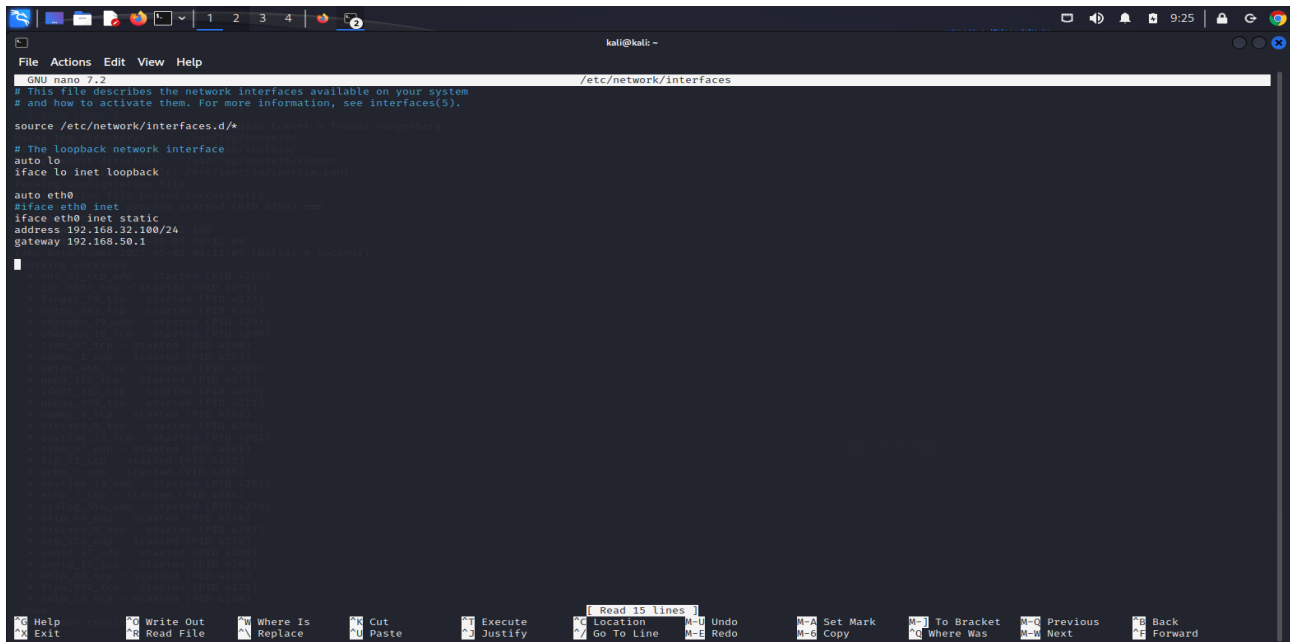


Report Mendolia Valerio – Compito Epicode 1

Assegnazione degli indirizzi IP richiesti nel compito:

Macchina Kali Linux (192.168.32.100):



