

# **SOC Analyst Simulation Project**

## **Hameet Benipal**

The SOC Analyst Simulation Project provided practical, hands-on experience replicating the core responsibilities of a Security Operations Center (SOC) analyst. This project involved setting up a virtualized environment comprising multiple Linux and Windows VMs, including an ELK stack for log collection and analysis, a fleet server for telemetry management, Kali Linux for conducting simulated attacks, and a Mythic server to execute command-and-control (C2) operations. Using tools like Crowbar for brute-force SSH and RDP attacks and Mythic agents for C2 attacks, the environment was configured to mimic real-world cyber threats. Elastic SIEM was leveraged to monitor and detect these attacks through custom detection rules, with logs and telemetry analyzed and visualized using Elastic dashboards. To simulate real-world incident response workflows, OSTicket was integrated with Elastic to automatically generate incident tickets for detected alerts, enabling the assignment, tracking, and resolution of security incidents. This project provided valuable experience in security monitoring, log analysis, incident response, and cybersecurity automation, aligning closely with the daily tasks of a modern SOC analyst.

# VM Configuration and Elastic SIEM Setup

Cloud Compute

Location

Q Search

+ Deploy

<input type="checkbox"/>	Name	OS	Location	Charges	Status	
<input type="checkbox"/>	<b>hameet-ELK</b> 16384.00 MB Optimized Cloud - 137.220.54.158		Toronto	\$21.25	Running	...
<input type="checkbox"/>	<b>Hameet-Fleet-Server</b> 4096.00 MB Optimized Cloud - 216.128.179.107		Toronto	\$0.54	Running	...
<input type="checkbox"/>	<b>HAMEET-LINUX</b> 1024.00 MB AMD High Performance - 155.138.129.112		Toronto	\$0.09	Running	...
<input type="checkbox"/>	<b>hameet-windows-server</b> 2048.00 MB Regular Cloud Compute - 216.128.185.53		Toronto	\$3.40	Running	...

← Manage Firewall Group

Group ID: 9b6b6237-1bbb-41a6-a74f-4aed8a5f49da Created: 2025-01-23 06:36:11 Updated: 2025-01-27 20:03:22

Description

hameet-soc-challenge

Group Rules

5/50

Linked Instances

1

IPv4 Rules

IPv6 Rules

Linked Instances

Inbound IPv4 Rules

Action	Protocol	Port (or range)	Source	Notes	Action
accept	SSH	22	Anywhere	0.0.0.0/0	Add note +
accept	TCP	1 - 65535	99.240.179.192/32		
accept	TCP	1 - 65535	155.138.129.112/32		
accept	TCP	1 - 65535	216.128.179.107/32		
accept	TCP	1 - 65535	216.128.185.53/32		
accept	SSH	22	99.240.179.192/32		
drop	any	0 - 65535	0.0.0.0/0		(default)

Ref 1: Spin up 4 VMs and configure all accordingly. Install ELK on corresponding VM, windows server on another, Fleet Server on another and Sysmon on Windows server. Set up basic firewall to prevent all traffic, allow from my IP and allow other VMs to communicate.

## Fleet

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 KQL syntax

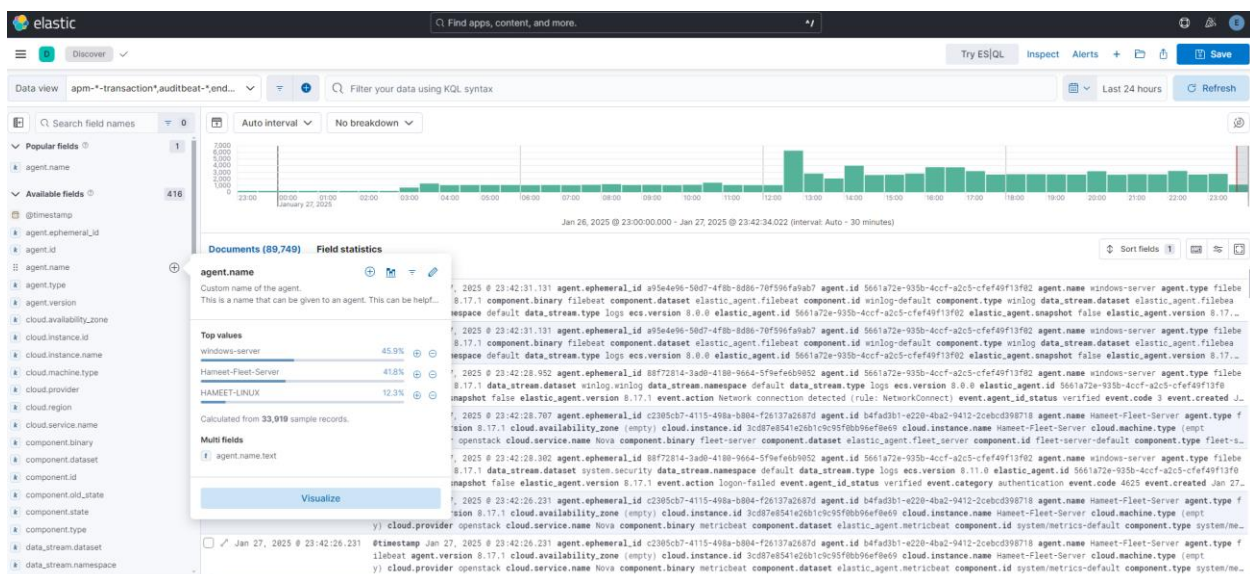
Status 4 Tags 0 Agent policy 3 Upgrade available

Showing 3 agents Clear filters Healthy 3 Unhealthy 0 Updating 0 Offline 0 Inactive 0 Unenrolled 0

Status	Host	Agent policy	CPU	Memory	Last activity	Version	Actions
Healthy	HAMEET-LINUX	HAMEET-LINUX-POLICY rev. 1	0.68 %	187 MB	20 seconds ago	8.17.1	...
Healthy	windows-server	Hameet-Windows-Policy rev. 4	2.06 %	180 MB	30 seconds ago	8.17.1	...
Healthy	Hameet-Fleet-Server	Fleet Server Policy rev. 2	0.54 %	209 MB	13 seconds ago	8.17.1	...

Rows per page: 20

Ref 2: Customize fleet policy to ingest logs from Windows Defender and Sysmon from both Linux machine and Windows Server.



Ref 3: Confirm telemetry is being ingested from both machines by checking logs and seeing both agent names.

