CKA Curriculum

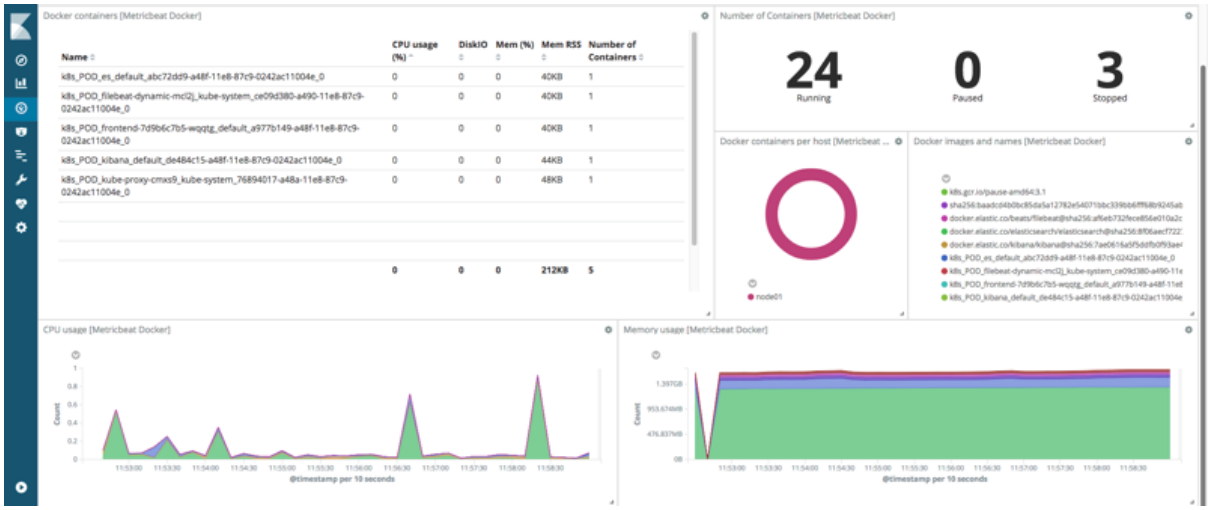
Logging and Monitoring Section

Kubernetes Monitoring using Elasticsearch and Beats

To monitor an application running in Kubernetes (k8s), you need logs and metrics from the app, as well as, the k8s environment it's running in. Using Elasticsearch, Kibana, and Beats allows you to collect, search, analyze and visualize all of this data about the app and the k8s (pods, containers, etc) in one place.

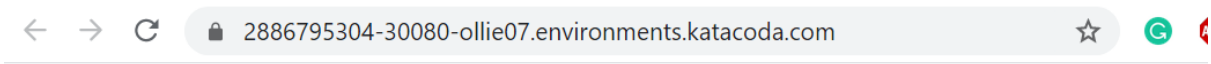
Goal

This is one of the out of the box dashboards that you will see once you deploy the Elastic Stack in this Katacoda environment. This is the Docker metrics dashboard that ships with Metricbeat. It shows an overview of the CPU and Memory use of every container, allows you to drill in to a specific container, and the containers per node. Looking at the dashboard is much easier than running the equivalent kubectl get, top, describe, etc. commands.



Make some entries in the Guestbook

Once the pods are all running, switch to the **Guestbook** tab and enter some messages into the Guestbook.



Guestbook

Messages

Submit

what is your name?
hello
how are doing today?
I hope covid19 has not made much impact to your loved ones?
which city you live in?

```
master $ kubectl get all
NAME                                READY   STATUS              RESTARTS   AGE
pod/es                              0/1     ContainerCreating   0          11s
pod/frontend-8bd5d6b48-9tk2d        1/1     Running             0          3m5s
pod/redis-master-5bc9f68944-4fpvx   1/1     Running             0          3m5s
pod/redis-slave-96685cfdb-9tpkz     1/1     Running             0          3m5s

NAME                    TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
service/elasticsearch   ClusterIP   10.102.67.41     <none>        9200/TCP       11s
service/frontend        NodePort    10.109.141.205   <none>        80:30080/TCP   3m5s
service/kubernetes      ClusterIP   10.96.0.1        <none>        443/TCP        3h3m
service/redis-master    ClusterIP   10.108.154.252   <none>        6379/TCP       3m6s
service/redis-slave     ClusterIP   10.106.34.54     <none>        6379/TCP       3m5s
```

```
NAME                           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/frontend       1/1     1            1           3m5s
deployment.apps/redis-master   1/1     1            1           3m6s
deployment.apps/redis-slave    1/1     1            1           3m5s

NAME                                      DESIRED   CURRENT   READY   AGE
replicaset.apps/frontend-8bd5d6b48        1         1         1       3m5s
replicaset.apps/redis-master-5bc9f68944   1         1         1       3m6s
replicaset.apps/redis-slave-96685cfdb     1         1         1       3m5s
```

