Cloud Computing Applications and Services

Monitoring

2023

ELK

The main goal of this guide is to deploy and use a modular system monitoring tool to analyse the execution of the Swap application. Along this exercise guide, we will go through the steps of configuring and deploying of the following ELK components:

- Elasticsearch: `https://www.elastic.co/elasticsearch`
- Kibana: `https://www.elastic.co/kibana`
- Metricbeat: `https://www.elastic.co/beats/metricbeat`

Tasks

VMs Setup

1. Create two VMs (*server1* and *server2*). *server1* should have at least 2GB of RAM. You may use the Vagrantfile provided along with this guide.
2. At each VM, run the script *install_g5.sh*, provided along with this guide, to install Docker:
   `bash install_g5.sh install_docker`
3. Next, we will use Elastic's official Docker images to install and run Elasticsearch and Kibana on *server1*, and Metricbeat on *server2* along with the Swap application.

Important: Do not destroy these VMs, as they will be needed for the next practical guide.

[*server1*] Elasticsearch and Kibana Installation

1. Create a Docker network: `docker network create elastic`
2. Create a folder to persist Elasticsearch data: `mkdir -p $HOME/esdata`
3. Install and run Elasticsearch:

   ```
   docker run --name elasticsearch --net elastic -p 9200:9200 -p 9300:9300 \
     -e "http.host=0.0.0.0" -e "transport.host=127.0.0.1" \
     -e "xpack.security.enabled=false" \
     -v $HOME/esdata:/usr/share/elasticsearch/data \
     docker.elastic.co/elasticsearch/elasticsearch:8.5.2
   ```

   Note: Since the container is running in the foreground, this command will hang until we stop it.
4. On another terminal, install and run Kibana:

   ```
   docker run --name kibana --net elastic -p 80:5601 \
     -e ELASTICSEARCH_HOSTS="http://<SERVER1_IP>:9200"  \
     -e "server.host=<SERVER1_IP>" docker.elastic.co/kibana/kibana:8.5.2
   ```

   Note: Since the container is running in the foreground, this command will hang until we stop it.

[*server2*] Swap and Metricbeat Installation

1. Install Swap using the provided script: `bash install_g5.sh install_swap`
2. Verify if Swap is running correctly (*i.e.,* access Swap webpage at `http://<SERVER2_IP>:80`)
3. Remotely setup Metricbeat's dashboards in Kibana:

   ```
   docker run --rm --network=host docker.elastic.co/beats/metricbeat:8.5.2 \
    setup --dashboards -E setup.kibana.host=<SERVER1_IP>:80
   ```

   Note: Metricbeat will connect to Kibana at *server1* to setup custom dashboards.
4. Start Metricbeat:

   ```
   docker run --name metricbeat --user="root" -v /proc:/hostfs/proc:ro \
     -v /sys/fs/cgroup:/hostfs/sys/fs/cgroup:ro -v /:/hostfs:ro \
     -v /var/run/docker.sock:/var/run/docker.sock:ro \
     -e LIBBEAT_MONITORING_CGROUPS_HIERARCHY_OVERRIDE=/hostfs \
     --network=host docker.elastic.co/beats/metricbeat:8.5.2 \
     --strict.perms=false -system.hostfs=/hostfs -e \
     -E output.elasticsearch.hosts=["<SERVER1_IP>:9200"]
   ```

   Note: Since the container is running in the foreground, this command will hang until we stop it.

[*browser*] Kibana's Dashboards

1. Open Kibana at `http://<SERVER1_IP>:80`
2. Observe summarized data in the `Analytics => Dashboard` page (e.g., *"[Metricbeat System] Host overview ECS"*).
3. At *server2*, seed Swap database and observe the alterations in Kibana's dashboards:

   ```
   docker exec -it swapapp /bin/sh -c "php artisan db:seed"
   ```
4. Explore the other menus from Kibana.

Extra

1. Create your own dashboard and include predefined and custom vizualizations in it.
   (a) Go to *"Dashboards"* and click on the *"Create dashboard"* button.
   (b) Add a predefined visualization from the library by clicking on the *"Add from library"* button and selecting one visualization (e.g., *"CPU Usage [Metricbeat System] ECS"*).
   (c) Create a custom visualization that shows the percentage of CPU used over time (per process):
      i. Click on the *"Create visualization"* button.
      ii. Fill in the right side as follows:
         - Change *"Bar Vertical stacked"* to *"Line"*.
         - On the *"Horizontal axis"*, select *"Date histogram"* as the *"Function"* and *"@timestamp"* as the *"Field"*.
         - On the *" Vertical axis"*, select *"Average"* as the *"Function"* and *"system.process.cpu.total.pct"* as the *"Field"*.
         - On the *"Breakdown"*, select *"Top values"* as the *"Function"* and *"process.name"* as the *"Field"*.
   (d) Save the dashboard by click on the button *"Save and return"*.

Learning Outcomes

Recognize different roles in a modular monitoring pipeline. Apply the ELK stack to monitor and vizualize server and application resources.