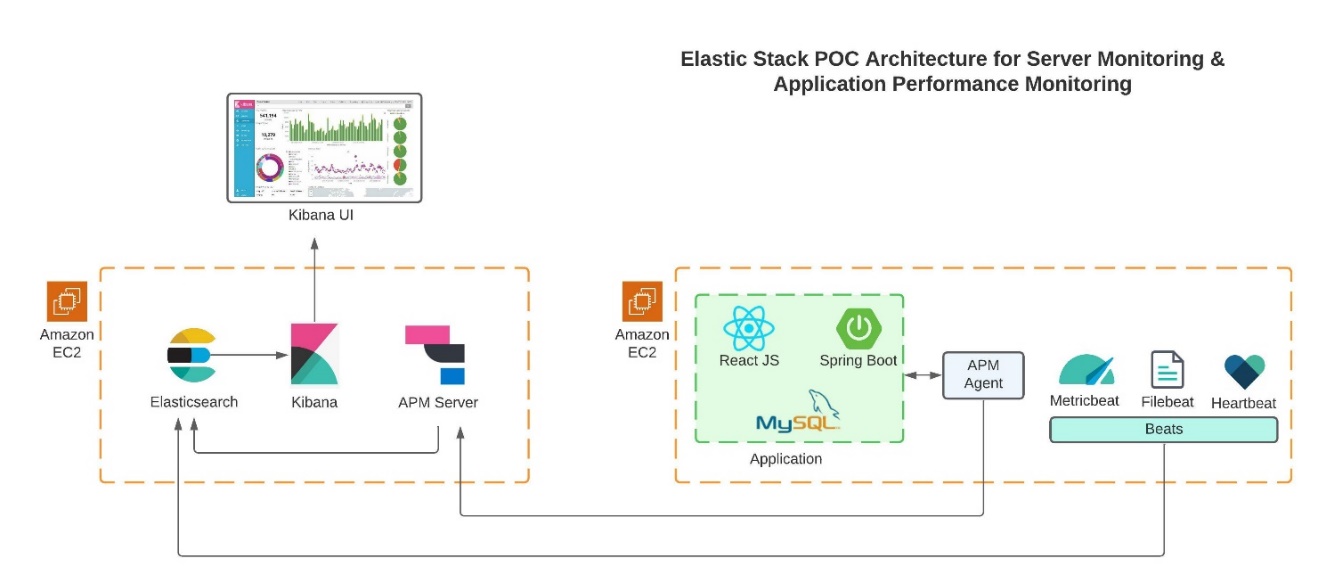
**ARCHITECTURE DIAGRAM**



**ELASTICSEARCH, KIBANA & APM-SERVER CONFIGURATION**

**(server 1)**

**Install Elasticsearch**

wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.17.8-amd64.deb

sudo dpkg -i elasticsearch-7.17.8-amd64.deb

sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable elasticsearch.service

sudo systemctl start elasticsearch.service

sudo systemctl stop elasticsearch.service

**Configure Elasticsearch**

sudo su

vi /etc/elasticsearch/elasticsearch.yml

cluster.name: my-cluster

network.host: 0.0.0.0

http.port: 9200

xpack.monitoring.collection.enabled: true

xpack.monitoring.elasticsearch.collection.enabled: true

**Install Kibana**

wget https://artifacts.elastic.co/downloads/kibana/kibana-7.17.8-amd64.deb

sudo dpkg -i kibana-7.17.8-amd64.deb

sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable kibana.service

sudo systemctl start kibana.service

sudo systemctl stop kibana.service

**Configure Kibana**

sudo su

vi /etc/kibana/kibana.yml

server.port: 5601

server.host: "0.0.0.0"

elasticsearch.hosts: ["http://public\_ip:9200"]

**Install APM Server**

wget https://artifacts.elastic.co/downloads/apm-server/apm-server-7.17.8-amd64.deb

sudo dpkg -i apm-server-7.17.8-amd64.deb

sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable apm-server

sudo systemctl start apm-server

sudo systemctl stop apm-server

**Configure APM Server**

apm-server:

host: "0.0.0.0:8200"

rum:

enabled: true

allow\_origins: ['\*']

allow\_headers: []

source\_mapping:

enabled: true

elasticsearch:

hosts: ["public\_ip:9200"]

