# Report Tecnico - Analisi HTTPS vs HTTP con Wireshark

## 1. Obiettivo dell'esercitazione

L'obiettivo del progetto è simulare in un ambiente virtuale un'architettura client-server, dove un client Windows accede tramite web browser a una risorsa pubblicata su un server Kali Linux, a nome di epicode.internal. L'intercettazione del traffico di rete viene effettuata con Wireshark ( Kali Linux), al fine di analizzare i pacchetti e confrontare la comunicazione su HTTPS e su HTTP.

## 2. Configurazione dell'ambiente

| | |
|---|---|
| Client | Windows - IP: 192.168.32.101 |
| Server | Kali Linux - IP: 192.168.32.100 |
| Hostname DNS | epicode.internal |
| Server Web | HTTP/HTTPS |
| Analisi | Wireshark su Kali Linux |

File hosts configurato su Windows: 192.168.32.100 epicode.internal

## 3. Fase 1 – Accesso HTTPS e analisi

Il server Apache su Kali è stato configurato con HTTPS. Il client ha effettuato una richiesta al sito: https://epicode.internal. Con Wireshark è stata poi intercettata la comunicazione effettuata da windows per poter verificare le varie informazioni di passaggio. Con il risultato che nella pagina epicode.internal verrà visualizzata la pagina di default di apache.

Risultati dell'analisi con Wireshark:

- MAC visibili

- Handshake visibile

- Contenuto cifrato e non leggibile

## 4. Fase 2 – Accesso HTTP e analisi

Il server è stato riconfigurato per rispondere su HTTP. Il client ha effettuato l'accesso a: http://epicode.internal. Utilizzando sempre Wireshark è stato possibile VEDERE tutte le informazioni che vendono trasferite

Risultati dell'analisi con Wireshark:

- MAC visibili

- Contenuto HTTP leggibile: richieste, cookie …

## 5. Differenze osservate

| Caratteristica | HTTPS | HTTP |
|---|---|---|
| Cifratura contenuto | Sì, dati cifrati | No, dati in chiaro |
| Analisi del contenuto | Non leggibile | Completamente leggibile |
| Visibilità MAC/IP | Visibili | Visibili |
| Sicurezza della comunicazione | Alta | Bassa |

## 6. Conclusioni

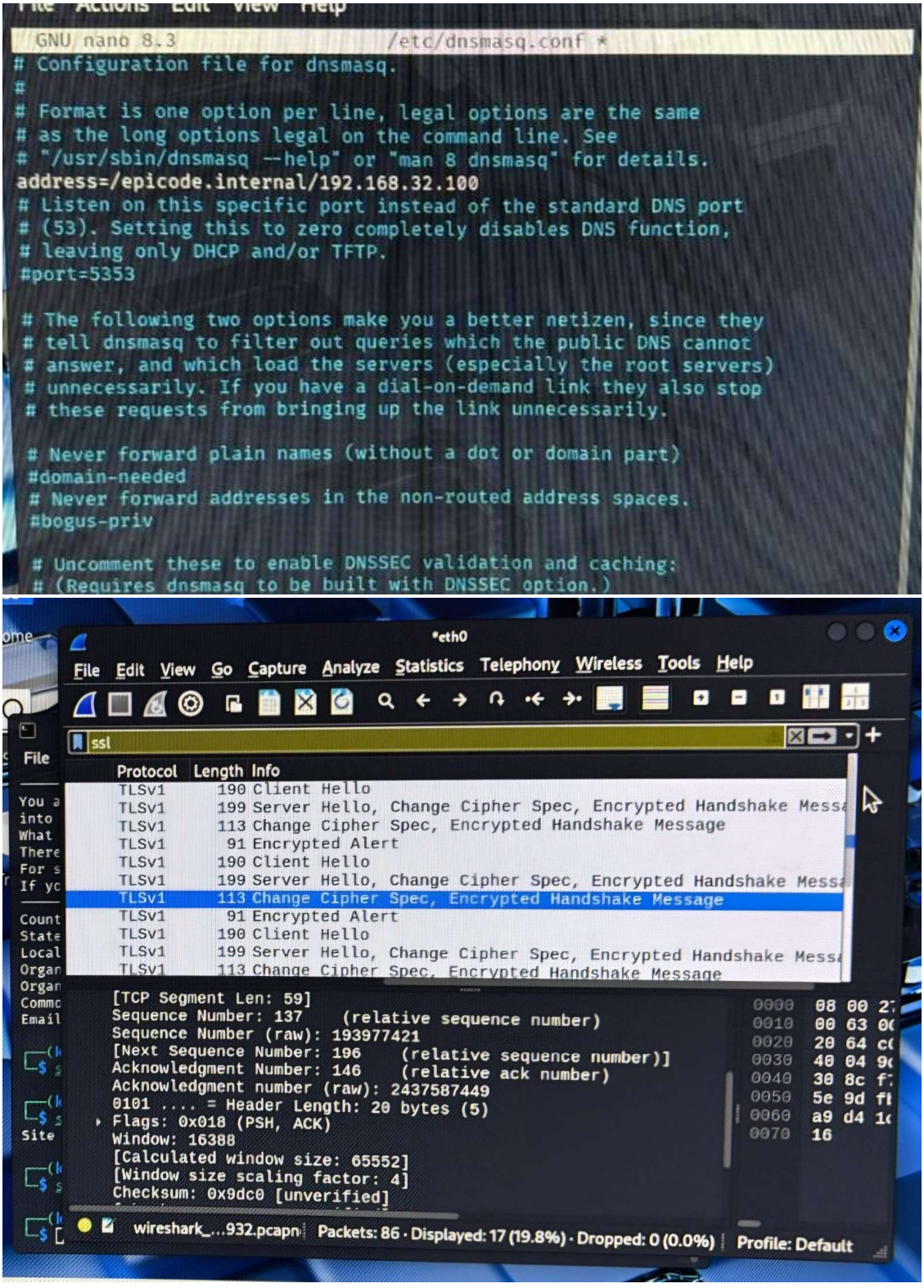
L'attività ha permesso di comprendere il funzionamento di una comunicazione client-server con risoluzione DNS locale, osservare con Wireshark le differenze tra traffico cifrato (HTTPS) e non cifrato (HTTP), e riflettere sull'importanza dell'uso di protocolli sicuri per proteggere la privacy degli utenti.

