Utilizzo di wireshark per esaminare il traffico HTTP e HTTPS

- **Parte 1: Cattura e visualizza il traffico HTTP**
- **Parte 2: Acquisizione e visualizzazione del traffico HTTPS**

1) Dal terminale di Kali usiamo il comando ifconfig e notiamo subito che con l' interfaccia di rete eth= abbiamo i relativi ip 192.168.50.100 e con l' interfaccia di rete lo l' ip 127.0.0.1

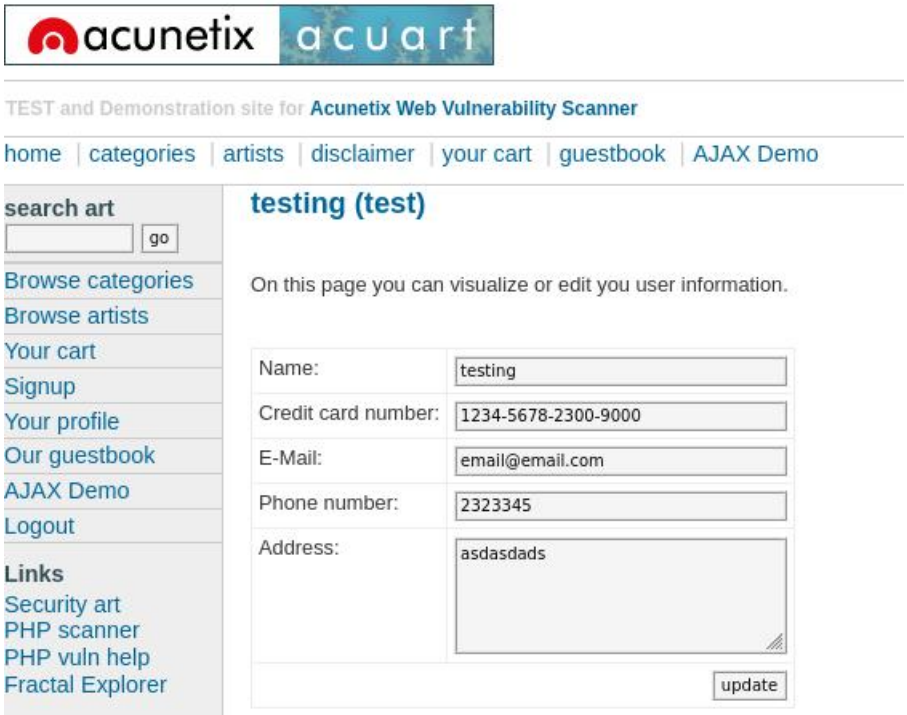
```
┌──(kali㉿kali)-[~/Desktop]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.50.100  netmask 255.255.255.0  broadcast 192.168.50.255
        ether 08:00:27:6e:13:6e  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 10  bytes 1189 (1.1 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 8  bytes 480 (480.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8  bytes 480 (480.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Andiamo dunque a digitare il comando sudo tcdump -i eth= -w httpdump.pcap per registrare il traffico di rete nell' interfaccia di rete eth0.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo tcpdump -i eth0 -s 0 -w httpdump.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

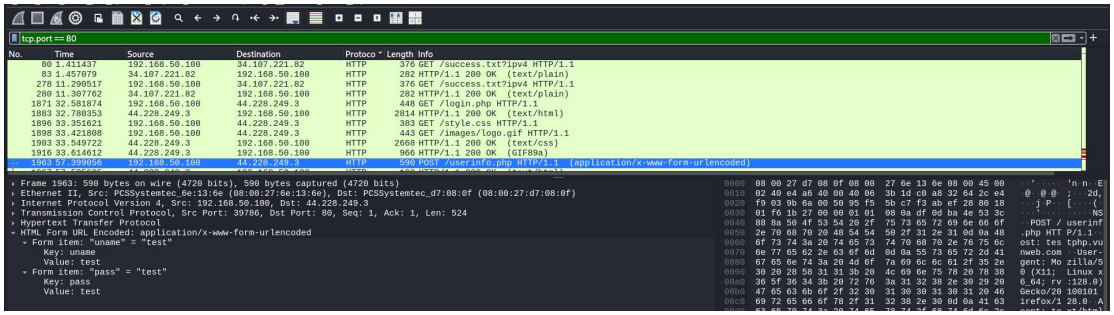
Ora facciamo l' accesso al sito http://testphp.vulnweb.com/login.php ed entriamo come user test e con la password test.



Ora chiudiamo il browser e torniamo nel terminale e con Ctrl+c interrompiamo l'acquisizione dei pacchetti precedentemente avviata, notando 2335 pacchetti ricevuti.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo tcpdump -i eth0 -s 0 -w httpdump.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C2335 packets captured
2335 packets received by filter
0 packets dropped by kernel
```

