**ELK User Manual**

**Summery:**

* **Zong ELK Stack Consist of below components**

1. **Master Node:**

Manages the overall operation of a cluster and keeps track of the cluster state. This includes creating and deleting indices, keeping track of the nodes that join and leave the cluster, checking the health of each node in the cluster (by running ping requests), and allocating shards to nodes.

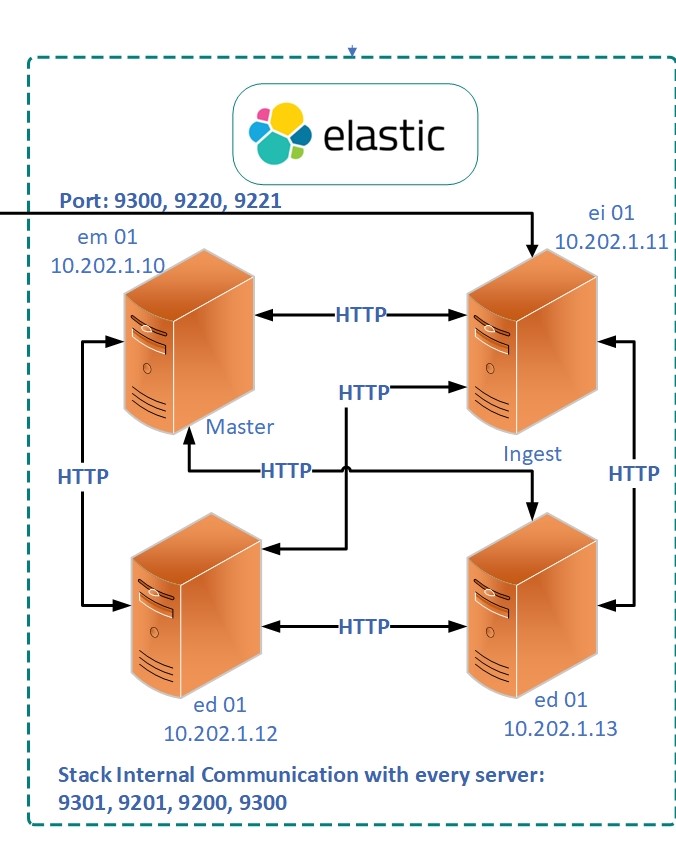
1. **Ingest Node:**

Preprocesses data before storing it in the cluster. Runs an ingest pipeline that transforms your data before adding it to an index.

1. **Data Node:**

Stores and searches data. Performs all data-related operations (indexing, searching, aggregating) on local shards. These are the worker nodes of your cluster and need more disk space than any other node type.

**Architecture Diagram:**



**Pre-Requisite:**

* Java JDK

**Common Installation and Configuration on all nodes in Cluster**

**Installation of Java:**

**#**sudo yum install java-11-openjdk.x86\_64

**Installation Of Elasticsearch**

1. Import Elastic search PGP key

#sudo rpm --import <https://artifacts.elastic.co/GPG-KEY-elasticsearch>

1. Create Elastic search repo

**#**sudo vi /etc/yum.repos.d/elastic.repo

[elasticsearch]

name=Elasticsearch repository for 7.x packages

baseurl=https://artifacts.elastic.co/packages/7.x/yum

gpgcheck=1

gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch

enabled=0

autorefresh=1

type=rpm-md

1. Install Elastic search

#sudo yum install --enablerepo=elasticsearch elasticsearch

1. Set Java Heap Size according to total installed RAM, as recommended it must be half of total size.

The previous configuration might cause node instability or even node death (with an *OutOfMemory* exception) if Elasticsearch tries to allocate more memory than is available. JVM heap limits will help us to define the memory usage and prevent this situation.

There are two rules to apply when setting the Elasticsearch heap size:

No more than 50% of available RAM.

No more than 32 GB.

In addition, you must take into account the memory usage by the operating system, services and software running on the host.

By default, Elasticsearch is configured with a 1 GB heap. You can change the heap size via JVM flags using the

#sudo vi /etc/elasticsearch/jvm.options

Uncomment Xms and Xmx

-Xms6g

-Xmx6g

Make Directory at root

#sudo mkdir estmp

#sudo chmod 777 -R /estmp

Now edit temp directry path in jvm.options

# sudo vi /etc/elasticsearch/jvm.options

# -Djava.io.tmpdir=/estmp

#:wq

1. Now assign data path and set permission. Data Path must have elastic search user rights.(If Data Path is default then skip this step)

#sudo vi /etc/elasticsearch/elasticsearch.yml

Make directory in data directory

#sudo mkdir /data/elasticsearch

#sudo chmod 775 -R /data/elasticsearch

#sudo chown -R elasticsearch:elasticsearch /data/elasticsearch

Path.data: /data/elasticsearch

: wq

1. Disable Swap

Swapping is very bad for performance, for node stability, and should be avoided at all costs. It can cause garbage collections to last for **minutes** instead of milliseconds and can cause nodes to respond slowly or even to disconnect from the cluster

**#sudo swapoff -a**

1. Disable Selinux

# sudo vi /etc/selinux/config

**SELINUX=disabled**

1. **Memory Locking Settings and verification on All Nodes in an Cluster**

Elasticsearch performs poorly when the system is swapping the memory. It is vitally important to the health of your node that none of the JVM is ever swapped out to disk.

