



SOC L1 Hands-On Lab:

Agent-Based Detection in Elastic-SIEM Using Custom Security Rule

Documented By:

David

Introduction to ELK Stack and Objective

The ELK Stack is a combination of three powerful tools — Elasticsearch, Logstash, and Kibana — which are used together for collecting, storing, analyzing, and visualizing log data in a centralized way.

Elasticsearch is the core engine that stores and searches the data.

Logstash helps in processing and sending data from multiple sources to Elasticsearch.

Kibana is the dashboard tool that allows us to visualize data, search logs, and monitor activity.

In cybersecurity, ELK Stack is especially useful for detecting suspicious behavior, tracking system activity, and performing threat hunting.

The main objective of this task was to:

Set up the **Elastic Agent** on a Linux system,

Connect it with the **ELK Stack**,

Use **Elastic Security** to monitor system activity, and

Create **custom detection rules** (like detecting ping commands) to generate alerts for unusual or malicious behavior.

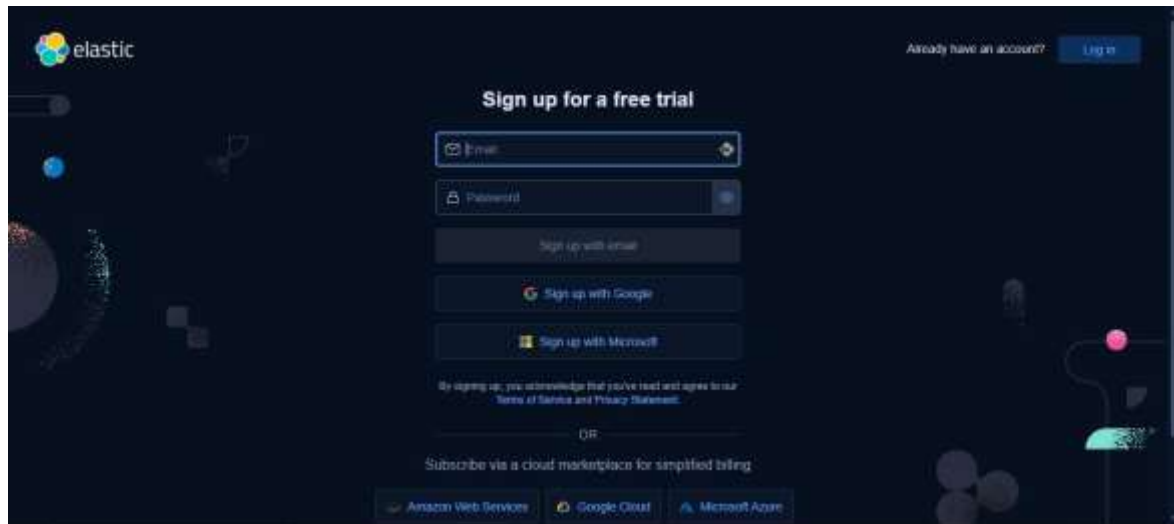
Through this, we will learn on how to build a basic detection system, which is a key skill in SOC (Security Operations Center) environments.

Sign Up on Elastic Cloud

1)- Go to: <https://www.elastic.co/cloud>

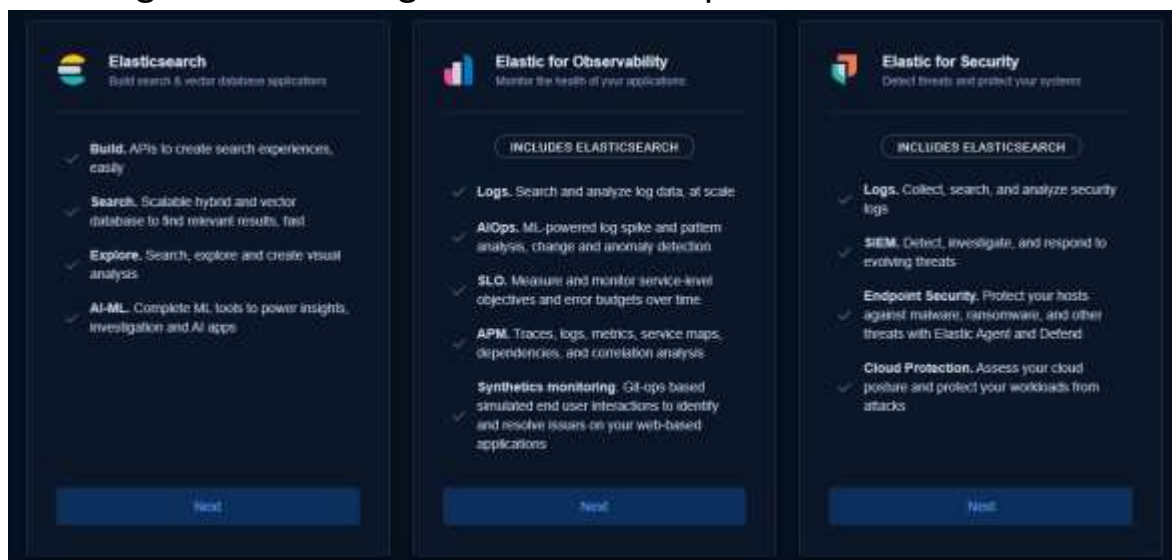
2)- Click on “Start Free Trial”

