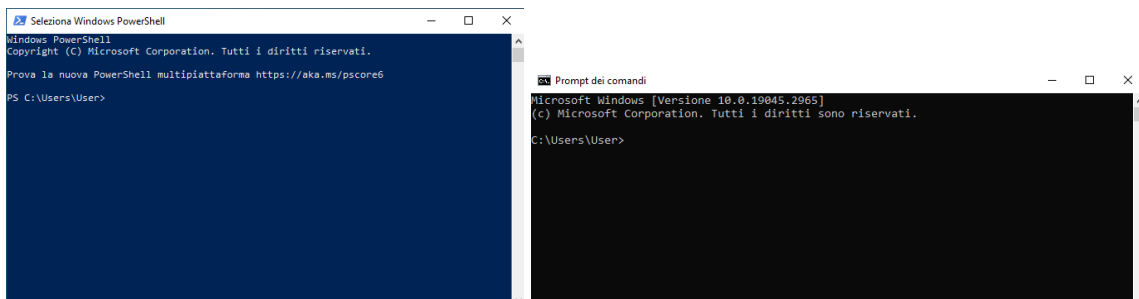


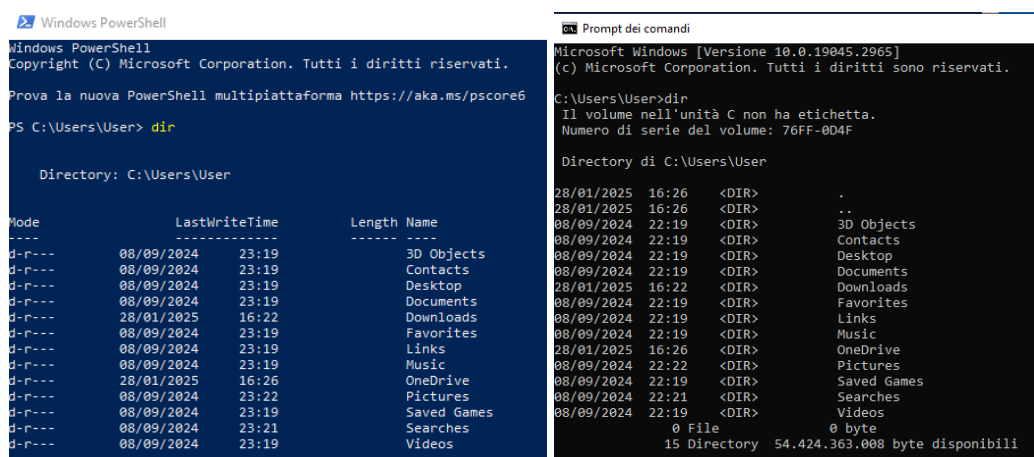
## Progetto S11-L5

### • Parte 1 - Windows Powershell

Avvio macchina virtuale Windows 10 pro ed avvio di **Windows Powershell** e **Prompt dei comandi**

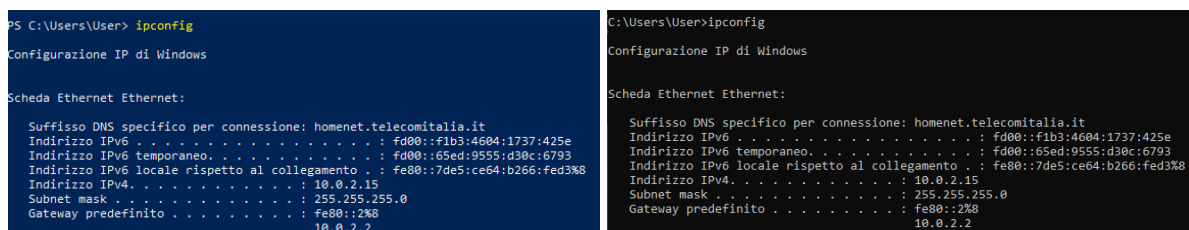


Inserimento comando **dir** in entrambe le finestre per individuare eventuali differenze



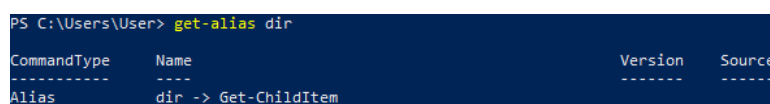
Come possiamo notare dalle immagini soprastanti, l'unica vera differenza degna di nota è la categoria **"Mode"** presente in Windows Powershell e non presente invece nel prompt dei comandi. Essa fornisce informazioni sugli attributi dei file: ad esempio la **d** indica che si tratta di una directory, mentre la **r** che si tratta di un file di sola lettura.

