Elasticsearch Installation & Configuration

Import the elasticsearch from gpg key using the command:

```
rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch
```

Installation:

Create a file called `elasticsearch.repo` in the `/etc/yum.repos.d/` directory for RedHat based distributions, containing:

```
[elasticsearch]
name=Elasticsearch repository for 8.8 packages
baseurl=https://artifacts.elastic.co/packages/8.8/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=0
autorefresh=1
type=rpm-md
```

Now install the elasticsearch from the rpm repository using the following command.

```
sudo yum install --enablerepo=elasticsearch elasticsearch
```

Runnning elasticsearch with systemd:

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

1. `sudo /bin/systemctl daemon-reload`
2. `sudo /bin/systemctl enable elasticsearch.service`

Elasticsearch can be started and stopped as follows:

1. `systemctl start elasticsearch.service`
2. `systemctl stop elasticsearch.service`

Configuration:

Open the elasticsearch.yml located at /etc/elasticsearch/elasticsearch.yml to configure elasticsearch.

In this file in the paths section uncomment

path.data: /var/lib/elasticsearch

path.logs: /var/log/elasticsearch

While uncommenting, take care of the indentations.

-coming to the network section
Uncomment
http.port: 9200 line. By default the port is set to 9200 and u can change it to whatever port u may wish to use.

U may also need to allow access to the port through firewall and set up tcp and udp connection on it using:
- Get a list of allowed ports in the current zone:
```
firewall-cmd --list-ports
```
- Add a port to the allowed ports to open it for incoming traffic:
```
sudo firewall-cmd --add-port=port-number/port-type
```
- Make the new settings persistent:
```
sudo firewall-cmd --runtime-to-permanent
```

Coming to the security auto configuration section

If you want to use self signed ssl certificates then follow these steps:
Put all these certificates in a folder at a secured location and then either add these lines or uncomment them in the yml file:
- xpack.security.enabled: true
- xpack.security.enrollment.enabled: true
- xpack.security.http.ssl:
    enabled: true
    key: path/to/your/ca.key
    certificate: path/to/your/ca.crt
- xpack.security.transport.ssl:
    enabled: true
    key: path/to/your/ca.key
    certificate: path/to/your/ca.crt
- if u do not want to join an existing cluster than set
    cluster.initial_master_nodes: ["node name”]
 -And then set the
     http.host: 0.0.0.0
To allow http api connection from anywhere. And if you want elasticsearch to be accessed from a specific ip address then set http.host to that ip.

Kibana Installation & Configuration

Installation:

Installing kibana with rpm

Import the elastic gpg signing key.Download and install the public signing key:

- `rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch`

Installing from the rpm repository:

Create a file called `kibana.repo` in the `/etc/yum.repos.d/` directory for RedHat based distributions, containing:

```
[kibana-8.x]
name=Kibana repository for 8.x packages
baseurl=https://artifacts.elastic.co/packages/8.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
type=rpm-md
```

And your repository is ready for use. You can now install Kibana with the following command:

```
sudo yum install kibana
```

To configure Kibana to start automatically when the system starts, run the following commands:

- `sudo /bin/systemctl daemon-reload`
- `sudo /bin/systemctl enable kibana.service`

Kibana can be started and stopped as follows:

- `sudo systemctl start kibana.service`
- `sudo systemctl stop kibana.service`

Configuration:

You may also need to allow access to the port through firewall and set up tcp and udp connection on it using:

- Get a list of allowed ports in the current zone:

```
firewall-cmd --list-ports
```

- Add a port to the allowed ports to open it for incoming traffic:

```
sudo firewall-cmd --add-port=port-number/port-type
```

- Make the new settings persistent:

```
sudo firewall-cmd --runtime-to-permanent
```

Uncomment or add these lines in the kibana.yml file located at /etc/kibana/kibana.yml

Add the port no you are wishing to use. We are using the default port 5601.

- server.port: 5601

Set the server.host: to 0.0.0.0 to make kibana listen on all ips(public and private) else specify your own ip.

- server.host: 0.0.0.0
- server.publicBaseUrl: “https://hostname:port_no”

If using ssl with self signed certificates, set

- server.ssl.enabled: true
- server.ssl.certificate: /path/to/ca.crt

- server.ssl.key: /path/to/ca.key
- elasticsearch.hosts: [“https://localhost:9200”]
- elasticsearch.ssl.verificationMode: none

To connect kibana to elasticsearch generate the authentication token and encrypt it using kibana keystore as follow:

-You can then generate an enrollment token for Kibana with the elasticsearch-create-enrollment-token tool:

- `bin/elasticsearch-create-enrollment-token -s kibana`

This will generate an enrollment token which u can keep stored and then add it to kibana keystore as follow:

`-bin/kibana-keystore add elasticsearch.serviceAccountToken`

It will ask you to give the value which you need to store, you can enter that value and you are good to go.

