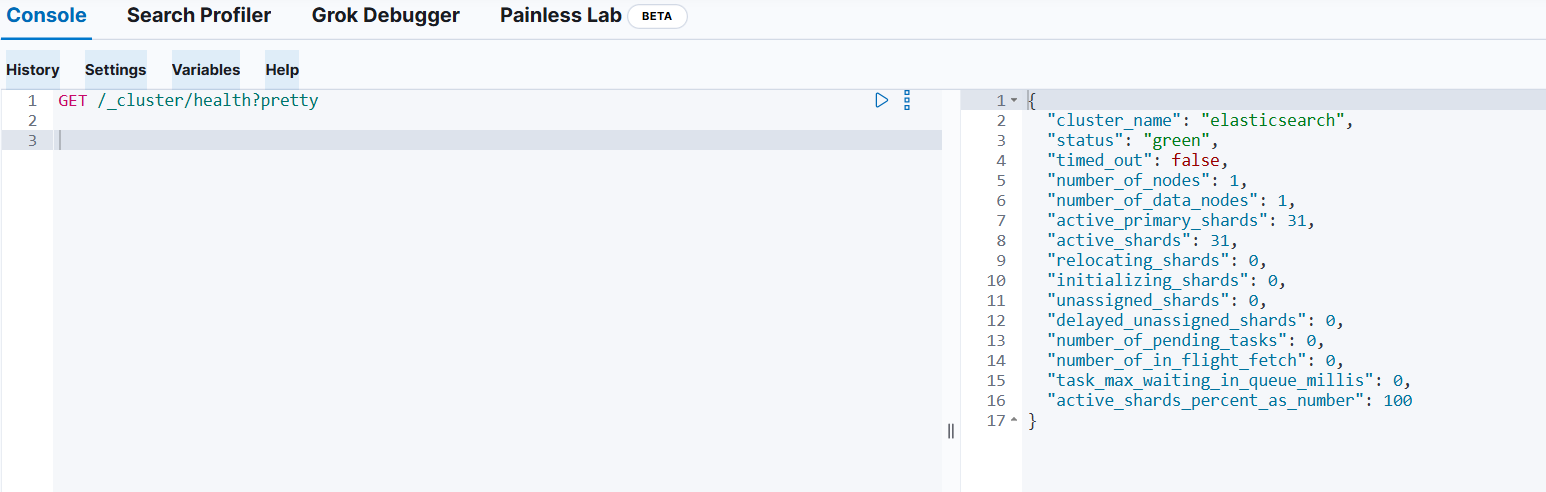
Downloaded elasticsearch and kibana

Copy 3 times of elasticsearch folder

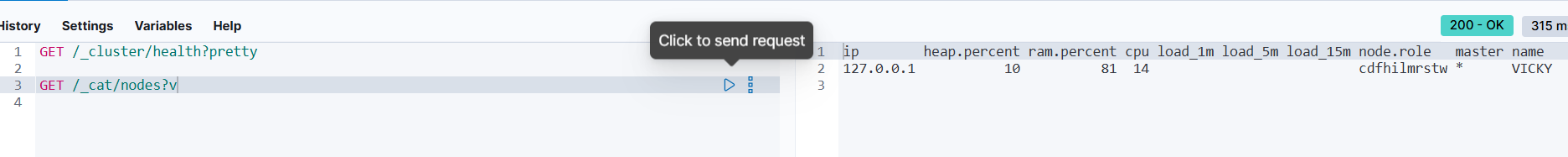
Run kibana and startic the elasticsearch port