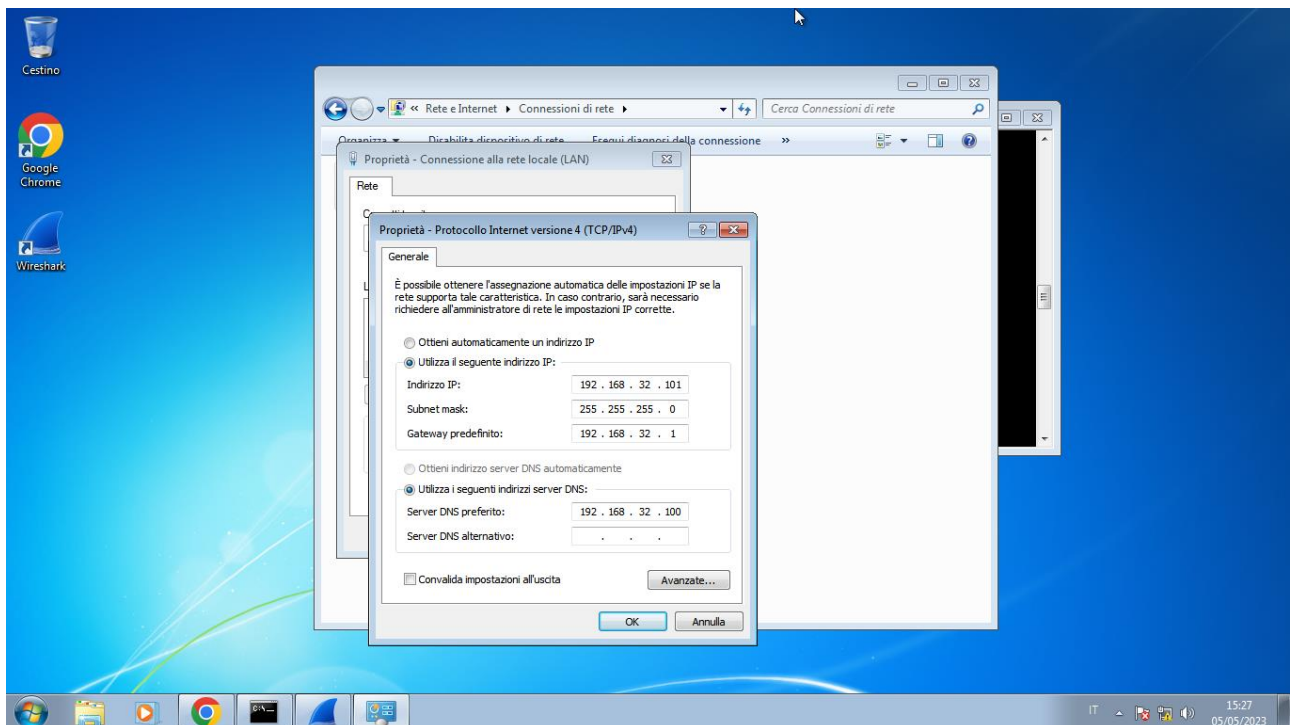
```
kali@kali ~$ nano /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

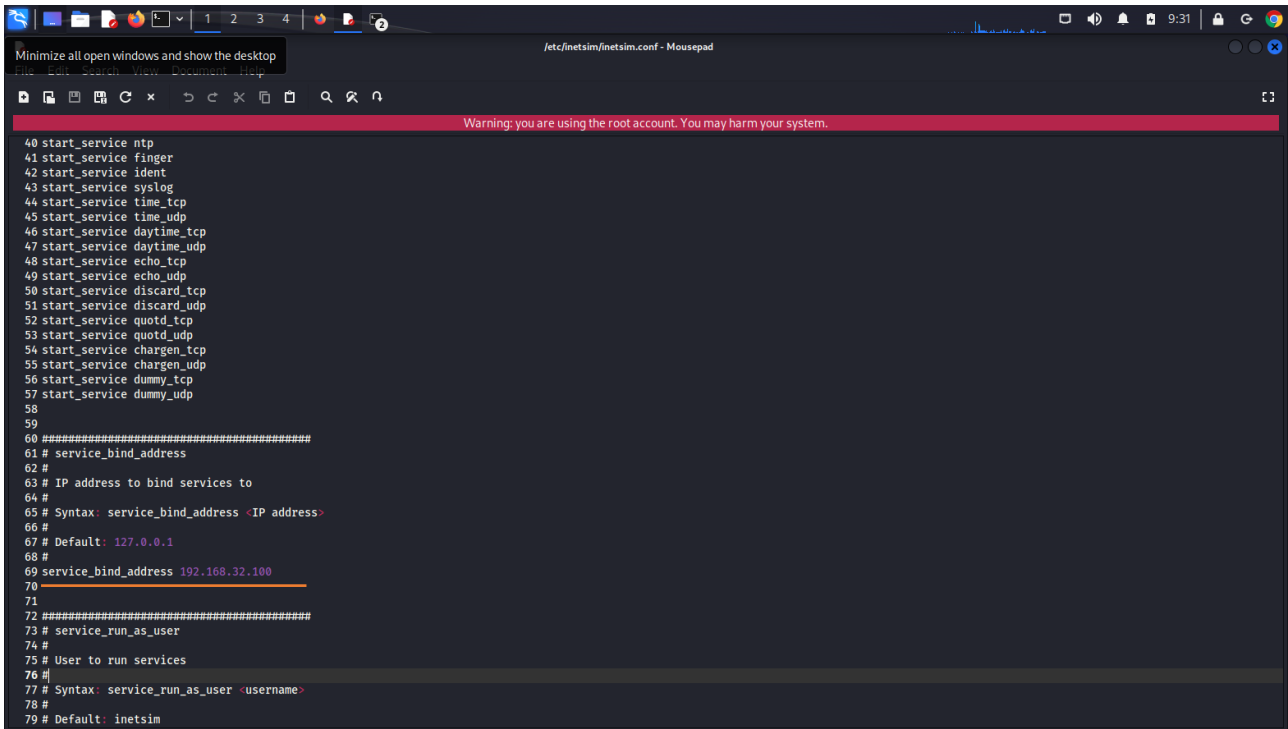
# The ethernet interface
auto eth0
iface eth0 inet static
address 192.168.32.100/24
gateway 192.168.50.1
```

