

Creating an Elastic Cluster



JANUARY 2024
Abdulfatah Abdillahi

Contents

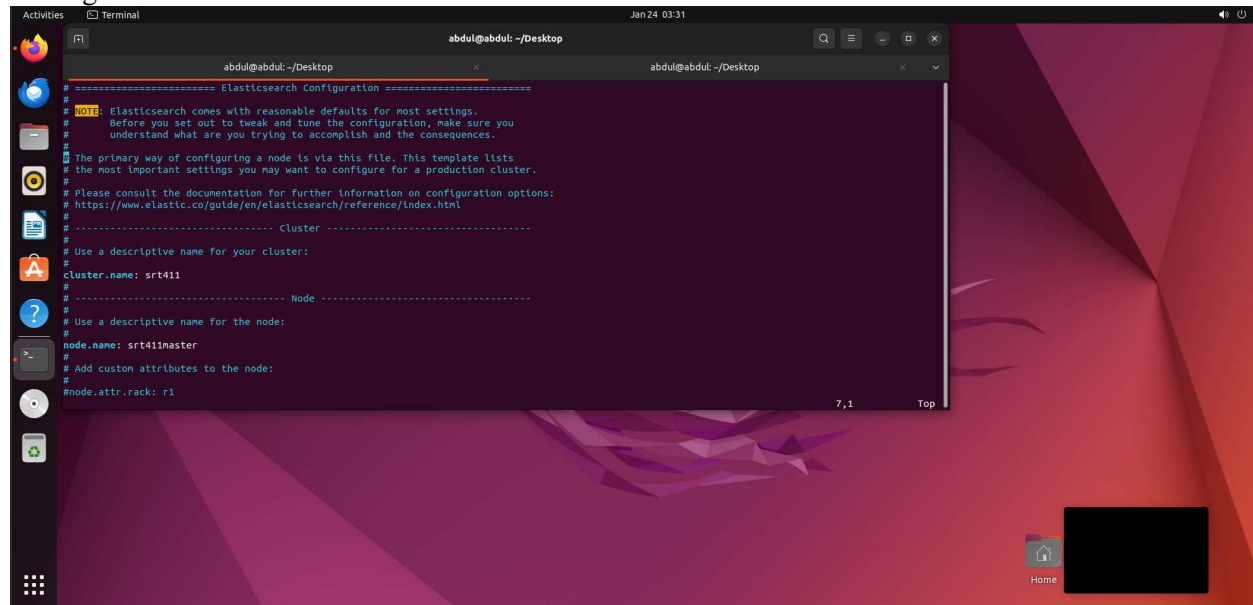
Introduction	3
Task 1: create and test the cluster.	3
Task 2: Data Query	7
Task 3: Index Manager	10

Introduction

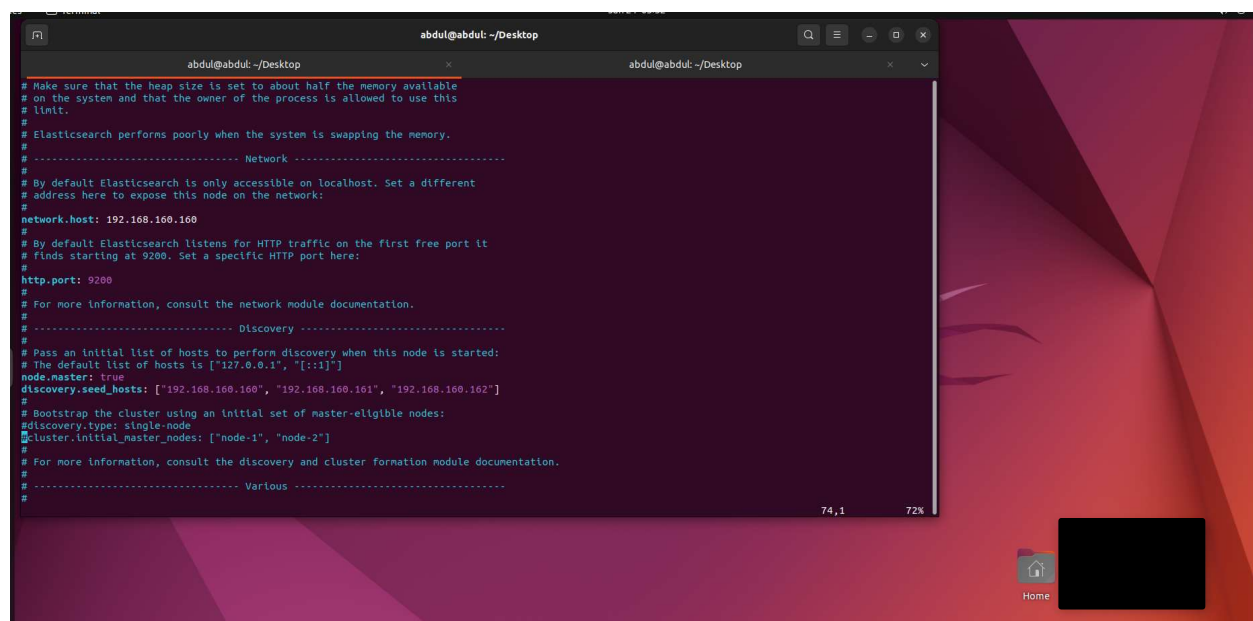
In this lab report, we outline the process of preparing your machine for data analytics by installing and configuring a Linux server with the ELK stack. Objectives include setting up Elasticsearch with Apache logs ingested via Filebeat and exploring data visualizations in Kibana. The report serves as a guide for understanding and implementing these crucial steps in establishing a functional data analytics environment.

