A report on :

" ELastic stack integration with Honeypot - Tpot "

" Hamza Jameel "
(dev.hamzaj@gmail.com)

Table of contents :

Introduction:

In this project, I successfully implemented a monitoring and detection setup using the Elastic Stack (ELK). This solution revolves around detecting OS ticket queries, streamlines incident tracking and management through features like customizable workflows, automated ticket routing, and real-time notifications. By integrating osTicket with Elastic SIEM, the security of organizations can elevate , enabling seamless incident detection, tracking, and resolution. This integration allows security alerts generated by Elastic SIEM to be automatically converted into actionable tickets in osTicket, ensuring prompt response and effective collaboration.

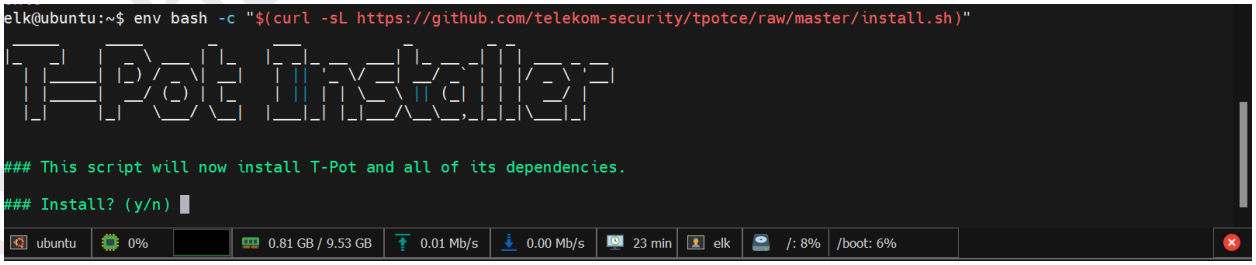
Installations and Configurations :

This setup requires two different installations of different stacks . In this lab, the installations are performed on the Ubuntu base system.

1. Honeypot - Tpot:

T-Pot Honeypot is an open-source, multi-layered honeypot solution designed to capture and analyze malicious activities on a network. It combines multiple honeypot technologies into a single platform, offering a wide range of decoy services to attract and monitor attackers. T-Pot is built on Docker containers and includes tools like **Cowrie**, **Dionaea**, **Elastic Stack (ELK)**, and **Suricata** to collect detailed logs and alerts from various attack vectors. To move forward , install the tpot with single command installation [script :]

env bash -c "$(curl -sLhttps://github.com/telekom-security/tpotce/raw/master/install.sh)"



