CKA Curriculum

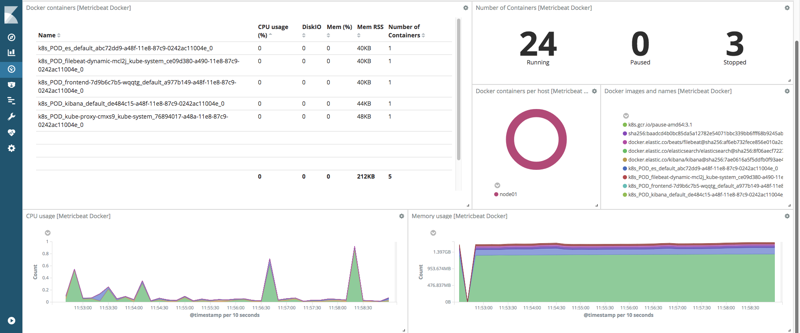
Logging and Monitoring Section

**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.

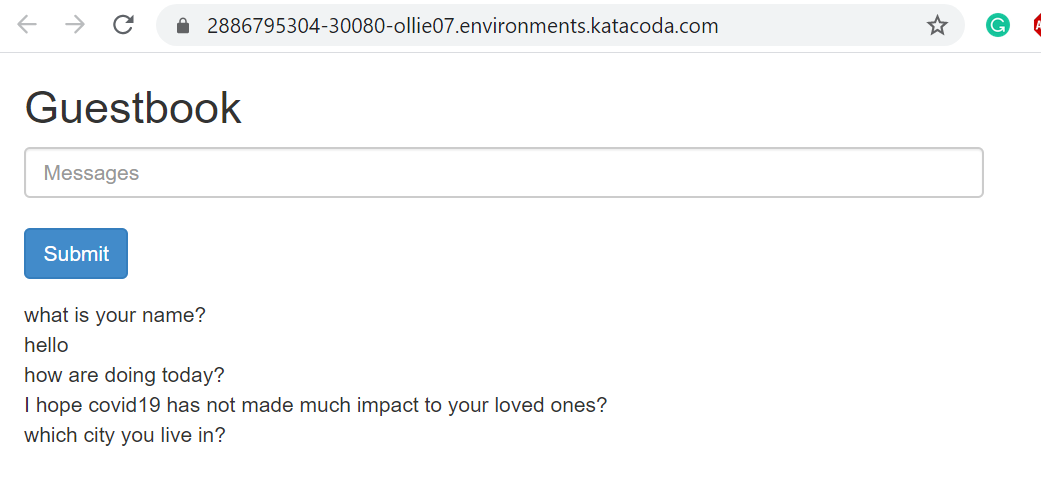
### **Goal**

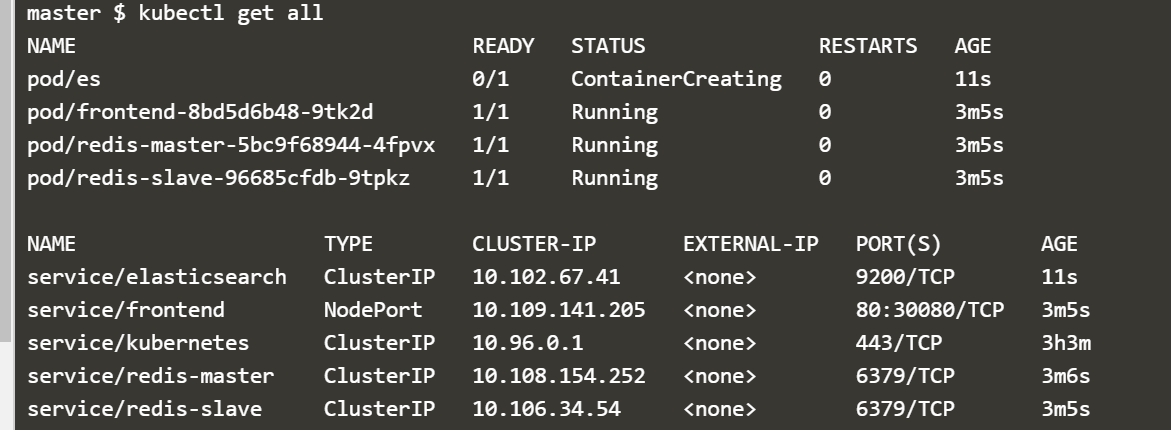
This is one of the out of the box dashboards that you will see once you deploy the Elastic Stack in this Katacoda environment. This is the Docker metrics dashboard that ships with Metricbeat. It shows an overview of the CPU and Memory use of every container, allows you to drill in to a specific container,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

