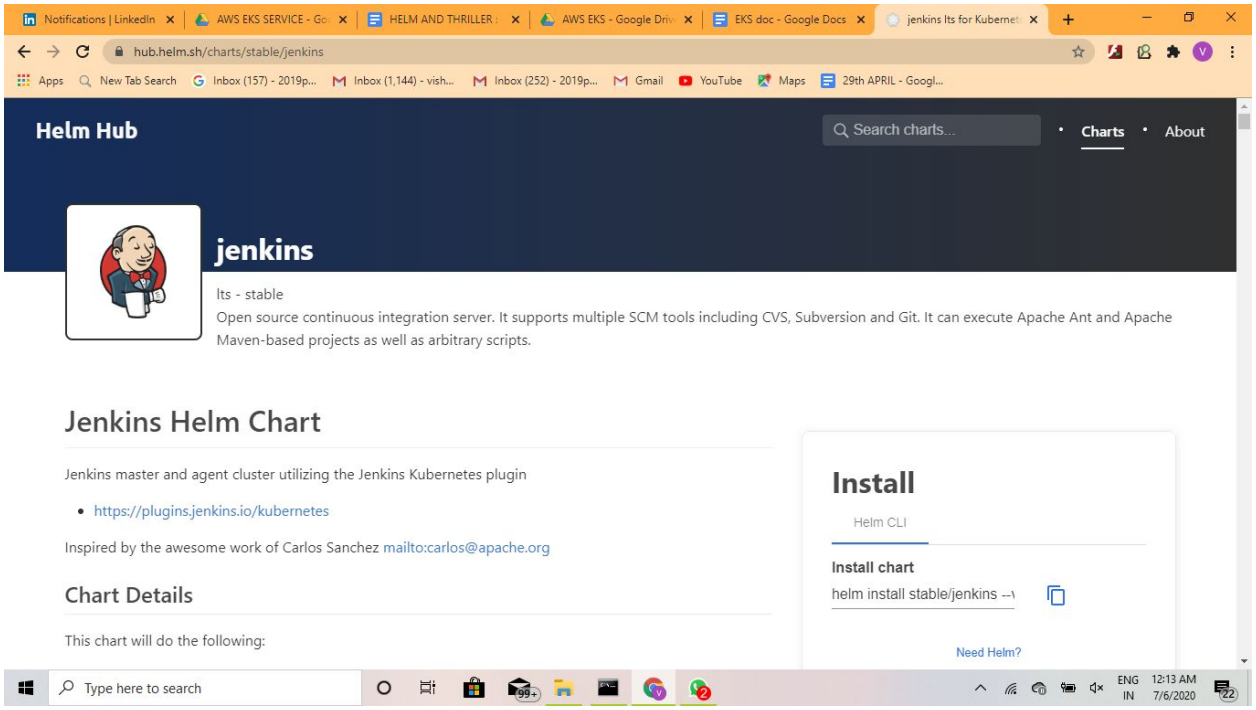
C:\Windows\system32>helm init
$HELM_HOME has been configured at C:\Users\user\.helm.

Tiller (the Helm server-side component) has been installed into your Kubernetes Cluster.

Please note: by default, Tiller is deployed with an insecure 'allow unauthenticated users' policy.
To prevent this, run `helm init` with the --tiller-tls-verify flag.
For more information on securing your installation see:
https://v2.helm.sh/docs/securing_installation/

C:\Windows\system32>kubectl get ns
NAME STATUS AGE
default Active 41d
kube-node-lease Active 41d
kube-public Active 41d
kube-system Active 41d
kubernetes-dashboard Active 19d
C:\Windows\system32>tiller version
[main] 2020/07/06 00:07:51 Starting Tiller v2.16.9 (tls=false)
[main] 2020/07/06 00:07:51 GRPC listening on :44134
[main] 2020/07/06 00:07:51 Probes listening on :44135
[main] 2020/07/06 00:07:51 Storage driver is ConfigMap
[main] 2020/07/06 00:07:51 Max history per release is 0
C:\Windows\system32>tiller -help
Usage of tiller:
-add_dir_header
If true, adds the file directory to the header
https://hub.helm.sh/