3)- Create a free Elastic Cloud account using email/password



You will 3 options appear on your interface. But we will choose **Elastic Security** because our main goal was to detect, investigate, and respond to security threats like ping sweeps, network scans, and suspicious system behavior. Besides Elastic Security provides built-in features for:

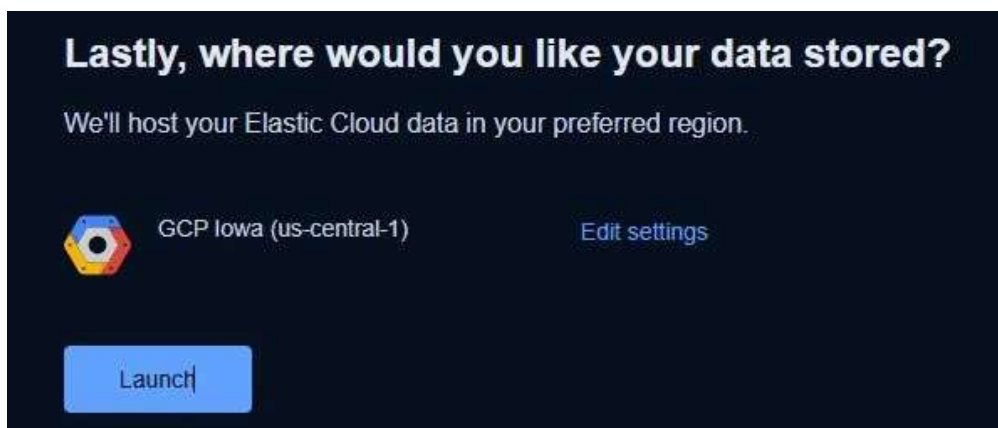
- 1)- Creating and managing detection rules
- 2)- Viewing alerts and timelines
- 3)- Tracking endpoint activity
- 4)- Performing threat hunting and incident response



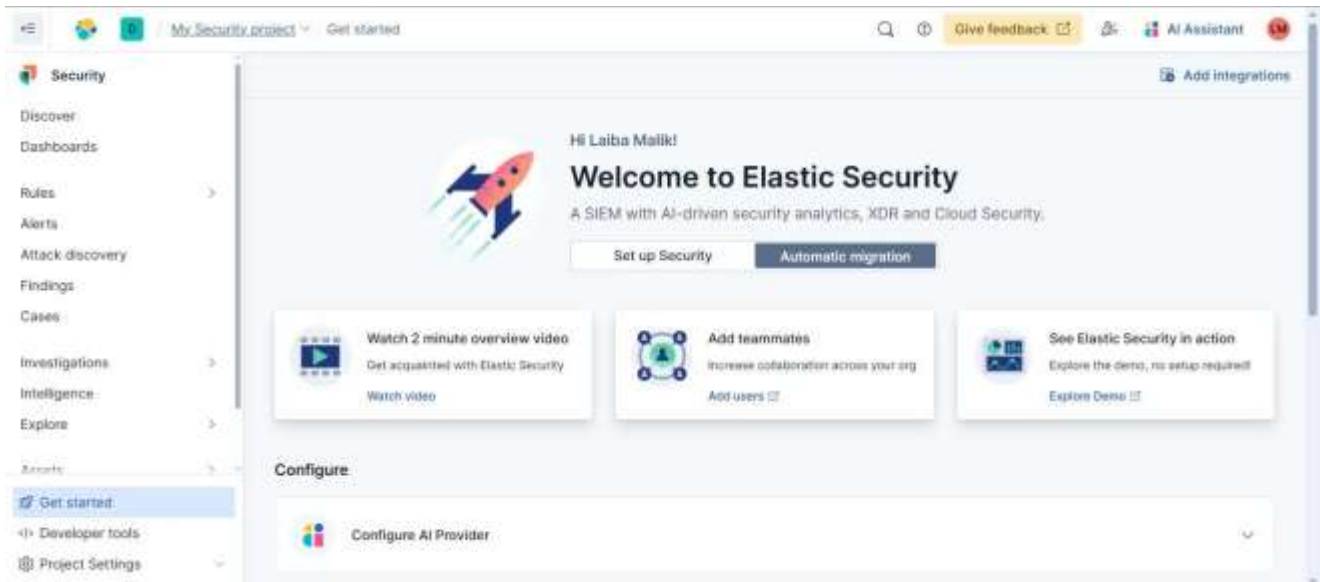
We used **Elastic Cloud Services** because it provides a ready-to-use, fully managed deployment of the ELK Stack without the need to set up or manage servers manually. It saved our time and effort, especially since we didn't have to deal with complex installation, configuration, or maintenance.



We chose **GCP Iowa (us-central1)** because it is a default and commonly available region in Elastic Cloud. It provides low-latency and stable performance for global access. Also, it ensured quick deployment without requiring advanced region selection.



The “**Welcome to Elastic Security**” interface under “**My Security Project**” is the main dashboard where we manage and monitor security operations. It provides access to features like alerts, rules, timelines, and integrations. From here, we can create detection rules, view threat data, and perform investigations — all in one place.



Virtual Box Installation:

- 1)- First, download the latest version of Oracle Virtual Box from its official website: <https://www.virtualbox.org>.
- 2)- Install it using the on-screen instructions. No need to change default settings during setup.

