Lab 5 - Monitoring
By Xiaogang Dong 101492108

1. Install Prometheus

```
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % brew install helm
Running `brew update --auto-update`...
==> Auto-updated Homebrew!
Updated 3 taps (homebrew/services, homebrew/core and homebrew/cask).
==> New Formulae
git-grab              libnsgif           limesuite        openjph          scnlib          terrapin-scanner   tomlplusplus
icloudpd              libspelling        nowplaying-cli   rsyncy           senpai          tfautomv           zigmod
==> New Casks
bitbox                cleanupbuddy       insomnium        nightshade       theiaide        znote
bugdom2               domzilla-caffeine  lyricsfinder     shadow-bot       ttu-base-suite

You have 8 outdated formulae installed.

==> Downloading https://ghcr.io/v2/homebrew/core/helm/manifests/3.14.0
######################################################################################################################## 100.0%
==> Fetching helm
==> Downloading https://ghcr.io/v2/homebrew/core/helm/blobs/sha256:9a473b31064afd43207e46fd21e4137565002e07d6da44ecdf9b3472585e1376
######################################################################################################################## 100.0%
==> Pouring helm--3.14.0.arm64_sonoma.bottle.tar.gz
==> Caveats
zsh completions have been installed to:
  /opt/homebrew/share/zsh/site-functions
==> Summary
🍺  /opt/homebrew/Cellar/helm/3.14.0: 65 files, 54.3MB
==> Running `brew cleanup helm`...
Disable this behaviour by setting HOMEBREW_NO_INSTALL_CLEANUP.
Hide these hints with HOMEBREW_NO_ENV_HINTS (see `man brew`).
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % kubectl get pods

E0129 20:18:18.697689   52809 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 1
27.0.0.1:8080: connect: connection refused
E0129 20:18:18.698463   52809 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 1
27.0.0.1:8080: connect: connection refused
E0129 20:18:18.699541   52809 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 1
27.0.0.1:8080: connect: connection refused
E0129 20:18:18.700004   52809 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 1
27.0.0.1:8080: connect: connection refused
E0129 20:18:18.701074   52809 memcache.go:265] couldn't get current server API group list: Get "http://localhost:8080/api?timeout=32s": dial tcp 1
27.0.0.1:8080: connect: connection refused
The connection to the server localhost:8080 was refused - did you specify the right host or port?
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 %
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm install prometheus prometheus-community/prometheus
```

```
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm install prometheus prometheus-community/prometheus


Error: INSTALLATION FAILED: repo prometheus-community not found
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm repo add prometheus-community https://prometheus-community.github.io/helm-charts


"prometheus-community" has been added to your repositories
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm install prometheus prometheus-community/prometheus


Error: INSTALLATION FAILED: Kubernetes cluster unreachable: Get "http://localhost:8080/version": dial tcp 127.0.0.1:8080: connect: connection refu
sed
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm repo update


Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "prometheus-community" chart repository
Update Complete. *Happy Helming!*
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm install prometheus prometheus-community/prometheus                                              ]
Error: INSTALLATION FAILED: Kubernetes cluster unreachable: Get "http://localhost:8080/version": dial tcp 127.0.0.1:8080: connect: connection refu
sed
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % minikube start                                                                                       ]
😄  minikube v1.32.0 on Darwin 14.2.1 (arm64)
✨  Using the docker driver based on existing profile
👍  Starting control plane node minikube in cluster minikube
🚜  Pulling base image ...
🔄  Restarting existing docker container for "minikube" ...
🐳  Preparing Kubernetes v1.28.3 on Docker 24.0.7 ...
🔗  Configuring bridge CNI (Container Networking Interface) ...
🔎  Verifying Kubernetes components...
    ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
🌟  Enabled addons: storage-provisioner, default-storageclass
🏄  Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm install prometheus prometheus-community/prometheus                                              ]
NAME: prometheus
LAST DEPLOYED: Mon Jan 29 20:21:03 2024
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
The Prometheus server can be accessed via port 80 on the following DNS name from within your cluster:
prometheus-server.default.svc.cluster.local


Get the Prometheus server URL by running these commands in the same shell:
  export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=prometheus,app.kubernetes.io/instance=prometheus" -o jsonpath=
"{.items[0].metadata.name}")
```

```
   export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=alertmanager,app.kubernetes.io/instance=prometheus" -o jsonpat
h="{.items[0].metadata.name}")
   kubectl --namespace default port-forward $POD_NAME 9093
###############################################################################
######   WARNING: Pod Security Policy has been disabled by default since   #####
######            it deprecated after k8s 1.25+. use                       #####
######            (index .Values "prometheus-node-exporter" "rbac"         #####
###### .          "pspEnabled") with (index .Values                        #####
######            "prometheus-node-exporter" "rbac" "pspAnnotations")      #####
######            in case you still need it.                               #####
###############################################################################


The Prometheus PushGateway can be accessed via port 9091 on the following DNS name from within your cluster:
prometheus-prometheus-pushgateway.default.svc.cluster.local


Get the PushGateway URL by running these commands in the same shell:
   export POD_NAME=$(kubectl get pods --namespace default -l "app=prometheus-pushgateway,component=pushgateway" -o jsonpath="{.items[0].metadata.na
me}")
   kubectl --namespace default port-forward $POD_NAME 9091

For more information on running Prometheus, visit:
https://prometheus.io/
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % kubectl get pods
NAME                                                 READY   STATUS            RESTARTS        AGE
mongodb-stateful-set-0                               1/1     Running           5 (22h ago)     9d
mongodb-stateful-set-1                               1/1     Running           3 (22h ago)     7d1h
note-deployment-74cc946cd8-657ng                     1/1     Running           6 (31s ago)     7d1h
note-deployment-74cc946cd8-wj874                     1/1     Running           11 (30s ago)    9d
note-server-deployment-6fb5fcb67f-5xxvw              1/1     Running           6 (22h ago)     9d
note-server-deployment-6fb5fcb67f-rf66l              1/1     Running           7 (22h ago)     9d
prometheus-alertmanager-0                            0/1     ContainerCreating 0               27s
prometheus-kube-state-metrics-745b475957-2j4hn       1/1     Running           0               27s
prometheus-prometheus-node-exporter-dkrf6            1/1     Running           0               27s
prometheus-prometheus-pushgateway-6ccd698d79-hsvtr   0/1     Running           0               27s
prometheus-server-bc7ccb595-m67tj                    0/2     ContainerCreating 0               27s
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % kubectl get svc
NAME                                  TYPE           CLUSTER-IP       EXTERNAL-IP   PORT(S)         AGE
kubernetes                            ClusterIP      10.96.0.1        <none>        443/TCP         9d
mongodb-service                       ClusterIP      10.110.83.89     <none>        27017/TCP       9d
note-server-service                   ClusterIP      10.110.101.56    <none>        5000/TCP        9d
note-service                          LoadBalancer   10.110.13.20     <pending>     3000:30444/TCP  9d
prometheus-alertmanager               ClusterIP      10.101.26.89     <none>        9093/TCP        43s
prometheus-alertmanager-headless      ClusterIP      None             <none>        9093/TCP        43s
prometheus-kube-state-metrics         ClusterIP      10.98.168.75     <none>        8080/TCP        43s
prometheus-prometheus-node-exporter   ClusterIP      10.101.193.199   <none>        9100/TCP        43s
prometheus-prometheus-pushgateway     ClusterIP      10.101.111.212   <none>        9091/TCP        43s
prometheus-server                     ClusterIP      10.102.138.220   <none>        80/TCP          43s
```

```
prometheus-prometheus-pushgateway     ClusterIP      10.101.111.212   <none>        9091/TCP        43s
prometheus-server                     ClusterIP      10.102.138.220   <none>        80/TCP          43s
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % kubectl expose service prometheus-server --type=NodePort --target-port=9090 --name=prometheus-server-ex
t
service/prometheus-server-ext exposed
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % kubectl get svc
NAME                                  TYPE           CLUSTER-IP       EXTERNAL-IP   PORT(S)         AGE
kubernetes                            ClusterIP      10.96.0.1        <none>        443/TCP         9d
mongodb-service                       ClusterIP      10.110.83.89     <none>        27017/TCP       9d
note-server-service                   ClusterIP      10.110.101.56    <none>        5000/TCP        9d
note-service                          LoadBalancer   10.110.13.20     <pending>     3000:30444/TCP  9d
prometheus-alertmanager               ClusterIP      10.101.26.89     <none>        9093/TCP        2m44s
prometheus-alertmanager-headless      ClusterIP      None             <none>        9093/TCP        2m44s
prometheus-kube-state-metrics         ClusterIP      10.98.168.75     <none>        8080/TCP        2m44s
prometheus-prometheus-node-exporter   ClusterIP      10.101.193.199   <none>        9100/TCP        2m44s
prometheus-prometheus-pushgateway     ClusterIP      10.101.111.212   <none>        9091/TCP        2m44s
prometheus-server                     ClusterIP      10.102.138.220   <none>        80/TCP          2m44s
prometheus-server-ext                 NodePort       10.104.11.167    <none>        80:30289/TCP    6s
xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % minikube ip
192.168.58.2
```