Macchina Windows 7 64bit (192.168.32.101):



Configurazione del servizio inetsim:

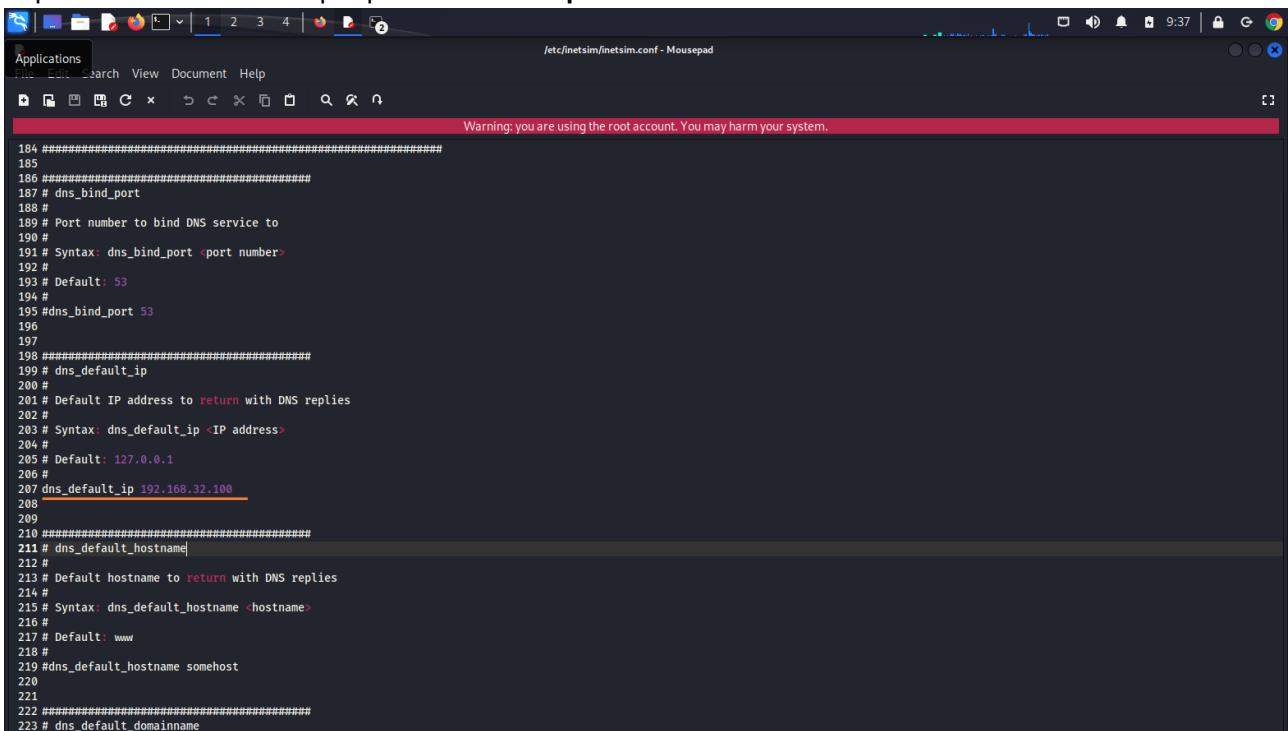
Comando : **sudo mousepad /etc/inetsim/inetsim.conf**



```
40 start_service ntp
41 start_service finger
42 start_service ident
43 start_service syslog
44 start_service time_tcp
45 start_service time_udp
46 start_service daytime_tcp
47 start_service daytime_udp
48 start_service echo_tcp
49 start_service echo_udp
50 start_service discard_tcp
51 start_service discard_udp
52 start_service quotd_tcp
53 start_service quotd_udp
54 start_service chargen_tcp
55 start_service chargen_udp
56 start_service dummy_tcp
57 start_service dummy_udp
58
59
60 #####
61 # service_bind_address
62 #
63 # IP address to bind services to
64 #
65 # Syntax: service_bind_address <IP address>
66 #
67 # Default: 127.0.0.1
68 #
69 service_bind_address 192.168.32.100
70
71
72 #####
73 # service_run_as_user
74 #
75 # User to run services
76 #
77 # Syntax: service_run_as_user <username>
78 #
79 # Default: inetsim
```