Task 1: create and test the cluster.

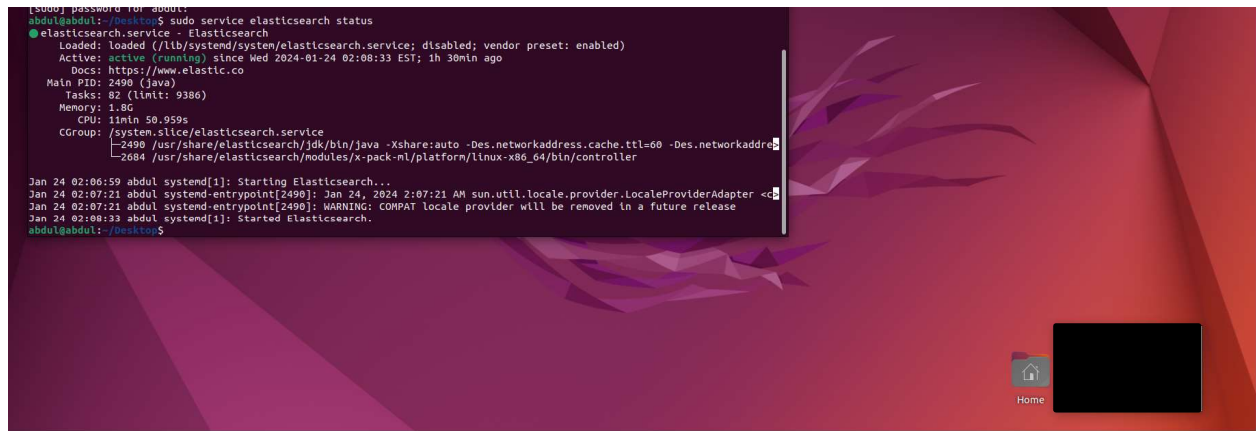
Editing the master node Elasticsearch file



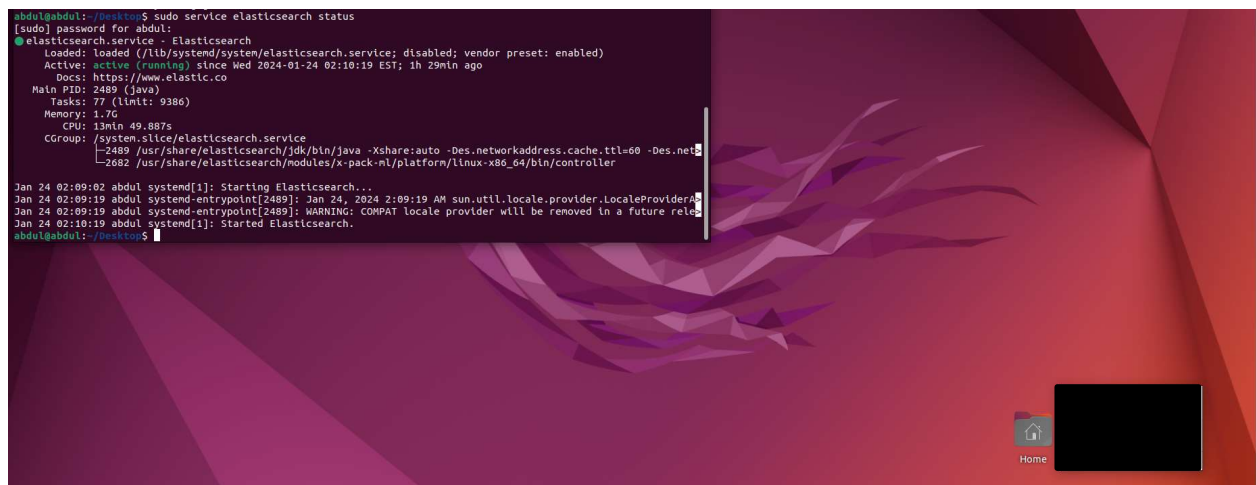
```
abdul@abdul: ~/Desktop
# ===== Elasticsearch Configuration =====
# NOTE: Elasticsearch comes with reasonable defaults for most settings.
# Before you set out to tweak and tune the configuration, make sure you
# understand what are you trying to accomplish and the consequences.
#
# The primary way of configuring a node is via this file. This template lists
# the most important settings you may want to configure for a production cluster.
# Please consult the documentation for further information on configuration options:
# https://www.elastic.co/guide/en/elasticsearch/reference/index.html
#
# ----- Cluster -----
# Use a descriptive name for your cluster:
#
cluster.name: srt411
#
# ----- Node -----
# Use a descriptive name for the node:
#
node.name: srt411master
# Add custom attributes to the node:
#
#node.attr.rack: r1
```