2.   Install Grafana

```
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm repo add grafana https://grafana.github.io/helm-charts
"grafana" has been added to your repositories
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "grafana" chart repository
...Successfully got an update from the "prometheus-community" chart repository
Update Complete. *Happy Helming!*
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % helm install grafana grafana/grafana
NAME: grafana
LAST DEPLOYED: Mon Jan 29 20:27:11 2024
NAMESPACE: default
STATUS: deployed
REVISION: 1
NOTES:
1. Get your 'admin' user password by running:

   kubectl get secret --namespace default grafana -o jsonpath="{.data.admin-password}" | base64 --decode ; echo


2. The Grafana server can be accessed via port 80 on the following DNS name from within your cluster:

   grafana.default.svc.cluster.local

   Get the Grafana URL to visit by running these commands in the same shell:
      export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=grafana,app.kubernetes.io/instance=grafana" -o jsonpath="{.
items[0].metadata.name}")
      kubectl --namespace default port-forward $POD_NAME 3000

3. Login with the password from step 1 and the username: admin
#################################################################################
######   WARNING: Persistence is disabled!!! You will lose your data when   #####
######            the Grafana pod is terminated.                            #####
#################################################################################
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % kubectl expose service grafana --type=NodePort --target-port=3000  --name=grafana-ext
service/grafana-ext exposed
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % kubectl get secret --namespace default grafana -o jsonpath="{.data.admin-password}" | base64 --decode ;]
 echo
ZJJ0wkyfaGNz4yzf3T1Wj6rKiwTG1AqOksA0oTq1
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % export POD_NAME=$(kubectl get pods --namespace default -l "app.kubernetes.io/name=grafana,app.kubernete]
s.io/instance=grafana" -o jsonpath="{.items[0].metadata.name}")
[xiaogangdong@Xiaogangs-MacBook-Air Lab 3 % kubectl --namespace default port-forward $POD_NAME 3000
Forwarding from 127.0.0.1:3000 -> 3000
Forwarding from [::1]:3000 -> 3000
Handling connection for 3000
Handling connection for 3000
Handling connection for 3000
Handling connection for 3000
```

3.   Screen of Grafana dashboard