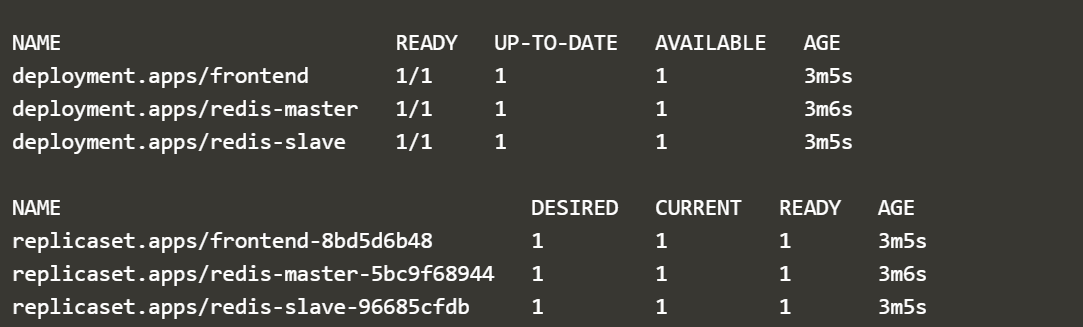


### Make some entries in the Guestbook

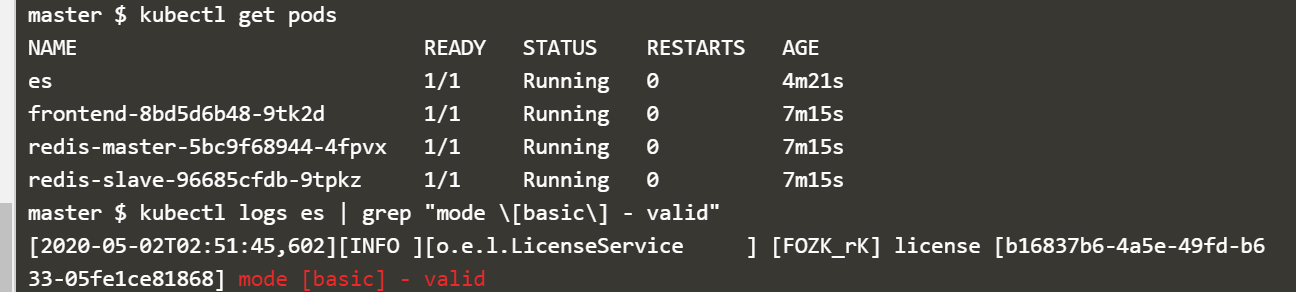
Once the pods are all running, switch to the **Guestbook** tab and enter some messages into the Guestbook.