Downloading the Kali Linux Image File:

- 3)- Pre-configured Kali Linux image in .7z format through Google Drive.
- 4)- File name: [kali-linux-2023.2-virtualbox-amd64.7z](#) 5)- Download it from the provided link.

Extracting the Image File:

- 6)- After download, use **7-Zip** or **WinRAR** to extract the .7z file.
- 7)- The extracted file will be a **.vbox** file and a **.vdi** file.

Importing the Kali Linux VM into Virtual Box:

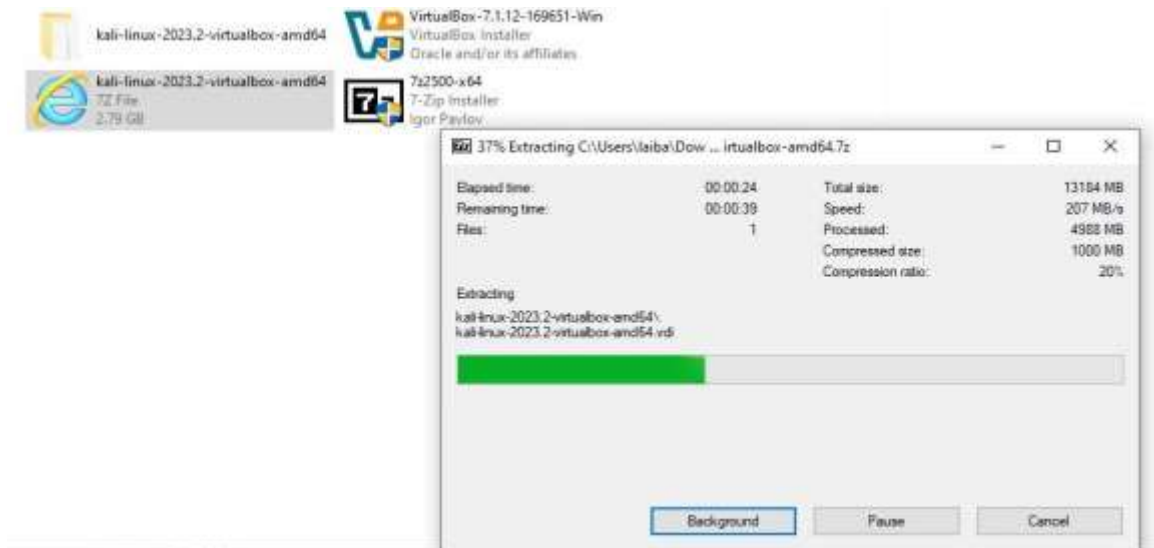
- 8)- Open Virtual Box.
- 9)- Click on **“Machine” > “Add”**, then navigate to the folder where you extracted the .vbox file.
- 10)- Select the .vbox file and click **Open**.
- 11)- The Kali Linux VM will be added to Virtual Box.

Starting the Virtual Machine:

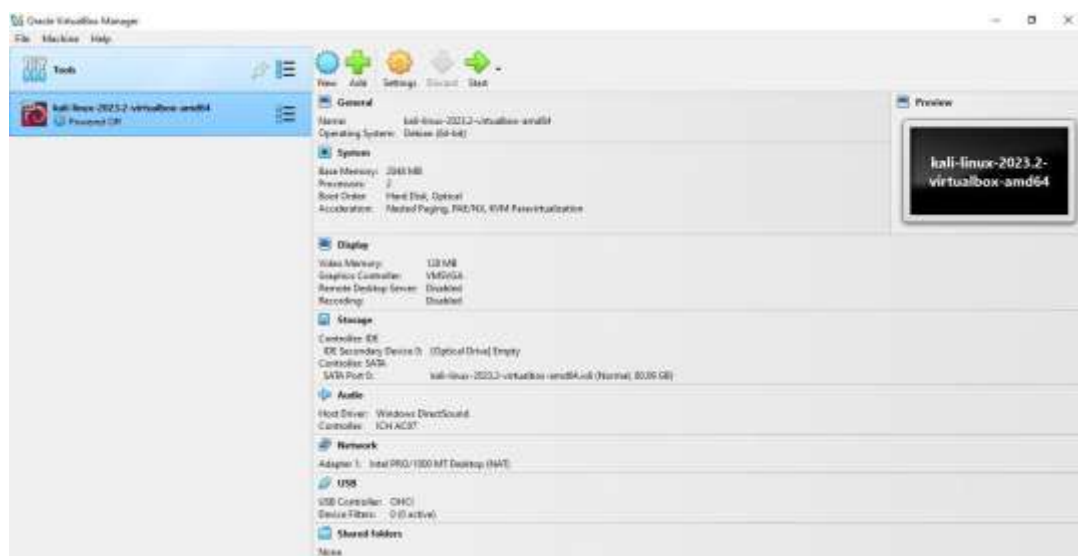
12)- Click on the newly added Kali Linux VM.

13)- Click **Start**.

14)- Kali Linux will boot up without needing manual installation.