https://github.com/helm/helm/releases

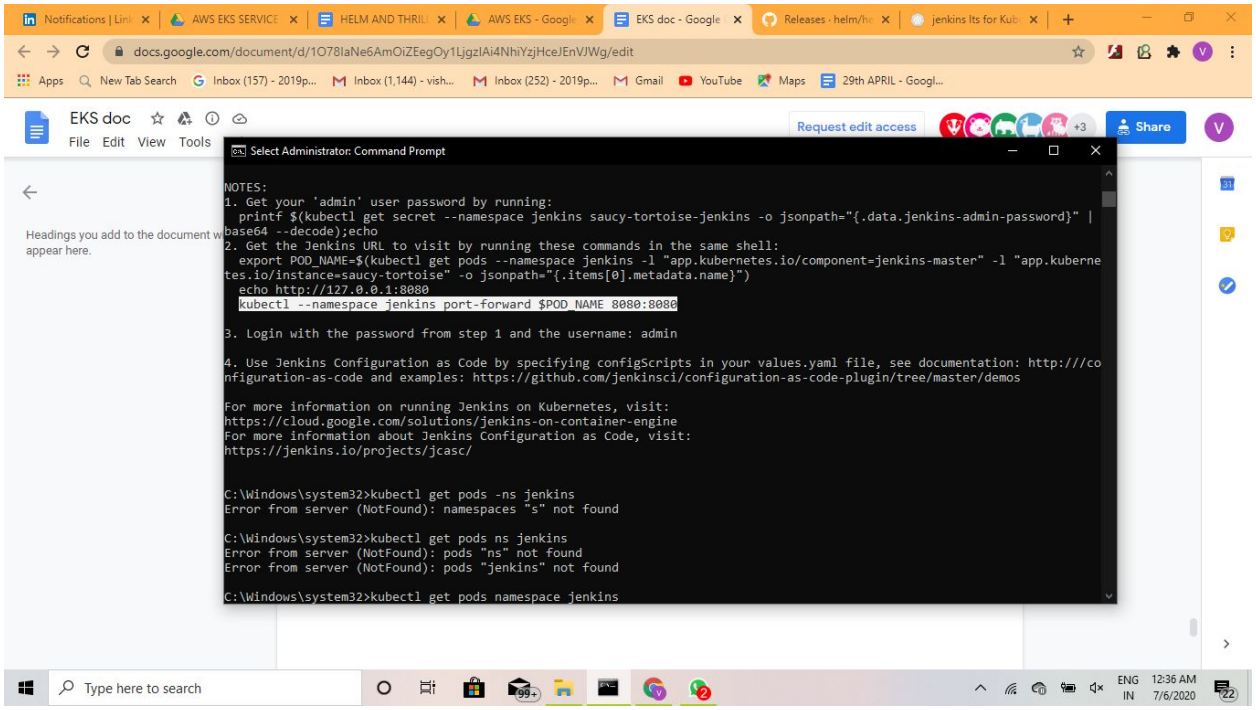
C:\Windows\system32>helm repo add stable https://kubernetes-charts.storage.googleapis.com/
"stable" has been added to your repositories