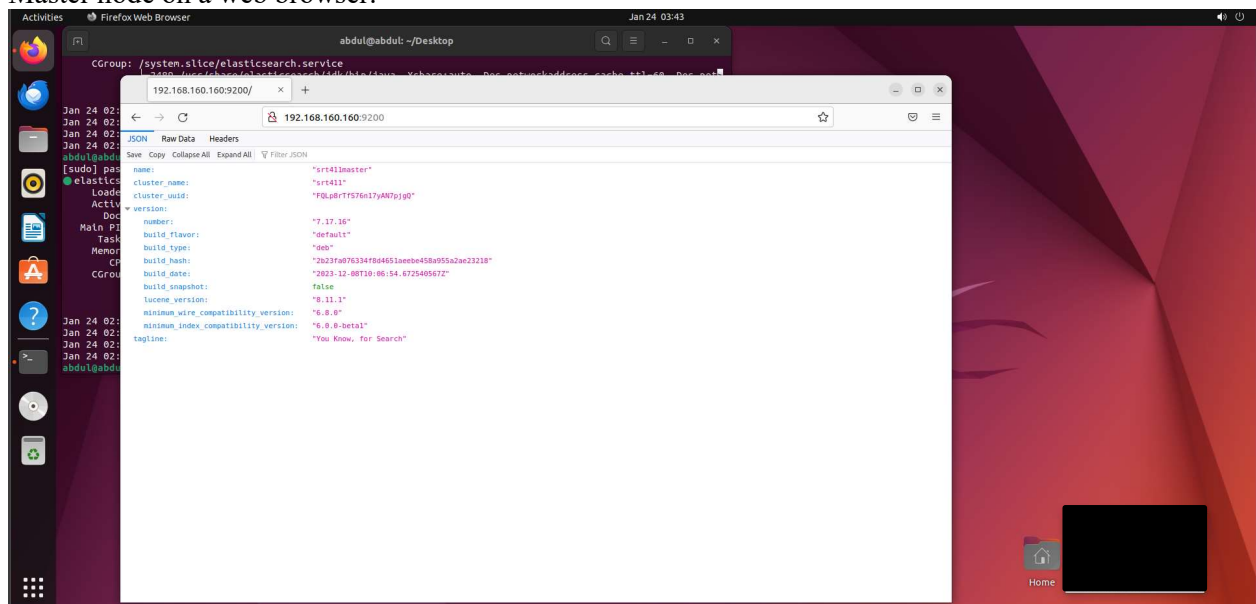
```
abdul@abdul: ~/Desktop
# Make sure that the heap size is set to about half the memory available
# on the system and that the owner of the process is allowed to use this
# limit.
#
# Elasticsearch performs poorly when the system is swapping the memory.
#
# ----- Network -----
# By default Elasticsearch is only accessible on localhost. Set a different
# address here to expose this node on the network:
#
network.host: 192.168.160.160
#
# By default Elasticsearch listens for HTTP traffic on the first free port it
# finds starting at 9200. Set a specific HTTP port here:
#
http.port: 9200
#
# For more information, consult the network module documentation.
#
# ----- Discovery -----
# Pass an initial list of hosts to perform discovery when this node is started:
# The default list of hosts is ["127.0.0.1", "127.0.0.1"]
#
node.master: true
discovery.seed_hosts: ["192.168.160.160", "192.168.160.161", "192.168.160.162"]
#
# Bootstrap the cluster using an initial set of master-eligible nodes:
#
#discovery.type: single-node
cluster.initial_master_nodes: ["node-1", "node-2"]
#
# For more information, consult the discovery and cluster formation module documentation.
#
# ----- Various -----
#
```

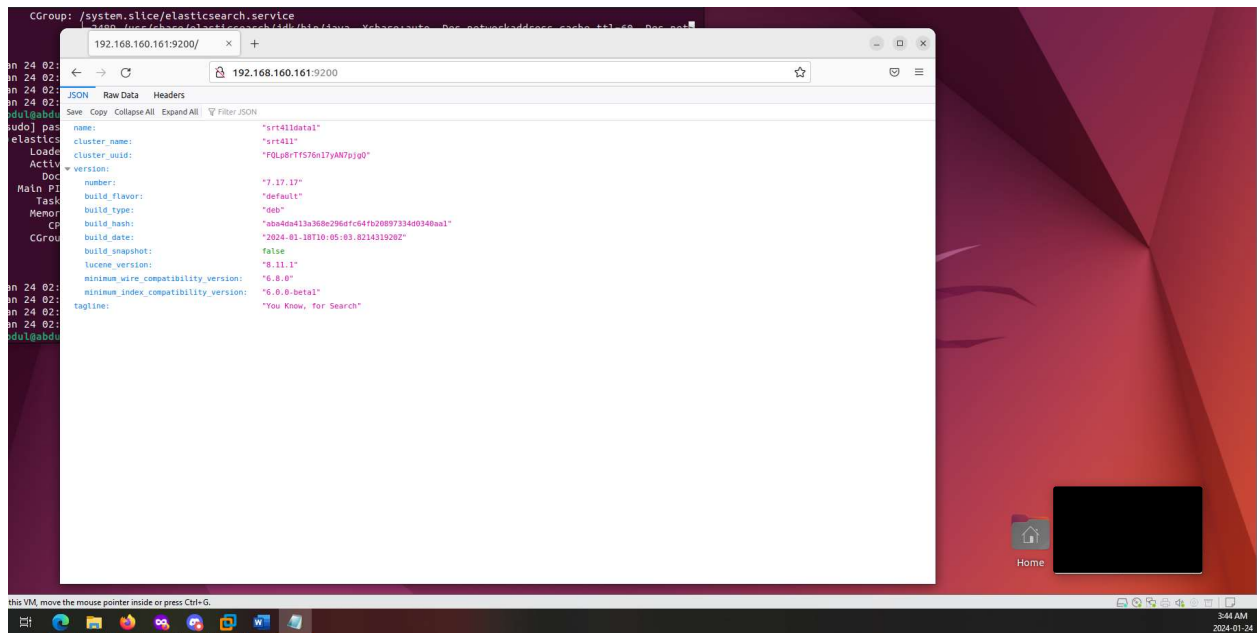
Node-2 installation with Elasticsearch confirmation