**METRICBEAT AND FILEBEAT CONFIGURATION**

**(server 2)**

**Install MetricBeat on Application server**

wget https://artifacts.elastic.co/downloads/beats/metricbeat/metricbeat-7.17.8-amd64.deb

sudo dpkg -i metricbeat-7.17.8-amd64.deb

sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable metricbeat.service

sudo systemctl start metricbeat.service

sudo systemctl stop metricbeat.service

**Configure Metricbeat**

sudo su

vi /etc/metricbeat/metricbeat.yml

live monitoring section

reload.enabled: true

reload.period: 10s

name: webserver

Tags: demo-web

### enable kibana dashboards

Setup.dashboards.enabled: true

host: public-ip:5601

### output configuration

output.elasticsearch:

# Array of hosts to connect to.

hosts: ["public-ip:9200"]

protocol: "https"

logging.level: debug

logging.selectors: ["\*"]

logging.to\_files: true

logging.files:

path: /var/log/metricbeat

name: metricbeat

keepfiles: 7

permissions: 0644

## enable interal collection of metricbeat

monitoring.enabled: true

monitoring.elasticsearch: true

sudo systemctl start metricbeat.service

sudo metricbeat modules enable apache

sudo metricbeat modules enable system

sudo metricbeat modules enable mysql

sudo metricbeat modules enable beat-xpack

**Install FileBeat on Application server**

wget https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-7.17.8-amd64.deb

sudo dpkg -i filebeat-7.17.8-amd64.deb

sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable filebeat.service

sudo systemctl start filebeat.service

sudo systemctl stop filebeat.service

**Configure Filebeat**

sudo su

vi /etc/filebeat/filebeat.yml

reload.enabled: true

reload.period: 10s

setup.dashboards.enabled: true

kibana:

host: "public-ip:5601"

elasticsearch.output

hosts: ["public-ip:9200"]

protocol: "http"

logging.to\_files: true

logging.files:

path: /var/log/filebeat

name: filebeat

keepfiles: 7

permissions: 0644

http.enabled: true

http.port: 5067

**Enable beat-xpack if not enabled**

sudo metricbeat modules enable beat-xpack

**Configure beat-xpack**

sudo nano /etc/metricbeat/modules.d/beat-xpack.yml

hosts: ["http://localhost:5067"]

sudo systemctl start filebeat

sudo filebeat modules enable apache

sudo filebeat modules enable system

sudo filebeat modules enable mysql

**DEMO APPLICATION CONFIGURATION**

**(server 2)**

**For testing purpose we used Java Backend and React frontend**

git clone https://github.com/Shubhamb7/employee-mgmt.git

Make changes to the ips -> localhost to public ip

Create a Jar file of the springboot application. First install maven in order to create jar file.

sudo apt-get install maven default-jre npm

sudo npm update npm -g

cd springboot-backend

mvn clean install

Jar file will be in target folder, move it to root folder

**Installing APM Agent for Java Backend**

Download the apm agent from below link:

<http://search.maven.org/#search%7Cga%7C1%7Ca%3Aelastic-apm-agent>

Run the Application jar file with the Java APM Agent:

sudo java -javaagent:elastic-apm-agent-1.26.0.jar \

-Delastic.apm.service\_name=my-application \

-Delastic.apm.server\_urls=http:// 43.204.38.64:8200 \

-Delastic.apm.secret\_token= \

-Delastic.apm.environment=dev \

-Delastic.apm.application\_packages=org.example \

-jar ~/springboot-backend/target/<name>.jar

**Installing APM Agent for Nodejs Backend**

npm install elastic-apm-node --save

Include below code in the App.js or server.js or index.js file

// Add this to the VERY top of the first file loaded in your app

var apm = require('elastic-apm-node').start({

// Override the service name from package.json

// Allowed characters: a-z, A-Z, 0-9, -, \_, and space

serviceName: '',

// Use if APM Server requires a secret token

secretToken: '',

// Set the custom APM Server URL (default: http:// localhost:8200)

serverUrl: 'http:// public\_ip:8200',

// Set the service environment

environment: 'production'

})

Nodejs APM agent should start when application is running - npm start

**Installing Real User Monitoring (RUM) Agent for frontend**

npm i @elastic/apm-rum-react

npm i @elastic/apm-rum

**Configuring RUM Agent with to work with React js code**

Insert following code in index.js or App.js or Server.js

import { init as initApm } from "@elastic/apm-rum";

import { ApmRoute } from "@elastic/apm-rum-react";

const apm = initApm({

serviceName: "web",

serverUrl: "http://public\_ip:8200",

logLevel: "debug",

serviceVersion: "1.0"

});

Instead of Route use ApmRoute in the code.

**Database setup**

sudo apt-get install mysql-server –y

sudo mysql\_secure\_installation

mysql –u root –p

-> create user ‘tom’@’%’ identified by ‘password’;

-> GRANT ALL PRIVILEGES ON mydb.\* TO ‘tom’@’%’;

sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf

bind-address : 0.0.0.0