Facendo un tentativo con altri comandi, come ad esempio **ipconfig**, i risultati sono invece simili.



Esistono una grande quantità di comandi in Powershell chiamati **cmdlets**.

Ad esempio, inserendo il comando **Get-Alias** in powershell, è possibile visualizzare gli alias disponibili per quel comando e i comandi a cui esso è associato.



Facendo un tentativo con il comando **dir**, possiamo vedere che esso è un alias del comando **Get-ChildItem**.

Un altro comando molto importante su Powershell è **netstat**. E' possibile inserire il comando **netstat -h** per visualizzare tutte le opzioni disponibili per tale comando.

```
PS C:\Users\User> 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
```

Inserendo il comando **netstat -r** è possibile visualizzare la tabella di route con i "percorsi" attivi.

```
PS C:\Users\User> netstat -r

=====
Elenco interfacce
8...08 00 27 96 c2 10 .....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    10.0.2.2     10.0.2.15    25
   10.0.2.0            255.255.255.0 On-link     10.0.2.15    281
   10.0.2.15           255.255.255.255 On-link     10.0.2.15    281
   10.0.2.255          255.255.255.255 On-link     10.0.2.15    281
   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
   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     10.0.2.15    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     10.0.2.15    281
=====
Route permanenti:
Nessuna

IPv6 Tabella route
=====
Route attive:
   Interf Metrica Rete Destinazione Gateway
   -----
   8      281 ::/0      fe80::2
   1      331 ::1/128    On-link
   8      281 fd00::/64   On-link
```

Grazie a questo comando, tra le altre cose, è possibile risalire all'indirizzo IP del gateway impostato, che in questo caso è 10.0.2.2.

Apriamo ora una nuova finestra di Windows Powershell con permessi di Amministratore ed inseriamo il comando **netstat -abno**.

```
PS C:\Windows\system32> netstat -abno

Connessioni attive

Proto  Indirizzo locale      Indirizzo esterno      Stato      PID
-----
TCP    0.0.0.0:135            0.0.0.0:0              LISTENING  888
RpcSs
[svchost.exe]
TCP    0.0.0.0:445            0.0.0.0:0              LISTENING  4
Impossibile ottenere informazioni sulla proprietà
TCP    0.0.0.0:5040           0.0.0.0:0              LISTENING  1056
CDPSvc
[svchost.exe]
TCP    0.0.0.0:7680           0.0.0.0:0              LISTENING  3016
Impossibile ottenere informazioni sulla proprietà
TCP    0.0.0.0:49664          0.0.0.0:0              LISTENING  664
[lsass.exe]
TCP    0.0.0.0:49665          0.0.0.0:0              LISTENING  512
Impossibile ottenere informazioni sulla proprietà
TCP    0.0.0.0:49666          0.0.0.0:0              LISTENING  504
EventLog
[svchost.exe]
TCP    0.0.0.0:49667          0.0.0.0:0              LISTENING  1256
Schedule
[svchost.exe]
TCP    0.0.0.0:49668          0.0.0.0:0              LISTENING  1892
[spoolsv.exe]
```

Questo comando ci permette di visualizzare le connessioni TCP attive. Ogni processo è identificato da un valore **PID**. Prendiamo come esempio il processo con PID 504.

```
TCP [::]:49666 [::]:0 LISTENING 504
EventLog
[svchost.exe]
```

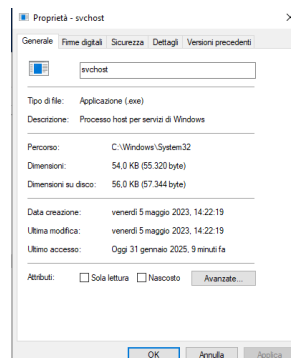
Ora apriamo il Task Manager (Gestione attività) ed individuiamo tale processo.