Documents (184)Field statistics

	@timestamp	system.auth.ssh.event	user.name	source.ip	source.geo.country_name
Jan 27, 2025 @ 19:43:25.197	Failed	sync	92.118.39.74	United States	
Jan 27, 2025 @ 19:42:38.491	Failed	ps	92.118.39.86	United States	
Jan 27, 2025 @ 19:36:11.409	Failed	snark	92.118.39.74	United States	
Jan 27, 2025 @ 19:38:41.582	Failed	v	92.118.39.86	United States	
Jan 27, 2025 @ 19:28:53.845	Failed	snarkOS	92.118.39.74	United States	
Jan 27, 2025 @ 19:21:35.566	Failed	grafana	92.118.39.74	United States	
Jan 27, 2025 @ 19:18:47.437	Failed	solx	92.118.39.86	United States	

*Ref 4: Apply appropriate filters and fields to filter results for brute force attack signs on Linux machine.*

## Create rule



### Elasticsearch query

Alert when matches are found during the latest query run. [Learn more](#)

Select a data view

**DATA VIEW** apm-\*transaction\*, auditbeat-\*, endgame-\*, filebeat-\*, logs-\*, packetbeat-\*, traces-apm\*, winlogbeat-\*, -\*elastic-cloud-logs-\*

### Define your query

+

Q

system.auth.ssh.event: "Failed"

×

agent.name: HAMEET-LINUX 

×

Set the group, threshold, and time window

**WHEN** count()

**OVER** all documents

**IS ABOVE** 5

**FOR THE LAST** 5 minutes

Set the number of documents to send

**SIZE** 100

☐ Exclude matches from previous runs

Add more fields to alert details

container.id 

×

 host.hostname 

×

 host.id 

×

 host.name 

×

×

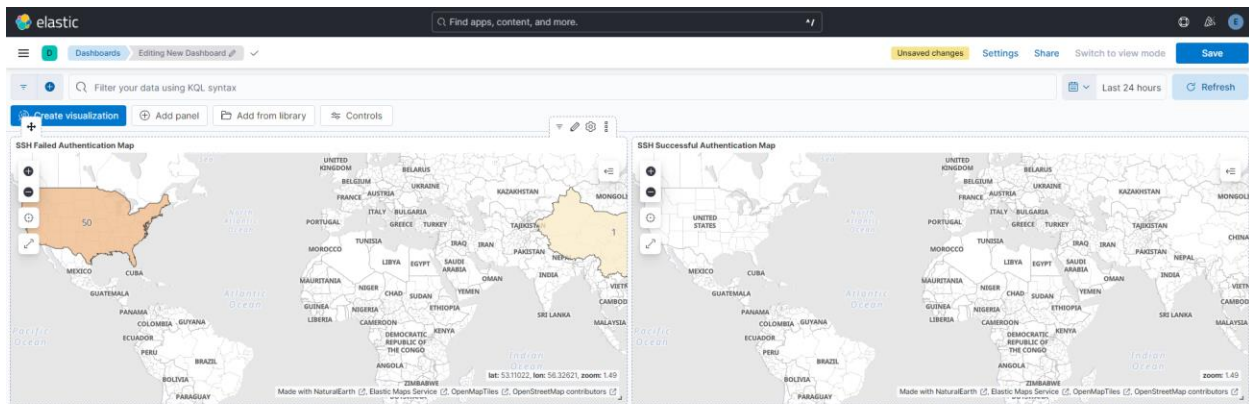
▼

Cancel

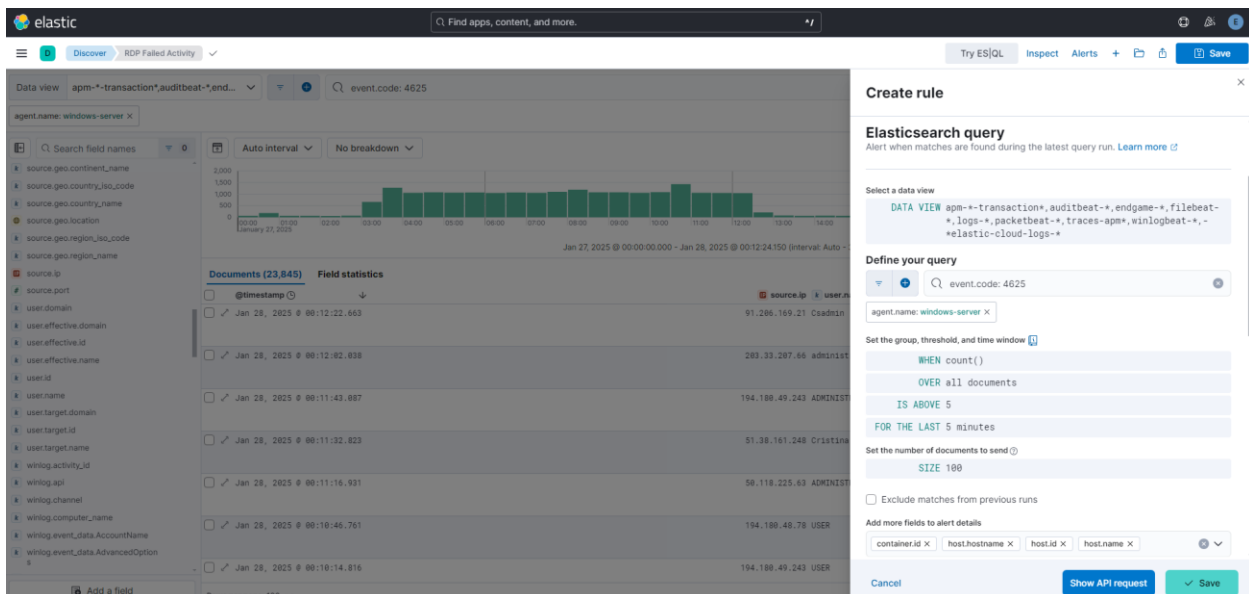
Show API request

✓ Save

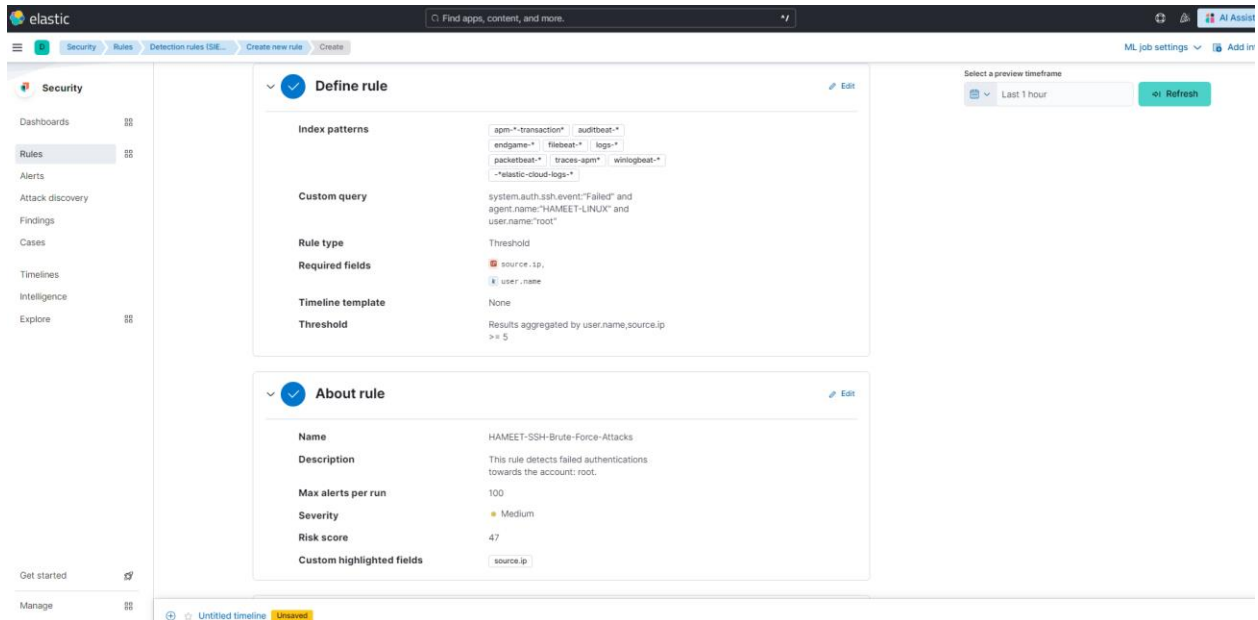
Ref 5: Create basic rule to send alert for brute force attack activity.