C:\Windows\system32>helm repo list
NAME URL
stable https://kubernetes-charts.storage.googleapis.com/
local http://127.0.0.1:8879/charts

```
C:\Windows\system32>helm repo update
Hang tight while we grab the latest from your chart repositories...
...Skip local chart repository
...Successfully got an update from the "stable" chart repository
Update Complete.
C:\Windows\system32>kubectl -n kube-system create serviceaccount tiller
serviceaccount/tiller created

C:\Windows\system32>kubectl get ns
NAME                   STATUS   AGE
default                Active   41d
kube-node-lease        Active   41d
kube-public            Active   41d
kube-system            Active   41d
kubernetes-dashboard   Active   19d

C:\Windows\system32>kubectl create clusterrolebinding tiller --clusterrole cluster-admin
--serviceaccount=kube-system:tiller
clusterrolebinding.rbac.authorization.k8s.io/tiller created

C:\Windows\system32>kubectl get all
NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   33h

C:\Windows\system32>kubectl get pods
No resources found in default namespace.

C:\Windows\system32>helm init --service-account tiller
$HELM_HOME has been configured at C:\Users\user\.helm.
Warning: Tiller is already installed in the cluster.
(Use --client-only to suppress this message, or --upgrade to upgrade Tiller to the current
version.)

C:\Windows\system32>kubectl get pods --namespace kube-system
NAME                                   READY   STATUS    RESTARTS   AGE
coredns-66bff467f8-9nq8k               1/1     Running   15         41d
