* Elastic search Installation:

Step 1: To use your own version of Java, set the ES\_JAVA\_HOME environment variable. If you must use a version of Java that is different from the bundled JVM, we recommend using a [supported](https://www.elastic.co/support/matrix) [LTS version of Java](https://www.oracle.com/technetwork/java/eol-135779.html). Elastic search will refuse to start if a known-bad version of Java is used. The bundled JVM directory may be removed when using your own JVM. Recommended version

Step 2 (Linux): <https://www.elastic.co/guide/en/elasticsearch/reference/current/targz.html>

Step 3 (Windows): <https://www.elastic.co/guide/en/elasticsearch/reference/current/zip-windows.html>

Step 4: after installation run for windows and collect elastic user password form comment prompt. And also collect a token for connect kibana dashboard.

.\bin\elasticsearch.bat

For Linux system: (Oracle Linux, Ubuntu):

To configure Elastic search to start automatically when the system boots up, run the following commands:

sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable elasticsearch.service

For start or stop manually:

sudo systemctl start elasticsearch.service

sudo systemctl stop elasticsearch.service

Step 4: Change below configuration as need. (Elasticsearch.yml)

1. node.name: elk-primary-node
2. network.host: 10.11.200.117
3. http.port: 9200
4. xpack.security.enabled: true
5. xpack.security.enrollment.enabled: true
6. xpack.security.http.ssl:
7. enabled: true
8. keystore.path: certs/http.p12
9. xpack.security.transport.ssl:
10. enabled: true
11. verification\_mode: certificate
12. keystore.path: certs/transport.p12
13. truststore.path: certs/transport.p12

Though all setup will automatically after elastic search installation if you want to change any parameters as your need you can to it from where.

Step 5: Go to any browser and provide https://<IP address>:9200 press enter then it will give a confirmation prompt, provide elastic search user name password that before you saved credential from the comment prompt.

* Kibana Installation:

Step 1 (installation): <https://www.elastic.co/guide/en/kibana/current/targz.html> , https://www.elastic.co/guide/en/kibana/current/rpm.html

Step 2: http:<IP address>:5602 then provide token that are found from elastic search installation.

Step 3: Open kibana.yml file and do some configuration.

server.port: 5602

server.host: "10.11.200.117"

elasticsearch.hosts: ['https://10.11.200.117:9201']

elasticsearch.serviceAccountToken: AAEAAWVsYXN0aWMva2liYW5hL2Vucm9sbC1wcm9jZXNzLXRva2VuLTE2NjkwODkyNzUyMzM6bEV4SVN4OE9SSXEyVU95VmxJQy1xUQ

elasticsearch.ssl.certificateAuthorities: ['E:\Elastic Cluster\kibana-8.4.0\data\ca\_1669089276037.crt']

xpack.fleet.outputs: [{id: fleet-default-output, name: default, is\_default: true, is\_default\_monitoring: true, type: elasticsearch, hosts: ['https://10.11.200.117:9200'], ca\_trusted\_fingerprint: a26cc1843a2e9a6b4e1de247a10ed7707e0ce084baa099913ae046965ee3448e}]

Step 5: To configure Kibana to start automatically when the system starts, run the following commands (Linux):

sudo /bin/systemctl daemon-reload

sudo /bin/systemctl enable kibana.service

Step 4: run kibana in windows

.\bin\kibana.bat

Kibana can be started and stopped (linux) as follows:

sudo systemctl start kibana.service

sudo systemctl stop kibana.service

* Logstash Installation:

Step 1(installation): <https://www.elastic.co/guide/en/logstash/current/installing-logstash.html>

Offline oracle data pass in Elastic search:

1. Pipeline Script: Create a pipeline script for data shipment.

input {

    jdbc {

            jdbc\_validate\_connection => true

            jdbc\_connection\_string => "jdbc:oracle:thin:@10.11.1.45:1521/emob"

            jdbc\_user => "asik"

            jdbc\_password => "AbsEra#123"

            jdbc\_driver\_library => "D:/elasticstack/doc/ojdbc8.jar"

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

            statement => "SELECT \* FROM(SELECT REQUEST\_DATE,

       INCENTIVE\_AMOUNT,

       REMARKS2,

       BEN\_ADDRESS,

       apex\_web\_service.blob2clobbase64 (b.DOC\_FRONT\_IMG) DOC\_FRONT,

       apex\_web\_service.blob2clobbase64 (b.DOC\_BACK\_IMG) DOC\_BACK,

       apex\_web\_service.blob2clobbase64 (b.DOC\_BACK\_IMG) IMG

  FROM archival.RM\_PAYMENT\_REQUEST\_HIST a, archival.RM\_REMITTANCE\_DOC\_HIST b

 WHERE a.REQUEST\_ID = b.REQUEST\_ID)"

        }

}

output {

    elasticsearch {

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

        index => "remittance\_doc"

        user => "elastic"

        password => "oNqFYft-c2tskbUFyHM9"

        ssl => true

        cacert => 'D:/elasticstack/elasticsearch-8.4.0/config/certs/http\_ca.crt'

    }

    stdout {}

}

1. Run: bin/logstash –f “/D:/elk/ora\_pipeline.conf”

Excel Data Process in Elastic search:

1. Pipeline Script: Create a pipeline script for data shipment. (named: excel.conf)

input {

    file {

        path => "D:/elasticstack/doc/\*.csv"

        start\_position => "beginning"

        sincedb\_path => "NULL"