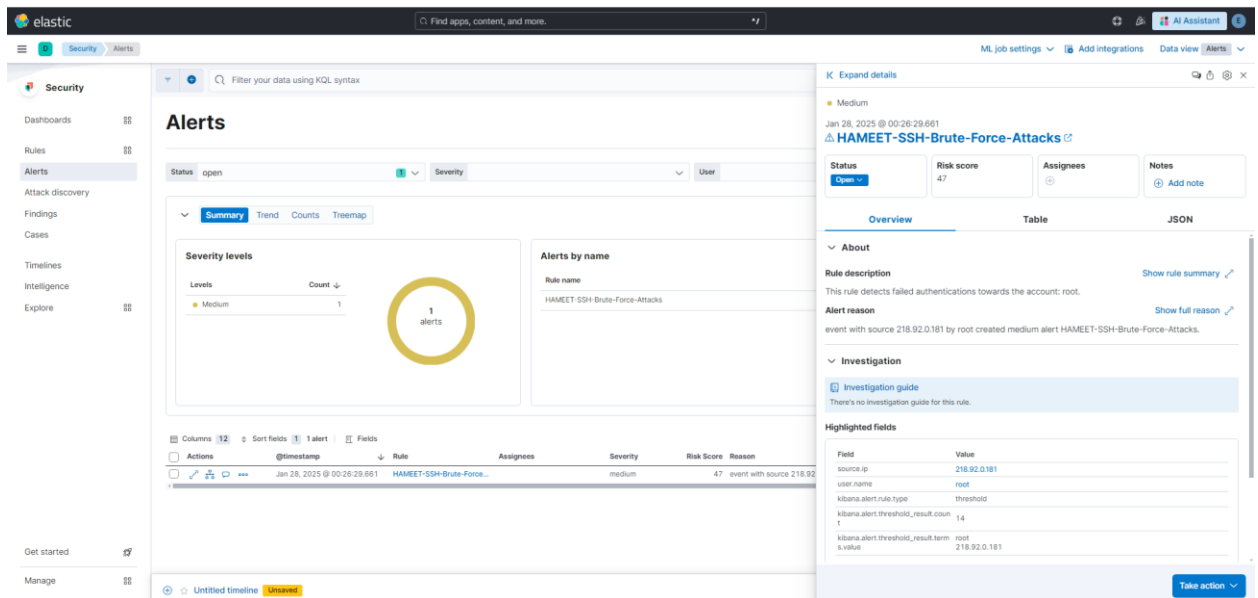
Ref 6: Create a simple dashboard to visualize the query above. Maps out attempts on a geolocation basis.



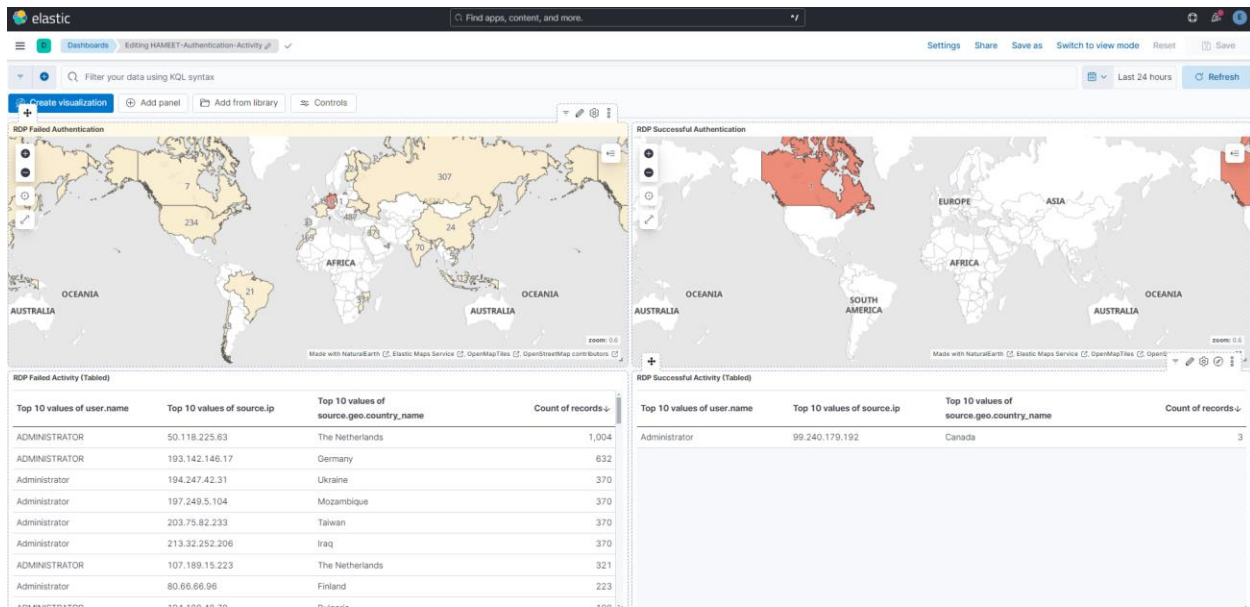
Ref 7: Create query and alert rule to filter for windows event code: 4625. This indicates a failed logon attempt. The alert and query is for RDP brute force attack warning.



Ref 8: That above rules were simple and provided little information. Creating a detection rule in elastic will allow us to provide more information with incoming alerts and are more customizable.



Ref 9: An example of an incoming alert for brute force attack.