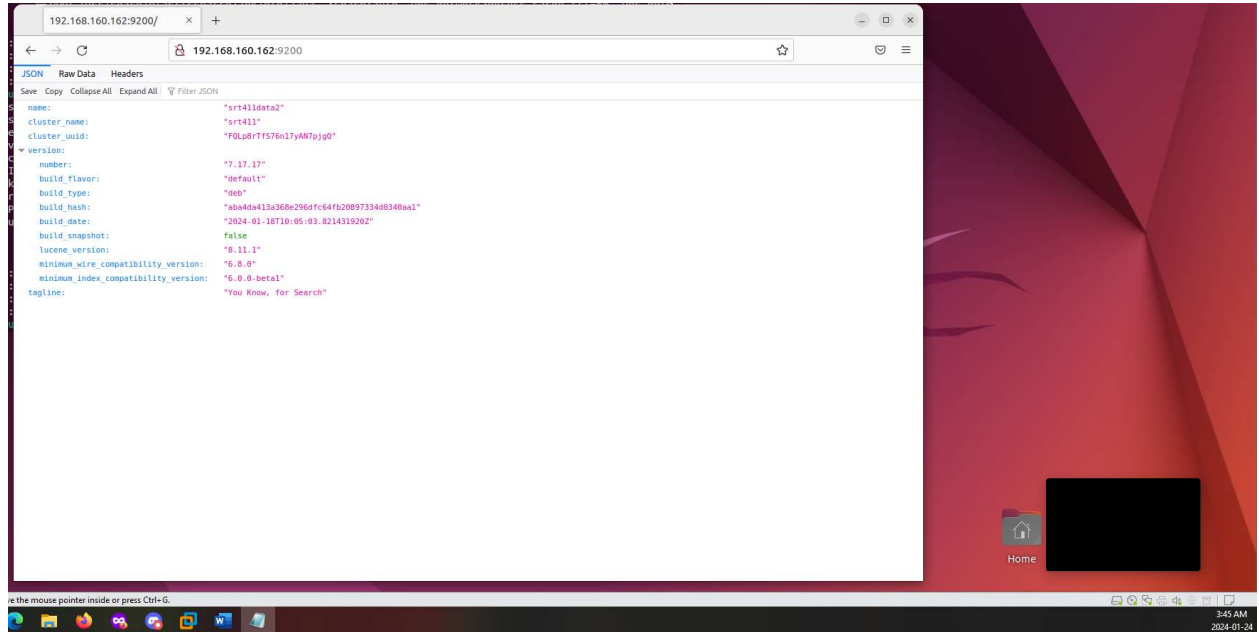
Master node on a web browser.