Nome	PID	Stato	Nome ute...	CPU	Memoria (...)	Virtualizzazion...
Interrupt sistema	-	In esecuzione	SYSTEM	01	0 K	
Processo di inattività...	0	In esecuzione	SYSTEM	29	8 K	
System	4	In esecuzione	SYSTEM	04	20 K	
Registry	92	In esecuzione	SYSTEM	00	12.004 K	Non consentito
msedge.exe	264	In esecuzione	User	00	1.200 K	Disabilitato
conhost.exe	328	In esecuzione	User	00	556 K	Disabilitato
smss.exe	344	In esecuzione	SYSTEM	00	0 K	Non consentito
svchost.exe	376	In esecuzione	SYSTEM	02	68.164 K	Non consentito
csrss.exe	436	In esecuzione	SYSTEM	00	508 K	Non consentito
svchost.exe	504	In esecuzione	SERVIZIO L...	00	8.080 K	Non consentito
wininit.exe	512	In esecuzione	SYSTEM	00	0 K	Non consentito
csrss.exe	524	In esecuzione	SYSTEM	00	612 K	Non consentito
svchost.exe	528	In esecuzione	SERVIZIO L...	00	3.128 K	Non consentito
winlogon.exe	604	In esecuzione	SYSTEM	00	556 K	Non consentito
services.exe	640	In esecuzione	SYSTEM	00	1.952 K	Non consentito
lsass.exe	664	In esecuzione	SYSTEM	01	3.628 K	Non consentito

Nome	PID	Stato	Nome utente	CPU	Memoria	Virtualizzazione
svchost.exe	504	In esecuzione	SERVIZIO LOCALE	00	8.072 K	Non consentito

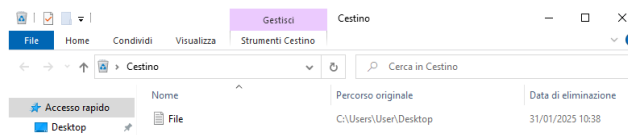
Da qui è possibile aprire la sezione Proprietà per tale processo e leggerne i dettagli.



Grazie a ciò, siamo in grado di risalire ad alcune informazioni:

1. Nome del processo associato al PID: in questo caso "svchost.exe"
2. L'utente che sta eseguendo questo processo: in questo caso "SERVIZIO LOCALE"
3. La quantità di memoria utilizzata: in questo caso 8072 K.

Creiamo ora un file di testo chiamato file.txt ed eliminiamolo. In questo modo sarà possibile visualizzare tale file nel Cestino.

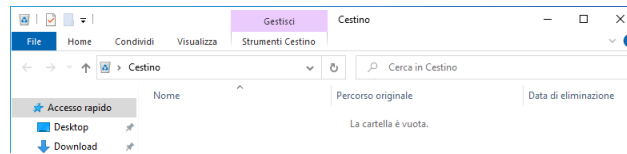


E' possibile "svuotare" il cestino tramite Powershell grazie al comando **clear-recyclebin**.

```
PS C:\Windows\system32> clear-recyclebin

Conferma
Eseguire l'operazione?
Esecuzione dell'operazione "Clear-RecycleBin" sulla destinazione "Tutto il contenuto del Cestino".
[S] Sì [T] Sì a tutti [N] No [U] No a tutti [O] Sospendi [?] Guida (il valore predefinito è "S"): sì
[S] Sì [T] Sì a tutti [N] No [U] No a tutti [O] Sospendi [?] Guida (il valore predefinito è "S"): t
PS C:\Windows\system32>
```

Tornando nel cestino, possiamo vedere che adesso risulterà vuoto.



Ecco ora una rapida panoramica su alcuni comandi in Powershell utili per un security analyst:

1. **Get-LocalUser**: per elencare tutti gli utenti locali.
2. **Get-LocalGroup**: per elencare tutti i gruppi locali.
3. **Get-Process**: per monitorare i processi in esecuzione sul sistema. E' possibile utilizzare anche il comando **Get-Process -Name "nome-processo"** per ottenere informazioni dettagliate su un processo specifico.
4. **Get-EventLog**: per controllare i log degli eventi ed identificare attività sospette. Ad esempio, il comando **Get-EventLog -LogName Security -Newest 10** mostrerà gli ultimi 10 eventi nel log di sicurezza di Windows.
5. **Get-MpPreference**: per monitorare la configurazione di Windows Defender.
6. **Get-Service**: per ottenere informazioni sui vari servizi. ad esempio, il comando **Get-Service -Name "wuauserv", "MpsSvc", "WinDefend"** mostrerà lo stato di Windows Update (wuauserv), Windows Firewall (MpsSvc), e Windows Defender (WinDefend).
7. **Get-LocalGroupPolicy**: per ottenere informazioni sulla configurazione delle policy di sicurezza di un sistema.
8. **Get-NetFirewallRule**: per monitorare e configurare il firewall di Windows.
9. **Clear-DnsClientCache**: per svuotare la cache DNS. Utile se si sospettano attacchi DNS spoofing.