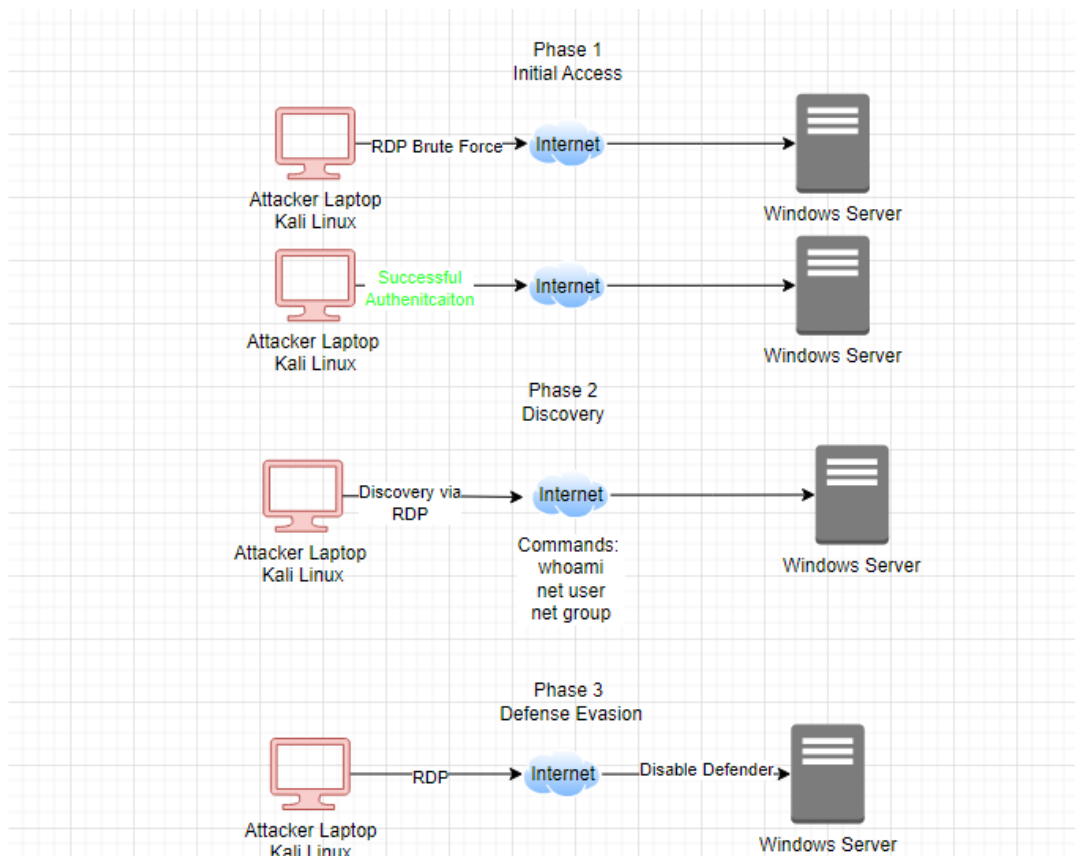


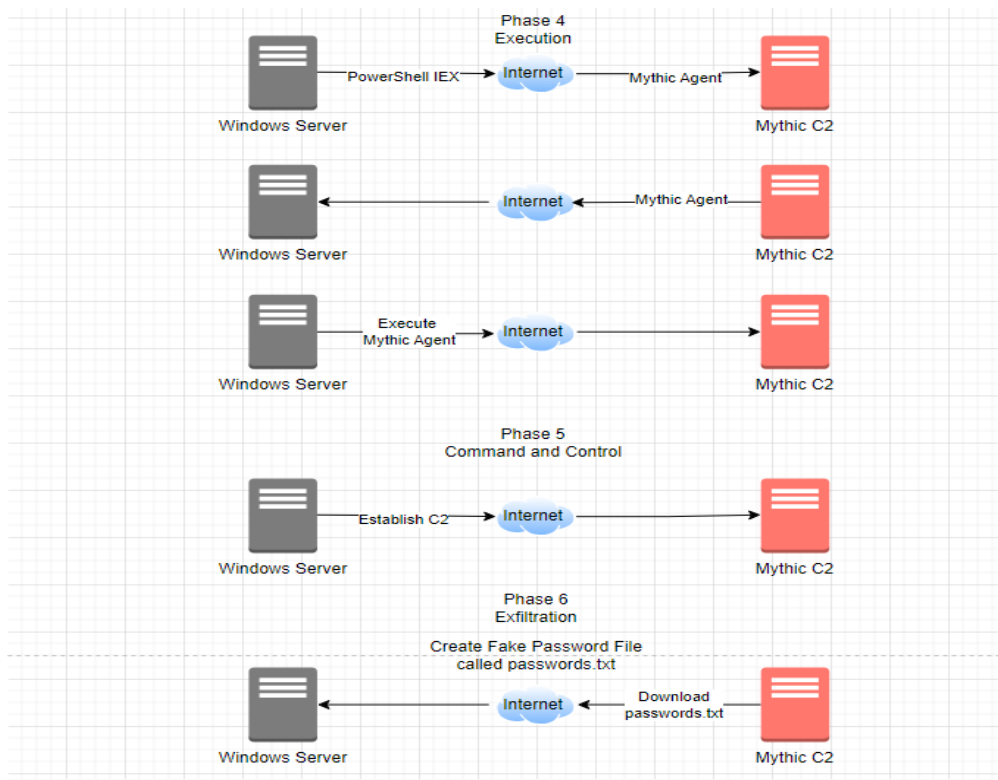
*Ref 10: Created a visualization of the attempts on Elastic SIEM, both unsuccessful and successful for RDP and SSH brute force attempts. There was both a visual map and a table outlining all activity.*

Now that our alerts and rules/policies are created. We are going to create an attack diagram to plan the command-and-control attack. Following that, the Mythic program was set up and loaded using an Ubuntu VM.

