ELK ---> Elastic, Logstack, Kinaba, filebeat

1. Install Filebeat on frontend.

https://logit.io/sources/configure/ubuntu -----> link for filebeat installtion

- curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-oss-7.15.1-amd64.deb

- sudo dpkg -i filebeat-oss-7.15.1-amd64.deb

2. to remove Filebeat

- sudo dpkg --purge filebeat

installation links for ELK stack

1. https://www.digitalocean.com/community/tutorials/how-to-install-elasticsearch-logstash-and-kibana-elastic-stack-on-ubuntu-18-04

2. https://asibin99.medium.com/install-elk-stack-on-ubuntu-20-04-18f6d748bbe0

\*Note ---> your filebeat and elasticsearch, logstash and kibana must have the same version.

$ cd /etc/filebeat/modules.d

In module enble nginx using follwing coomand

$ filebeat modules enable nginx

after enable some change with nginx.yml uncomment following give path of access log

----------------------------------------------------------------------------------------------------------------------------------------------------

- module: nginx

access:

enabled: true

var.paths: ["/path/to/log/nginx/access.log\*"]

error:

enabled: true

var.paths: ["/path/to/log/nginx/error.log\*"]

ingress\_controller:

enabled: false

-------------------------------------------------------------------------------------------------------------------------------------------

after that change configuration of filebeat.yml as per below

$ sudo nano /etc/filebeat/filebeat.yml

-------------------------------------------------------------------------------------------------------------------------------------------------

filebeat.inputs:

- type: log

paths:

- /var/log/nginx/access.log.1

filebeat.config.modules:

path: ${path.config}/modules.d/\*.yml

reload.enabled: true

reload.period: 10s

setup.template.settings:

index.number\_of\_shards: 1

output.logstash:

hosts: ["ELK server private IP:5044"]

processors:

- add\_host\_metadata:

when.not.contains.tags: forwarded

- add\_cloud\_metadata: ~

- add\_docker\_metadata: ~

- add\_kubernetes\_metadata: ~

-----------------------------------------------------------------------------------------------------------------------------------------------------------

Restart your filebeat now and this will end with frontend configuration

$ sudo systemctl restart filebeat

$ sudo systemctl enable filebeat

check error log file of filebeat and also sending log data logstash

$ journalctl -u filebeat.service

some other command in filebeat

$ sudo filebeat setup -e ----> not use just for reference

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*End with frontend Now go to ELK server\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Launch new instance, Install ELK stack on that server which should be t2.medium/t2.large.

ELK server ---> security group -----> routed All traffic

\*Note ---> your filebeat and elasticsearch, logstash and kibana must have the same version.

\*\*must --> Before Installing ELK we have install JDK-8 and nginx on ELK server

$ sudo apt update

$ sudo apt install openjdk-8-jdk

$ sudo apt install nginx

$ sudo systemctl restart nginx

$ sudo systemctl enable nginx

after successfull nginx installation we have to configure nginx for accessing kibana form browser

$ sudo nano /etc/nginx/sites-available/nginx.conf

-----------------------------------------------------------------------------------------------------------------------------------------------------------

server {

listen 80;

server\_name 10.0.1.202;

auth\_basic "Restricted Access";

auth\_basic\_user\_file /etc/nginx/htpasswd.users;

location / {

proxy\_pass http://localhost:5601;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection 'upgrade';

proxy\_set\_header Host $host;

proxy\_cache\_bypass $http\_upgrade;

}

}

-------------------------------------------------------------------------------------------------------------------------------------------------------------

$ sudo ln /etc/nginx/sites-available/nginx.conf /etc/nginx/sites-enabled/nginx.conf

$ sudo rm -rf /etc/nginx/sites-enabled/default

$ sudo systemctl restart nginx

Installation process of ELK

$ wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

$ echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list

$ sudo apt update

$ sudo apt install elasticsearch

after install elasticsearch, do changes with elasticsearch.yml file

$ sudo nano /etc/elasticsearch/elasticsearch.yml

\*uncomment the following from this YML file

-----------------------------------------------------------------------------------------------------------------------------------------------

cluster.name: my-application

node.name: node-1

path.logs: /var/log/elasticsearch

path.data: /var/lib/elasticsearch

network.host: localhost

http.port: 9200

----------------------------------------------------------------------------------------------------------------------------------------------------

$ sudo systemctl start elasticsearch

$ sudo systemctl enable elasticsearch

You can test whether your Elasticsearch service is running by sending an HTTP request:

$ curl -X GET "localhost:9200"

Now that Elasticsearch is up and running, **let’s install Kibana**, the next component of the Elastic Stack.

$ sudo apt install kibana

$ sudo systemctl enable kibana

$ sudo systemctl start kibana

The following command will create the administrative Kibana user and password, and store them in the htpasswd.users file.

You will configure Nginx to require this username and password and read this file momentarily:

$ echo "kibanaadmin:`openssl passwd -apr1`" | sudo tee -a /etc/nginx/htpasswd.users

Enter and confirm a password at the prompt. Remember or take note of this login, as you will need it to access the Kibana web interface.

after password, do changes in kibana.yml file uncomment following

---------------------------------------------------------------------------------------------------------------------------------------------------------

server.port: 5601

server.host: "localhost"

--------------------------------------------------------------------------------------------------------------------------------------------------------

after that restart ypur nginx and kibana.

$ sudo systemctl restart nginx

$ sudo systemctl restart kibana

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

now we have to work on logstash

install logstash following command.

$ sudo apt install logstash

now we have configure logstash file

create new file like logstash.conf

$ sudo nano /etc/logstash/conf.d/logstash.conf

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

input {

beats {

port => 5044

}

}

filter {

grok {

match => [ "message", "%{GREEDYDATA}\|%{GREEDYDATA}%{INT}m%{GREEDYDATA:host\_requested}%{SPACE} %{IP:client\_ip}%{SPACE}\-%{SPACE}\-%{SPACE}\[%{HTTPDATE:timestamp}\]%{GREEDYDATA:body}"]

}

}

output {

elasticsearch {

hosts => ["http://localhost:9200"]

manage\_template => false

index => "nginx-index"

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Test your Logstash configuration with this command:

$ sudo -u logstash /usr/share/logstash/bin/logstash --path.settings /etc/logstash -t

To check index pattern created or not

curl 'localhost:9200/\_cat/indices?v'

Restart your logstash and enble it.

$ sudo systemctl start logstash

$ sudo systemctl enable logstash

check your error log file of logstash

$ less /var/log/logstash/logstash-plain.log

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Now go to browser and paste your ELK ip to access kibana

Access your kibana using username and password

do following options in kibana

Discover ------> stack management -------> index pattern

you can find here your index pattern

if not their

create new index pattern

after that we have to create Dashboard

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*