Devtools in Elasticsearch-> Go to console

GET /\_cluster/health?pretty

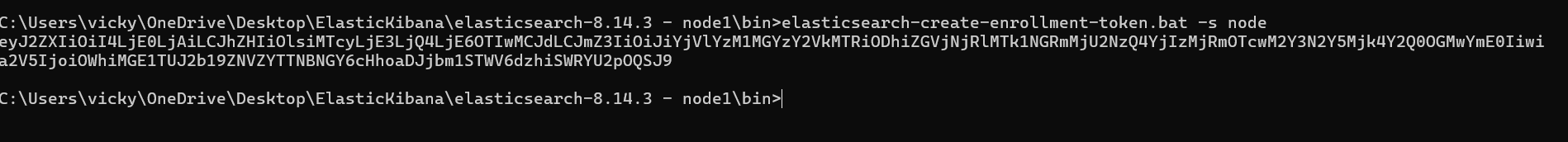


GET /\_cat/nodes?v



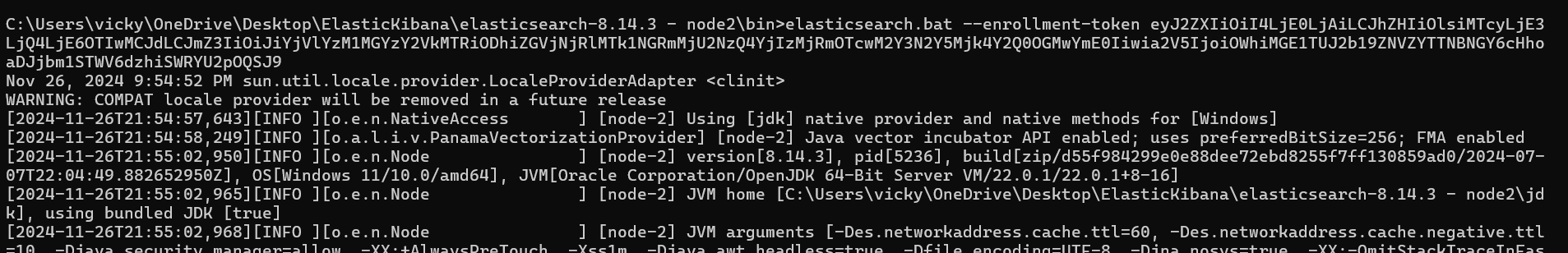
Node 1:

elasticsearch-create-enrollment-token.bat -s node



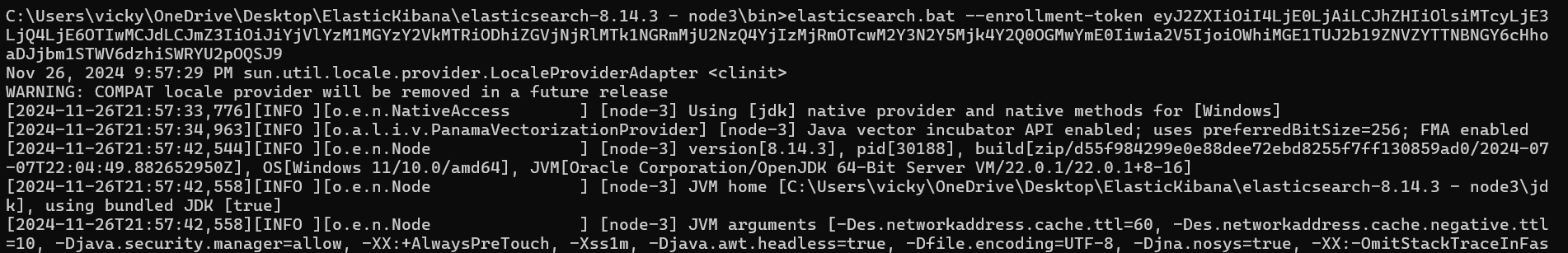
Node 2:

elasticsearch.bat --enrollment-token eyJ2ZXIiOiI4LjE0LjAiLCJhZHIiOlsiMTcyLjE3LjQ4LjE6OTIwMCJdLCJmZ3IiOiJiYjVlYzM1MGYzY2VkMTRiODhiZGVjNjRlMTk1NGRmMjU2NzQ4YjIzMjRmOTcwM2Y3N2Y5Mjk4Y2Q0OGMwYmE0Iiwia2V5IjoiOWhiMGE1TUJ2b19ZNVZYTTNBNGY6cHhoaDJjbm1STWV6dzhiSWRYU2pOQSJ9



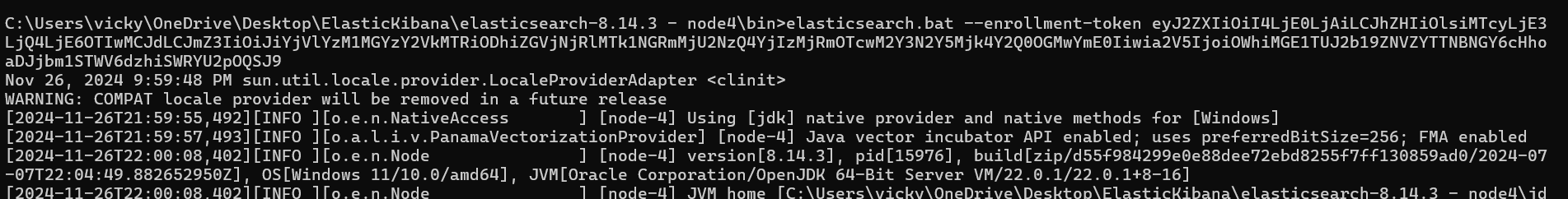
Node 3:

elasticsearch.bat --enrollment-token eyJ2ZXIiOiI4LjE0LjAiLCJhZHIiOlsiMTcyLjE3LjQ4LjE6OTIwMCJdLCJmZ3IiOiJiYjVlYzM1MGYzY2VkMTRiODhiZGVjNjRlMTk1NGRmMjU2NzQ4YjIzMjRmOTcwM2Y3N2Y5Mjk4Y2Q0OGMwYmE0Iiwia2V5IjoiOWhiMGE1TUJ2b19ZNVZYTTNBNGY6cHhoaDJjbm1STWV6dzhiSWRYU2pOQSJ9