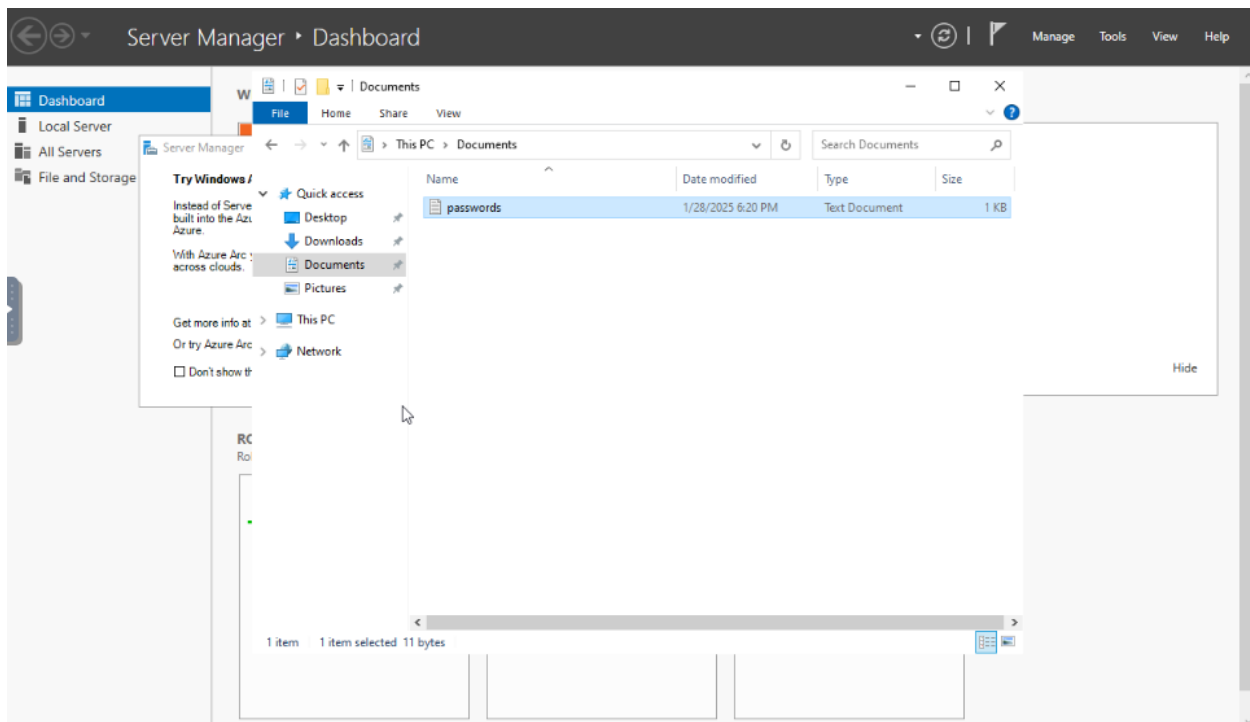


## Command-and-Control Attack Phase



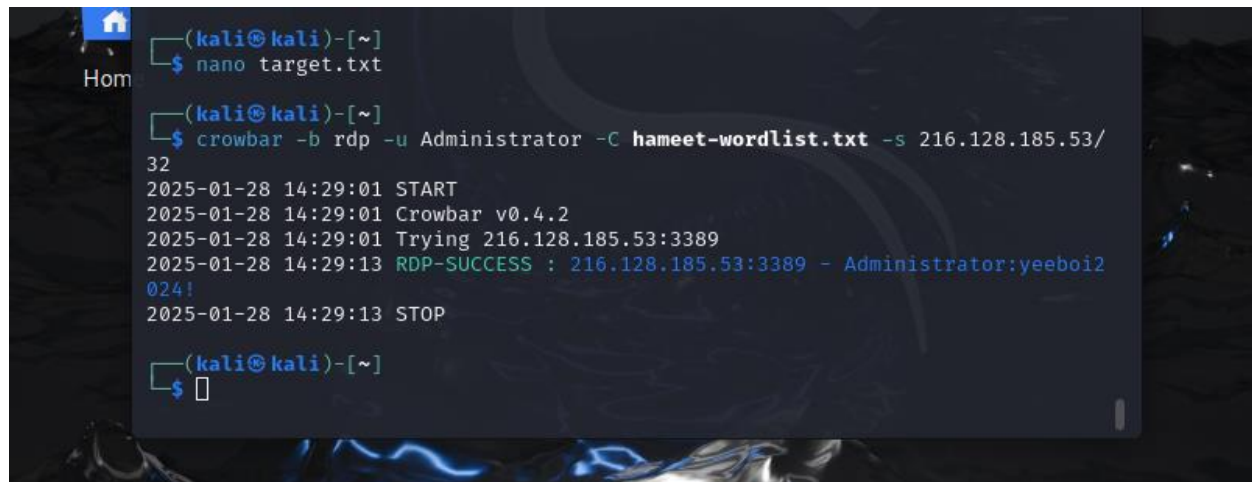


Ref 11: Attack diagram for Mythic command-and-control attack.



Ref 12: As we prepare to simulate the attack, a dummy text file called passwords was created. Additionally, the password for the windows logon for this machine was changed to the one in the text file.

A wordlist with our target password was created for use for the brute force attack on our Kali Linux system.

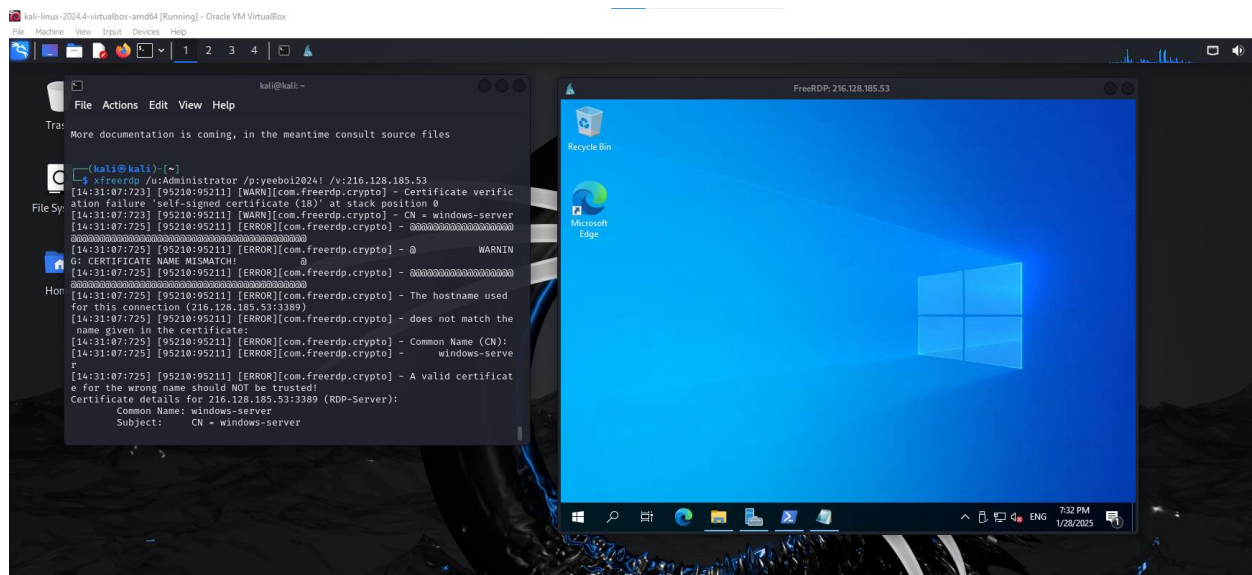


```
(kali㉿kali)-[~]
$ nano target.txt

(kali㉿kali)-[~]
$ crowbar -b rdp -u Administrator -C hameet-wordlist.txt -s 216.128.185.53/32
2025-01-28 14:29:01 START
2025-01-28 14:29:01 Crowbar v0.4.2
2025-01-28 14:29:01 Trying 216.128.185.53:3389
2025-01-28 14:29:13 RDP-SUCCESS : 216.128.185.53:3389 - Administrator:yeeboi2024!
2025-01-28 14:29:13 STOP

(kali㉿kali)-[~]
$
```