## ● Parte 2 - Analisi traffico HTTP e HTTPS

Scarichiamo, importiamo ed avviamo la CyberOps WS.



Apriamo il terminale ed inseriamo il comando **ip address** per ottenere informazioni sulla scheda di rete.

```
[analyst@secOps ~]$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:f7:f4:9b brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 86293sec preferred_lft 86293sec
    inet6 fd00::a00:27ff:fe7f:f49b/64 scope global dynamic mngtaddr noprefixroute
        valid_lft 86295sec preferred_lft 14295sec
    inet6 fe80::a00:27ff:fe7f:f49b/64 scope link
        valid_lft forever preferred_lft forever
```

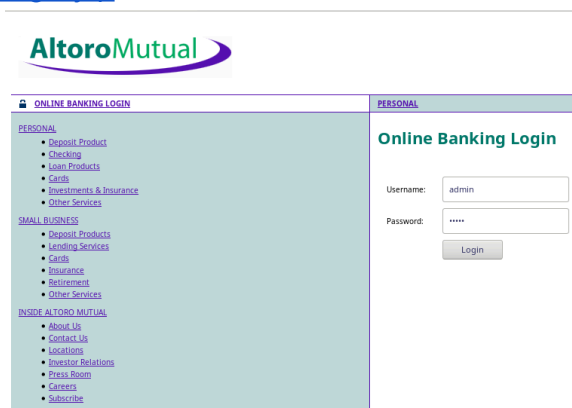
Dall'immagine soprastante possiamo vedere che alla scheda di rete **enp0s3** è associato l'indirizzo IP 10.0.2.15 con netmask 255.255.255.0 (come indica il **/24**), mentre l'indirizzo di loopback è 127.0.0.1/8.

Avviamo ora **tcpdump**, per la cattura dei pacchetti di rete, utilizzando il seguente comando:

```
[analyst@secOps ~]$ sudo tcpdump -i enp0s3 -s 0 -w httpdump.pcap
```

1. **sudo**: permette di eseguire il comando con permessi da superuser.
2. **-i**: permette di selezionare l'interfaccia di rete.
3. **-s**: permette di definire la lunghezza di cattura di ogni pacchetto. Quando impostato su 0 si setta in automatico sul valore di default.
4. **-w**: permette di salvare la cattura in un file, in questo caso chiamato "httpdump.pcap".

Una volta avviato tcpdump, apriamo il browser e colleghiamoci al sito "[www.altoromutual.com/login.jsp](http://www.altoromutual.com/login.jsp)".

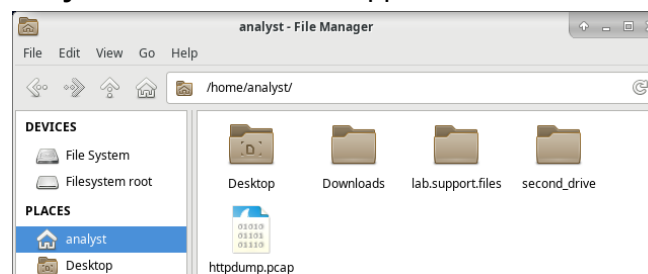


Tale sito utilizza il protocollo **HTTP**.

Una volta collegati, effettuiamo il login e successivamente terminiamo la cattura tramite tcpdump.

```
[analyst@secOps ~]$ sudo tcpdump -i enp0s3 -s 0 -w httpdump.pcap
[sudo] password for analyst:
tcpdump: listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
^C2686 packets captured
2686 packets received by filter
0 packets dropped by kernel
```

Nel percorso **/home/analyst** individuiamo il file appena creato.



Apriamo il file con **Wireshark** ed applichiamo un filtro di ricerca per "http".