Verify that Elasticsearch is running

```
master $ kubectl get pods
NAME                           READY   STATUS    RESTARTS   AGE
es                             1/1     Running   0          4m21s
frontend-8bd5d6b48-9tk2d       1/1     Running   0          7m15s
redis-master-5bc9f68944-4fpvx  1/1     Running   0          7m15s
redis-slave-96685cfdb-9tpkz    1/1     Running   0          7m15s
master $ kubectl logs es | grep "mode \[basic\] - valid"
[2020-05-02T02:51:45,602][INFO ][o.e.l.LicenseService     ] [FOZK_rK] license [b16837b6-4a5e-49fd-b6
33-05fe1ce81868] mode [basic] - valid
```

Deploy Kibana

```
master $ kubectl apply -f /root/course/kibana.yaml
pod/kibana created
service/kibana created
service/kibana-internal created
master $
```

Verify Kibana is running

```
master $ kubectl get pods
NAME                           READY   STATUS    RESTARTS   AGE
es                             1/1     Running   0          8m24s
frontend-8bd5d6b48-9tk2d       1/1     Running   0          11m
kibana                         1/1     Running   0          38s
redis-master-5bc9f68944-4fpvx  1/1     Running   0          11m
redis-slave-96685cfdb-9tpkz    1/1     Running   0          11m
master $
```

Check the Kibana logs:

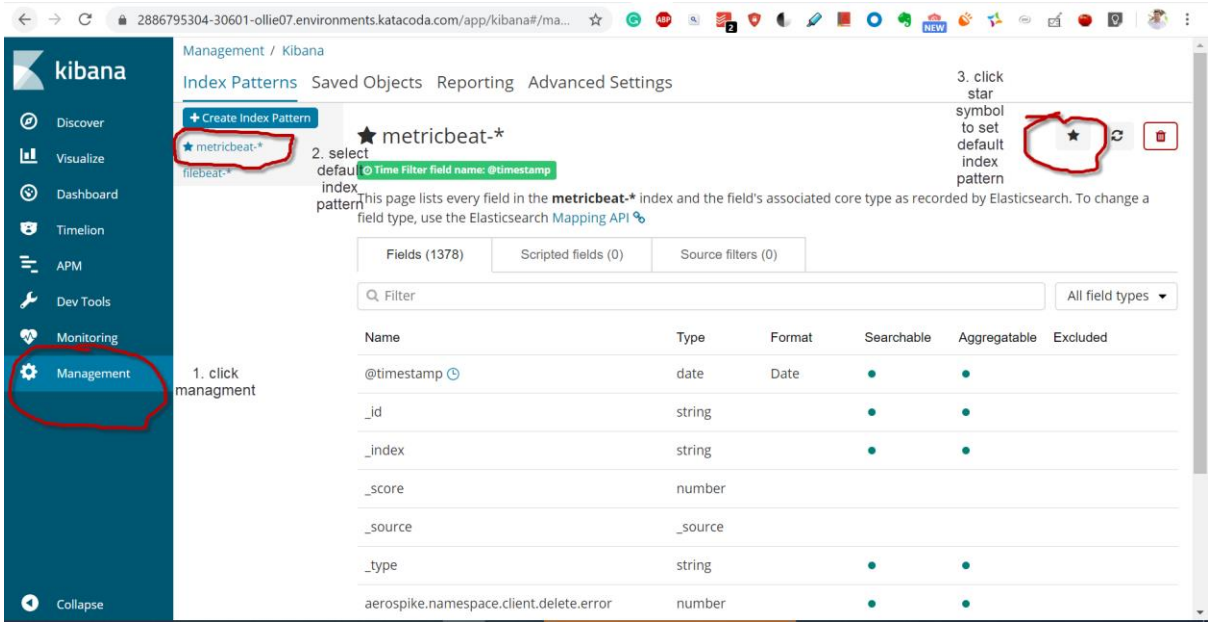
```
master $ kubectl logs kibana|grep "Status changed from yellow to green"
{"type":"log","@timestamp":"2020-05-02T02:59:40Z","tags":["status","plugin:elasticsearch@6.3.2","inf
o"],"pid":1,"state":"green","message":"Status changed from yellow to green - Ready","prevState":"yel
low","prevMsg":"Waiting for Elasticsearch"}
{"type":"log","@timestamp":"2020-05-02T02:59:40Z","tags":["status","plugin:xpack_main@6.3.2","info"]
,"pid":1,"state":"green","message":"Status changed from yellow to green - Ready","prevState":"yellow
","prevMsg":"Waiting for Elasticsearch"}
{"type":"log","@timestamp":"2020-05-02T02:59:40Z","tags":["status","plugin:searchprofiler@6.3.2","in
fo"],"pid":1,"state":"green","message":"Status changed from yellow to green - Ready","prevState":"ye
llow","prevMsg":"Waiting for Elasticsearch"}
{"type":"log","@timestamp":"2020-05-02T02:59:40Z","tags":["status","plugin:ml@6.3.2","info"],"pid":1
,"state":"green","message":"Status changed from yellow to green - Ready","prevState":"yellow","prevM
sg":"Waiting for Elasticsearch"}
{"type":"log","@timestamp":"2020-05-02T02:59:40Z","tags":["status","plugin:tilemap@6.3.2","info"],"p
id":1,"state":"green","message":"Status changed from yellow to green - Ready","prevState":"yellow","
prevMsg":"Waiting for Elasticsearch"}
```