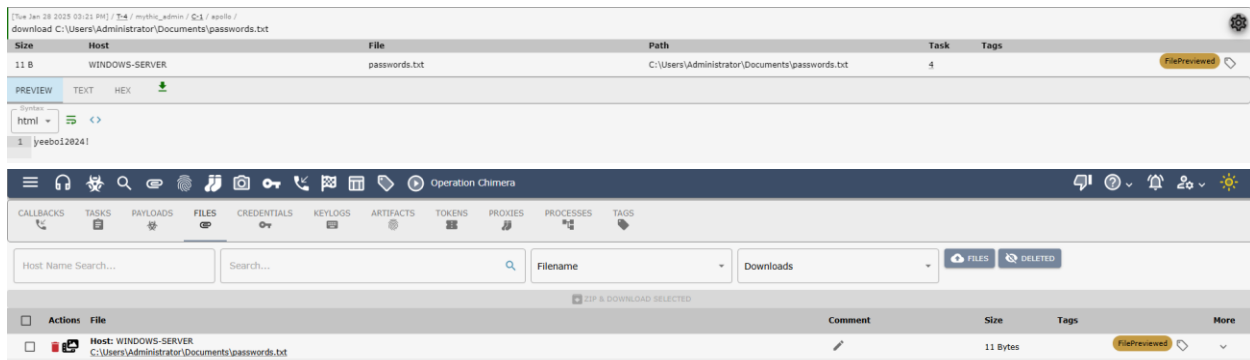
Ref 13: Using crowbar in Kali Linux, we execute a brute force attack on our windows server. Using the appropriate IP, username, and wordlist.



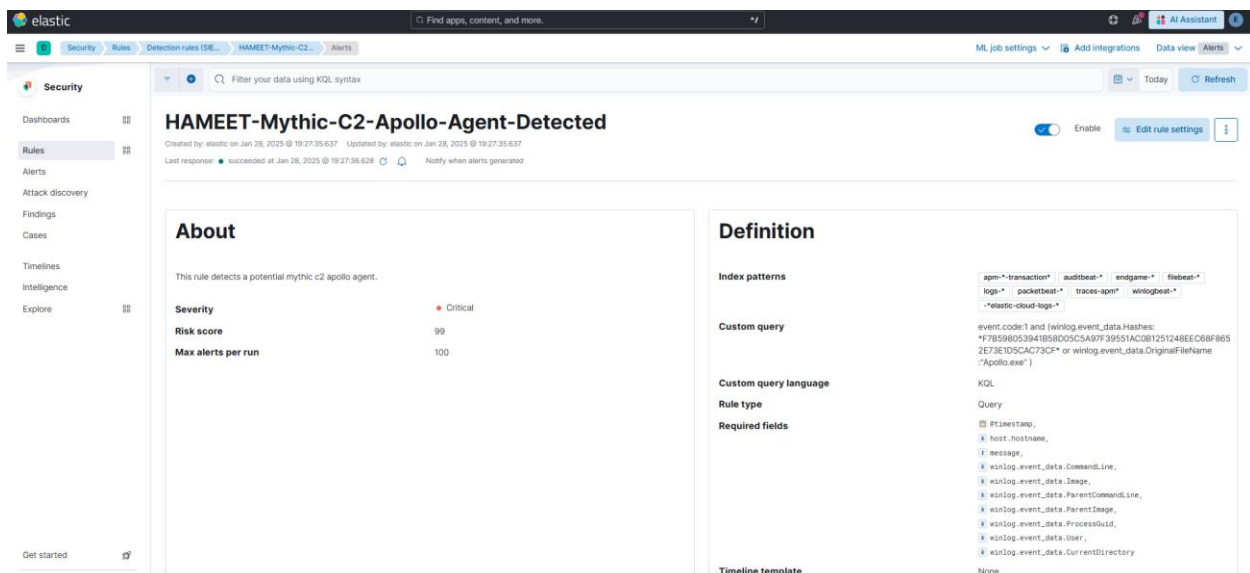
Ref 14: Now that we have the password from the brute force attack, we can establish an RDP connection using xfreerdp.

The next phase of the attack is now commenced. Using command prompt, we type in commands such as `ipconfig`, `whoami`, `net user/ group`, etc to gain knowledge and discovery. Then we proceed to disable windows defender to further compromise the machine.





Ref 18: The dummy text file was downloaded and that completes all phases of the attack. The text file shows up in the attachments tab and has our dummy password inside of it.



Ref 19: Create a rule in elastic in which if the SHA256 file hash matches Apollo.exe, or if original file name matches Apollo.exe then it will be detected. An alert with the above fields will be generated.

elastic Find apps, content, and more.

Dashboards Editing New Dashboard

Filter your data using KQL syntax

Create visualization Add panel Add from library Controls

Unsaved changes Settings Share Switch to view mode Last 7 days Refresh

Process Created (PowerShell, cmd, rundll32)

User	ParentImage	ParentCommandLine	Image	CommandLine	CurrentDirectory	Count of records
NT AUTHORITY\SYSTEM	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	"C:\Windows\System32\cmd.exe"	C:\Windows\System32	3
NT AUTHORITY\SYSTEM	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	"C:\Windows\System32\cmd.exe"	C:\Windows\System32	1
NT AUTHORITY\SYSTEM	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	"C:\Windows\System32\cmd.exe"	C:\Windows\System32	1
WINDOWS-SERVICE	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	C:\Windows\System32\cmd.exe	"C:\Windows\System32\cmd.exe"	C:\Windows\System32	1

Process Initiated Network Connections

Image	DestinationIp	SourceIp	DestinationPort	Count of records
C:\Windows\System32\svchost.exe	0.0.0.0/0.0.0.1	0.0.0.0/0.0.0.1	5985	6
C:\Windows\System32\svchost.exe	224.0.0.251	216.128.185.53	5353	3
C:\Windows\System32\svchost.exe	ff02:0:0:0:0:0:0:0	fe80:0:0:0:5400:5ff:fe42:8f9b	5353	3
C:\Users\Public\Downloads\svchost-name.exe	155.138.129.151	216.128.185.53	80	2
C:\Windows\System32\rundll32.exe	20.106.86.13	216.128.185.53	443	1
C:\Windows\System32\rundll32.exe	52.183.220.149	216.128.185.53	443	1
C:\Windows\System32\WindowsPowerShell\1.0\powershell.exe	155.138.129.151	216.128.185.53	9999	1

Microsoft Defender Disabled

hostname	Product Name	event_code	Count of records
----------	--------------	------------	------------------

Ref 20: Now we are going to create a dashboard that looks for suspicious activity, including event id 3: Network Connection (external) - any processes creating a network connection outbound. Event id 1- process creates – powershell,cmd,etc. Event id 5001- windows defender disabled.

## Investigation and Ticketing System

We will now create a ticketing system using OSTicket. This will simulate a SOC scenario and give me hands-on experience.