No.	Time	Source	Destination	Protocol	Length	Info
11	0.063856	10.0.2.15	34.107.221.82	HTTP	342	GET /success.txt HTTP/1.1
13	0.088476	34.107.221.82	10.0.2.15	HTTP	270	HTTP/1.1 200 OK (text/plain)
57	1.754045	10.0.2.15	95.100.171.40	OCSP	485	Request
61	1.757996	10.0.2.15	95.100.171.40	OCSP	485	Request
66	1.785286	95.100.171.40	10.0.2.15	OCSP	944	Response
68	1.791032	95.100.171.40	10.0.2.15	OCSP	944	Response
104	1.924600	10.0.2.15	95.100.171.40	OCSP	485	Request
110	1.959143	95.100.171.40	10.0.2.15	OCSP	943	Response
209	3.044039	10.0.2.15	95.100.171.40	OCSP	485	Request
211	3.044463	10.0.2.15	95.100.171.40	OCSP	485	Request
218	3.077164	95.100.171.40	10.0.2.15	OCSP	943	Response
220	3.077254	95.100.171.40	10.0.2.15	OCSP	943	Response
224	3.082430	10.0.2.15	95.100.171.40	OCSP	485	Request
248	3.114370	95.100.171.40	10.0.2.15	OCSP	943	Response
290	3.234560	10.0.2.15	216.58.204.227	OCSP	482	Request
294	3.236297	10.0.2.15	216.58.204.227	OCSP	482	Request
326	3.366448	216.58.204.227	10.0.2.15	OCSP	756	Response
328	3.368572	216.58.204.227	10.0.2.15	OCSP	756	Response

Frame 11: 342 bytes on wire (2736 bits), 342 bytes captured (2736 bits)  
 Ethernet II, Src: PcsCompu\_f7:f4:9b (08:00:27:f7:f4:9b), Dst: 52:55:0a:00:02:02 (52:55:0a:00:02:02)  
 Internet Protocol Version 4, Src: 10.0.2.15, Dst: 34.107.221.82  
 Transmission Control Protocol, Src Port: 59704, Dst Port: 80, Seq: 1, Len: 288  
 Hypertext Transfer Protocol

Scorriamo la lista di operazioni fino ad individuare una richiesta **POST**.

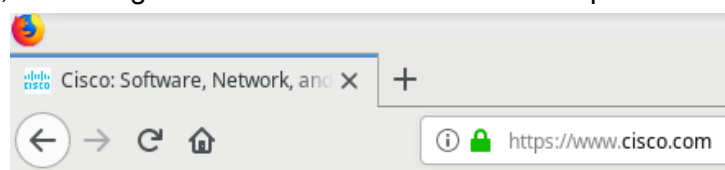
2491	85.216541	10.0.2.15	65.61.137.117	HTTP	589	POST /doLogin HTTP/1.1 (application/x-www-form-urlencoded)
------	-----------	-----------	---------------	------	-----	--

Cliccando su tale riga, nella tendina in basso ci vengono rilevate delle informazioni su tale operazione tra cui anche il nome utente e la password inseriti.

Frame 2491: 589 bytes on wire (4712 bits), 589 bytes captured (4712 bits)
Ethernet II, Src: PcsCompu_f7:f4:9b (08:00:27:f7:f4:9b), Dst: 52:55:0a:00:02:02 (52:55:0a:00:02:02)
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 65.61.137.117
Transmission Control Protocol, Src Port: 35782, Dst Port: 80, Seq: 1, Ack: 1, Len: 535
Hypertext Transfer Protocol
HTML Form URL Encoded: application/x-www-form-urlencoded
Form item: "uid" = "admin"
Form item: "passw" = "admin"
Form item: "btnSubmit" = "Login"

Passiamo ora ad analizzare il traffico relativo al protocollo **HTTPS**.

Per fare ciò, avviamo come prima una sessione di **tcpdump**, salvando la cattura nel file "httpsdump.pcap", e ci colleghiamo a un sito web che utilizza il protocollo HTTPS.



In questo caso il sito è "[www.cisco.com](https://www.cisco.com)" e la presenza del lucchetto accanto al nome indica l'utilizzo di un protocollo sicuro (HTTPS).