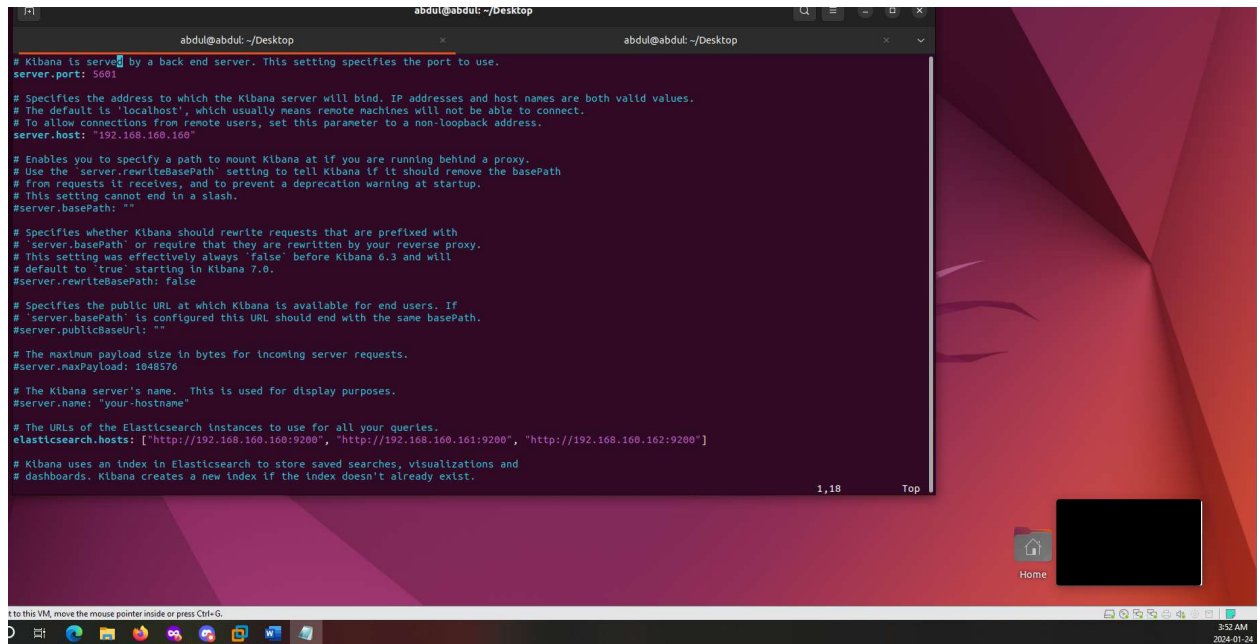
Node-1 status on web browser



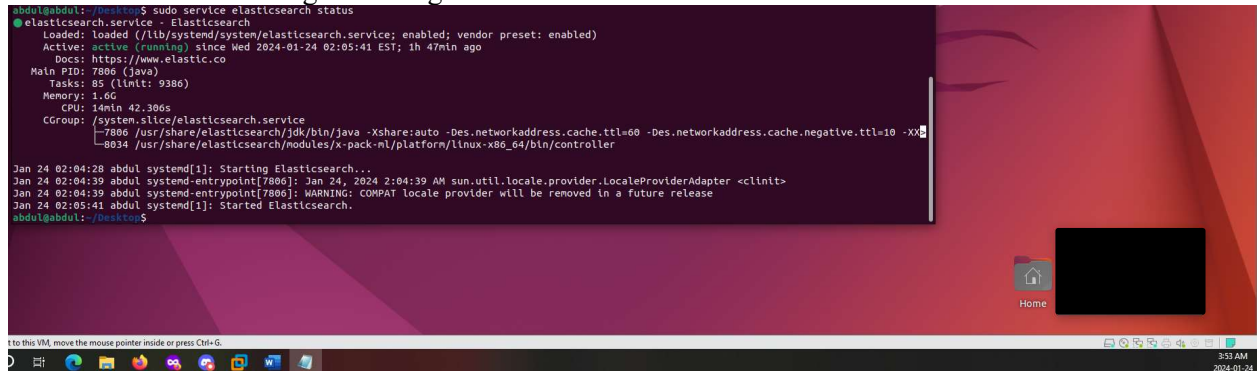
Node-2 status on web browser



Editing Kibana Configuration File

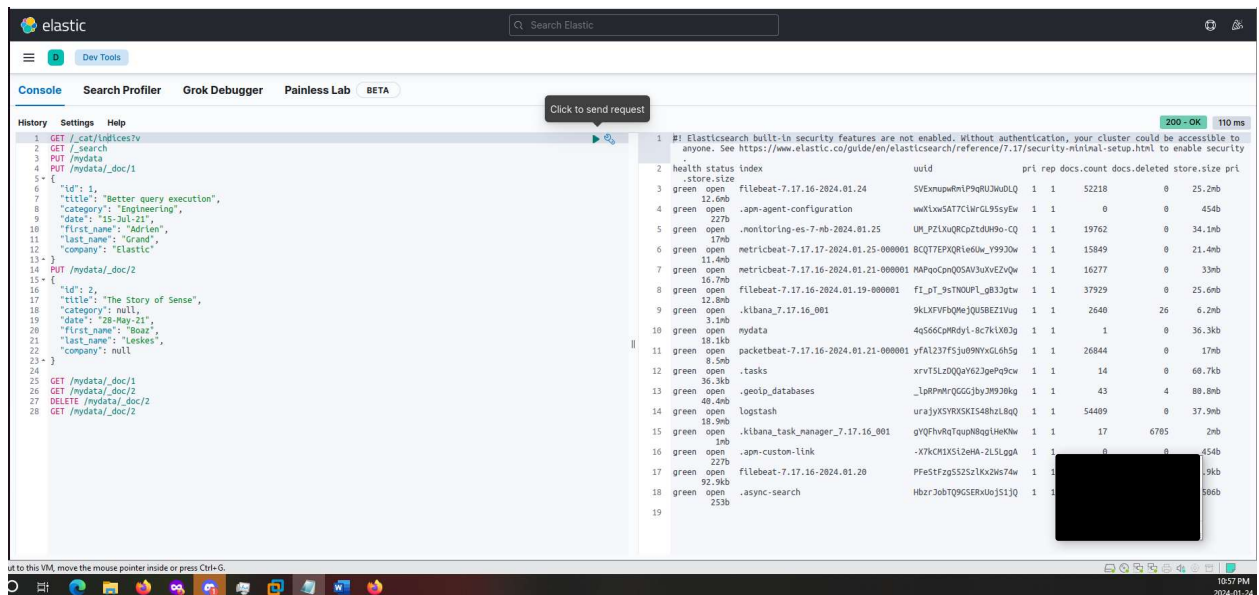


Kibana status after editing the configuration.



Task 2: Data Query

1. Write the command to query how many indices are there in the cluster and what are their names?



2. Write the command to query all documents in all indices of your cluster. By default, only the first 10 rows will be displayed.

The screenshot shows the Elastic UI Console with a 'Click to send request' button. The console displays a list of commands on the left and the response on the right. The response is a JSON object representing a search result.

```
1 GET /_cat/indices?v
2 GET /_search
3 PUT /mydata
4 PUT /mydata/_doc/1
5 {
6   "id": 1,
7   "title": "Better query execution",
8   "category": "Engineering",
9   "date": "15-Jul-21",
10  "first_name": "Adrian",
11  "last_name": "Grand",
12  "company": "Elastic"
13 }
14 PUT /mydata/_doc/2
15 {
16   "id": 2,
17   "title": "The Story of Sense",
18   "category": null,
19   "date": "28-May-21",
20   "first_name": "Boaz",
21   "last_name": "Leskes",
22   "company": null
23 }
24 GET /mydata/_doc/1
25 GET /mydata/_doc/2
26 DELETE /mydata/_doc/2
27 GET /mydata/_doc/2
```

```
{
  "took": 449,
  "timed_out": false,
  "_shards": {
    "total": 15,
    "successful": 15,
    "skipped": 0,
    "failed": 0
  },
  "hits": {
    "total": {
      "value": 10000,
      "relation": "gte"
    },
    "max_score": 1.0,
    "hits": [
      {
        "_index": ".kibana_7.17.16_001",
        "_type": "_doc",
        "_id": "searchKafka stacktraces-ecs",
        "_score": 1.0,
        "_source": {
          "search": {
            "columns": [
              "kafka.log.class",
              "kafka.log.trace.class",
              "kafka.log.trace.full"
            ],
            "description": "",
            "hits": 0,
            "kibanaSavedObjectMeta": {
              "searchSourceJSON": ""
            }
          }
        }
      }
    ]
  }
}
```