Power On [kali-linux-2023.2-virtualbox-amd64](#)

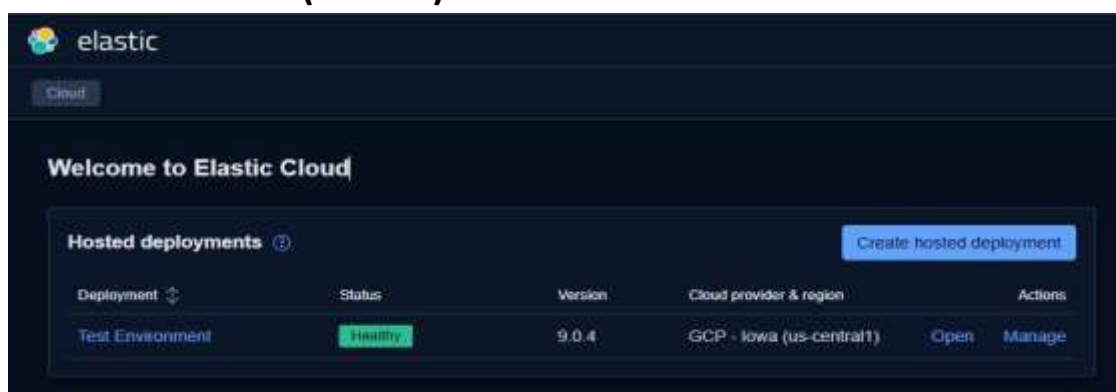


Finally the setup is **completed**.



Go on given link as <https://cloud.elastic.co/home> as we are going to deploy the **Elastic Agent** on the Cloud

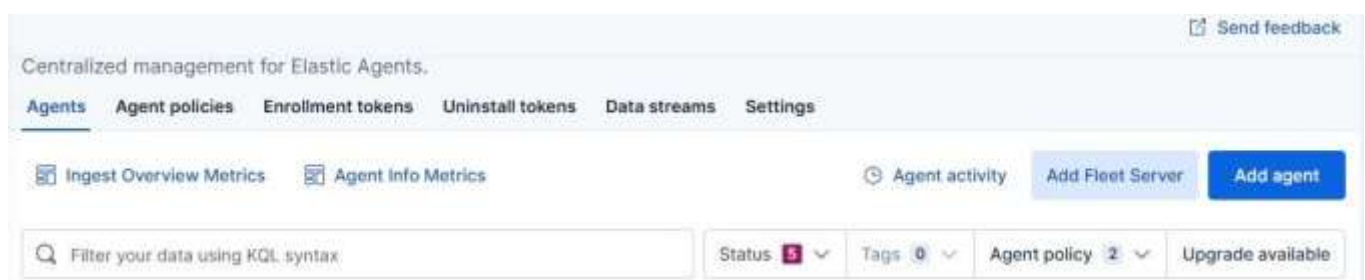
Choose the **Create hosted Deployment** option , from here you will deploy and choose any name of your environment I have selected **“Test Environment”** it will take a few moments (**5 min**).



We scrolled down to find the **Asset option** to view and manage connected hosts, endpoints, and network data. This helped us confirm that our **Linux system and agents** were properly integrated. It ensured that data was being collected and the system was ready for **threat detection**.



We chose the **Fleet** option to manage and monitor all agents from a centralized interface. Here, we saw the default **Elastic Cloud Agent Policy**, which defines what data the agent collects. To deploy our own agent, we clicked on “**Add Agent**” to connect our Linux machine. This step was essential to start sending system and network logs to Elastic Security.



Now Select the Policy, if mentioned, leave it as default (**mine: Agent Policy 1**)

Choose **Elastic Fleet Option (Recommended)**

Wait to proceed Further

A side command will appear copy it and that you will run on your setup [kali-linux-2023.2-virtualbox-amd64](#)

Here below are the shown commands that I Run on Kali:

At First it might show you error, Paste the Command as it is by using **Ctrl+Shift+C**


```

File Actions Edit View Help
(kali@kali)-[~]
└─$ curl -L -O https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-9.0.4-linux-x86_64.tar.gz
└─$ tar xzvf elastic-agent-9.0.4-linux-x86_64.tar.gz
└─$ cd elastic-agent-9.0.4-linux-x86_64
└─$ sudo ./elastic-agent install --url=https://4e39a0da5f114e81bce6c983171f167e.fleet.us-central1.gcp.cloud.es.io:443 --enrollment-token=VFHVX1ZaZ0J2QjI2T3BJT0ZFNTI6Wkx5RFhMVVR3QWZyYakxCcm5XMHR5Zw==
[200-curl] -L -O https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-9.0.4-linux-x86_64.tar.gz
zsh: bad pattern: ^[[200-curl

(kali@kali)-[~]
└─$ tar xzvf elastic-agent-9.0.4-linux-x86_64.tar.gz
tar (child): elastic-agent-9.0.4-linux-x86_64.tar.gz: Cannot open: No such file or directory
tar (child): Error is not recoverable: exiting now
tar: Child returned status 2
tar: Error is not recoverable: exiting now

(kali@kali)-[~]
└─$ cd elastic-agent-9.0.4-linux-x86_64
cd: no such file or directory: elastic-agent-9.0.4-linux-x86_64

(kali@kali)-[~]
└─$ sudo ./elastic-agent install --url=https://4e39a0da5f114e81bce6c983171f167e.fleet.us-central1.gcp.cloud.es.io:443 --enrollment-token=VFHVX1ZaZ0J2QjI2T3BJT0ZFNTI6Wkx5RFhMVVR3QWZyYakxCcm5XMHR5Zw==
[sudo] password for kali:
sudo: ./elastic-agent: command not found