Come prima, effettuiamo il login sul sito e terminiamo la nostra cattura.

```
[analyst@sec0ps ~]$ sudo tcpdump -i enp0s3 -s 0 -w httpsdump.pcap
[sudo] password for analyst:
tcpdump: listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes
^C3018 packets captured
3018 packets received by filter
0 packets dropped by kernel
```

Apriamo ora la nostra nuova cattura con Wireshark e filtriamo i risultati per porta con il comando **tcp.port==443**. Successivamente individuiamo la riga relativa ad **Application Data**.

No.	Time	Source	Destination	Protocol	Length	Info
21	1.123951	10.0.2.15	34.120.5.221	TCP	74	46488 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=304239
22	1.150000	34.120.5.221	10.0.2.15	TCP	60	443 → 46488 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
23	1.150026	10.0.2.15	34.120.5.221	TCP	54	46488 → 443 [ACK] Seq=1 Ack=1 Win=29200 Len=0
24	1.166778	10.0.2.15	34.120.5.221	TLSv1.2	256	Client Hello
25	1.167067	34.120.5.221	10.0.2.15	TCP	60	443 → 46488 [ACK] Seq=1 Ack=203 Win=65535 Len=0
26	1.194602	34.120.5.221	10.0.2.15	TLSv1.2	1466	Server Hello
27	1.194619	10.0.2.15	34.120.5.221	TCP	54	46488 → 443 [ACK] Seq=203 Ack=1413 Win=31064 Len=0
28	1.196615	34.120.5.221	10.0.2.15	TLSv1.2	1627	Certificate, Server Key Exchange, Server Hello Done
29	1.196631	10.0.2.15	34.120.5.221	TCP	54	46488 → 443 [ACK] Seq=203 Ack=2986 Win=34560 Len=0
30	1.201124	10.0.2.15	34.120.5.221	TLSv1.2	147	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
31	1.201604	34.120.5.221	10.0.2.15	TCP	60	443 → 46488 [ACK] Seq=2986 Ack=296 Win=65535 Len=0

No.	Time	Source	Destination	Protocol	Length	Info
33	1.228614	34.120.5.221	10.0.2.15	TLSv1.2	123	Application Data

In questo caso possiamo notare che nella sezione protocollo troviamo la sigla **TLS1.2** invece di HTTP.

Inoltre, esaminando le informazioni relative a tale riga dal menù sottostante, possiamo notare come i dati non siano più visibili in quanto criptati.

▶ Frame 33: 123 bytes on wire (984 bits), 123 bytes captured (984 bits)
▶ Ethernet II, Src: 52:55:0a:00:02:02 (52:55:0a:00:02:02), Dst: PcsCompu_f7:f4:9b (08:00:27:f7:f4:9b)
▶ Internet Protocol Version 4, Src: 34.120.5.221, Dst: 10.0.2.15
▶ Transmission Control Protocol, Src Port: 443, Dst Port: 46488, Seq: 3296, Ack: 296, Len: 69
▼ Secure Sockets Layer
▼ TLSv1.2 Record Layer: Application Data Protocol: http2
Content Type: Application Data (23)
Version: TLS 1.2 (0x0303)
Length: 64
Encrypted Application Data: 00000000000000001907310a282c8519ac186c684517e5290...

## Vantaggi di utilizzare HTTPS invece di HTTP:

### 1. Crittografia:

HTTPS utilizza TLS (Transport Layer Security) per crittografare la comunicazione tra il browser e il server, impedendo a terzi di intercettare i dati trasmessi. Questo è fondamentale per proteggere informazioni sensibili come credenziali, dati bancari e dati personali.

### 2. Autenticazione:

HTTPS garantisce che il sito web con cui stai comunicando è autentico e non un sito fasullo o di phishing. Infatti i certificati SSL/TLS verificano l'identità del sito, riducendo il rischio di attacchi "man-in-the-middle" (MITM).

### 3. Integrità dei dati:

HTTPS assicura che i dati trasmessi non siano stati alterati o corrotti durante il trasferimento.

## Tutti i siti web che utilizzano HTTPS sono considerati sicuri?

NO! Anche se HTTPS garantisce crittografia e autenticazione, non significa automaticamente che il sito sia affidabile o privo di pericoli.

Gli hacker possono ottenere certificati SSL validi e creare siti dannosi con HTTPS, sfruttando domini simili a quelli reali.

Inoltre, HTTPS non protegge da download dannosi, quindi anche siti con questo protocollo potrebbero contenere malware.