    }

}

filter {

    csv {

        separator =>","

        columns => ["id","name","host\_id","host\_name","neighbourhood\_group","neighbourhood","latitude","longitude","room\_type","price","minimum\_nights","number\_of\_reviews","last\_review","reviews\_per\_month","calculated\_host\_listings\_count","availability\_365"]

    }

}

output {

    elasticsearch {

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

        index => "ab\_nyc\_2019"

        user => "elastic"

        password => "oNqFYft-c2tskbUFyHM9"

        ssl => true

        cacert => 'D:/elasticstack/elasticsearch-8.4.0/config/certs/http\_ca.crt'

    }

    stdout {}

}

1. Run: bin/logstash –f “/D:/elk/ excel.conf”

Real-time Log process Config:

File Beat Configuration: (version – 8.5.3)

1. First of all change in filebeat.yml file in the section of logstash output on set here host and port –

output.logstash:

hosts: ["10.11.202.10:5244"]

1. Enable apache module using this command – (using bin)

./filebeat.exe modules enable apache

1. After Enabling apache module you will find a file named apache.yml, open this file and do some changes – ( for access log)

-module: apache

# Access logs

access:

enabled: true

# Set custom paths for the log files. If left empty,

# Filebeat will choose the paths depending on your OS.

var.paths: ["/var/log/httpd/access\_log\*"]

Provide valid log location on var.paths options.

Logstash Configuration:

1. Create a file on location:../logstash/conf.d/ <conf file name> (i.e. apache\_pipeline.conf).
2. Write a data shipper code in this file (apache\_pipeline.conf) based on your needs.

input{

    beats{

        port => 5244

        host => "10.11.202.10"

    }

}

filter{

    grok {

            match => {"message" => '%{HTTPD\_COMMONLOG} "%{GREEDYDATA:referrer}" "%{GREEDYDATA:agent}"'}

    }

    if "\_grokparsefailure" in [tags]{

                drop {}

    }

    useragent{

        source => "[event][original]"

        target => "uos"

    }

    # ip2location {

    #     source => "%{[source][address]}"

    # }

    # geoip{

    #     source => "%{[source][address]}"

    #     target => "geo"

    # }

    mutate {

        add\_field =>{

            "host\_name" =>"%{[host][name]}"

            "verb" =>"%{[http][request][method]}"

            "http\_version" =>"%{[http][version]}"

            "status\_code" =>"%{[http][response][status\_code]}"

            "log\_time" =>"%{[timestamp]}"

            "req\_ip" =>"%{[source][address]}"

            "os\_name" => "%{[uos][os][name]}"

            "os\_full\_name" => "%{[uos][os][full]}"

            "os\_version" => "%{[uos][os][version]}"

            "browser\_name" => "%{[uos][name]}"

            "browser\_varsion" =>"%{[uos][version]}"

        }

        remove\_field => [ "http","@version", "log","host","url","uos","message","agent","tags" ]

    }

    ruby{

        code => "

            str = event.get('referrer');

            v =  (str =~ /=/);

            v\_data = str[v+1, str.length];

            v\_app = (v\_data =~ /:/i);

            a = v\_data[0,v\_app];

            b = v\_data[v\_app+1,v\_data.length];

            v\_page\_id = (b =~ /:/i);

            c = b[0,v\_page\_id];

            d = b[v\_page\_id+1,b.length];

            v\_session\_id = (d =~ /:/i);

            e = d[0,v\_session\_id];

            event.set('app\_id',a);

            event.set('page\_id',c);

            event.set('app\_session',e);

        "

    }

    if "\_rubyexception" in [tags]{

                 drop {}

    }

}

#output{

#    stdout {

#        codec => rubydebug

#    }

#}

output {

     elasticsearch {

         hosts => ["https://10.11.200.109:9200"]

         manage\_template => false

         index => "apache\_log-%{+YYYY.MM.dd}"

         user => "fardaus"

         password => "123456"

         ssl => true

         cacert => '/home/elk-stack/elk/certs/http\_ca.crt'

     }

     stdout {}

 }

1. Change pipeline.yml file –

- pipeline.id: pipeline\_apache\_log

path.config: "/etc/logstash/conf.d/apache\_pipeline.conf"

For Oracle Apex (http.d) Configuration:

1. Change /etc/httpd/conf/httpd.conf adding –

ServerRoot "/etc/httpd"

Listen 80

Include conf.modules.d/\*.conf

#

# "/var/www/cgi-bin" should be changed to whatever your ScriptAliased

# CGI directory exists, if you have that configured.

#

Alias /j/ "/etc/httpd/j/"

<Directory "/etc/httpd/j">

#Options Indexes MultiViews FollowSymLinks

AllowOverride None

Options None

Require all granted

</Directory># Supplemental configuration

#

# Load config files in the "/etc/httpd/conf.d" directory, if any.

IncludeOptional conf.d/\*.conf

ProxyPass /cblagent http://10.11.201.82:8080/cblagent

ProxyPassReverse /cblagent http://10.11.201.82:8080/cblagent

RequestHeader unset Origin

1. Provide oracle apex image file (i.e i or j) in the location of /etc/httpd

Run modules one by one:

1. Run elasticsearch: /bin/elasticsearch
2. Run kibana: /bin/kibana
3. Run logstash: /bin/logstash –f Or /bin/logstash –f “config file location”
4. Run filebeat: /bin/filebeat –e
5. Run application using : <localhost>/servicename