ELK Stack Documentation

- 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 LTS version of Java. 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)

```
node.name: elk-primary-node
network.host: 10.11.200.117
http.port: 9200
xpack.security.enabled: true
xpack.security.enrollment.enabled: true
xpack.security.http.ssl:
  enabled: true
  keystore.path: certs/http.p12
xpack.security.transport.ssl:
  enabled: true
  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:
AAEAAWVsYXN0aWMva2liYW5hL2Vucm9sbC1wcm9jZXNzLXRva2VuLTE2NjkwODkyNzUyMzM6bEV4SVN4O
E9SSXEyVU95VmxJQy1xUQ
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
```

ELK Stack Documentation

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 {}
}
```

ELK Stack Documentation

2) 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","lat
itude","longitude","room_type","price","minimum_nights","number_of_reviews","
last_review","reviews_per_month","calculated_host_listings_count","availabili
ty_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 {}
}
```

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

ELK Stack Documentation

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"]

2) Enable apache module using this command – (using bin)
   ./filebeat.exe modules enable apache

3) 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]{
```

ELK Stack Documentation

```
                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];
```

ELK Stack Documentation

```
            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 {}
 }
```

3) Change pipeline.yml file –

```
- pipeline.id: pipeline_apache_log
  path.config: "/etc/logstash/conf.d/apache_pipeline.conf"
```

ELK Stack Documentation

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>

# 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
```

2) 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