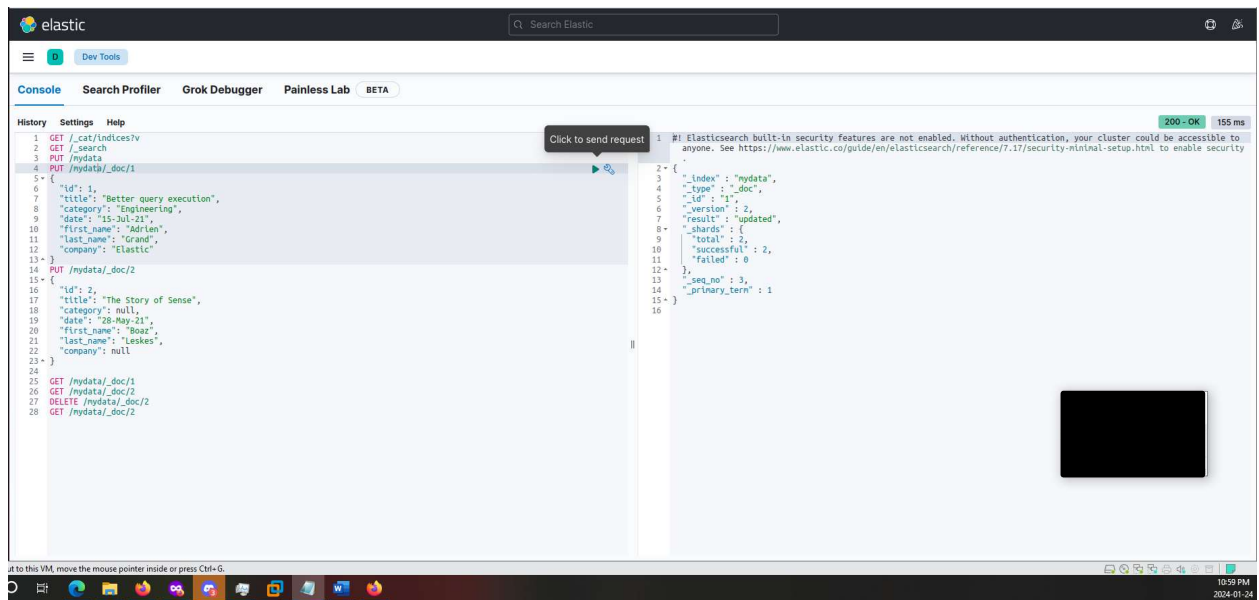
3. Write the command to create a new index named “mydata”.

The screenshot shows the Elastic UI Console with a 'Click to send request' button. The console displays a list of commands on the left and the response on the right. The response is a JSON object representing an error.

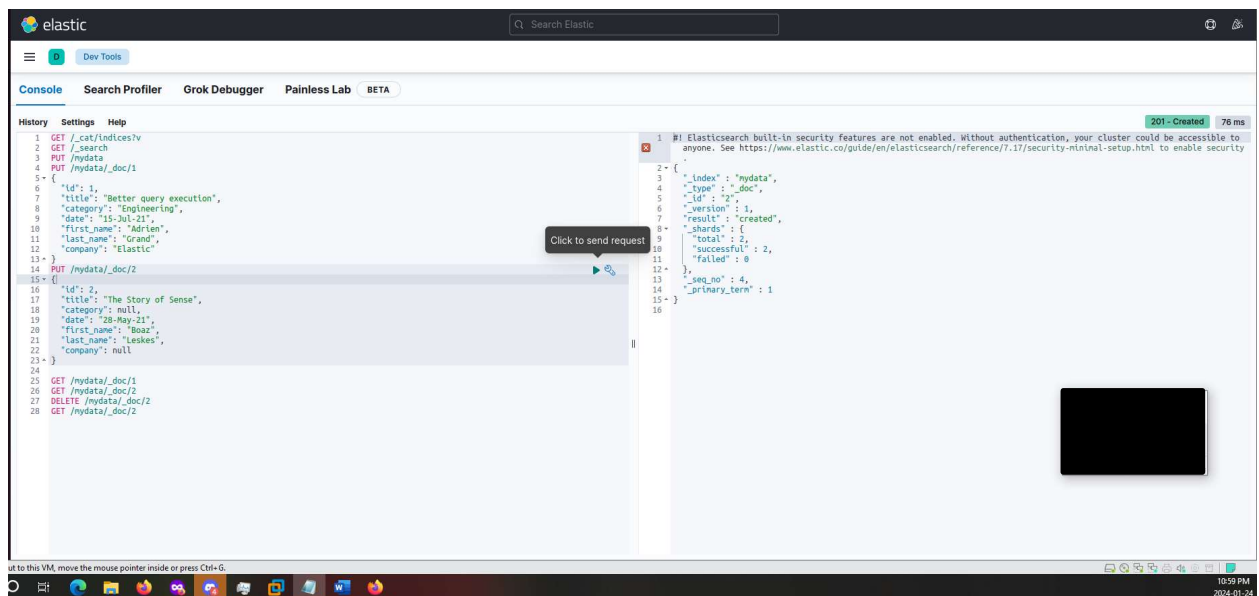
```
1 GET /_cat/indices?v
2 GET /_search
3 PUT /mydata
4 PUT /mydata/_doc/1
5 {
6   "id": 1,
7   "title": "Better query execution",
8   "category": "Engineering",
9   "date": "15-Jul-21",
10  "first_name": "Adrian",
11  "last_name": "Grand",
12  "company": "Elastic"
13 }
14 PUT /mydata/_doc/2
15 {
16   "id": 2,
17   "title": "The Story of Sense",
18   "category": null,
19   "date": "28-May-21",
20   "first_name": "Boaz",
21   "last_name": "Leskes",
22   "company": null
23 }
24 GET /mydata/_doc/1
25 GET /mydata/_doc/2
26 DELETE /mydata/_doc/2
27 GET /mydata/_doc/2
```

```
{
  "error": {
    "root_cause": [
      {
        "type": "resource_already_exists_exception",
        "reason": "index [mydata/4q566cpMRdyL-8C7kLX03g] already exists",
        "index_uuid": "4q566cpMRdyL-8C7kLX03g",
        "index": "mydata"
      }
    ],
    "type": "resource_already_exists_exception",
    "reason": "index [mydata/4q566cpMRdyL-8C7kLX03g] already exists",
    "index_uuid": "4q566cpMRdyL-8C7kLX03g",
    "index": "mydata"
  },
  "status": 400
}
```