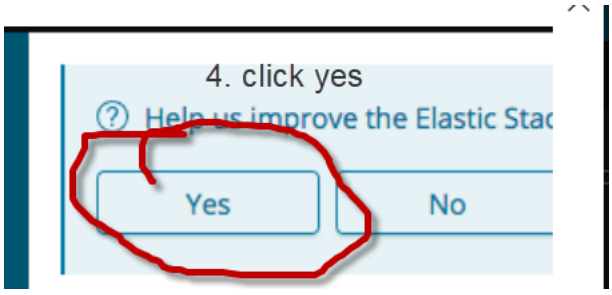
Deploy Filebeat

Filebeat will automatically discover the running pods, find the proper files, configure Elasticsearch to parse the logs, and configure Kibana with sample visualizations and dashboards by looking at the available metadata and applying technology specific modules.

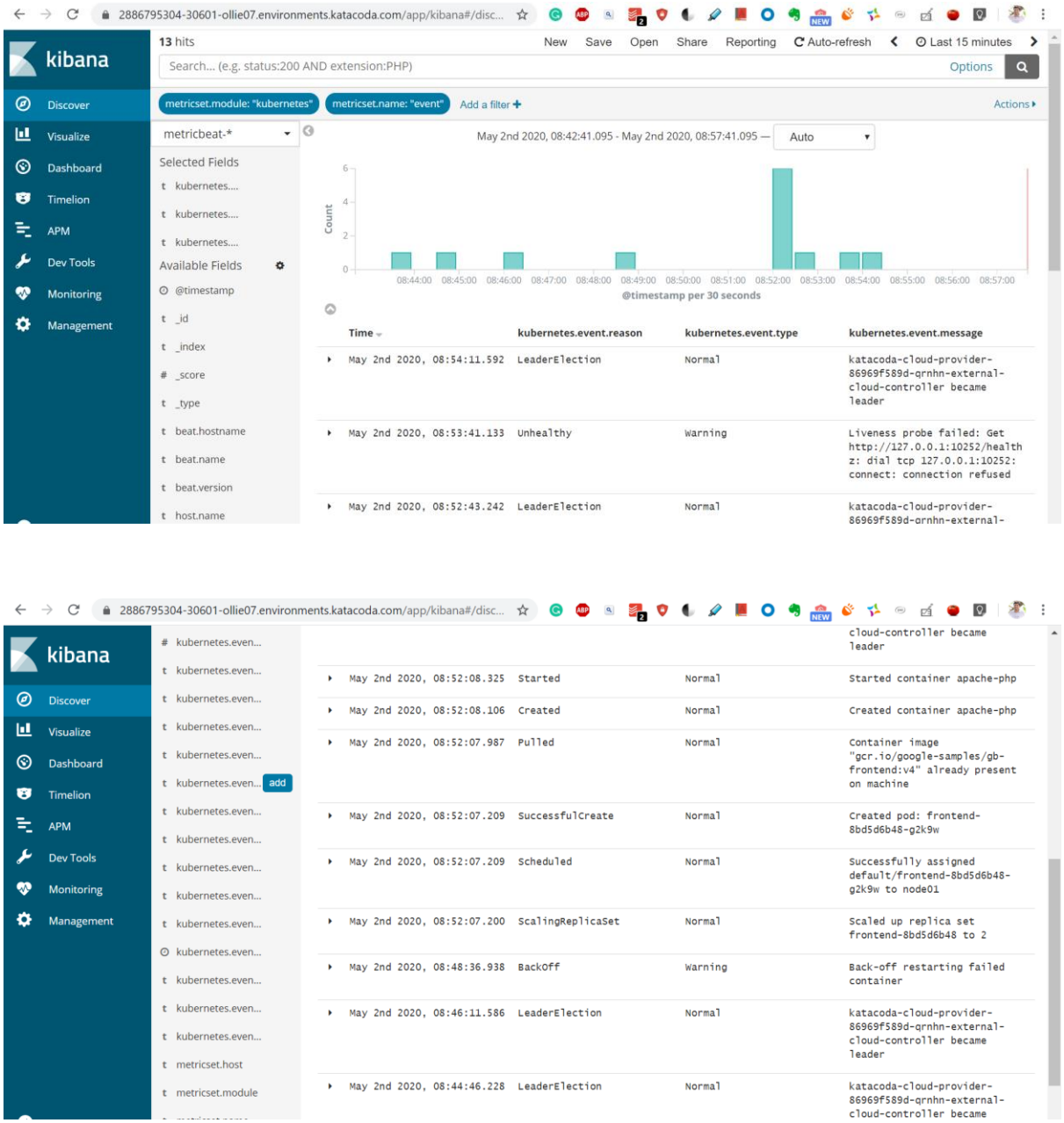
```
master $ kubectl get pods -n kube-system | grep filebeat
filebeat-dynamic-5pptn                   1/1     Running            0          17s
master $
```