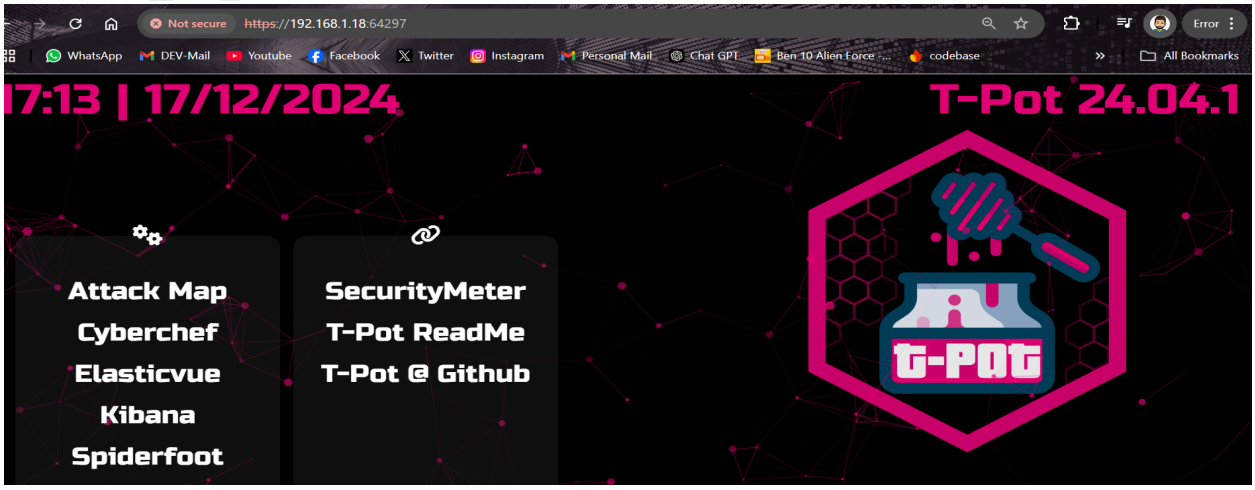
After this , the installation script will start running and user need to enter some credentials for this , remember this installation script is running on the top of docker engine .

```
### Creating base64 encoded htpasswd username and password for T-Pot config file: /home/elk/tpotce/.env

### Now pulling images ...
[+] Pulling 14/26
 ✔ conpot_ipmi Skipped - Image is already being pulled by conpot_guardian_ast                    0.0s
 ✔ map_data Skipped - Image is already being pulled by map_web                                   0.0s
 ✔ conpot_kamstrup_382 Skipped - Image is already being pulled by conpot_guardian_ast            0.0s
 ✔ conpot_IEC104 Skipped - Image is already being pulled by conpot_guardian_ast                  0.0s
 ⠿ dicompot [   ] Pulling                                                                        4.9s
 ⠿ conpot_guardian_ast [   ] Pulling                                                             4.9s
 ⠿ suricata [   ] Pulling                                                                        4.9s
 ✔ medpot Pulled                                                                                 3.6s
 ⠿ logstash [⣿⣿⣿] 189.1MB / 500.5MB Pulling                                                     4.9s
 ⠿ elasticsearch [   ] Pulling                                                                   4.9s
 ⠿ p0f [   ] Pulling                                                                             4.9s
 ⠿ honeytrap [   ] Pulling                                                                       4.9s
 ⠿ spiderfoot [   ] Pulling                                                                      4.9s
 ⠿ ciscoasa [   ] Pulling                                                                        4.9s
 ⠿ map_web [⣿⣿] 16.35MB / 39.98MB Pulling                                                       4.9s
 ⠿ kibana [   ] Pulling                                                                          4.8s
 ✔ nginx Pulled                                                                                  4.1s
 ⠿ ewsposter [   ] Pulling                                                                       4.8s
 ⠿ tpotinit [   ] Pulling                                                                        4.8s
 ⠿ honeypots [   ] Pulling                                                                       4.8s
 ⠿ map_redis [   ] Pulling                                                                       4.8s
 ⠿ fatt [   ] Pulling                                                                            4.8s
 ⠿ adbhoney [   ] Pulling                                                                        4.8s
```

The tpot is installed and running now . Navigate to the Ubuntu command line and type the netstat command for open ports .

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       User       Inode      PID/Program name
tcp        0      0 127.0.0.1:6010          0.0.0.0:*               LISTEN      1000       12923      1072/sshd: elk@pts/
tcp        0      0 127.0.0.1:6791          0.0.0.0:*               LISTEN      0          1635       723/elastic-agent
tcp        0      0 127.0.0.1:6789          0.0.0.0:*               LISTEN      0          7821       723/elastic-agent
tcp        0      0 0.0.0.0:64295           0.0.0.0:*               LISTEN      0          49289      14797/sshd: /usr/sb
tcp6       0      0 ::1:6010                :::*                    LISTEN      1000       12922      1072/sshd: elk@pts/
tcp6       0      0 :::64295                :::*                    LISTEN      0          49291      14797/sshd: /usr/sb

### Done. Please reboot and re-connect via SSH on tcp/64295.
```

Note : **SSH** port has been moved from port **22** to port **64295**. It is now necessary to connect from this port to access the system.



Although the dashboards are already provided in Tpot, the custom installation is recommended for hands-on lab and custom parsing .

2. ELK Stack:

Download the elastic search latest file from the official repository or from their webpage . In my case I am using the base operating system linux ubuntu. For ELK setup , two different modules needed to be downloaded: the elastic search and kibana . Then they both will be interlinked to share and visualize the data .
Use the wget command to download the package and then use dpkg -i command to install the elastic search into the target system.