coredns-66bff467f8-sdptx               1/1     Running   14         41d
etcd-minikube                          1/1     Running   12         41d
kube-apiserver-minikube                1/1     Running   12         41d
kube-controller-manager-minikube       1/1     Running   12         41d
kube-proxy-f2qgq                       1/1     Running   12         41d
kube-scheduler-minikube                1/1     Running   12         41d
```

```
storage-provisioner              1/1     Running   17        41d
tiller-deploy-fc55974f-rcj45     1/1     Running   0         21m
C:\Windows\system32>kubectl create ns jenkins
namespace/jenkins created

C:\Windows\system32>kubectl get ns
NAME                 STATUS   AGE
default              Active   41d
jenkins              Active   4s
kube-node-lease      Active   41d
kube-public          Active   41d
kube-system          Active   41d
kubernetes-dashboard   Active   19d
```

> helm install stable/jenkins --version 2.1.2

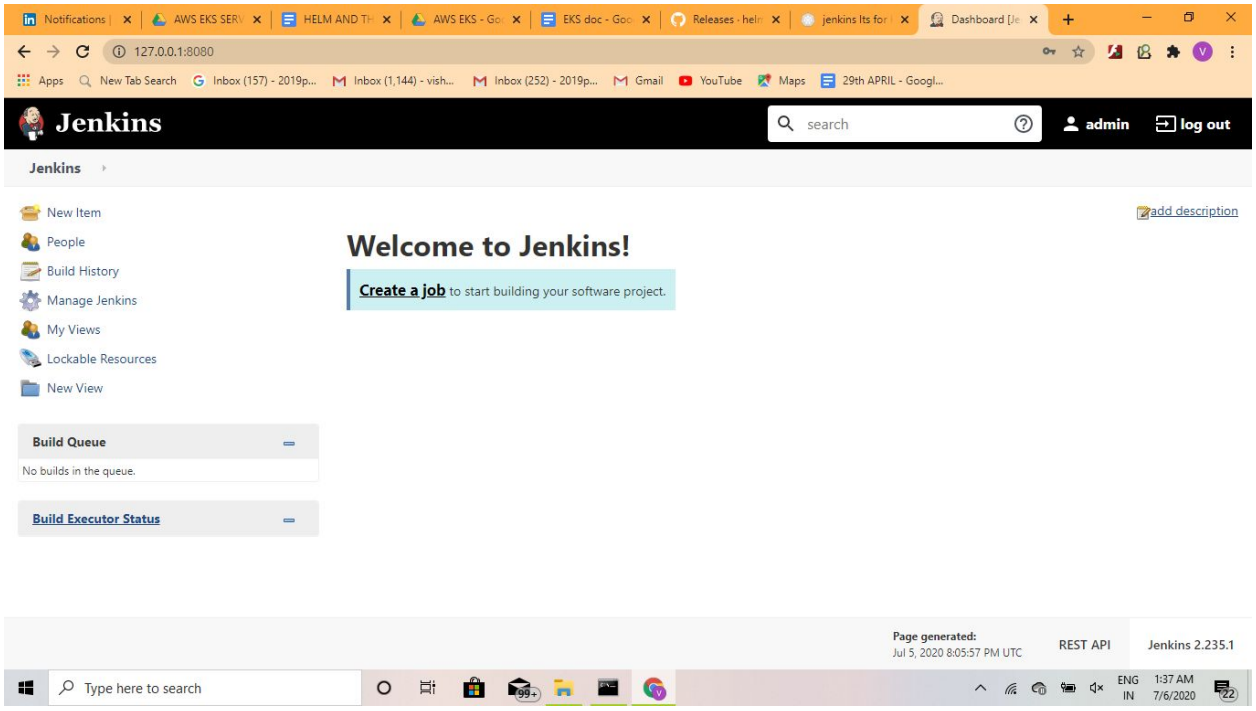



```
C:\Windows\system32>kubectl --namespace jenkins port-forward
saucy-tortoise-jenkins-6684b45d98-7cjnt 8080:8080
C:\Windows\system32>kubectl delete all --all --namespace jenkins
pod "saucy-tortoise-jenkins-6684b45d98-7cjnt" deleted
service "saucy-tortoise-jenkins" deleted
service "saucy-tortoise-jenkins-agent" deleted
deployment.apps "saucy-tortoise-jenkins" deleted
replicaset.apps "saucy-tortoise-jenkins-6684b45d98" deleted
C:\Windows\system32>kubectl get all --namespace jenkins
No resources found in jenkins namespace.
```