Assegnazione dell'interfaccia di rete su **service_bind_address** con **192.168.32.100**, così usciamo dal localhost e andiamo a comunicare con Windows 7.

Imposto le modifiche DNS per poter utilizzare **epicode.internal** sulla macchina Windows:

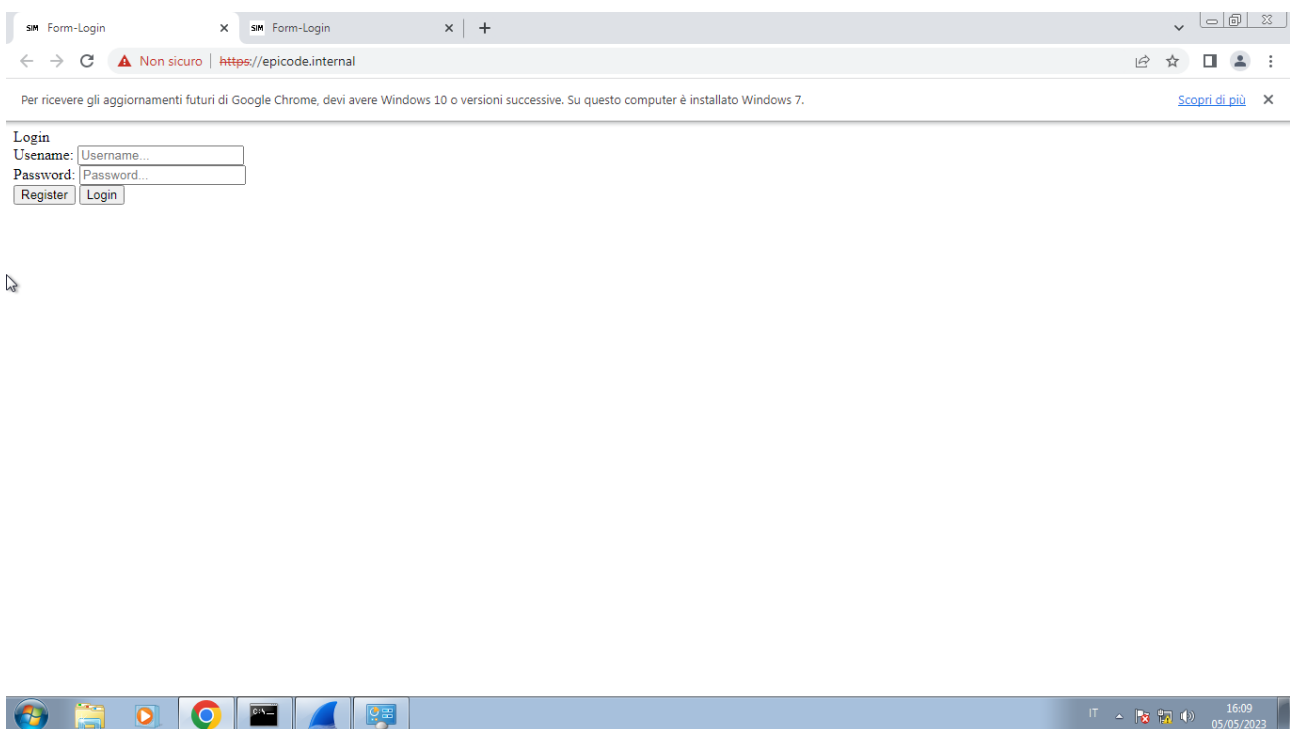


```
184 #####
185
186 #####
187 # dns_bind_port
188 #
189 # Port number to bind DNS service to
190 #
191 # Syntax: dns_bind_port <port number>
192 #
193 # Default: 53
194 #
195 #dns_bind_port 53
196
197
198 #####
199 # dns_default_ip
200 #
201 # Default IP address to return with DNS replies
202 #
203 # Syntax: dns_default_ip <IP address>
204 #
205 # Default: 127.0.0.1
206 #
207 dns_default_ip 192.168.32.100
208
209
210 #####
211 # dns_default_hostname
212 #
213 # Default hostname to return with DNS replies
214 #
215 # Syntax: dns_default_hostname <hostname>
216 #
217 # Default: www
218 #
219 #dns_default_hostname somehost
220
221
222 #####
223 # dns_default_domainname
```