```
Preparing to unpack elasticsearch-8.15.0-amd64.deb ...
Creating elasticsearch group... OK
Creating elasticsearch user... OK
Unpacking elasticsearch (8.15.0) ...
Setting up elasticsearch (8.15.0) ...
--------------------------- Security autoconfiguration information ------------------------------

Authentication and authorization are enabled.
TLS for the transport and HTTP layers is enabled and configured.

The generated password for the elastic built-in superuser is : cju3j08XZNLHUzoKZ7rr

If this node should join an existing cluster, you can reconfigure this with
'/usr/share/elasticsearch/bin/elasticsearch-reconfigure-node --enrollment-token <token-here>'
after creating an enrollment token on your existing cluster.

You can complete the following actions at any time:

Reset the password of the elastic built-in superuser with
'/usr/share/elasticsearch/bin/elasticsearch-reset-password -u elastic'.

Generate an enrollment token for Kibana instances with
 '/usr/share/elasticsearch/bin/elasticsearch-create-enrollment-token -s kibana'.

Generate an enrollment token for Elasticsearch nodes with
'/usr/share/elasticsearch/bin/elasticsearch-create-enrollment-token -s node'.

--------------------------------------------------------------------------------------------
### NOT starting on installation, please execute the following statements to configure elasticsearch service to start automatically using systemd
 sudo systemctl daemon-reload
```

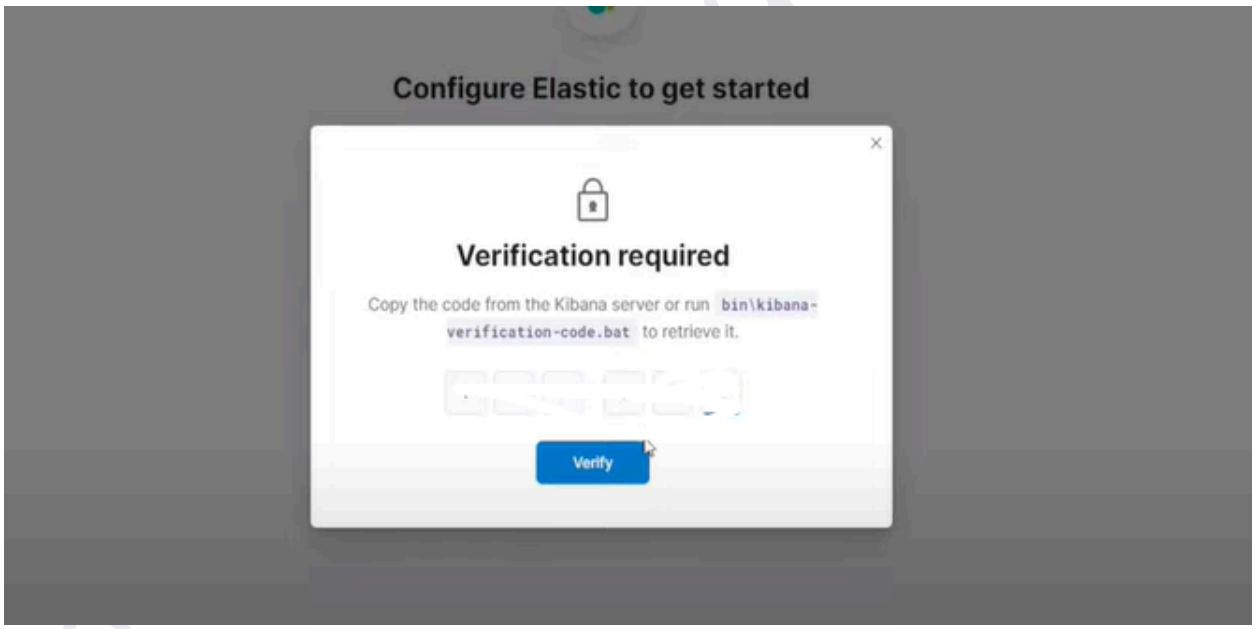
After successful installation it will show some useful credentials that we will need to set up our kibana instance. Note these credentials and save them in a notepad file. After successful setup of our elastic instance we will setup kibana instance.
Download the kibana debian file from the official webpage and use the command dpkg -i to unpack it.
After that , move towards the elastic search directory and create an enrollment token for kibana as shown:

