Monitor ELK stack

Containerized ELK stack can be monitored by using Nagios. Install Nagios core (any version) on EC2 instance and NRPE on the client machine on which the ELK containers are running. Basically, Nagios uses check\_tcp plugin to monitor respective ports on which these containers are running.

Below is the ELK-stack.cfg file script which can be used to monitor. Let us see which ports we must monitor.

TCP ports to monitor. Below diagram gives a conceptual understanding on how this stack will be monitored.

1. Elasticsearch – runs on 9200
2. Logstash – runs on 5044
3. Kibana – runs on 5601
4. NRPE plugin – runs on 5666

Note 🡪 Both the instances are of t2. large size, to avoid high CPU utilization.

NRPE (TCP 5666)

Elasticsearch (TCP 9200)

Logstash (TCP 5044)

Kibana (TCP 5601)

Nagios Core (EC2 Instance)

define host {

use linux-server

host\_name ELK-Stack

alias My client server

address 3.109.210.110

max\_check\_attempts 20

check\_period 24x7

notification\_interval 1

notification\_period 24x7

}

define service {

use generic-service

host\_name ELK-Stack

service\_description CPU Load

check\_interval 2

retry\_interval 1

check\_command check\_nrpe!check\_load -a -w .15,.10,.05 -c .30,.25,.20

}

define service {

use generic-service

host\_name ELK-Stack

service\_description Elasticsearch service

check\_interval 1

retry\_interval 1

check\_command check\_tcp!9200

}

define service {

use generic-service

host\_name ELK-Stack

service\_description Kibana service

check\_interval 1

retry\_interval 1

check\_command check\_tcp!5601

}

define service {

use generic-service

host\_name ELK-Stack

service\_description Logstash service

check\_interval 1

retry\_interval 1

check\_command check\_tcp!5044

}

define service {

use generic-service

host\_name ELK-Stack

service\_description NRPE service

check\_interval 1

retry\_interval 1

check\_command check\_tcp!5666

}