```

```

(kali@kali)-[~]
└─$ curl -L -O https://artifacts.elastic.co/downloads/beats/elastic-agent/elastic-agent-9.0.4-linux-x86_64.tar.gz
└─$ tar xzvf elastic-agent-9.0.4-linux-x86_64.tar.gz
└─$ cd elastic-agent-9.0.4-linux-x86_64
└─$ sudo ./elastic-agent install --url=https://4e39a0da5f114e81bce6c983171f167e.fleet.us-central1.gcp.cloud.es.io:443 --enrollment-token=VFHVX1ZaZ0J2QjI2T3BJT0ZFNTI6Wkx5RFhMVVR3QWZyYakxCcm5XMHR5Zw==

```

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload	Upload	Total	Spent	Left
0	0	0	0	0	0	0:00:00	0
0	0	0	0	0	0	0:00:00	0
0	423M	0 29172	0	0 22650	0	5:26:41	0:00:01 5:26:40 2264
0	423M	0 4015k	0	0 1970k	0	0:03:40	0:00:02 0:03:38 1970
1	423M	1 5996k	0	0 1973k	0	0:03:39	0:00:03 0:03:36 1973
2	423M	2 12.3M	0	0 3121k	0	0:02:18	0:00:04 0:02:14 3122
4	423M	4 18.3M	0	0 3731k	0	0:01:56	0:00:05 0:01:51 3787
5	423M	5 23.8M	0	0 4049k	0	0:01:47	0:00:06 0:01:41 5140
6	423M	6 28.9M	0	0 4207k	0	0:01:43	0:00:07 0:01:36 5119
7	423M	7 33.5M	0	0 4275k	0	0:01:41	0:00:08 0:01:33 5674
9	423M	9 38.9M	0	0 4416k	0	0:01:38	0:00:09 0:01:29 5461
10	423M	10 44.4M	0	0 4539k	0	0:01:35	0:00:10 0:01:25 5353
11	423M	11 49.9M	0	0 4637k	0	0:01:33	0:00:11 0:01:22 5347
13	423M	13 55.7M	0	0 4743k	0	0:01:31	0:00:12 0:01:19 5496
14	423M	14 60.9M	0	0 4789k	0	0:01:30	0:00:13 0:01:17 5616
15	423M	15 65.4M	0	0 4770k	0	0:01:30	0:00:14 0:01:16 5412
16	423M	16 70.5M	0	0 4799k	0	0:01:30	0:00:15 0:01:15 5320
17	423M	17 75.5M	0	0 4822k	0	0:01:29	0:00:16 0:01:13 5232
19	423M	19 80.5M	0	0 4839k	0	0:01:29	0:00:17 0:01:12 5069
20	423M	20 85.5M	0	0 4854k	0	0:01:29	0:00:18 0:01:11 5023
21	423M	21 89.8M	0	0 4834k	0	0:01:29	0:00:19 0:01:10 5014
22	423M	22 94.8M	0	0 4844k	0	0:01:29	0:00:20 0:01:09 4981
23	423M	23 100M	0	0 4867k	0	0:01:29	0:00:21 0:01:08 5011