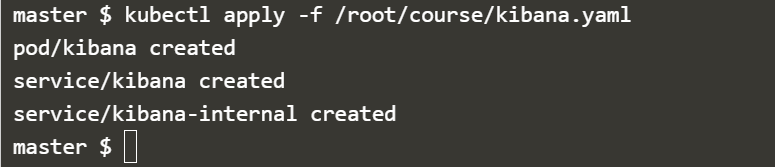




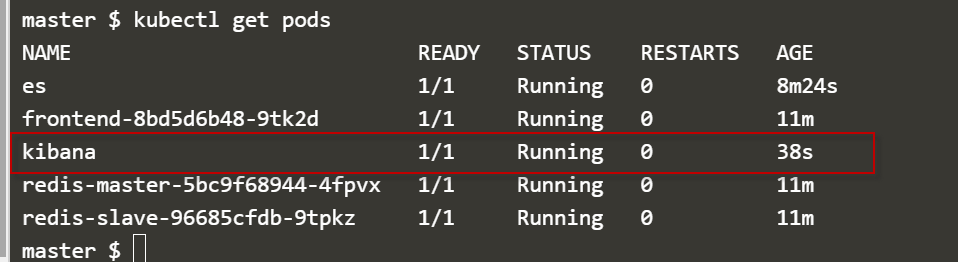




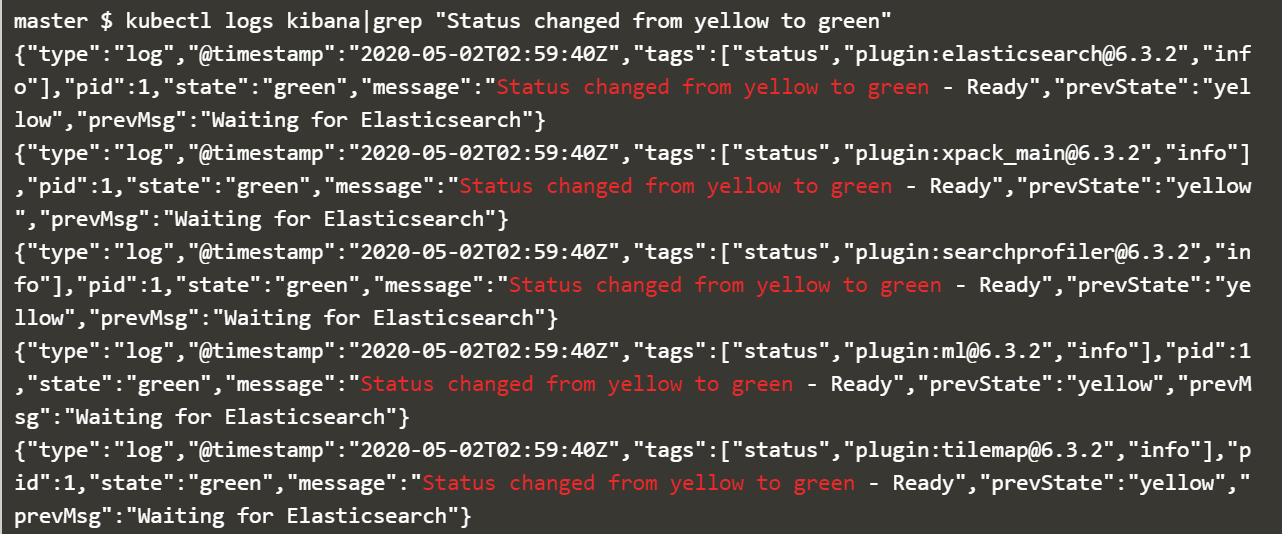
Deploy Kibana



Verify Kibana is running



Check the Kibana logs:

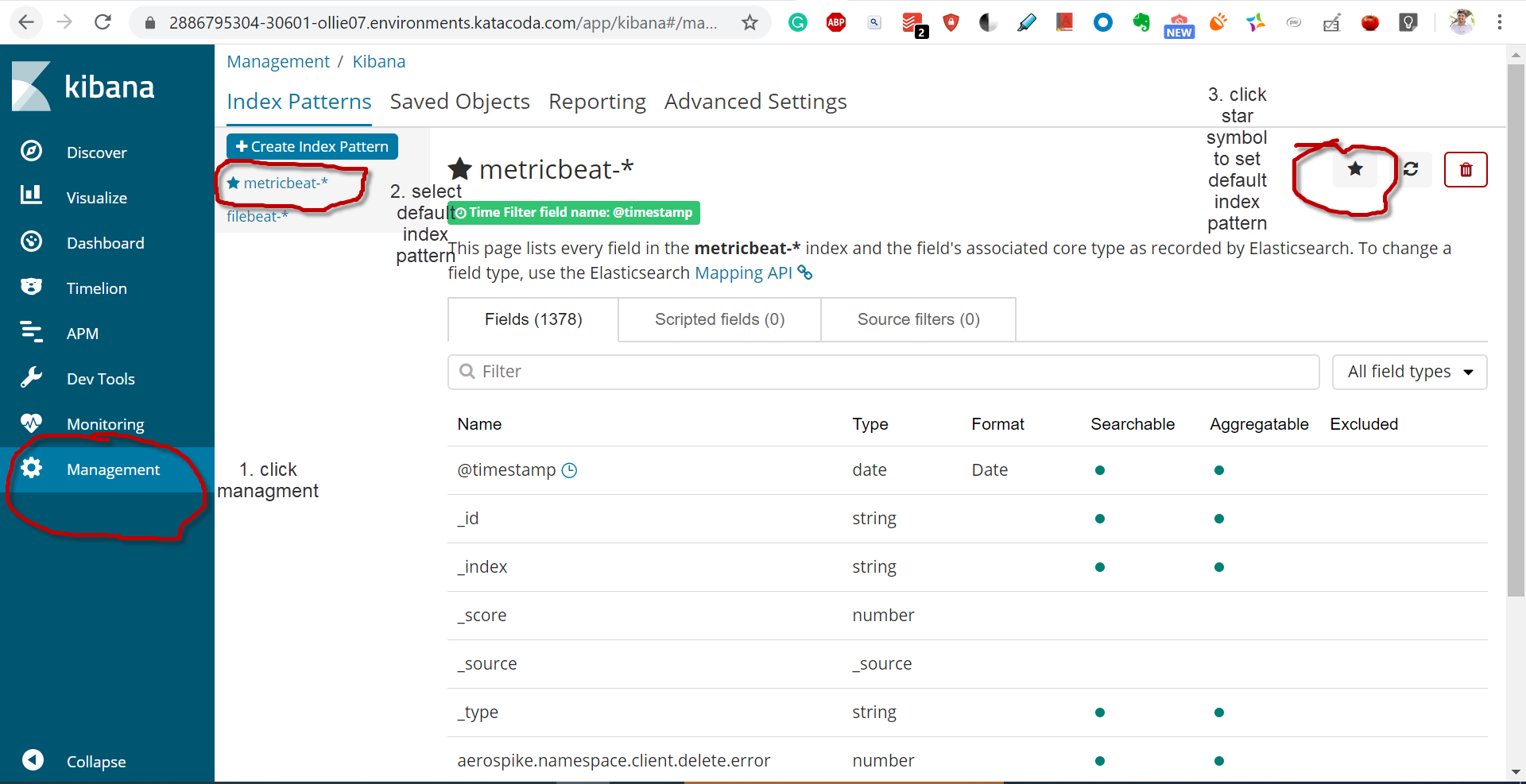


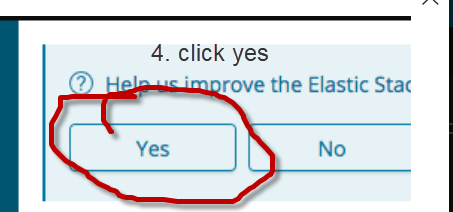
Deploy Filebeat

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.