The Alertmanager handles alerts sent by client applications such as the Prometheus server

helm install stable/prometheus --namespace prometheus --set alertmanager.persistentVolume.storageClass="gp2" --set server.persistentVolume.storageClass="gp2"

>> kubectl --namespace jenkins --set alertmanager.persistentVolume.storageClass=”gp2”
>> kubectl --namespace jenkins --set server.persistentVolume.storageClass=”gp2”



C:\Windows\system32>helm install stable/jenkins --version 2.1.2 --namespace jenkins --set master.usePodSecurityContext=True --set master.adminPassword="redhat"
NAME: quieting-billygoat
LAST DEPLOYED: Mon Jul 6 01:33:48 2020
NAMESPACE: jenkins
STATUS: DEPLOYED

RESOURCES:
==> v1/ConfigMap
NAME DATA AGE
quieting-billygoat-jenkins 2 0s
quieting-billygoat-jenkins-jenkins-jcasc-config 1 0s
quieting-billygoat-jenkins-tests 1 0s

```
==> v1/Deployment
NAME                    READY  UP-TO-DATE  AVAILABLE  AGE
quieting-billygoat-jenkins  0/1    1           0          0s

==> v1/PersistentVolumeClaim
NAME                    STATUS  VOLUME                                     CAPACITY  ACCESS MODES
STORAGECLASS  AGE
quieting-billygoat-jenkins  Bound   pvc-5970cc3d-3269-48c8-8eba-d0aef89b6ab7  8Gi       RWO
standard      0s

==> v1/Pod(related)
NAME                                READY  STATUS    RESTARTS  AGE
quieting-billygoat-jenkins-8654c5dc75-kzx24  0/2    Init:0/1  0         0s

==> v1/Role
NAME                                CREATED AT
quieting-billygoat-jenkins-casc-reload      2020-07-05T20:03:48Z
quieting-billygoat-jenkins-schedule-agents  2020-07-05T20:03:48Z

==> v1/RoleBinding
NAME                                ROLE                                      AGE
quieting-billygoat-jenkins-schedule-agents   Role/quieting-billygoat-jenkins-schedule-agents  0s
quieting-billygoat-jenkins-watch-configmaps  Role/quieting-billygoat-jenkins-casc-reload      0s

==> v1/Secret
NAME                    TYPE    DATA  AGE
quieting-billygoat-jenkins  Opaque  2     0s

==> v1/Service
NAME                         TYPE       CLUSTER-IP     EXTERNAL-IP  PORT(S)    AGE
quieting-billygoat-jenkins       ClusterIP  10.102.22.195  <none>       8080/TCP   0s
quieting-billygoat-jenkins-agent  ClusterIP  10.97.36.11    <none>       50000/TCP  0s

==> v1/ServiceAccount
NAME                    SECRETS  AGE
quieting-billygoat-jenkins  1        0s


NOTES:
1. Get your 'admin' user password by running:
  printf $(kubectl get secret --namespace jenkins quieting-billygoat-jenkins -o
jsonpath="{.data.jenkins-admin-password}" | base64 --decode);echo
2. Get the Jenkins URL to visit by running these commands in the same shell:
```

```
  export POD_NAME=$(kubectl get pods --namespace jenkins -l
"app.kubernetes.io/component=jenkins-master" -l
"app.kubernetes.io/instance=quieting-billygoat" -o jsonpath="{.items[0].metadata.name}")
  echo http://127.0.0.1:8080
  kubectl --namespace jenkins port-forward $POD_NAME 8080:8080

3. Login with the password from step 1 and the username: admin

4. Use Jenkins Configuration as Code by specifying configScripts in your values.yaml file, see
documentation: http:///configuration-as-code and examples:
https://github.com/jenkinsci/configuration-as-code-plugin/tree/master/demos

For more information on running Jenkins on Kubernetes, visit:
https://cloud.google.com/solutions/jenkins-on-container-engine
For more information about Jenkins Configuration as Code, visit:
https://jenkins.io/projects/jcasc/

C:\Windows\system32>kubectl get pods -n jenkins
NAME                                   READY   STATUS    RESTARTS   AGE
quieting-billygoat-jenkins-8654c5dc75-kzx24   1/2     Running   0          90s

C:\Windows\system32>kubectl --namespace jenkins port-forward
quieting-billygoat-jenkins-8654c5dc75-kzx24 8080:8080
Forwarding from 127.0.0.1:8080 -> 8080
Forwarding from [::1]:8080 -> 8080

C:\Windows\system32>kubectl get secret quieting-billygoat-jenkins --namespace jenkins -o
yaml
apiVersion: v1
data:
  jenkins-admin-password: cmVkaGF0
  jenkins-admin-user: YWRtaW4=
kind: Secret
metadata:
  creationTimestamp: "2020-07-05T20:03:48Z"
  labels:
    app.kubernetes.io/component: jenkins-master
    app.kubernetes.io/instance: quieting-billygoat
    app.kubernetes.io/managed-by: Tiller
    app.kubernetes.io/name: jenkins
    helm.sh/chart: jenkins-2.1.2
  managedFields:
  - apiVersion: v1
```