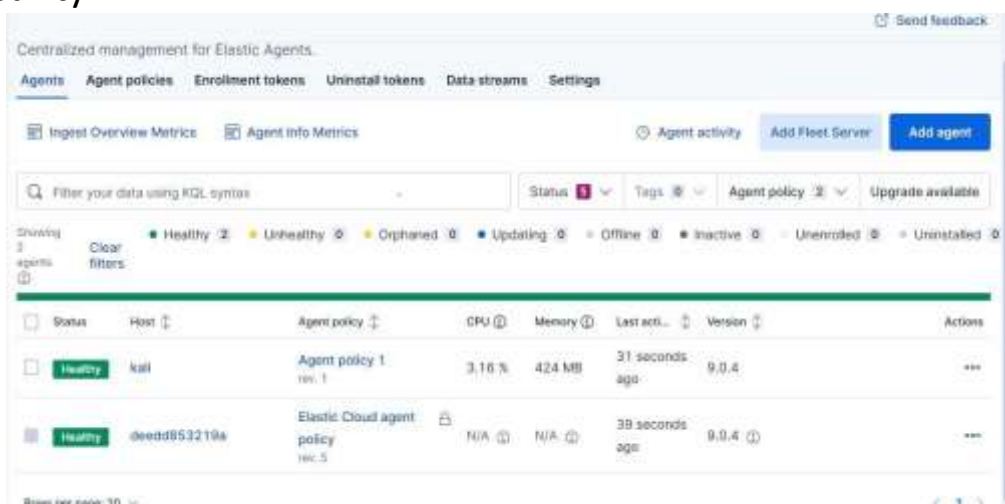
After completing the setup process, we saw a confirmation message that the **Elastic Agent was successfully installed**. This meant that our Linux system was now connected to Elastic Security and ready to send logs. It confirmed that the agent was active and following the assigned integration policy.

```

-agent
elastic-agent-9.0.4-linux-x86_64/data/elastic-agent-d49717/components/pf-host
-agent.spec.yml
elastic-agent-9.0.4-linux-x86_64/.elastic-agent.active.commit
elastic-agent-9.0.4-linux-x86_64/data/elastic-agent-d49717/elastic-agent
elastic-agent-9.0.4-linux-x86_64/LICENSE.txt
elastic-agent-9.0.4-linux-x86_64/elastic-agent.reference.yml
elastic-agent-9.0.4-linux-x86_64/data/elastic-agent-d49717/otelcol
elastic-agent-9.0.4-linux-x86_64/.build_hash.txt
elastic-agent-9.0.4-linux-x86_64/elastic-agent
Elastic Agent will be installed at /opt/Elastic/Agent and will run as a service. Do you want to continue? [Y/n]:Y
[+] Service Started [17s] Elastic Agent successfully installed, starting enrollment.
[+] Waiting For Enroll ... [18s] {"log.level":"info","@timestamp":"2025-07-29T04:17:35.983-0400","log.origin":{"function":"github.com/elastic/elastic-agent/internal/pkg/agent/cmd.(*enrollCmd).enrollWithBackoff","file.name":"cmd/enroll_cmd.go","file.line":534},"message":"Starting enrollment to URL: https://4e39a0da5f114e81bce6c983171f167e.fleet.us-central1.gcp.cloud.es.io:443/","ecs.version":"1.6.0"}
[+] Waiting For Enroll ... [22s] {"log.level":"info","@timestamp":"2025-07-29T04:17:39.549-0400","log.origin":{"function":"github.com/elastic/elastic-agent/internal/pkg/agent/cmd.(*enrollCmd).daemonReloadWithBackoff","file.name":"cmd/enroll_cmd.go","file.line":497},"message":"Restarting agent daemon, attempt 0","ecs.version":"1.6.0"}
[+] Waiting For Enroll ... [22s] {"log.level":"info","@timestamp":"2025-07-29T04:17:39.606-0400","log.origin":{"function":"github.com/elastic/elastic-agent/internal/pkg/agent/cmd.(*enrollCmd).Execute","file.name":"cmd/enroll_cmd.go","file.line":315},"message":"Successfully triggered restart on running Elastic Agent.","ecs.version":"1.6.0"}
Successfully enrolled the Elastic Agent.
[+] Done [22s]
Elastic Agent has been successfully installed.

```

Now go back to **Elastic Cloud**, and you will see your deployed agent listed on the screen. If the status shows **“Healthy”**, it means the agent is working correctly, sending data, and following the assigned policies without any issues. This confirms that the agent is **active**, connected, and successfully integrated with Elastic Security.



Centralized management for Elastic Agents.

Agents Agent policies Enrollment tokens Uninstall tokens Data streams Settings

Ingest Overview Metrics Agent Info Metrics Agent activity Add Fleet Server Add agent

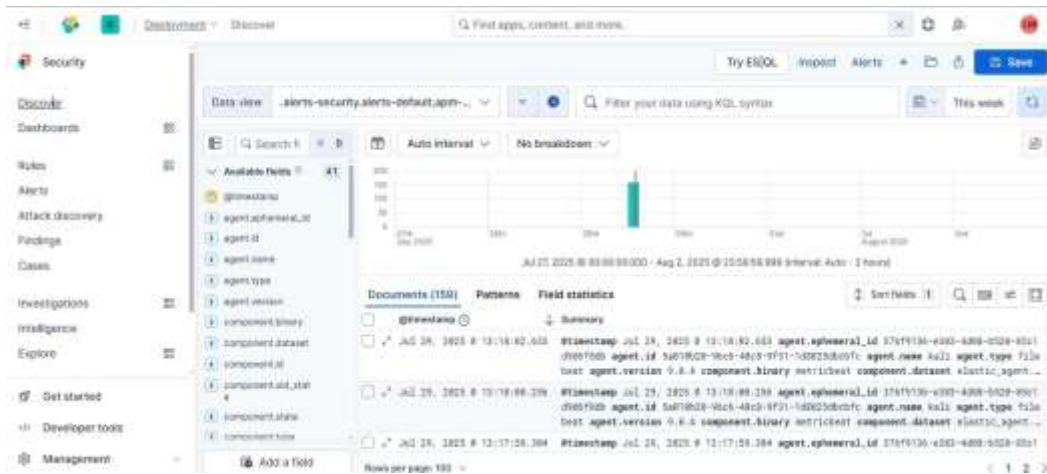
Filter your data using RQL syntax Status Tags Agent policy Upgrade available

Showing 2 agents Clear filters

Status	Host	Agent policy	CPU	Memory	Last acti...	Version	Actions
Healthy	kali	Agent policy 1 rev. 1	3.16 %	424 MB	31 seconds ago	9.0.4	...
Healthy	deedd853219a	Elastic Cloud agent policy rev. 5	N/A	N/A	39 seconds ago	9.0.4	...

Rows per page: 20

Scroll up and click on the **Discover** option to access raw log data from connected agents. An interface will appear where you can **search, filter, and analyze logs** in real-time.



Expand the first one **July 29, 2025 13:18:02.653** and search the Host Name and also the IP.