Avvio inetsim : **sudo inetsim**

```
Applications Edit View Help
kali@kali ~
$ sudo mousepad /etc/inetsim/inetsim.conf
(kali@kali)~$ sudo inetsim
InetSim 1.3.2 (2020-05-19) by Matthias Eckert & Thomas Hungenberg
Using log directory: /var/log/inetsim/
Using data directory: /var/lib/inetsim/
Using report directory: /var/log/inetsim/report/
Using configuration file: /etc/inetsim/inetsim.conf
Parsing configuration file.
Configuration file parsed successfully.
InetSim main process started (PID 4255)
Session ID: 4255
Listening on: 192.168.32.100
Real Date/Time: 2023-05-05 09:12:09
Fake Date/Time: 2023-05-05 09:12:09 (Delta: 0 seconds)
Forking services ...
* dns_53_tcp_udp - started (PID 4265)
* irc_6667_tcp - started (PID 4275)
* finger_79_tcp - started (PID 4277)
* https_443_tcp - started (PID 4267)
* charge0_19_udp - started (PID 4291)
* charge0_19_tcp - started (PID 4290)
* time_37_tcp - started (PID 4280)
* dummy_1_udp - started (PID 4293)
* smtps_465_tcp - started (PID 4269)
* pop3_110_tcp - started (PID 4270)
* ident_113_tcp - started (PID 4278)
* pop3s_995_tcp - started (PID 4271)
* dummy_1_tcp - started (PID 4292)
* discard_9_tcp - started (PID 4286)
* daytime_13_tcp - started (PID 4282)
* time_37_udp - started (PID 4281)
* ftp_21_tcp - started (PID 4272)
* echo_7_udp - started (PID 4285)
* daytime_13_udp - started (PID 4283)
* echo_7_tcp - started (PID 4284)
* syslog_514_udp - started (PID 4279)
* tftp_69_udp - started (PID 4274)
* discard_9_udp - started (PID 4287)
* ntp_123_udp - started (PID 4276)
* quotd_17_udp - started (PID 4289)
* quotd_17_tcp - started (PID 4288)
* http_80_tcp - started (PID 4266)
* ftps_990_tcp - started (PID 4273)
* smtp_25_tcp - started (PID 4268)
done.
Simulation running.
```

Cambio il file HTML default di inetsim con un login con user e password e testo sulla mia macchina di windows 7 l'url con SSL <https://epicode.internal> e l'url senza SSL <http://epicode.internal> .