Node 4:

elasticsearch.bat --enrollment-token eyJ2ZXIiOiI4LjE0LjAiLCJhZHIiOlsiMTcyLjE3LjQ4LjE6OTIwMCJdLCJmZ3IiOiJiYjVlYzM1MGYzY2VkMTRiODhiZGVjNjRlMTk1NGRmMjU2NzQ4YjIzMjRmOTcwM2Y3N2Y5Mjk4Y2Q0OGMwYmE0Iiwia2V5IjoiOWhiMGE1TUJ2b19ZNVZYTTNBNGY6cHhoaDJjbm1STWV6dzhiSWRYU2pOQSJ9



GET /\_cat/nodes?v

A screenshot of a computer

Description automatically generated

Users :

Run the users.py file

After uploading data run the below command in console

**1. Check Shard Distribution in Elasticsearch**

To inspect the shard distribution of your index and how they are allocated across your cluster's nodes, you can use the following Elasticsearch API call:

This will return a table showing the shard distribution for the users index, with columns like:

* **index**: The name of the index (users).
* **shard**: The shard number.
* **prirep**: Whether the shard is a primary (p) or replica (r).
* **state**: The state of the shard (e.g., STARTED, RELOCATING, etc.).
* **docs**: The number of documents in that shard.
* **store**: The disk space used by the shard.
* **node**: The node where the shard is located.

GET /\_cat/shards/users?v&h=index,shard,prirep,state,docs,store,node



A screenshot of a computer

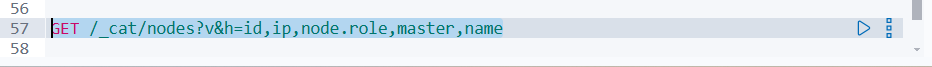
Description automatically generated

**2. Check Node Allocation (Which Node Holds Which Shards)**

To check how the shards are allocated across the nodes in your Elasticsearch cluster, you can use:

This will give you a list of all the nodes in your cluster, their roles, and which one is the master node.

GET /\_cat/nodes?v&h=id,ip,node.role,master,name



A screenshot of a computer

Description automatically generated

**3. Verify Index Shard Settings**

You can verify the shard settings of your index using

GET /users/\_settings

This will return information about the users index settings, including the number of primary shards and replicas.



A screenshot of a computer program

Description automatically generated

**4. Verify Data is Routed Correctly to Shards**

If you want to see which document is in which shard, you can query a specific document and check its routing:

This will return documents and their respective shard number (although it's not as clear as directly querying the shards themselves). However, the shard distribution will be most evident from the previous commands.

GET /users/\_search

{

"query": {

"match\_all": {}

}

}

A blue rectangle with white border

Description automatically generated

A screenshot of a computer program

Description automatically generated

Regional\_trends:

Run the regional\_trends.py file

GET /\_cat/shards/regional\_trends?v&h=index,shard,prirep,state,docs,store,node



A screenshot of a computer

Description automatically generated

GET /regional\_trends/\_settings



A screenshot of a computer code

Description automatically generated

GET /regional\_trends/\_search

{

"query": {

"match\_all": {}

}

}

A blue rectangle with white border

Description automatically generated

A screen shot of a computer code

Description automatically generated

Content:

Run the content.py file

GET /\_cat/shards/content?v&h=index,shard,prirep,state,docs,store,node



A screenshot of a computer

Description automatically generated

GET /content/\_settings



A screenshot of a computer program

Description automatically generated

GET /content/\_search

{

"query": {

"match\_all": {}

}

}

A white rectangular object with a black stripe

Description automatically generated

A computer screen shot of a computer code

Description automatically generated

Interaction History:

Run the interaction\_history.py file

GET /\_cat/shards/interaction\_history?v&h=index,shard,prirep,state,docs,store,node



A screenshot of a computer

Description automatically generated

GET /interaction\_history/\_settings



A screenshot of a computer program

Description automatically generated

GET /interaction\_history/\_search

{

"query": {

"match\_all": {}

}

}

A blue screen with a white border

Description automatically generated

A computer screen shot of a code

Description automatically generated