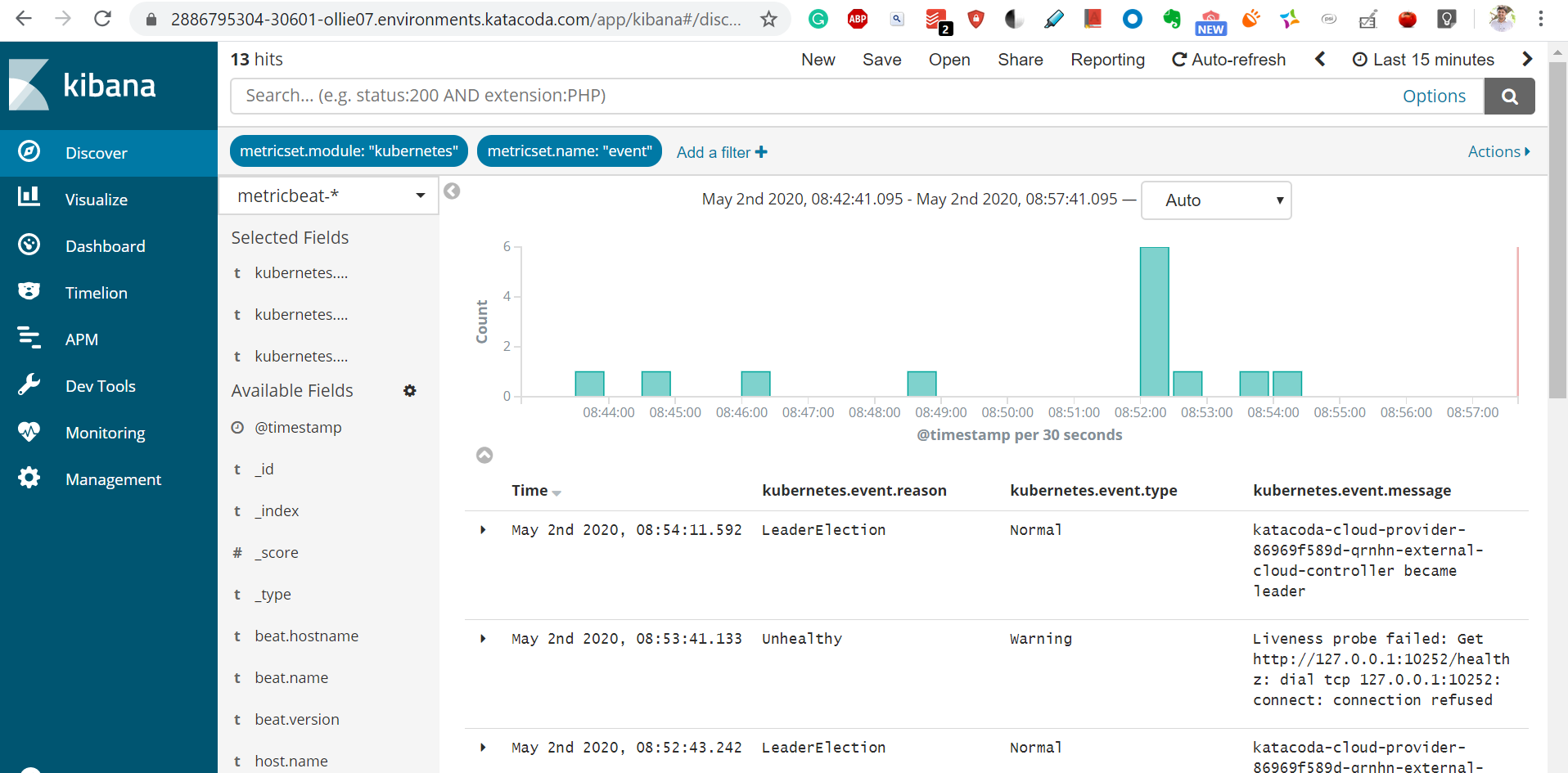


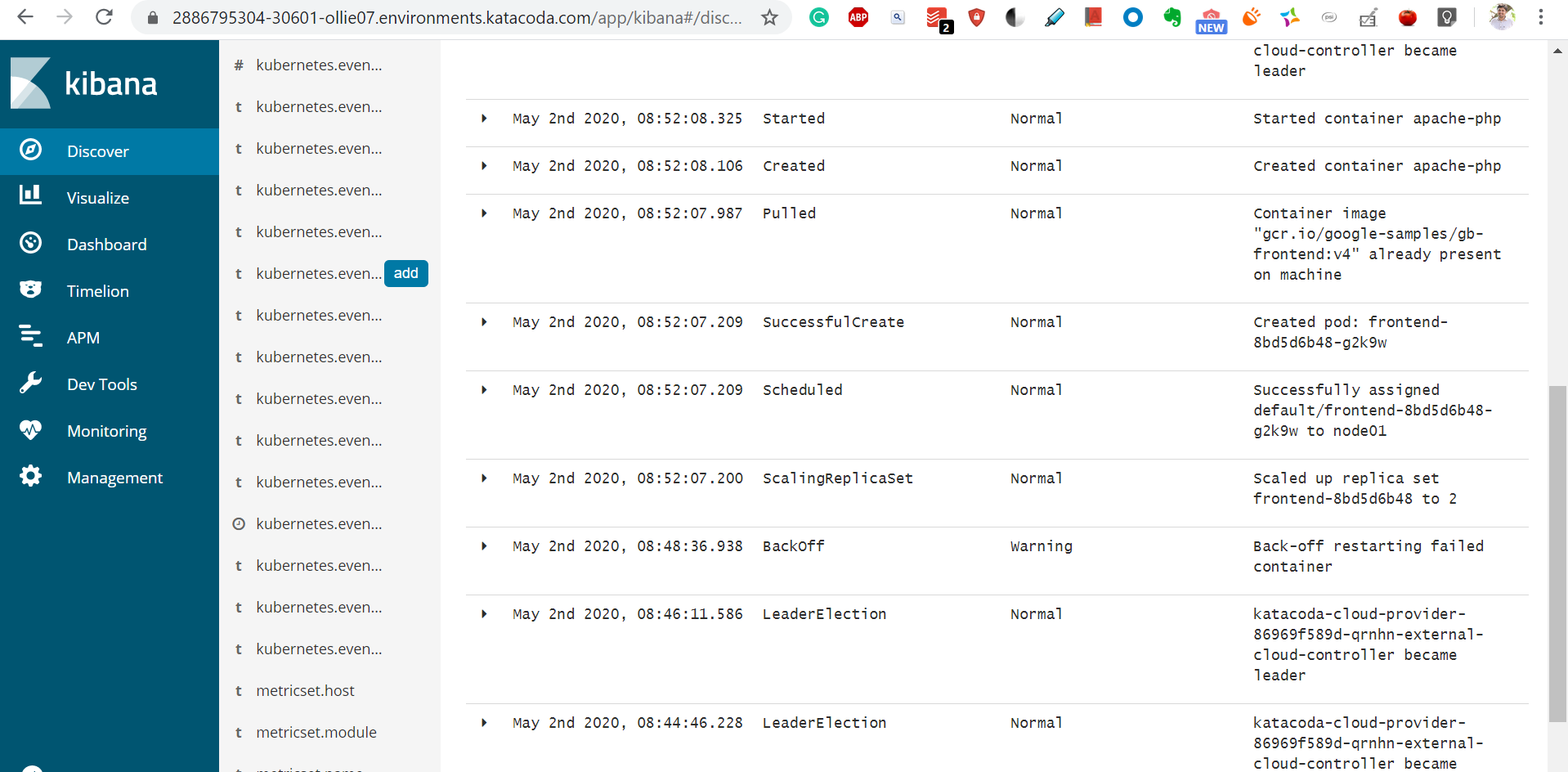
#### Explore your logs & metrics in Kibana