Invio una richiesta POST dal browser chrome con Windows 7(192.168.32.101) e nel login metto username: **adam** e password: **adam**

e imposto i filtri di wireshark con

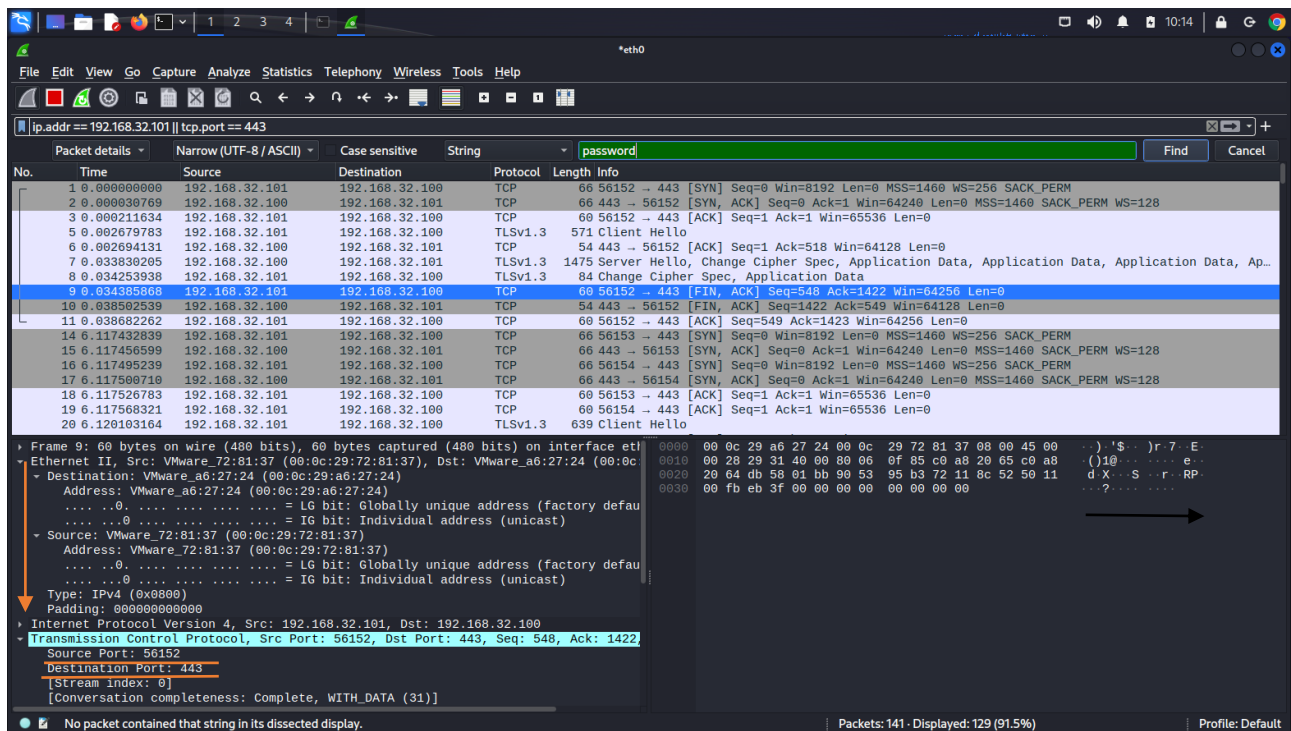
ip.addr == 192.168.32.101 || tcp.port == 443

così vado a vedere la richiesta che mi interessa.