## 7. Risoluzione dei problemi

Non trovando soluzioni per quanto riguarda il DNS che risulta complicato “reperire” è stato scelto l’utilizzo del DNSmusq quindi andando a configurare si riesce a fornire il servizio disabilitando il firewall.

Utilizzo di Apache http Server per la condivisione di informazioni, e collegamento e per poter configurare il server (funzionalità) httpsconfig viene poi visualizzato al completamento

A causa di ripetuti problemi con le varie macchine virtuali sono stata costretta a trasferire il lavoro su un’altra macchina (pc) per la risoluzione del progetto.


GNU nano 8.3 /etc/dnsmasq.conf *
# Configuration file for dnsmasq.
#
# Format is one option per line, legal options are the same
# as the long options legal on the command line. See
# "/usr/sbin/dnsmasq --help" or "man 8 dnsmasq" for details.
address=/epicode.internal/192.168.32.100
# Listen on this specific port instead of the standard DNS port
# (53). Setting this to zero completely disables DNS function,
# leaving only DHCP and/or TFTP.
#port=5353
# The following two options make you a better netizen, since they
# tell dnsmasq to filter out queries which the public DNS cannot
# answer, and which load the servers (especially the root servers)
# unnecessarily. If you have a dial-on-demand link they also stop
# these requests from bringing up the link unnecessarily.
# Never forward plain names (without a dot or domain part)
#domain-needed
# Never forward addresses in the non-routed address spaces.
#bogus-priv
# Uncommend these to enable DNSSEC validation and caching:
# (Requires dnsmasq to be built with DNSSEC option.)
*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ssl
Protocol Length Info
TLSv1 190 Client Hello
TLSv1 199 Server Hello, Change Cipher Spec, Encrypted Handshake Messa
TLSv1 113 Change Cipher Spec, Encrypted Handshake Message
TLSv1 91 Encrypted Alert
TLSv1 190 Client Hello
TLSv1 199 Server Hello, Change Cipher Spec, Encrypted Handshake Messa
TLSv1 113 Change Cipher Spec, Encrypted Handshake Message
TLSv1 91 Encrypted Alert
TLSv1 190 Client Hello
TLSv1 199 Server Hello, Change Cipher Spec, Encrypted Handshake Messa
TLSv1 113 Change Cipher Spec, Encrypted Handshake Message
[TCP Segment Len: 59]
Sequence Number: 137 (relative sequence number)
Sequence Number (raw): 193977421
[Next Sequence Number: 196 (relative sequence number)]
Acknowledgment Number: 146 (relative ack number)
Acknowledgment number (raw): 2437587449
0101 .... = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 16388
[Calculated window size: 65552]
[Window size scaling factor: 4]
Checksum: 0x9dc0 [unverified]
0000 08 00 27
0010 00 63 0
0020 20 64 c
0030 40 04 9
0040 30 8c f
0050 5e 9d f
0060 a9 d4 1
0070 16
wireshark_...932.pcapn
Packets: 86 · Displayed: 17 (19.8%) · Dropped: 0 (0.0%)
Profile: Default