Explore your logs & metrics in Kibana

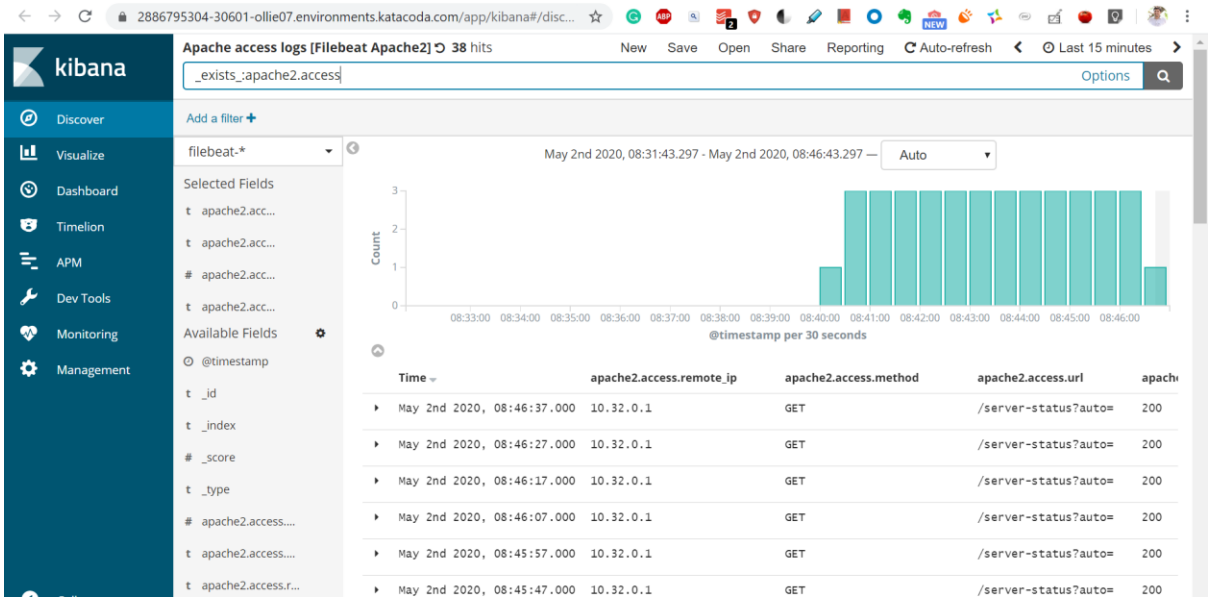




Examine events from kube-state-metrics



Apache Dashboards



During this tutorial we created a Kubernetes cluster, deployed a sample application, deployed Filebeat from Elastic, configured Filebeat to connect to an Elasticsearch Service deployment running in Elastic Cloud, and viewed logs and metrics in the Elasticsearch Service Kibana.

References:

https://www.katacoda.com/dan_roscigno/scenarios/logs-and-metrics-elasticsearch-kibana

https://www.elastic.co/beats/filebeat

https://www.elastic.co/

https://github.com/swaroopcs88/k8s_hands_on

https://bit.ly/3fdcBeY

https://bit.ly/3faRdqS

https://bit.ly/2KOWUg8