Report di Laboratorio ES4: Analisi del Traffico HTTP e HTTPS con tcpdump e Wireshark

Oggetto: Esercizio 4 - Usare Wireshark per Esaminare il Traffico HTTP e HTTPS

Strumenti Utilizzati: Kali Linux, terminale (tcpdump), Wireshark, Web Browser.

Obiettivi del Laboratorio

Il presente laboratorio ha lo scopo di dimostrare, tramite l'acquisizione e l'analisi dei pacchetti di rete, la differenza fondamentale in termini di confidenzialità tra il protocollo HTTP (in chiaro) e il protocollo HTTPS (cifrato).

L'attività è divisa in due fasi:

1. Cattura e visualizzazione delle credenziali trasmesse in chiaro via HTTP.
2. Cattura e visualizzazione del traffico cifrato tramite HTTPS (TLS).

Ambiente di Lavoro e Setup Iniziale

Come prima operazione, è stato necessario identificare l'interfaccia di rete corretta su cui porre in ascolto lo sniffer.

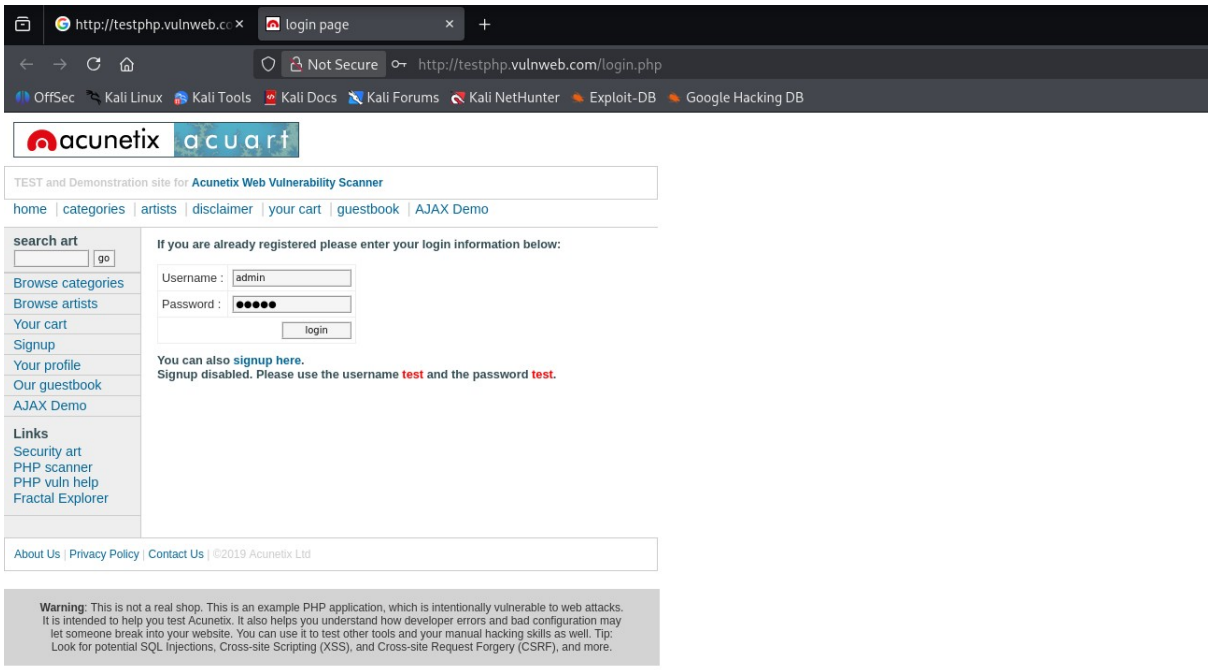
```
┌──(kali㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1f:b7:23 brd ff:ff:ff:ff:ff:ff
    inet 192.168.50.100/24 brd 192.168.50.255 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever

┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 -s 0 -w httpdump.pcap
[sudo] password for kali:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C6746 packets captured
6750 packets received by filter
0 packets dropped by kernel
```