For generating alerting documents in kibana:

1. Go to /usr/share/kibana/bin and add the xpack.encryptedSavedObjects.encryptionKey this in kibana keystore.value of this key must be atleast 32 characters. This will help in enabling the rules section in the kibana.after this restart kibana. Now create rules section must appear in the observability-alerts section.
2. You can store this encryption key in kibana-keystore as follows:
   `bin/kibana-keystore add xpack.encryptedSavedObjects.encryptionKey`
   It will ask you to give the value which you need to store, you can enter that value and yo are good to go.

Metricbeat and Filebeat Installation

Prerequisites:

1. Metricbeat API:
    a. Go to Hamburger menu (≡) > Management > Stack management.
    b. From the list on the left side go to Security > Roles.
    c. Click on “Create a new role” button and create a new role as following:
        i. Role name: metricbeat-user
        ii. Cluster privileges:
            1. Monitor
            2. Read_ilm
        iii. Index privileges:
            1. Indices -> metricbeat-* , Privileges -> create_doc
    d. Click on the “Save role” button.
    e. Now, On the same page, Go to Security > Users.
    f. Click on “Create user” button and create a new user as following:
        i. Username: metric
        ii. Password: //Any password You like
        iii. Confirm password: //Same password as before
        iv. Privileges:
            1. Metricbeat-user
            2. Editor
    g. Click on “Save user”.
    h. Now, Go to Hamburger menu (≡) > Management > Dev Tools.
    i. Type in the code:

```
POST /_security/api_key/grant
{
 "grant_type": "password",
 "username": "metric",
 "password": "//Password you set above",
 "api_key": {
  "name": "metric-api"
 }
}
```

    j. Play the code snippet and save the response you get back to a text file. Also make a word in format “id:api_key” where both “id” and “api_key” are taken from the response object that you just saved. We will be referencing this newly generated word as **metric_api_key** in this documentation.
2. Filebeat API:
    a. Go to Hamburger menu (≡) > Management > Stack management.
    b. From the list on the left side go to Security > Roles.
    c. Click on “Create a new role” button and create a new role as following:
        i. Role name: filebeat-user
        ii. Cluster privileges:

1. Monitor
2. Manage_ilm

iii. Index privileges:

1. Indices -> filebeat-* , Privileges -> all

d. Click on the “Save role” button.
e. Now, On the same page, Go to Security > Users.
f. Click on “Create user” button and create a new user as following:
   i. Username: file
   ii. Password: //Any password You like
   iii. Confirm password: //Same password as before
   iv. Privileges:
      1. Filetricbeat-user
      2. Editor
g. Click on “Save user”.
h. Now, Go to Hamburger menu (≡) > Management > Dev Tools.
i. Type in the code:

```
POST /_security/api_key/grant
{
 "grant_type": "password",
 "username": "file",
 "password": "//Password you set above",
 "api_key": {
  "name": "file-api"
 }
}
```

j. Play the code snippet and save the response you get back to a text file. Also make a word in format “id:api_key” where both “id” and “api_key” are taken from the response object that you just saved. We will be referencing this newly generated word as **file_api_key** in this documentation.

3. Cluster Data:
   a. Check the address of the elasticsearch host for your cluster. Eg: https://slazintern01.francecentral.cloudapp.azure.com:9200
   We will be referencing this address as **elastic_host** in this documentation.
   b. Check the address of the kibana host for your cluster. Eg: https://slazintern01.francecentral.cloudapp.azure.com:5601
   We will be referencing this address as **kibana_host** in this documentation.

Installation:

There are two methods to install metricbeat and filebeat on the machine.

1. Install manually by following this documentation and executing commands by yourself.
2. Install automatically by executing a script.

Method 1:

1. Open terminal and switch to root user using command:
   `sudo -i`

2. Add public signing key using command:
   rpm --import https://packages.elastic.co/GPG-KEY-elasticsearch

3. Go to /etc/yum.repos.d/ directory and create a new file named elastic.repo using following command:
   cd /etc/yum.repos.d
   nano elastic.repo
4. Now copy the following text into the nano editor:
   [elastic-8.x]
   name=Elastic repository for 8.x packages
   baseurl=https://artifacts.elastic.co/packages/8.x/yum
   gpgcheck=1
   gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
   enabled=1
   autorefresh=1
   type=rpm-md
5. Once done, just execute following command:
   yum install metricbeat-8.8.0
   yum install filebeat-8.8.0
   (Type y when prompted to install metricbeat & filebeat)
6. Enable metricbeat by following command:
   systemctl enable metricbeat
   systemctl enable filebeat
7. Add **metric_api_key** & **file_api_key** into metricbeat keystore by following commands:
   metricbeat keystore add api_key
   filebeat keystore add api_key
8. Now, enter the **metric_api_key** (for metricbeat) and **file_api_key** (for filebeat) created previously into the terminal when prompted and hit enter.
9. Configure metricbeat.yml & filebeat.yml file by opening it in nano editor one by one by following command and make the changes suggested in next step:
   nano /etc/metricbeat/metricbeat.yml
   nano /etc/filebeat/filebeat.yml
10. Now, change or add following lines in both the yml files:
    a. Modules configuration section:
       i. reload.enabled: true
    b. Kibana section:
       i. Uncomment setup.kibana
       ii. host: “**kibana_host**” //**kibana_host** is the one mentioned above
       iii. ssl.verification_mode: “none”
    c. Elasticsearch output section:
       i. hosts: [“**elastic_host**”] //**elastic_host** is the one mentioned above
       ii. api_key: “${api_key}”
11. Enable some modules for metricbeat & filebeat by following commands:
    metricbeat modules enable elasticsearch
    metricbeat modules enable elasticsearch-xpack
    metricbeat modules enable kibana
    metricbeat modules enable kibana-xpack
    metricbeat modules enable linux

`metricbeat modules enable system`
`filebeat modules enable elasticsearch`
`filebeat modules enable kibana`
`filebeat modules enable system`

12. Go to metricbeat modules.d using following command:
    `cd /etc/metricbeat/modules.d`
13. Change following lines from specified files:
    (Use “nano *file_name*” to edit content of files)
    a. In elasticsearch.yml & elasticsearch-xpack.yml
        i. hosts: [“**elastic_host**”] //**elastic_host** is the one mentioned above
    b. In kibana.yml & kibana-xpack.yml
        i. hosts: [“**kibana_host**”] //**kibana_host** is the one mentioned above
14. Now, the setup is complete. Just start the metricbeat & filebeat service using:
    `systemctl start metricbeat.service`
    `systemctl start filebeat.service`

Method 2:

For this step, you must have the *beats.zip* file on the machine where you want to install & configure the beats. For this you can use `scp` command to copy the file from your local machine to the target machine.
After having the zip file in place:

1. Move *beats.zip* to your home directory and then unzip it using following command:
   `unzip beats.zip`
2. From the home directory itself, run the setup script using following command:
   `sudo sh beats/setup.sh`
3. Follow on-screen instructions and provide all the data asked during installation. The names of fields asked during installation are to be referred from this documentation itself (eg. **elastic_host**, **kibana_host** etc.)
4. Now, the setup is complete. Both the beats are up and running which you can cross check by following command:
   `systemctl status metricbeat`
   `systemctl status filebeat`

Kibana Dashboards

Host Metrices Pro Max

- Contains the list of all the active serves along with relevant metrics
- Contains tables for servers with Warnings and Critical condition
- To edit the warning and critical conditions in the table, edit the filter applied on the particular table using kibana lens

List of Hosts

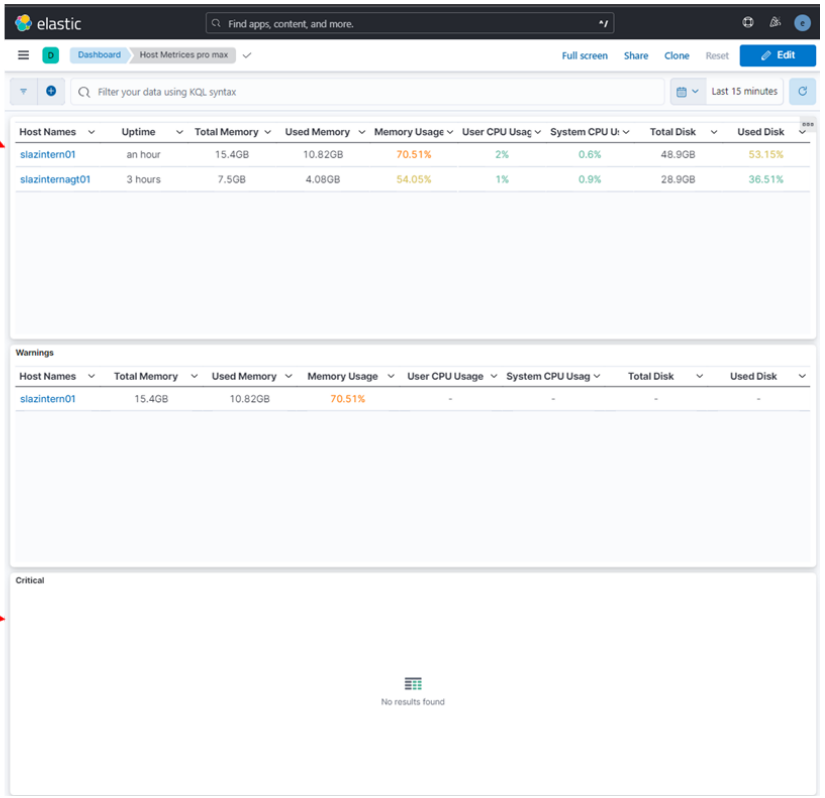
- List of connected clients along with relevant metrices
- Detailed Host Dashboard accessible by clicking Host's name
- Color-coded metrices for quick overview

