**ELK INSTALLATION on CENTOS .7**

**Prerequisite:--**

**Java has to be installed .**

**Java can be checked using java –version.**

**Kernel parameter has to be set in below file**

**/etc/sysctl.conf**

**vm.max\_map\_count=262144**

**Save it**

**Sysctl -w**

**Simple Steps –**

**1 – Downloaded the RPM for Elasticsearch,Logstash and kibana from URL** [**https://www.elastic.co**](https://www.elastic.co)

**2- Create a new user on centos server with sudo access.**

**3-Now create Directory with the name /opt/elasticsearch,/opt/logstash and /opt/kibana**

**4 – Install each rpm using below command**

**yum localinstall elasticsearch-6.3.2.rpm**

**yum localinstall logstash-6.3.2.rpm**

**yum localinstall kibana-6.3.2-x86\_64.rpm**

**This is how ELK will be installed .**

**Now we need to edit kibana.yml file to change the localhost to ip address so that we can access it outside .**

**ELK services can be started and stopped using below commands**

**Systemctl status elasticsearch**

**Systemctl status logstash**

**Systemctl status kibana**

**If not started .we can start**

**Systemctl start elasticsearch**

**Systemctl start logstash**

**Systemct start kibana**

**Make sure to start the service at os boots.**

**Systemctl enable elasticsearch**

**Systemctl enable logstash**

**Systemctl enable kibana**

**/etc/security/limits.conf**:

elasticsearch - nofile 65535

elasticsearch - memlock unlimited

**/etc/default/elasticsearch (on CentOS/RH: /etc/sysconfig/elasticsearch )**:

ES\_HEAP\_SIZE=512m

MAX\_OPEN\_FILES=65535

MAX\_LOCKED\_MEMORY=unlimited

**/etc/elasticsearch/elasticsearch.yml**:

bootstrap.mlockall: true

/etc/sysconfig/elasticsearch

Uncomment the ES\_HEAP\_SIZE line, and set a value, eg:

# Heap Size (defaults to 256m min, 1g max)

ES\_HEAP\_SIZE=16g

**To Precent Swapping**

bootstrap.mlockall: true

**We need to make this entry as well to disable system call filter:**

You need to open your configuration file /etc/elasticsearch/elasticsearch.yml and add the below value

bootstrap.system\_call\_filter: false

and then restart your service

service elasticserach restart

**JDBC Settings in Logstash**

**input {**

**jdbc {**

**jdbc\_driver\_library => "/usr/local/src/elk/ojdbc6.jar"**

**jdbc\_driver\_class => "Java::oracle.jdbc.driver.OracleDriver"**

**jdbc\_connection\_string => "jdbc:oracle:thin:@10.100.32.243:1521/orcl"**

**jdbc\_user => "zic\_prod\_dump"**

**jdbc\_password => "test"**

**statement => "select**

**bwr.id bwr\_id, bwr.status bwr\_status, bwr.created bwr\_created,**

**bwr.started bwr\_started, bwr.finished bwr\_finished, (bwr.finished - bwr.started) bwr\_duration,**

**swr.id swr\_id, swr.status swr\_status, swr.created swr\_created,**

**swr.started swr\_started, swr.finished swr\_finished, (swr.finished - swr.started) swr\_duration,**

**(swr.ORDERPREPARATIONENDED - swr.ORDERPREPARATIONSTARTED) duration\_ordering,**

**(swr.PLAUSIBILITYSTARTED - swr.NORMALIZATIONSTARTED) duration\_norm\_selection,**

**(swr.PLAUSIBILITYENDED - swr.PLAUSIBILITYSTARTED) duration\_plausibility**

**from s\_businessworkrequest bwr**

**inner join s\_scrubbingprocesslink lnk**

**on lnk.FKBUSINESSPROCESSWORKREQUESTID = bwr.id**

**inner join s\_scrubbingworkrequest swr**

**on lnk.SCRUBBINGPROCESSWORKREQUEST001 = swr.id**

**where bwr.created > TO\_DATE('03-09-2018 00:01:00', 'DD.MM.YYYY HH24:MI:SS')"**

**schedule => "\* \* \* \* \*"**

**}**

**}**

**filter {**

**mutate {**

**add\_field => { "logstash\_timestamp"=> "%{@timestamp}" }**

**}**

**}**

**output {**

**elasticsearch {**

**hosts => "10.100.76.90:9200"**

**index => "data"**

**}**

**}**

**How to create multiple Indexes in Elasticsearch using Logstah**

1 –First of all We need to make the entry of fields in filebeat like below .

- type: log

enabled: true

paths:

- /var/log/\*.log

fields:

env: test

On Logstash:

input {

beats {

port => 5044

}

}

output {

if [fields][level] == "debug" {

elasticsearch {

hosts => "10.11.19.183:9200"

index => "test\_index"

}

}

else if [fields][env] == "uat" {

elasticsearch {

hosts => "10.11.19.183:9200"

index => "uat\_index"

}

}

else {

elasticsearch {

hosts => "10.11.19.183:9200"

index => "prod\_index"

}

}

}