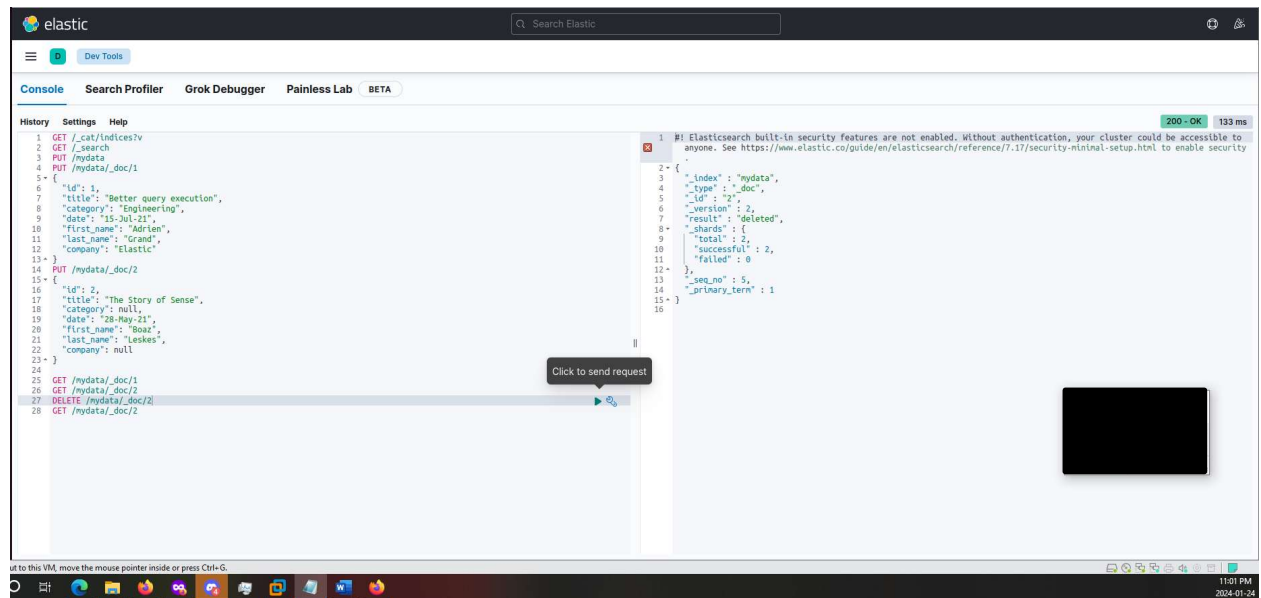
4. Write the command to add the following two documents in the index called “mydata” one by one. Use `_doc` for the type and their respective ids. You have to provide ids while putting the data in Elasticsearch. Look for PUT and POST commands and should be able to identify the difference and pick the right command for this step.