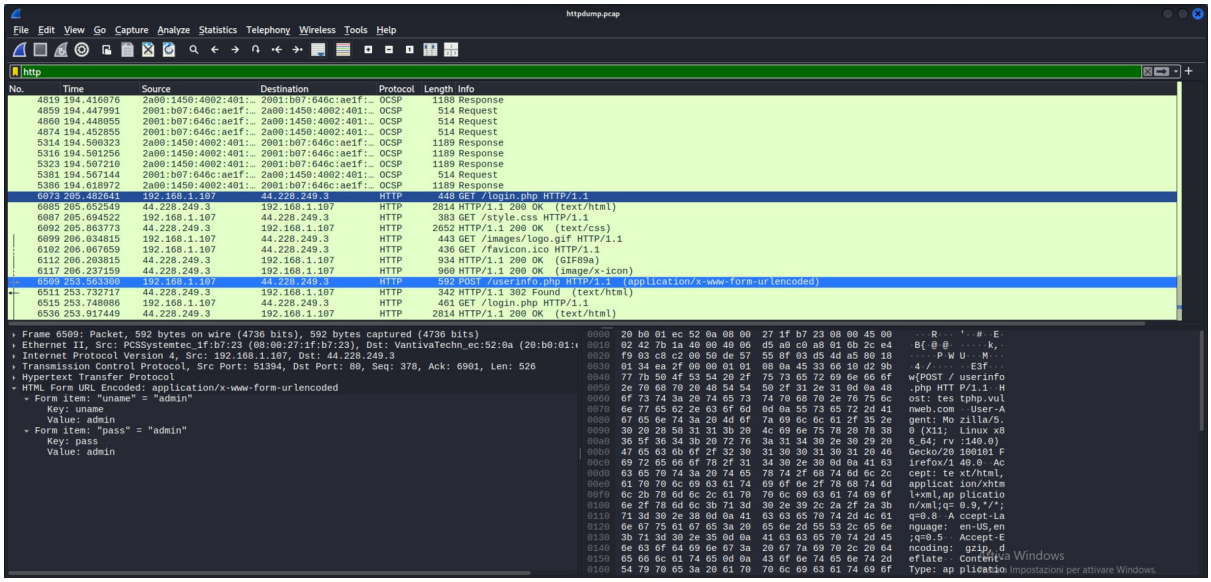
Parte 1: Catturare e Visualizzare il Traffico HTTP

Svolgimento Pratico

1. **Avvio della cattura:** È stato avviato il tool a riga di comando tcpdump con privilegi di root sull'interfaccia eth0, salvando l'output nel file httpdump.pcap:
   sudo tcpdump -i eth0 -s 0 -w httpdump.pcap
2. **Generazione del traffico:** Tramite browser, è stata raggiunta la URL vulnerabile di test http://testphp.vulnweb.com/login.php. Come mostrato nello screenshot, il browser segnala correttamente la connessione come "Not Secure". È stato effettuato il login inserendo admin come Username e admin come Password.



3. **Analisi con Wireshark:** La cattura è stata interrotta (raccogliendo 6746 pacchetti) ed è stato aperto il file httpdump.pcap con Wireshark. Filtrando per http, è stato individuato il pacchetto di tipo POST diretto a /userinfo.php.



Risposta alla domanda della traccia (Parte 1)

Domanda: Espandendo la sezione *HTML Form URL Encoded: application/x-www-form-urlencoded*, quali due informazioni vengono visualizzate?

Risposta: Come chiaramente visibile nello screenshot nel riquadro inferiore di Wireshark, vengono visualizzate in chiaro le credenziali di accesso inserite dall'utente:

- Form item: "uname" = "admin" (Nome utente)
- Form item: "pass" = "admin" (Password)
  Questo dimostra l'assenza di cifratura nel protocollo HTTP, esponendo i dati sensibili ad attacchi di tipo *Man-in-the-Middle (MitM)* o di *sniffing* passivo.