Table JSON

Host Name

Selected only

Field	Value
host.hostname	kali
host.name	kali
host.os.codename	kali-rolling
host.os.name	Kali GNU/Linux

To check the IP address, go to your **Kali Linux terminal** and run the command **ip a**. You will see that the **same IP address** displayed in Elastic is also shown here. This confirms that the agent is correctly installed on your Kali system.

Table JSON

IP

Selected only

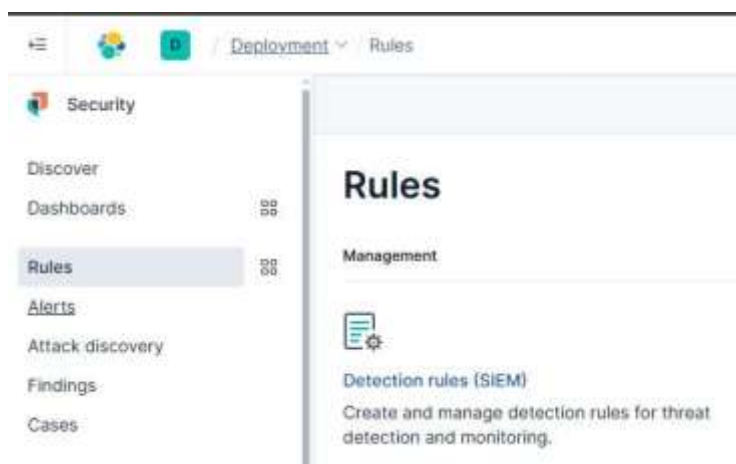
Field	Value
host.ip	[10.0.2.15, fd17:625c:f037:2:234b:4e0f:a52a:aab3, fe80::99f6:d87:f5ea:dc8a]
log.logger	publisher_pipeline_output
log.origin.file.name	pipeline/client_worker.go
log.origin.function	github.com/elastic/beats/v7/libbeat/publisher/pipeline.(*netClientWorker).run

IP is **10.0.2.15** (highlighted).

```
(kali㉿kali)-[~/elastic-agent-9.0.4-linux-x86_64]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def
  ault qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP g
  roup default qlen 1000
    link/ether 08:00:27:53:0c:ba brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
        valid_lft 84996sec preferred_lft 84996sec
    inet6 fd17:625c:f037:2:234b:4e0f:a52a:aab3/64 scope global dynamic nopref
  ixroute
        valid_lft 86295sec preferred_lft 14295sec
    inet6 fe80::99f6:d87:f5ea:dc8a/64 scope link noprefixroute
```

We went to the **Rules tab** to view and manage detection rules that help identify.

These rules automatically generate **alerts** when certain conditions are met in the logs. A **custom rule** is a rule that we create manually based on our own specific use case. We created a custom rule to **detect ping (ICMP) activity** from our Linux system for security monitoring.



Create New Rule.



Select Custom Query Alerts option to define our own specific detection condition. This allowed us to **target ICMP (ping) traffic** that default rules might not cover.

Create new rule

1 Define rule

Rule type



Custom query

Use KQL or Lucene to detect issues across indices.

✓ Selected



Machine Learning

Select ML job to detect anomalous activity.

Select



Threshold

Aggregate query results to detect when number of matches exceeds threshold.

Select

Define rule

 Edit

Index patterns

apm-*transaction* auditbeat-*
endgame-* filebeat-* logs-*
packetbeat-* traces-apm* winlogbeat-*
-*elastic-cloud-logs-*

Custom query

event.action : "ping"

Rule type

Query

Timeline template

None

About rule

 Edit

Name

Linux Ping Detection

Description

Triggers alert when a ping (ICMP) is executed on the Linux system using Sysmon logs.

Max alerts per run

100

Severity

● Low

Risk score

21

Indicator prefix override

Tags

ping linux sysmon icmp


Schedule rule
Edit

Runs every	5m
Additional look-back time	1m

For further Proceed of result,Click **Create and Enable Rule**. Go on Linux Machine and then run this command

```
(kali@kali)-[/elastic-agent-9.0.4-linux-x86_64]
$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data:
 64 bytes from 8.8.8.8: icmp_seq=1 ttl=255 time=44.0 ms
 64 bytes from 8.8.8.8: icmp_seq=2 ttl=255 time=47.4 ms
 64 bytes from 8.8.8.8: icmp_seq=3 ttl=255 time=46.6 ms
 64 bytes from 8.8.8.8: icmp_seq=4 ttl=255 time=44.9 ms
 64 bytes from 8.8.8.8: icmp_seq=5 ttl=255 time=45.7 ms
 64 bytes from 8.8.8.8: icmp_seq=6 ttl=255 time=61.7 ms
 64 bytes from 8.8.8.8: icmp_seq=7 ttl=255 time=43.1 ms
 64 bytes from 8.8.8.8: icmp_seq=8 ttl=255 time=46.6 ms
 64 bytes from 8.8.8.8: icmp_seq=9 ttl=255 time=44.8 ms
 64 bytes from 8.8.8.8: icmp_seq=10 ttl=255 time=48.1 ms
 64 bytes from 8.8.8.8: icmp_seq=11 ttl=255 time=45.7 ms
 64 bytes from 8.8.8.8: icmp_seq=12 ttl=255 time=47.3 ms
 64 bytes from 8.8.8.8: icmp_seq=13 ttl=255 time=45.2 ms
 64 bytes from 8.8.8.8: icmp_seq=14 ttl=255 time=43.1 ms
 64 bytes from 8.8.8.8: icmp_seq=15 ttl=255 time=50.7 ms
 64 bytes from 8.8.8.8: icmp_seq=16 ttl=255 time=42.6 ms
 64 bytes from 8.8.8.8: icmp_seq=17 ttl=255 time=49.1 ms
 64 bytes from 8.8.8.8: icmp_seq=18 ttl=255 time=44.2 ms
```