```
root@MyDFIR-ELK:~# root@MyDFIR-ELK:~# cd /usr/share/elasticsearch/bin
root@MyDFIR-ELK:/usr/share/elasticsearch/bin# ls
elasticsearch                        elasticsearch-env            elasticsearch-reconfigure-node  elasticsearch-sql-cli
elasticsearch-certgen                elasticsearch-env-from-file  elasticsearch-reset-password    elasticsearch-sql-cli-8.15.0.jar
elasticsearch-certutil               elasticsearch-geoip          elasticsearch-saml-metadata     elasticsearch-syskeygen
elasticsearch-cli                    elasticsearch-keystore       elasticsearch-service-tokens    elasticsearch-users
elasticsearch-create-enrollment-token elasticsearch-node          elasticsearch-setup-passwords   systemd-entrypoint
elasticsearch-croneval               elasticsearch-plugin         elasticsearch-shard
root@MyDFIR-ELK:/usr/share/elasticsearch/bin# ./elasticsearch-create-enrollment-token --scope kibana
eyJ2ZXIiOiI4LjE0LjAiLCJhZHIiOlsiMjE2LjEyOC4xNzYuMTk3OjkyMDAiXSwiZmdyIjoiODBiYzVhYzViZmE1NTJlMGM1YTY3Y2U4MDhkMTY2MmRhZDljMzU3OTJkNTUyMWExNmMxYTY0Y
ThlODc3ZjBkMCIsImtleSI6Il8zR01QcEVCUDBkZTcxQW5QNTF4Om5GVU5LUEItUmx5Y1c0alp5LVJoQkEifQ==
root@MyDFIR-ELK:/usr/share/elasticsearch/bin#
```

After creating the token from elastic enter it into the kibana dashboard and then create a verification code from kibana's backend. Enter it into the kibana front page as shown:

```
root@MyDFIR-ELK:/usr/share/kibana/bin# root@MyDFIR-ELK:/usr/share/kibana/bin# ls
kibana  kibana-encryption-keys  kibana-health-gateway  kibana-keystore  kibana-plugin  kibana-setup  kibana-verification-code
root@MyDFIR-ELK:/usr/share/kibana/bin#
root@MyDFIR-ELK:/usr/share/kibana/bin# ./kibana-verification-code
Your verification code is:  115 341
root@MyDFIR-ELK:/usr/share/kibana/bin#
```

Add the verification :



Now after these steps all the verifications are done and the instances are created and configured and ready to use now .

HoneyPot Integration with ElasticSearch:

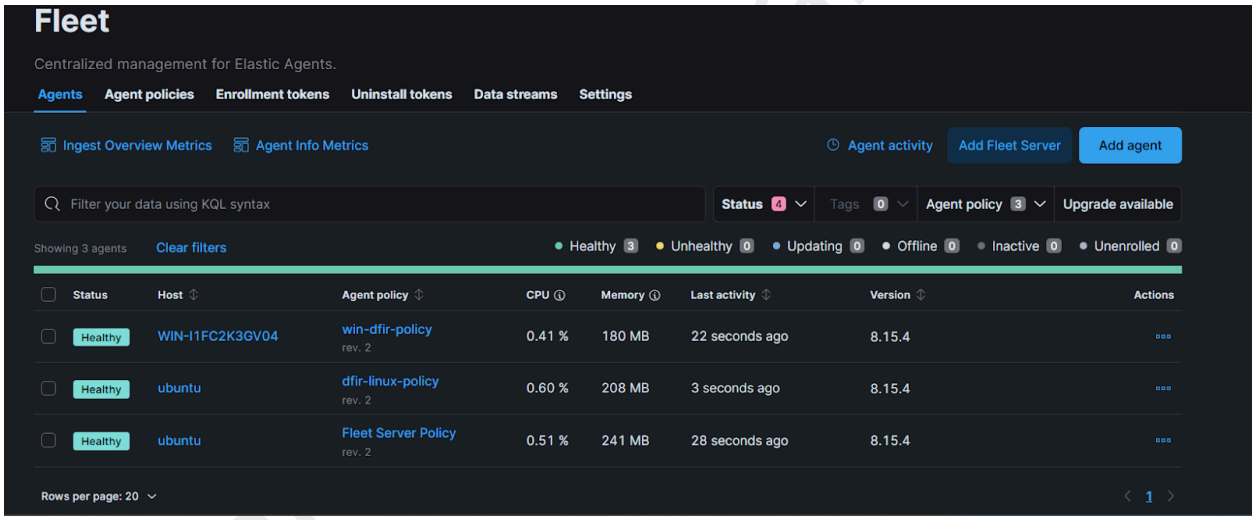
To parse the data generated by mythic server into elastic stack you need the following things :

1. Pre requisites:

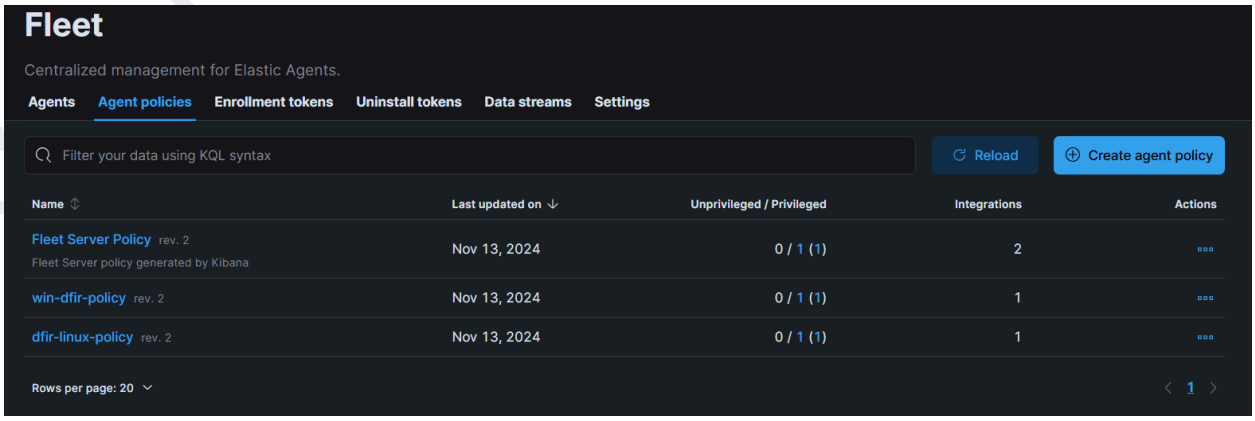
Some of the components which are required before the practical are:

- A fleet server on ELK up and running

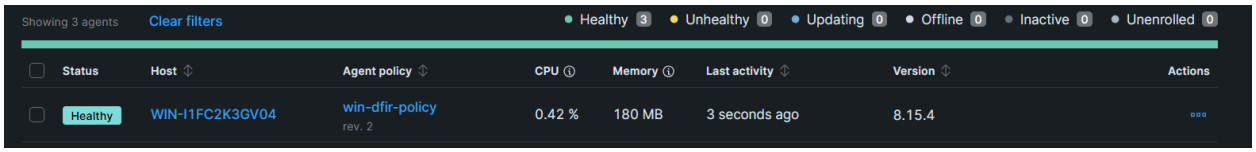
  Navigate to your Kibana instance and then click on 3 lines on the top left corner. Navigate to Fleet at last and set up the fleet policies. From the agent option click on install agent at any endpoint and then also add the flag –insecure in case you are not using the ssl certificates.



- A windows and Linux Fleet policies