Parte 2: Catturare e Visualizzare il Traffico HTTPS

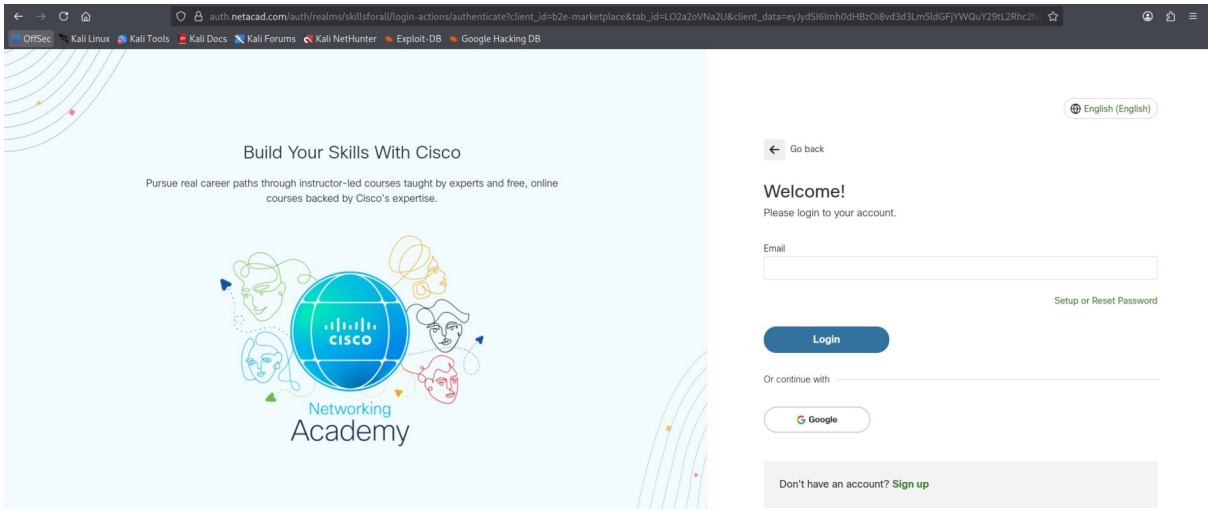
Svolgimento Pratico

1. **Avvio della cattura:** È stata avviata una nuova sessione di tcpdump salvando l'output nel file httpsdump.pcap:
   sudo tcpdump -i eth0 -s 0 -w httpsdump.pcap

```
┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 -s 0 -w httpdump.pcap
[sudo] password for kali:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C6746 packets captured
6750 packets received by filter
0 packets dropped by kernel

┌──(kali㉿kali)-[~]
└─$ sudo tcpdump -i eth0 -s 0 -w httpsdump.pcap
[sudo] password for kali:
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C6159 packets captured
6159 packets received by filter
0 packets dropped by kernel
```

2. **Generazione del traffico:** È stato aperto il browser navigando sul portale di autenticazione Cisco NetAcad / Skills For All (auth.netacad.com).