XAMPP Control Panel v3.3.0 [ Compiled: Apr 6th 2021 ]

XAMPP Control Panel v3.3.0

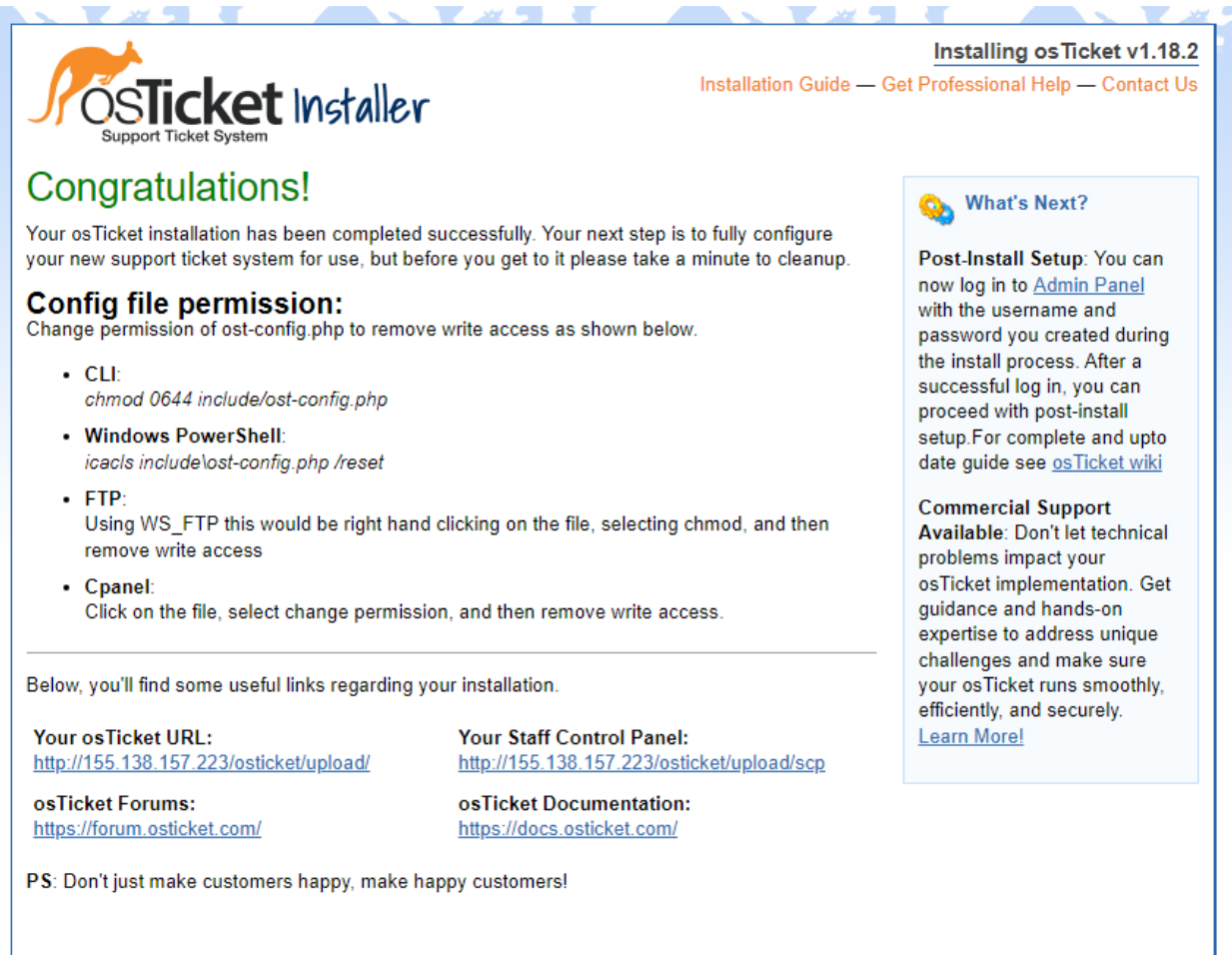
Modules

Service	Module	PID(s)	Port(s)	Actions
Apache	Apache	816 2132	80, 443	Stop Admin Config Logs
MySQL	MySQL	5284	3306	Stop Admin Config Logs
FileZilla	FileZilla			Start Admin Config Logs
Mercury	Mercury			Start Admin Config Logs
Tomcat	Tomcat			Start Admin Config Logs

Config Netstat Shell Explorer Services Help Quit

12:47:57 AM [Apache] Attempting to stop Apache (PID: 1496)  
 12:47:58 AM [Apache] Status change detected: stopped  
 12:47:59 AM [mysql] Attempting to stop MySQL app...  
 12:47:59 AM [mysql] Status change detected: stopped  
 12:48:00 AM [Apache] Attempting to start Apache app...  
 12:48:00 AM [Apache] Status change detected: running  
 12:48:01 AM [mysql] Attempting to start MySQL app...  
 12:48:01 AM [mysql] Status change detected: running

Ref 21: First a windows VM was spun up, and a XAMPP Apache server was installed and booted. Next OSTicket was installed on this VM.



**Installing osTicket v1.18.2**  
Installation Guide — Get Professional Help — Contact Us

## Congratulations!

Your osTicket installation has been completed successfully. Your next step is to fully configure your new support ticket system for use, but before you get to it please take a minute to cleanup.

### Config file permission:

Change permission of ost-config.php to remove write access as shown below.

- **CLI:**  
`chmod 0644 include/ost-config.php`
- **Windows PowerShell:**  
`icacls include\ost-config.php /reset`
- **FTP:**  
Using WS\_FTP this would be right hand clicking on the file, selecting chmod, and then remove write access
- **Cpanel:**  
Click on the file, select change permission, and then remove write access.

Below, you'll find some useful links regarding your installation.

<b>Your osTicket URL:</b> <a href="http://155.138.157.223/osticket/upload/">http://155.138.157.223/osticket/upload/</a>	<b>Your Staff Control Panel:</b> <a href="http://155.138.157.223/osticket/upload/scp">http://155.138.157.223/osticket/upload/scp</a>
<b>osTicket Forums:</b> <a href="https://forum.osticket.com/">https://forum.osticket.com/</a>	<b>osTicket Documentation:</b> <a href="https://docs.osticket.com/">https://docs.osticket.com/</a>

PS: Don't just make customers happy, make happy customers!


**What's Next?**  
**Post-Install Setup:** You can now log in to [Admin Panel](#) with the username and password you created during the install process. After a successful log in, you can proceed with post-install setup. For complete and upto date guide see [osTicket wiki](#)

**Commercial Support Available:** Don't let technical problems impact your osTicket implementation. Get guidance and hands-on expertise to address unique challenges and make sure your osTicket runs smoothly, efficiently, and securely. [Learn More!](#)

Copyright © 2025 osTicket.com

Ref 22: Our SQL database was created and OSTicket was successfully installed and configured.

Now the next step was to use an Elastic connector/webhook to connect our OSTicket to our ELK stack.

**Webhook connector**  
Send a request to a web service.

Compatibility: Alerting Rules Security Solution

Connector name

OSTicket

Connector settings

Method

POST

URL

http://155.138.157.223/osticket/upload/api/tickets.xml

Authentication

☒ None

☐ Basic authentication

☐ SSL authentication

☒ Add HTTP header

Headers in use

Key	Value
X-API-Key	1F833A6B583ADA04022685FB38D30B75

+ Add

☐ Add certificate authority

Save & test

Save

Back

Ref 23: Create webhook connector from Elastic to OSTicket.