Warning:

- List of connected clients containing warnings
- Warning limits are specified in terms of memory usage and CPU usage
- Current Warning Limits:
    - Memory Usage: 60% - 80%
    - CPU usage: 60%- 80%

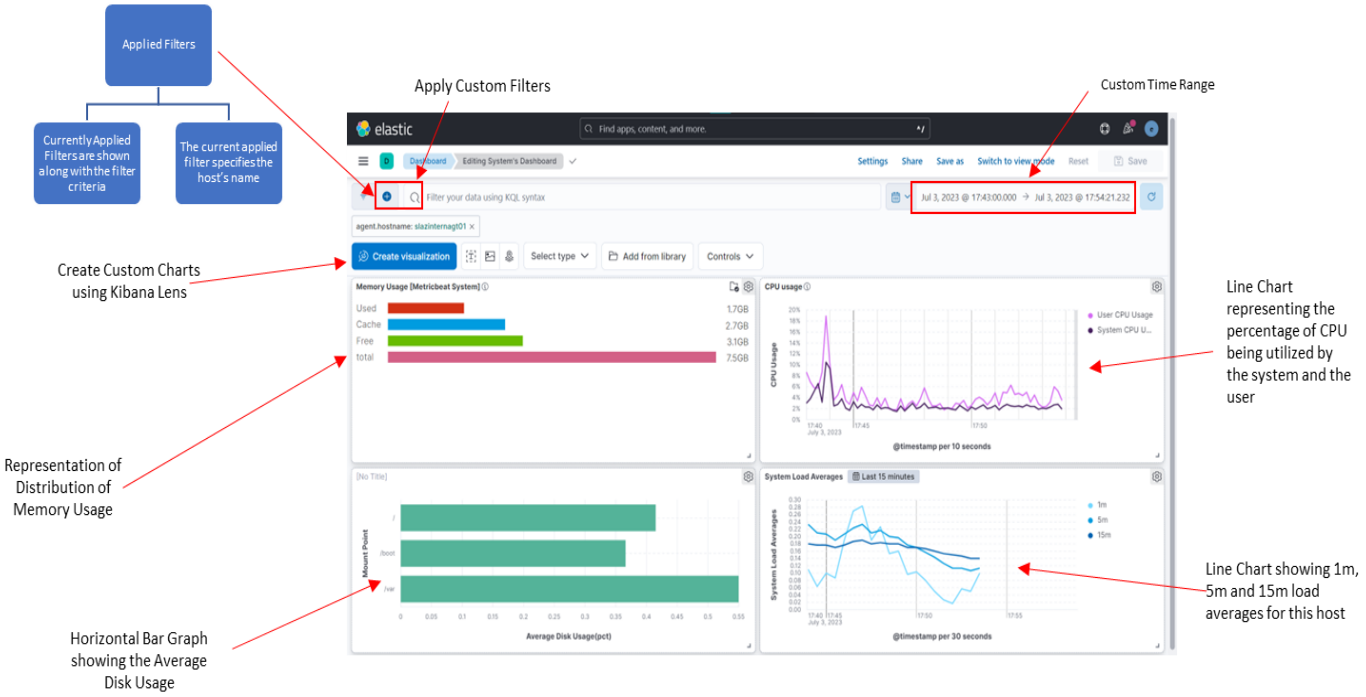
Critical:

- List of connected clients that are in critical state
- Critical limits are specified in terms of memory usage and CPU usage
- Current Critical Limits:
    - Memory Usage: 80% and above
    - CPU usage: 80% and above



System Dashboard

- Contains detailed charts for each server
- New charts and tables can be added using ‘create visualisation’



Create Visualisation:

- Create new charts and tables
- Charts with custom filters and time range can also be created

Filebeat Dashboard:

- Dashboard for logs generated by all the servers
- Includes charts and graphs for logs related to different processes of a server