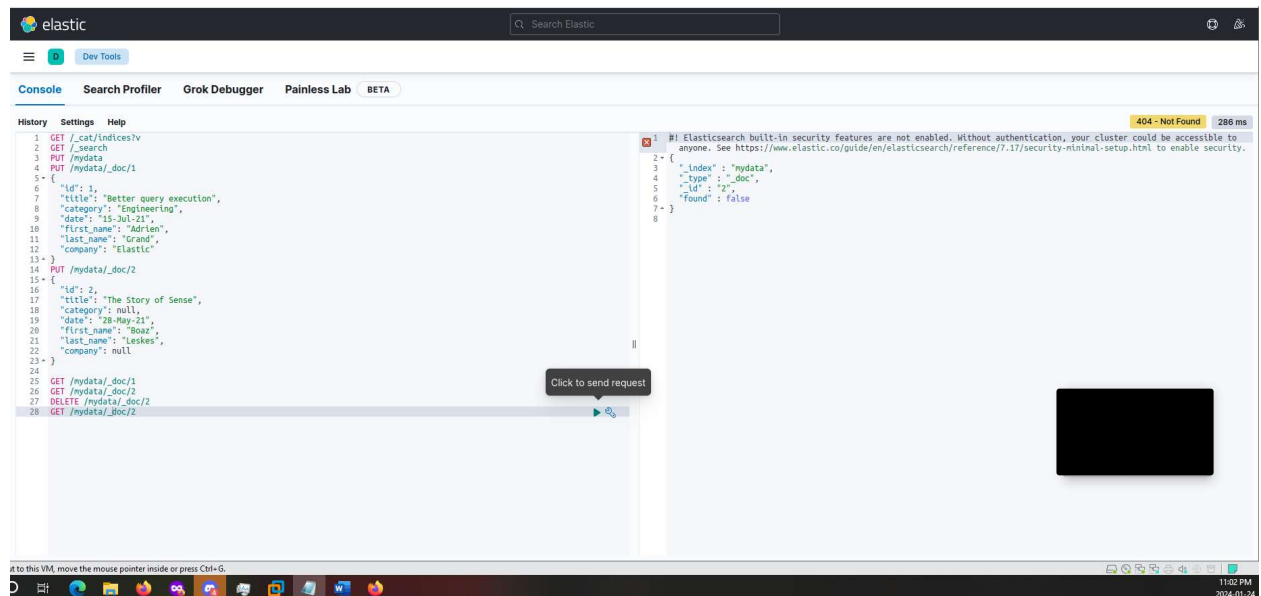
5. Write the GET command to retrieve the document with id of 1 and type _doc from the “myData” index.



6. Write the command to delete the document with id 2 from the “myData” index



7. Verify it was deleted by trying to GET it again, record the response.



Task 3: Index Manager

Activities Firefox Web Browser Jan 24 22:19

Stack Monitoring - Elastic x Start Metricbeat | Metric x +

192.168.160.160:5601/app/monitoring#/elasticsearch/nodes?_g={cluster_uuid:FQLp8rTF576n17yAN7pigQ,refreshInterval:(pauseIfValue:10000),time:(from:now-15)}







elastic Search Elastic

Clusters Alerts and rules

Status Green Alerts 0 Nodes 3 Indices 18 JVM Heap 1.3 GB / 3.0 GB Total shards 36 Unassigned shards 0 Documents 200,149 Data 255.6 MB

Metricbeat is monitoring all nodes.

Filter Nodes...

Name ↑	Alerts	Status	Shards	CPU Usage	Load Average	JVM Heap	Disk Free Space
 srt411data1 http://192.168.160.161:9200  Monitored with Metricbeat		Online	13	↓ 4%	↓ 0.22	↑ 47%	↓ 9.0 GB
 srt411data2 http://192.168.160.162:9200  Monitored with Metricbeat		Online	12	↓ 5%	↑ 1.22	↓ 21%	↓ 9.0 GB
 srt411master http://192.168.160.160:9200  Monitored with Metricbeat		Online	11	↑ 11%	↑ 1.86	↑ 40%	↓ 7.9 GB

Rows per page: 20

Set up monitoring for new node

You are in setup mode. The (P) icon indicates configuration options.

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

10:19 PM 2024-01-24

Activities Firefox Web Browser Jan 24 22:51


Dev Tools - Elastic x Index Management - Elastic x Start Metricbeat | Metric x +


192.168.160.160:5601/app/management/data/index_management/indices


elastic Search Elastic


Stack Management Index Management

Management

Ingest  Ingest Pipelines

Data  **Index Management**
Index Lifecycle Policies
Snapshot and Restore
Rollup Jobs
Transforms
Remote Clusters

Alerts and Insights  Rules and Connectors
Reporting
Machine Learning Jobs

Kibana  Index Patterns
Saved Objects
Tags
Search Sessions
Spaces
Advanced Settings

Indices Data Streams Index Templates Component Templates

Update your Elasticsearch indices individually or in bulk. [Learn more.](#)

☐ Include rollup indices ☐ Include hidden indices

2 indices have lifecycle errors [Show errors](#)

Search Lifecycle status Lifecycle phase Reload indices

<input type="checkbox"/>	Name	Health	Status	Primaries	Replicas	Docs count	Storage size	Data stream
<input type="checkbox"/>	filebeat-71716-2024.01.24	green	open	1	1	52218	25.2mb	
<input type="checkbox"/>	metricbeat-71717-2024.01.25-000001	green	open	1	1	14521	25.3mb	
<input type="checkbox"/>	metricbeat-71716-2024.01.21-000001	green	open	1	1	14926	32.9mb	
<input type="checkbox"/>	filebeat-71716-2024.01.19-000001	green	open	1	1	37929	25.6mb	
<input type="checkbox"/>	mydata	green	open	1	1	1	35.7kb	
<input type="checkbox"/>	packetbeat-71716-2024.01.21-000001	green	open	1	1	26844	17mb	
<input type="checkbox"/>	logstash	green	open	1	1	54409	37.9	
<input type="checkbox"/>	filebeat-71716-2024.01.20	green	open	1	1	118	185	

Rows per page: 10

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

10:51 PM 2024-01-24