Monitoring Your EKS Cluster with ELK: A Step-by-Step Guide

I use Terraform to provision an EKS cluster.

1: Deploy Elasticsearch, Logstash, and Kibana on EKS

The ELK stack consists of three main components:

Elasticsearch: Stores logs and metrics.
Logstash: Processes and forwards logs.
Kibana: Provides a visual interface for data exploration.

To simplify the deployment process, I will use **Helm**, a package manager for Kubernetes.

1. Install Helm

```
curl https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash
```

Now, adding the **Elastic Helm repository** and update it:

```
helm repo add elastic https://helm.elastic.co
helm repo update
```

2. Deploy Elasticsearch

```
helm install elasticsearch elastic/elasticsearch --namespace logging --create-namespace
```

3. Deploy Kibana

```
helm install kibana elastic/kibana --namespace logging
```

4. Deploy Logstash

Before deploying Logstash, we need to create a **ConfigMap** to define its configuration:

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: logstash-config
  namespace: logging
data:
  logstash.conf: |
    input {
      beats {
        port => 5044
      }
    }
    filter {
      grok {
        match => { "message" => "%{COMBINEDAPACHELOG}" }
      }
    }
    output {
      elasticsearch {
        hosts => ["http://elasticsearch-master:9200"]
        index => "eks-logs-%{+YYYY.MM.dd}"
      }
    }
```

Apply the ConfigMap:

```
kubectl apply -f logstash-config.yaml
```

Now deploy Logstash using Helm:

```
helm install logstash elastic/logstash --namespace logging
```

2: Collect CPU and Memory Metrics with Metricbeat

To track **CPU and memory usage**, we use **Metricbeat**, an Elastic Agent that collects and forwards system metrics.

1. Deploy Metricbeat

```
helm install metricbeat elastic/metricbeat --namespace logging
```

2. Configure Metricbeat

To enable **Kubernetes monitoring**, I modify the **Metricbeat ConfigMap**:

```
kubectl edit configmap metricbeat -n logging
```

Ensure the following configuration is present:

```
processors:
  - add_kubernetes_metadata:
      host: ${NODE_NAME}
      matchers:
        - logs_path:
            logs_path: "/var/log"
output.elasticsearch:
  hosts: ["http://elasticsearch-master:9200"]
```

After making changes, restart Metricbeat:

```
kubectl rollout restart daemonset metricbeat -n logging
```

3: View Metrics in Kibana

1. Get Kibana's Service URL

To find Kibana's external IP, run:

```
kubectl get svc -n logging
```