We will set the *bootstrap.memory\_lock* setting to true, so Elasticsearch will lock the process address space into RAM, preventing any Elasticsearch memory from being swapped out.

**#sudo vi /etc/elasticsearch/elasticsearch.yml**

**#Bootstrap memory lock: true(Uncomment )**

**#sudo vi /etc/sysconfig/elasticsearch**

**MAX-LOCK-MEMORY=unlimited (**uncomment this line**)**

**Vi /etc/security/limits.conf**

Insert at bottom of file

elasticsearch soft memlock unlimited

elasticsearch hard memlock unlimited

**#sudo vi /usr/lib/systemd/system/elasticsearch.service**

Insert at [Service] block

LimitMEMLOCK=infinity

**9- Elastic search Master Node Configuration**

**#sudo vi /etc/elasticsearch/elasticsearch.yml**

**Set below parameters**

**cluster.name: es-cluster //Cluster Name will same across all nodes**

**node.name: es-master-node //Name of Node will be uniques for each node**

**node.master: true //this node will be Master Node**

**node.data: false //Will not store data**

**(uncomment this line)**

**Network.host:10.250.30.42 (Set Server IP) //Set IP for this Node**

**http.port:9220**

**discovery-seed-host: [“10.250.30.42” “10.250.30.144” “10.250.30.145” “10.250.30.146”] (insert all nodes IP,s) //Mention all nodes in this discovery seed so each node in a cluster recognize each other and create cluster**

**cluster.initial.master.node: [“es-master-node”] (set master Node)**

**:wq**

**10- Data Node Installation**

**cluster.name: es-cluster (Cluster name will be same)**

**node.name: es-data-node1 (Set Node name accordingly)**

**node.master: false**

**node.data: true**

**Network.host:10.250.30.42 (Set Server IP) //Set IP for this Node**

**http.port:9220**

**discovery-seed-host: [“10.250.30.42” “10.250.30.144” “10.250.30.145” “10.250.30.146”] (insert all nodes IP,s) //Mention all nodes in this discovery seed so each node in a cluster recognize each other and create cluster**

**cluster.initial.master.node: [“es-master-node”] (set master Node)**

**11- Ingest Node Installation**

Follow above all steps only change below parameters at Step 15:

**cluster.name: es-cluster (Cluster name will be same)**

**node.name: es-ingest-node (Set Node name accordingly)**

**node.master: false**

**node.data: false**

**node.voting\_only: false (this node cannot be elected as master node)**

**node.ingest: true** //will Ingest data from Applications

**cluster.remote.connect: false**

**Network.host:10.250.30.42 (Set Server IP) //Set IP for this Node**

**http.port:9220**

**discovery-seed-host: [“10.250.30.42” “10.250.30.144” “10.250.30.145” “10.250.30.146”] (insert all nodes IP,s) //Mention all nodes in this discovery seed so each node in a cluster recognize each other and create cluster**

**cluster.initial.master.node: [“es-master-node”] (set master Node)**

**12-Reload Daemon and elasticsearch services on all nodes**

**#sudo Systemctl daemon-reload**

**start elasticsearch services**

**#sudo Systemctl start elasticsearch**

**Security**

1- we need to modify elasticsearch.yml file on each node in the cluster to enable security features:

#sudo vi /etc/elasticsearch/elasticsearch.yml

Insert

xpack.security.enabled: true

xpack.security.transport.ssl.enabled: true

2- Elasticsearch comes with a file called elasticsearch-certutil The utility can be used to generate a self-signed certificate for internal communication encryption Elasticsearch cluster.

Now create certs on master nodes following below instruction

#cd /usr/share/elasticsearch/

#bin/elasticsearch-certutil ca

ENTER ENTER

#bin/elasticsearch-certutil cert --ca elastic-stack-ca.p12

ENTER ENTER ENTER

3- We will get TLS / SSL certificates can be used to encrypt communicationselastic-certificates.p12, Copy the certificate to config/certs Directory, and then elasticsearch.yml file configuration

Create directory on each node.

#mkdir /etc/elasticsearch/certs

Copy elastic-certificates.p12 certificate on each node certs directory including master node

Then

#cd /etc/elasticsearch/elasticsearch.yml

# xpack.security.transport.ssl.enabled: true

#xpack.security.transport.ssl.verification\_mode: certificate

#xpack.security.transport.ssl.keystore.path: certs/elastic-certificates.p12

#xpack.security.transport.ssl.truststore.path: certs/elastic-certificates.p12

4-Restart Elasticsearch service on all nodes

#sudo systemctl restart elasticsearch

5- Set Built-in User Password

#cd /usr/share/elasticsearch

# bin/elasticsearch-setup-passwords interactive

6- to enable traffic over https:

#sudo vi /etc/elasticsearch/elasticsearch.yml

Insert

xpack.security.http.ssl.enabled: true

xpack.security.http.ssl.keystore.path: certs/elastic-certificates.p12

xpack.security.http.ssl.truststore.path: certs/elastic-certificates.p12

then restart services

#sudo systemctl restart elasticsearch

**Completed**