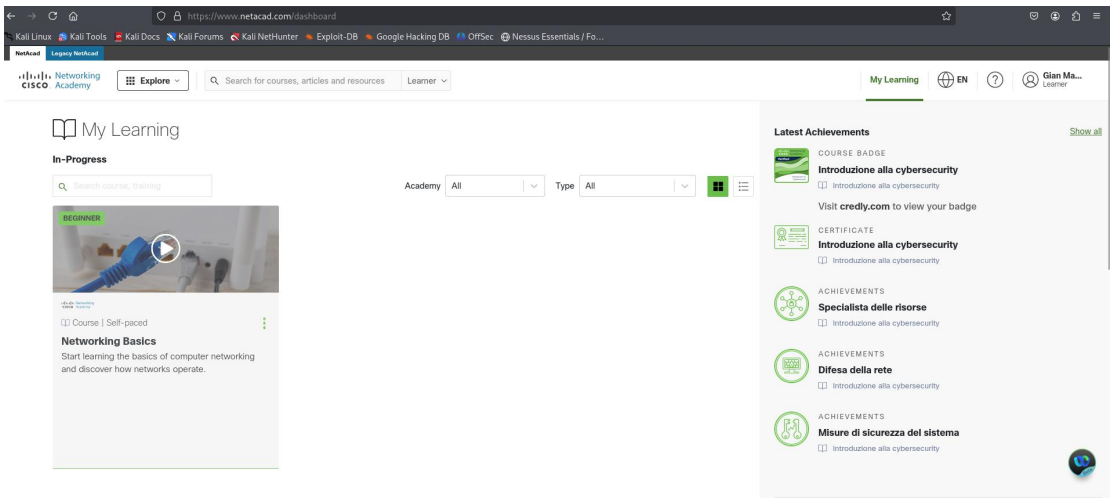
Ora apriamo il file precedentemente creato dal nome httpdump.pcap che ci porterà direttamente nell' interfaccia di Whireshark . Ora nella barra delle ricerche digitiamo il filtro tcp.port==80 per visualizzare la cattura del traffico HTTP sulla porta 80 e, espandendo la sezione URL del modulo HTML codificato: application/x-www-form-urlencoded possiamo notare l' UID e la password dell' amministratore.



2) Rifacciamo tutto lo stesso percorso ma con HTTPS.

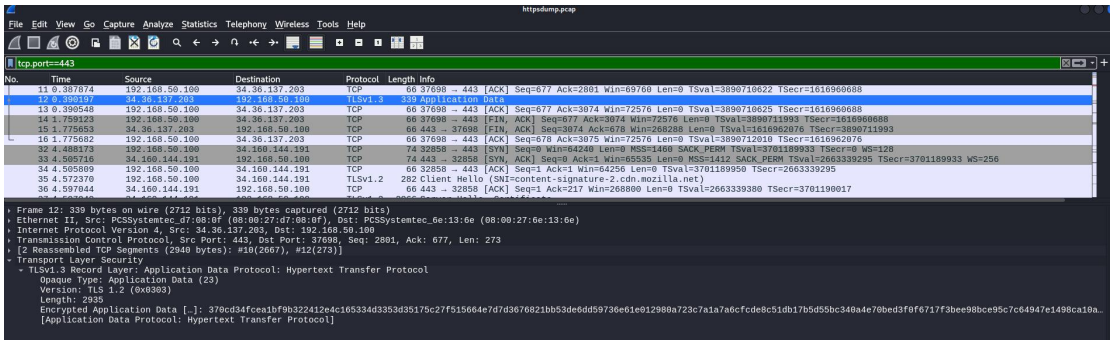
```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo tcpdump -i eth0 -s 0 -w httpsdump.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

Questa volta utilizzeremo il sito http://www.netacad.com/ e dopo esserci logati chiudiamo il browser e terminiamo la ricezione dei pacchetti dal terminale.



```
┌──(kali㉿kali)-[~/Desktop]
└─$ sudo tcpdump -i eth0 -s 0 -w httpsdump.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C15559 packets captured
15559 packets received by filter
0 packets dropped by kernel
```

Notiamo che sono stati ricevuti 1559 pacchetti.
Ora apriamo il file httpsdump.pcap e filtriamo con tcp.port==443.



Qui possiamo notare la differenza fondamentale con HTTP, espandendo la sezione Secure Sokets Layer notiamo la stringa Encrypted application data. Qui il payload dei dati è crittografato tramite TLS versione 1.2 e quindi non può essere visualizzato.
Possiamo concludere che il protocollo HTTPS risulta più sicuro in quanto presenta crittografia dei dati proteggendo l' integrità dei dati stessi e la privacy.