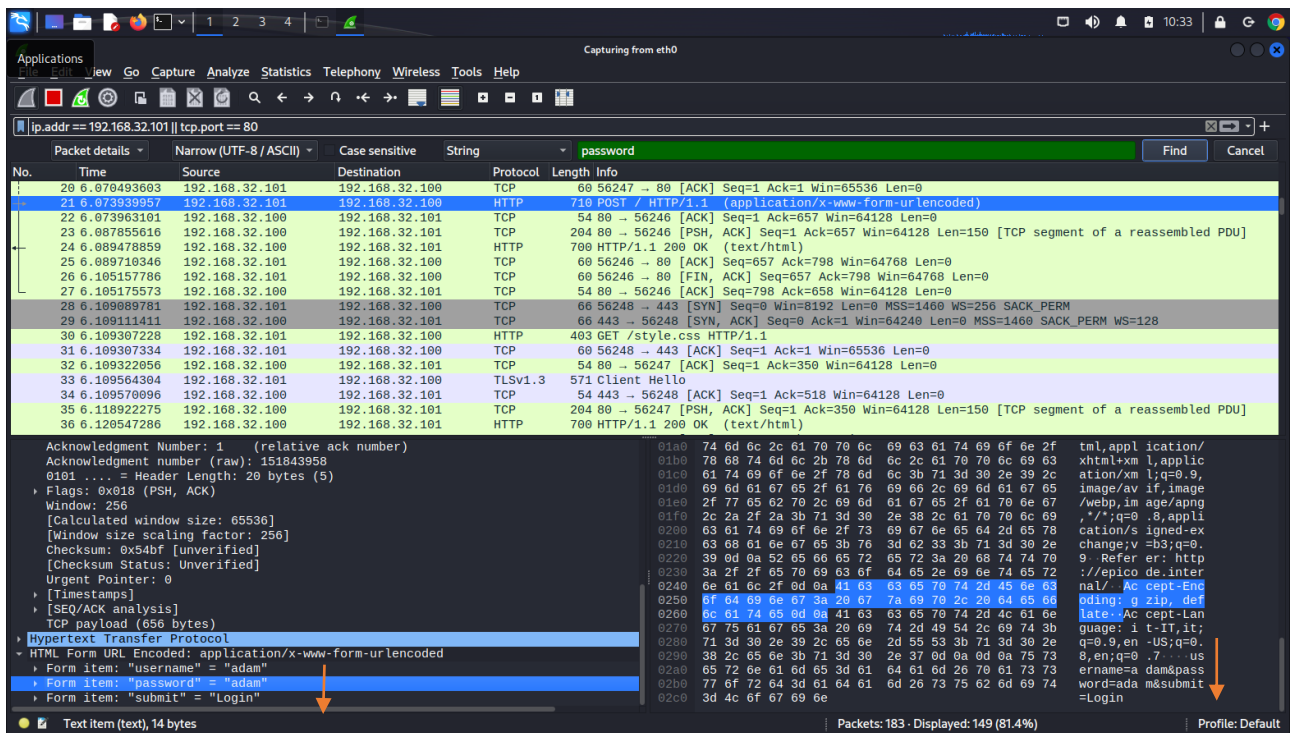
Provo a cercare la stringa del POST '**password**' tra i pacchetti ma ovviamente non viene trovata perché sono sotto la crittografia SSL/TLS (Sennò ruberebbero tutte le carte di credito online). Ovviamente il certificato non è verificato da nessun ente, quindi mi dà errore di certificato.

Qua il mio Sniffing è fallito, non vedo alcun dato sensibile dell'utente.

Sullo screen viene visualizzato l'indirizzo ip di origine e la sua porta, l'indirizzo ip di destinazione e la sua porta e gli eventuali mac address.



Adesso proverò la stessa cosa con la porta 80 di http (Non crittografata):



Come si vede dalla foto il mio sniffing qua ha avuto successo, sono riuscito a prendere la password di Adam soltanto perché il passaggio di informazioni con il POST non aveva alcun tipo di crittografia. La crittografia in SSL/TSL fa una differenza sostanziale in questo esempio o nel web perché ci consente di capire quanto possa essere pericoloso non utilizzarla e che i nostri dati possano essere sempre in pericolo.

Mendolia Valerio.