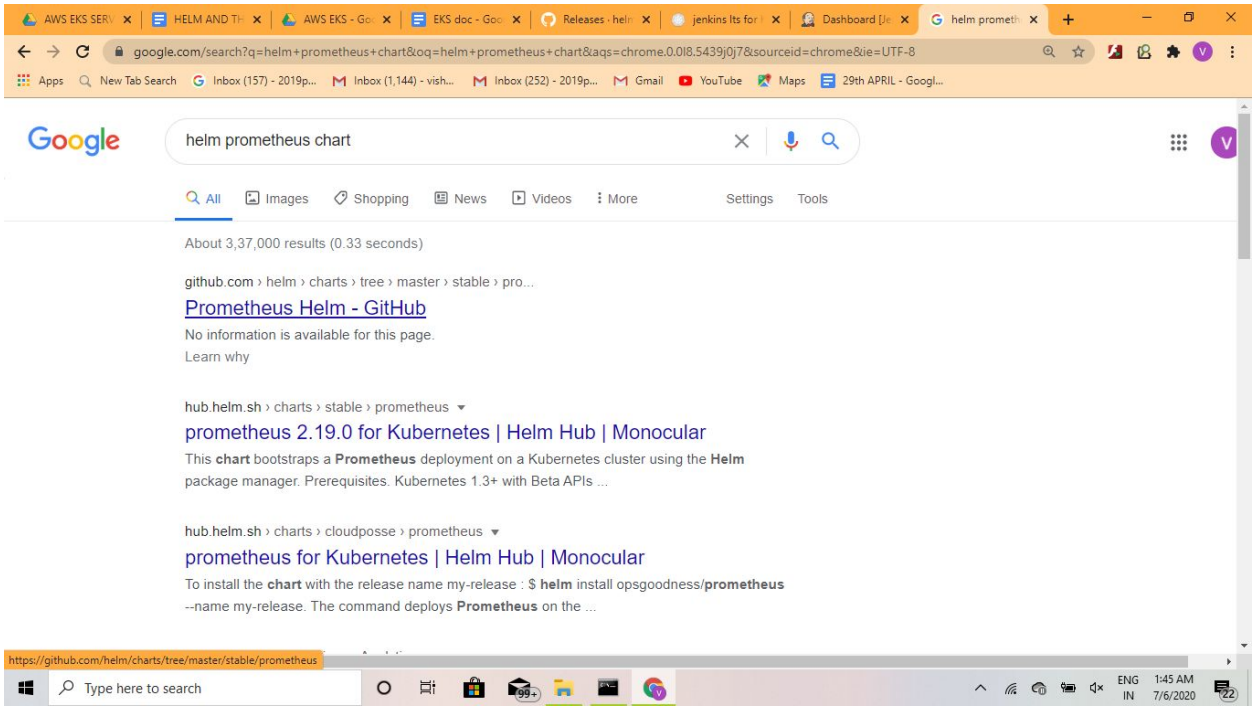
```
    fieldsType: FieldsV1
    fieldsV1:
      f:data:
        .: {}
        f:jenkins-admin-password: {}
        f:jenkins-admin-user: {}
      f:metadata:
        f:labels:
          .: {}
          f:app.kubernetes.io/component: {}
          f:app.kubernetes.io/instance: {}
          f:app.kubernetes.io/managed-by: {}
          f:app.kubernetes.io/name: {}
          f:helm.sh/chart: {}
      f:type: {}
    manager: Go-http-client
    operation: Update
    time: "2020-07-05T20:03:48Z"
  name: quieting-billygoat-jenkins
  namespace: jenkins
  resourceVersion: "428698"
  selfLink: /api/v1/namespaces/jenkins/secrets/quieting-billygoat-jenkins
  uid: 3af17e1d-d26d-4a7b-8ec3-2950076e77d1
type: Opaque
```

For prometheus:

The Alertmanager handles alerts sent by client applications such as the Prometheus server

```
helm install  stable/prometheus     --namespace prometheus     --set
alertmanager.persistentVolume.storageClass="gp2"     --set server.persistentVolume.storageClass="gp2"
```

```
>> kubectl --namespace jenkins --set alertmanager.persistentVolume.storageClass=”gp2”
>> kubectl --namespace jenkins --set server.persistentVolume.storageClass=”gp2”
```


google.com/search?q=helm+prometheus+chart&oq=helm+prometheus+chart&aqs=chrome.0.0l8.5439j0j7&sourceid=chrome&ie=UTF-8
Google
helm prometheus chart
All
Images
Shopping
News
Videos
More
Settings
Tools
About 3,37,000 results (0.33 seconds)
github.com › helm › charts › tree › master › stable › pro...
Prometheus Helm - GitHub
No information is available for this page.
Learn why
hub.helm.sh › charts › stable › prometheus
prometheus 2.19.0 for Kubernetes | Helm Hub | Monocular
This chart bootstraps a Prometheus deployment on a Kubernetes cluster using the Helm package manager. Prerequisites. Kubernetes 1.3+ with Beta APIs ...
hub.helm.sh › charts › cloudposse › prometheus
prometheus for Kubernetes | Helm Hub | Monocular
To install the chart with the release name my-release : $ helm install opsgoodness/prometheus --name my-release. The command deploys Prometheus on the ...
https://github.com/helm/charts/tree/master/stable/prometheus
Type here to search
ENG
IN
1:45 AM
7/6/2020