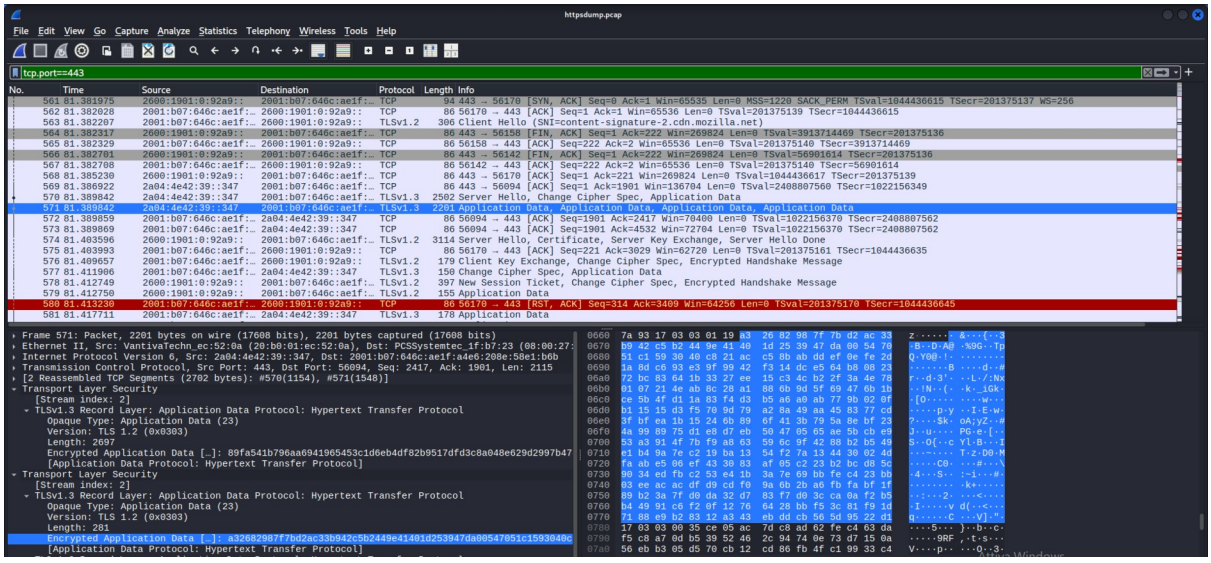


Risposte alle domande della traccia (Parte 2)

Domanda: Cosa noti riguardo all'URL del sito web?

Risposta: L'URL inizia con il prefisso https:// ed è accompagnato dall'icona di un lucchetto nel browser. Questo indica che la connessione tra il client (Kali Linux) e il server web è protetta tramite un tunnel crittografico (SSL/TLS), garantendo un canale di comunicazione sicuro.

Analisi con Wireshark: Il file httpsdump.pcap è stato aperto e filtrato utilizzando la sintassi tcp.port == 443. È stato selezionato un pacchetto di tipo Application Data.



Domanda: Cosa ha sostituito la sezione HTTP che era nel file di cattura precedente?

Risposta: La sezione HTTP (HyperText Transfer Protocol) è stata sostituita dal livello **Transport Layer Security (TLS)**. Analizzando lo screenshot, si nota in particolare che la negoziazione è avvenuta utilizzando **TLSv1.3**, il protocollo crittografico più recente e sicuro (un aggiornamento rispetto al TLSv1.2 mostrato nell'esempio della traccia).

Domanda: I dati dell'applicazione (Encrypted Application Data) sono in formato plaintext o leggibile?

Risposta: Assolutamente no. I dati sotto la voce Encrypted Application Data si presentano come una stringa esadecimale pseudocasuale illeggibile. Senza le chiavi crittografiche di sessione (simmetriche) concordate tra client e server durante l'Handshake TLS, è computazionalmente impossibile per un attaccante (o per il nostro sniffer) decifrare e leggere il contenuto del payload (es. le credenziali di accesso).

Risposte alle Domande di Riflessione Finali

1. Quali sono i vantaggi dell'uso di HTTPS invece di HTTP?

I vantaggi principali (i tre pilastri della sicurezza in transito) sono:

- **Confidenzialità (Cifratura):** Tutti i dati trasmessi (URL esatte, parametri POST, cookie di sessione, credenziali) sono cifrati. Nessun intermediario può leggerli.
- **Integrità dei dati:** Il protocollo TLS utilizza codici MAC (Message Authentication Code) per assicurare che i dati non vengano alterati o manomessi durante il transito.
- **Autenticazione:** Tramite l'uso di Certificati Digitali X.509 emessi da CA (Certificate Authorities) fidate, il client può verificare l'identità del server, mitigando il rischio di comunicare con un server truffaldino.

2. Tutti i siti web che usano HTTPS sono considerati affidabili?

No. Questo è un falso mito molto pericoloso per gli utenti finali. La presenza di HTTPS garantisce esclusivamente che il canale di comunicazione tra il browser e il server sia cifrato e privato. Non garantisce in alcun modo le intenzioni di chi possiede il server.

Oggi, gli attori malevoli ottengono facilmente certificati SSL/TLS gratuiti (es. tramite Let's Encrypt) per i loro siti di phishing o per distribuire malware. Pertanto, un sito HTTPS garantisce solo una "comunicazione sicura con un sito che potrebbe essere malevolo". L'affidabilità del sito deve essere stabilita tramite altri fattori (reputazione del dominio, assenza di typosquatting, contesto, ecc.).

Conclusioni: L'esercizio ha dimostrato con successo, a livello di packet inspection, la vulnerabilità intrinseca del protocollo HTTP e l'efficacia del livello di incapsulamento e cifratura offerto da TLS (HTTPS) per la protezione dei dati sensibili in reti non sicure.