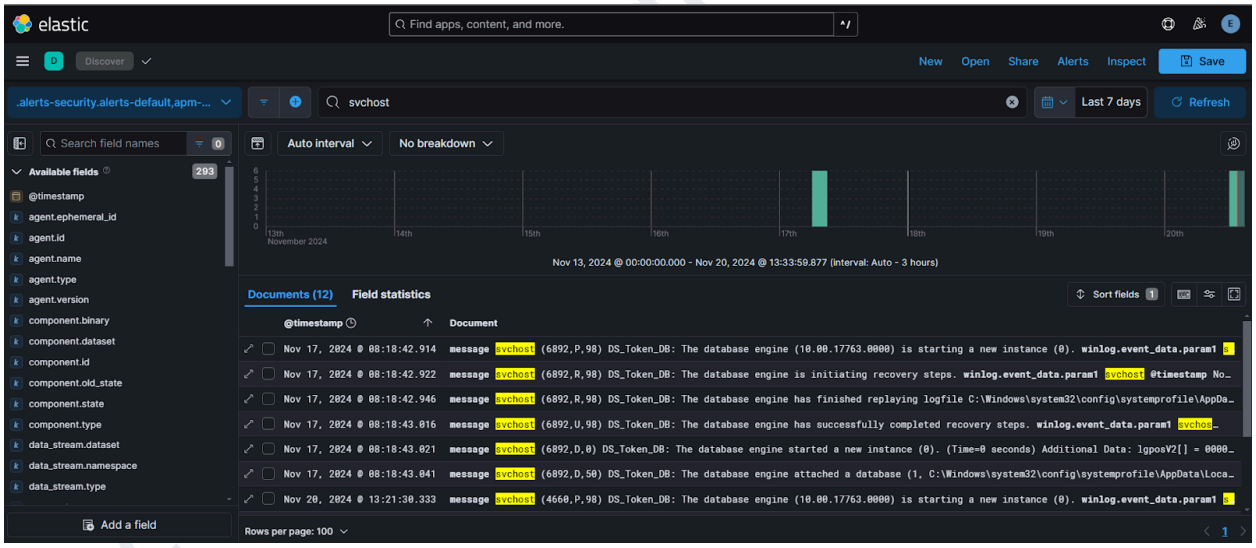
- An ELK agent installed on the Tpot installed endpoint.



This will allow you to parse the syslogs from honeypot os endpoint to the ELK deployed stack. After ensuring these instances , you can proceed with the next steps.

2. Data Filtration on ELK:

Once an agent is installed on the endpoint , we can cross check whether the data from the endpoint is coming to the kibana interface or not. For this , select the index pattern is discover tab and use the keywords of the service we have installed on the endpoint.

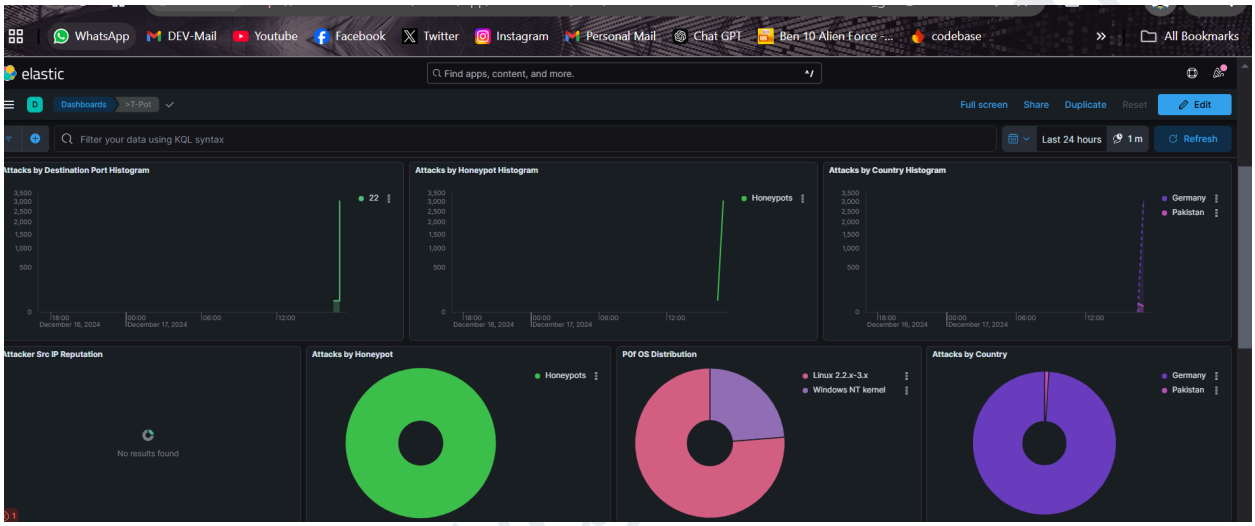


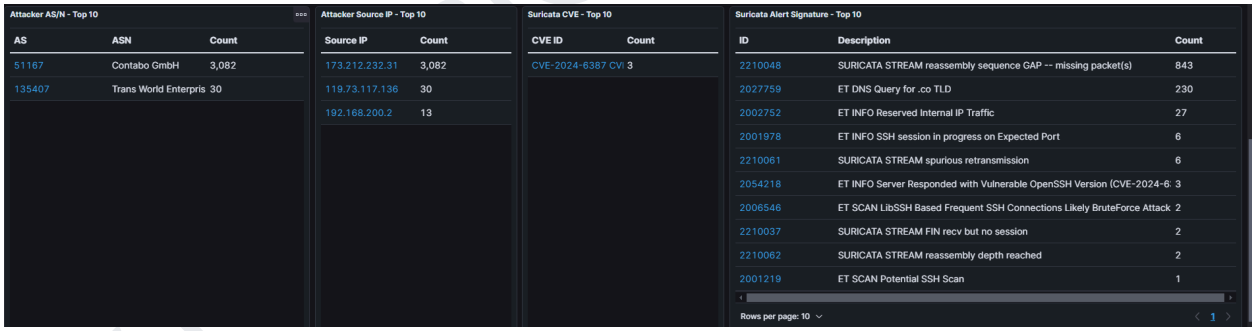
As visible in the screenshot the data is coming and parsing is enabled . Now we can apply different queries to filter the required data . After applying some queries and filtering the data you will get the data in the discover tab . Save the data with queries and now the custom dashboards on use cases can be made

Important note : Although the dashboards are already provided in Tpot but the custom installation is recommended for hands-on lab and custom parsing .

3. Visualization:

The live events and attacks can be visualized at kibana like this :





Attacker AS/N - Top 10

AS	ASN	Count
51167	Contabo GmbH	3,082
135407	Trans World Enterpris	30