At first, you won't see any alerts in the **Discover tab** because no data is being collected yet.To fix this, you need to **add integrations** to the policy linked with your installed agent.These integrations tell the agent what kind of data to collect, like network traffic or system logs. The added integrations are:

Integrations

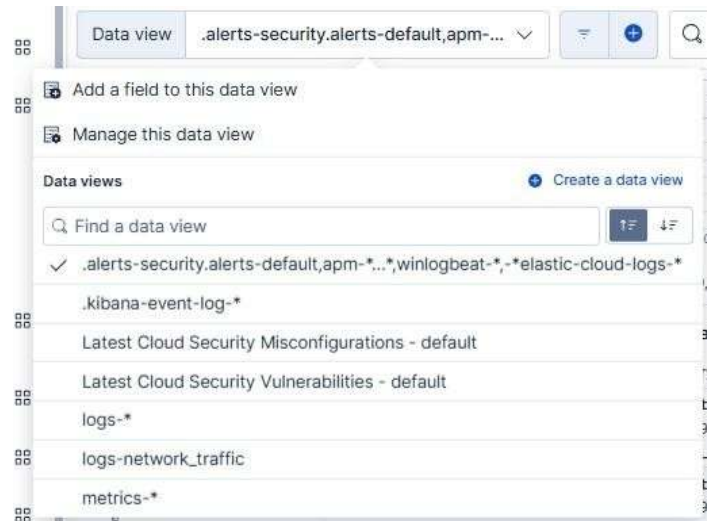
>  system-1

>  sysmon_linux-1

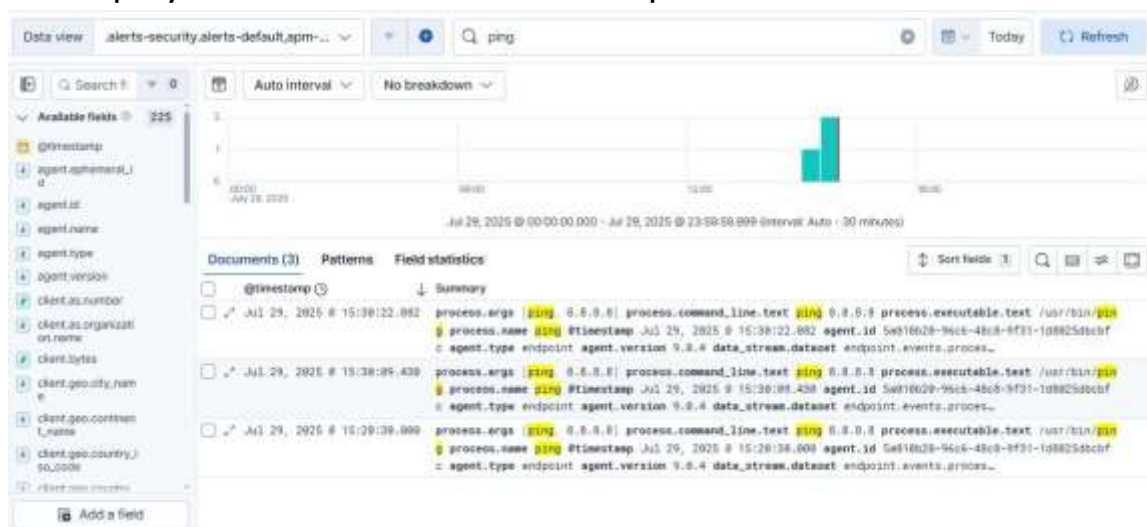
>  network_traffic-1

>  elastic defend

In-order to Add Integrations go on the tab **Assets > Fleet > Add Integration**. Go back to the Linux and run the command **ping 8.8.8.8**. Now move to the Discover Tab and choose the Data Review as marked shown below:



The **alerts** are displayed below shown in the Snap-Shot.



By expanding any log entry, we can view its detailed information.

We can also add relevant fields to the view based on what we need.

For example: **user.name**, **process.command_line**, **host.os.type**, **process.name**.



To set up email notifications for a detection rule in Elastic, go to the **Actions** tab while creating or editing the rule. Click **"Add action"** and choose **Elastic-CloudSMTP** as the notification method. Select **"Summary of alerts per rule run"** as the action type. In the message field, you can write something like:
"The detection rule '{{context.rule.name}}' has triggered {{state.signals_count}} alerts."



Email Received



Conclusion: Through this task, we learned how to set up and use the Elastic Security platform for threat detection. We understood how to install and connect an agent to start collecting logs from a Linux system. We explored the Discover tab to view real-time data and verify system activity. We also created a custom detection rule to monitor **ping (ICMP) traffic**. Overall, this activity helped us gain hands-on experience in log analysis and security monitoring.