**Edit connector**
×

[Configuration](#)
[Rules](#)
[Test](#)

1

Create an action

Body

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <ticket alert="true" autorespond="true" source="API">
3   <name>Angry User</name>
4   <email>api@osticket.com</email>
5   <subject>Testing API</subject>
6   <phone>318-555-8634X123</phone>
7   <message type="text/plain"><![CDATA[Message content here]]></message>
8   <attachments>
9     <file name="file.txt" type="text/plain"><![CDATA[
10       File content is here and is automatically trimmed

```

2

Run the test

▶ Run

3

Results

When you run the test, the results will show up here.

Welcome, **hameet**. | [Admin Panel](#) | [Profile](#) | [Log Out](#)

[Dashboard](#)
[Users](#)
[Tasks](#)
[Tickets](#)
[Knowledgebase](#)

[Open](#)
[My Tickets](#)
[Closed](#)
[Search](#)
[New Ticket](#)

[Open](#)

	Ticket	Last Updated	Subject	From	Priority	Assigned To
<input type="checkbox"/>	<a href="#">299422</a>	02/20/25 19:04:52	osTicket Installed!	osTicket Team		
<input type="checkbox"/>	<a href="#">190920</a>	02/20/25 19:12:28	Testing API	Angry User	Normal	

Select: [All](#) [None](#) [Toggle](#)

Page: **[1]** [Export](#)

Showing 1 - 2 of about 2

Ref 24: Once connected through the API key from osTicket, it was tested successfully. This was indicated by the agent panel in osTicket. Next, we are going to use our previously set-up alerts in Elastic (Brute force attacks) and use them to create tickets for the alerts.

## Actions

Choose when to perform actions or snooze them. Notifications are not created for snoozed actions. [Learn more](#).



Notify when alerts generated

osTicket

Webhook connector

osTicket

Add connector

Action frequency

For each alert

Per rule run

☐ If alert matches a query

☐ If alert is generated during timeframe

Body

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ticket alert="true" autorespond="true" source="API">
3   <name>Elastic</name>
4   <email>api@osticket.com</email>
5   <subject>{{rule.name}}</subject>
6   <phone>318-555-8634X123</phone>
7   <message type="text/plain"><![CDATA[Please investigate the rule: {{rule.name}}]]></message>
8 </ticket>
```

Ref 25: Create action to create ticket for alerts on Elastic.

The screenshot shows a web browser window with the URL `155.138.157.223/osticket/upload/scp/tickets.php?id...`. The page is the oSTicket interface, which has a navigation bar with links for Dashboard, Users, Tasks, Tickets (active), and Knowledgebase. Below the navigation bar are buttons for Open, My Tickets, Closed, Search, and New Ticket. The main content area displays 'Ticket #937239' with the title 'HAMEET-SSH-Brute-Force-Attacks'. The ticket details include: Status: Open, Priority: Normal, Department: Support, Create Date: 02/24/25 19:54:31, User: Angry User (27) (Manage Collaborators), Email: api@osticket.com, Source: API, Assigned To: — Unassigned —, SLA Plan: Default SLA, Due Date: 02/27/25 08:00:00, Help Topic: None, Last Message: 02/24/25 19:54:31, and Last Response: (empty). Below the details is a 'Ticket Thread (1)' section showing a post from 'Angry User' at 02/24/25 19:54:31. The post content is: 'Please investigate the rule: HAMEET-SSH-Brute-Force-Attacks' followed by a long URL: `http://137.220.54.158:5601/app/security/detections/rules/id/31d94ce9-5cc8-4fb8-aa68-2130dd2b1e1d?timerange=(global:(linkTo:(timeline),timerange:(from:1740426569024,kind:absolute,to:1740426889024)),timeline:(linkTo:(global),timerange:(from:1740426569024,kind:absolute,to:1740426889024)))`. Below the post is a 'Created by' field showing 'Angry User' at 02/24/25 19:54:31. At the bottom is a 'Post Reply' form with a 'From' dropdown set to 'Support <test@test.com>' and 'Recipients' set to 'Angry User' <api@osticket.com> and Collaborators.

Ref 26: As can be seen, the alerts in Kibana now open a ticket as well. This includes the rule name from Kibana as well as a link to investigate the incident further. The ticket can be assigned to someone, and comments can be made under the ticket. In a real-world scenario, once the proper investigation was done, the ticket can also be resolved and closed.

This step was repeated for the RDP brute force attack and the Mythic agent rules. After this, taking a slightly deeper dive into the investigation of the brute force attack we can check for the reputation of an IP address using intelligence websites like abuseipdb.com and greynoise.com.

GREYNOISE

III K

[TRENDS](#)
[TODAY](#)
[TAGS](#)
[ANALYSIS](#)
[ALERTS](#)

[LOG IN](#)
[SIGN UP](#)

MALICIOUS
HOSTING

92.51.2.47

ORGANIZATION

Flyservers S.A.

ACTOR

unknown

Not Spoofable

Observed Activity

Shows the ports & protocols that this IP scanned, along with fingerprints of the SSH & TLS negotiation between this IP and the target.

SUMMARY

TIMELINE

PROTOCOL

TCP

WEB PATH

/fawcon.io

1000ms

sharding.cgi

JAVA FINGERPRINT

faw33b0d7633ad62b64ad7b0bc3488

WEB PATH

View Similar IPs

FIRST SEEN

2025-02-10

LAST SEEN

2025-02-24

COUNTRY

Russia

REGION

Moscow

CITY

Moscow

ASN

AS209588

Tags

RDP Crawler
RDP BruteForce Attempt

EXPAND DETAILS

GREYNOISE IP CLASSIFICATION

This is a Malicious IP

GreyNoise has identified malicious activity from this IP. It is still considered internet background noise since it is scanning the entire internet and NOT targeting you specifically.

For more details on our classifications, click [here](#)

Next

Create a free account or log in to view activity

[LOGIN](#)
[SIGN UP](#)

[Home](#)
[Report IP](#)
[Bulk Reporter](#)
[Pricing](#)
[About](#)
[FAQ](#)
[Documentation](#)
[Statistics](#)
[IP Tools](#)
[Contact](#)

## AbuseIPDB » 92.51.2.47

Check an IP Address, Domain Name, or Subnet  
e.g. 2607:fe88:5292:6100:744a:ca7e:7e0f:4385, microsoft.com, or 5.188.10.0/24

[CHECK](#)

92.51.2.47 was found in our database!

This IP was reported **37** times. Confidence of Abuse is **100%**:

100%

ISP

Alviva Holding Limited

Usage Type

Data Center/Web Hosting/Transit

ASN

AS209588

Domain Name

digi-cloud.net

Country

Russian Federation

City

Moscow, Moscow

IP Info including ISP, Usage Type, and Location provided by [IPInfo](#). Updated biweekly.

[REPORT 92.51.2.47](#)
[WHOIS 92.51.2.47](#)

[feedback](#)

Ref 27: Searching the source IP address of the latest RDP brute force attack shows that this is definitely a malicious IP and is a threat.