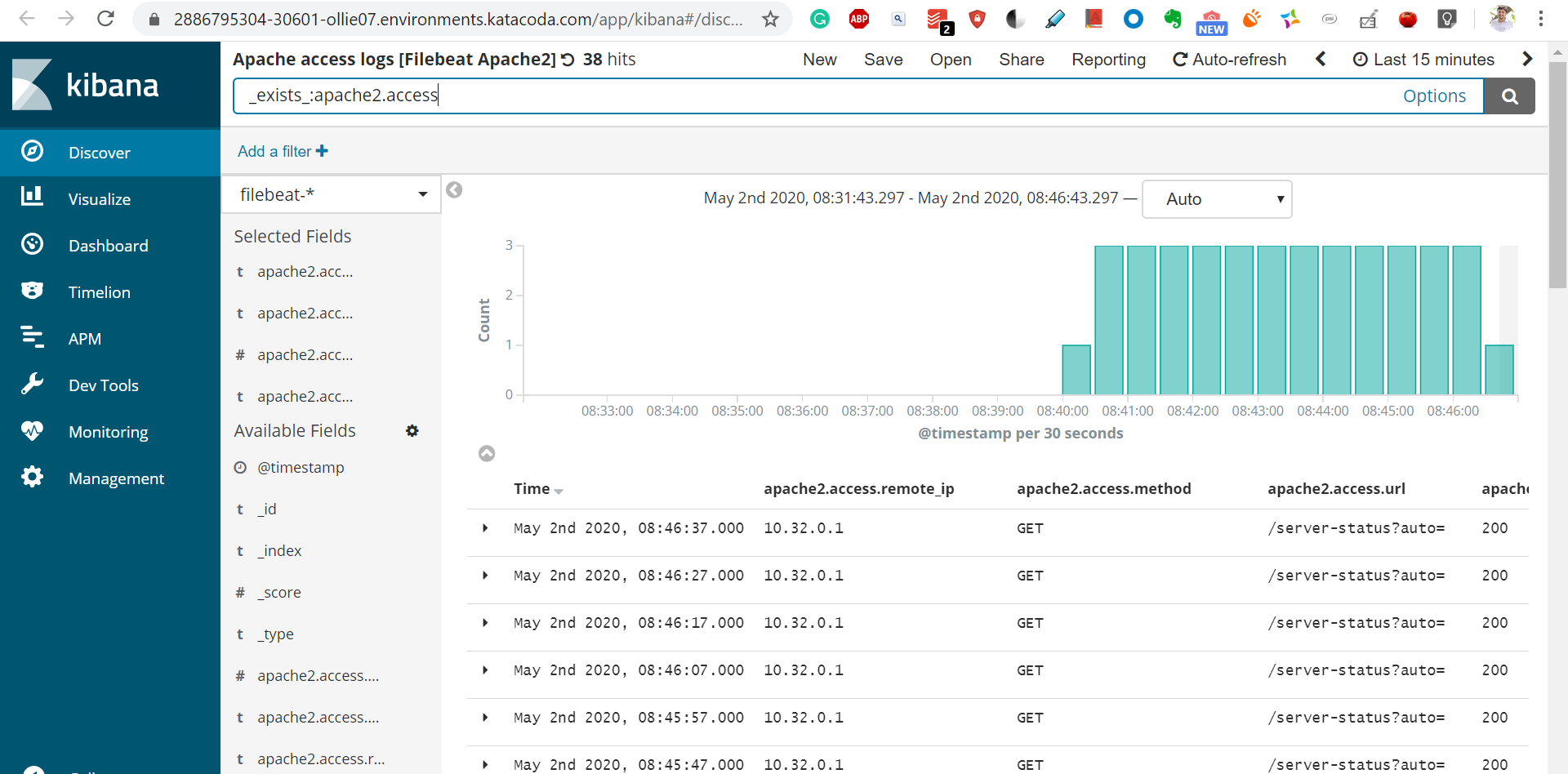


#### Examine events from kube-state-metrics





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.