Attacker Source IP - Top 10

Source IP	Count
173.212.232.31	3,082
119.73.117.136	30
192.168.200.2	13

Suricata CVE - Top 10

CVE ID	Count
CVE-2024-6387 CVI	3

Suricata Alert Signature - Top 10

ID	Description	Count
2210048	SURICATA STREAM reassembly sequence GAP -- missing packet(s)	843
2027759	ET DNS Query for .co TLD	230
2002752	ET INFO Reserved Internal IP Traffic	27
2001978	ET INFO SSH session in progress on Expected Port	6
2210061	SURICATA STREAM spurious retransmission	6
2054218	ET INFO Server Responded with Vulnerable OpenSSH Version (CVE-2024-6	3
2006546	ET SCAN LibSSH Based Frequent SSH Connections Likely BruteForce Attack	2
2210037	SURICATA STREAM FIN recv but no session	2
2210062	SURICATA STREAM reassembly depth reached	2
2001219	ET SCAN Potential SSH Scan	1

Rows per page: 10

Furthermore, this project can be extended to capture the IOC's like users can add their own threat intelligence databases and can use python scripts to train the machine learning models for future use .

Summary:

The project involved setting up a Fleet Server on ELK for centralized agent management, configuring a honeypot server to simulate adversarial techniques, and installing an Elastic Agent on Windows endpoints for log collection. Using ELK's detection rules and alerting mechanisms.