Cuore Andrea

28-10-2022

TASK

Nell'esercizio di oggi metteremo insieme le competenze acquisite finora. Lo studente verrà valutato sulla base della risoluzione al problema seguente.

Requisiti e servizi:

- Kali Linux IP 192.168.32.100
- Windows 7 IP 192.168.32.101
- HTTPS server: attivo
- Servizio DNS per risoluzione nomi di dominio: attivo

Traccia:

Simulare, in ambiente di laboratorio virtuale, un'architettura client server in cui un client con indirizzo 192.168.32.101 richiede tramite web browser una risorsa all'hostname epicode.internal che risponde all'indirizzo 192.168.32.100.

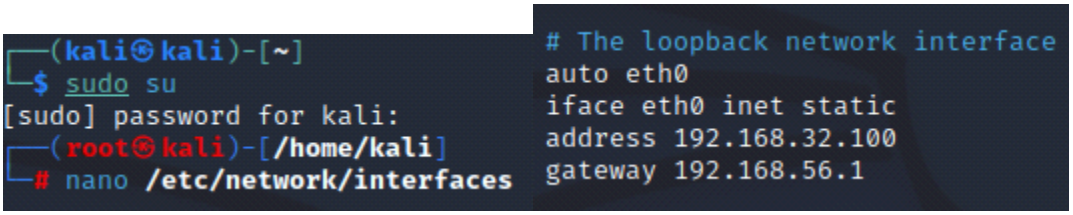
Si intercetti poi la comunicazione con Wireshark, evidenziando i MAC address di sorgente e destinazione ed il contenuto della richiesta HTTPS.

Ripetere l'esercizio, sostituendo il server HTTPS, con un server HTTP. Si intercetti nuovamente il traffico, evidenziando le differenze tra il traffico appena catturato in HTTP ed il traffico precedente in HTTPS. Spiegare, motivandole, le principali differenze.

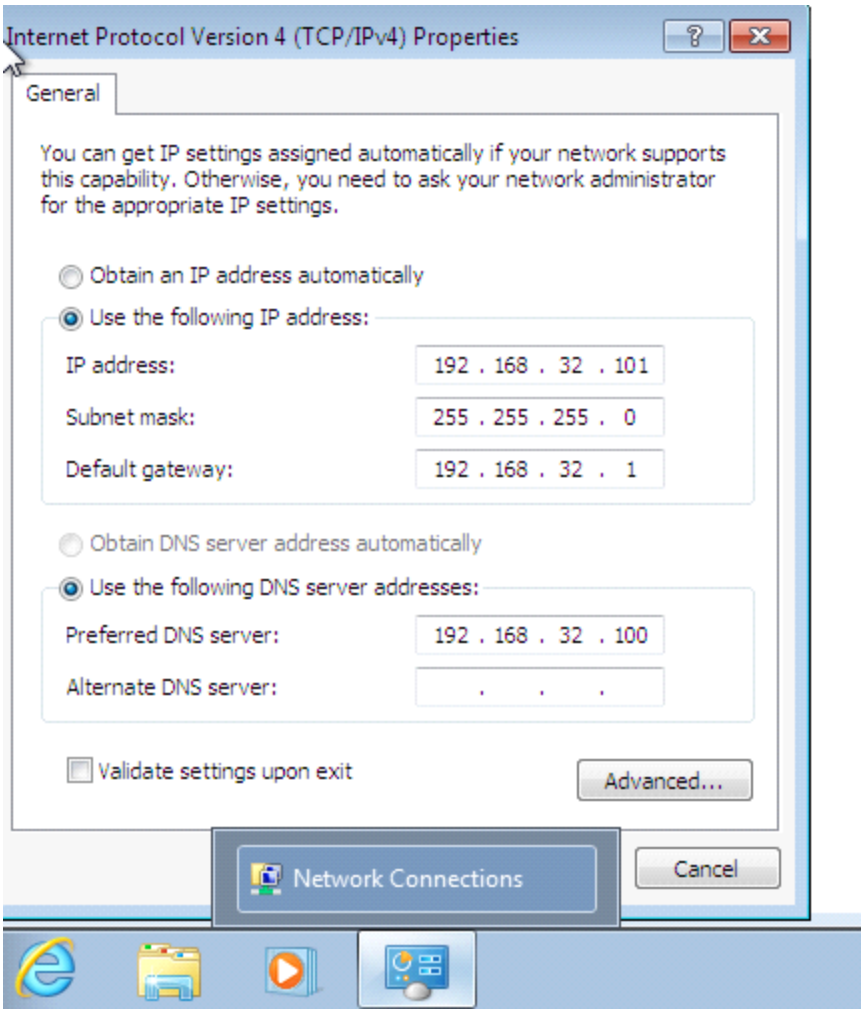
NUOVA CONFIGURAZIONE IP LINUX E WINDOWS

Come richiesto dalla traccia uno dei passi che abbiamo effettuato per la risoluzione di questa task, è quello di assegnare dei nuovi codici IP alle due macchine virtuali precedentemente create.

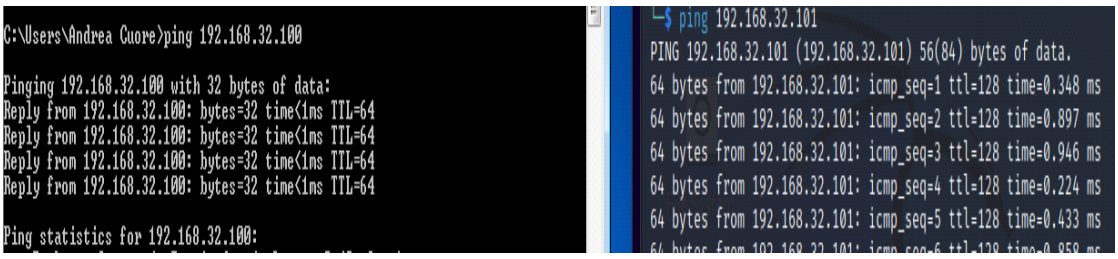
Linux : Per la procedura di configurazione del nuovo IP per la macchina Linux ci siamo recati nella shell accedendo come super user (sudo su), e con il comando nano /etc/network/interfaces siamo riusciti a cambiare l'indirizzo IP assegnandone uno nuovo con i seguenti valori:



Windows: Per il cambio IP della macchina Windows è stato sufficiente cambiare l'ip tramite le opzioni del sistema operativo. Una volta impostato l'IP assegnatoci dalla traccia è stato necessario anche impostare il DNS che ci servirà nelle fasi successive della task



Di seguito riportata la fase di test ping per le due macchine :



CONFIGURAZIONE SERVER CON INETSIM

Per simulare un server, e successivamente una richiesta HTTP e HTTPS ci siamo avvalsi del tool usato in precedenza INETSIM.

La configurazione di tale tool è avvenuta in 4 fasi

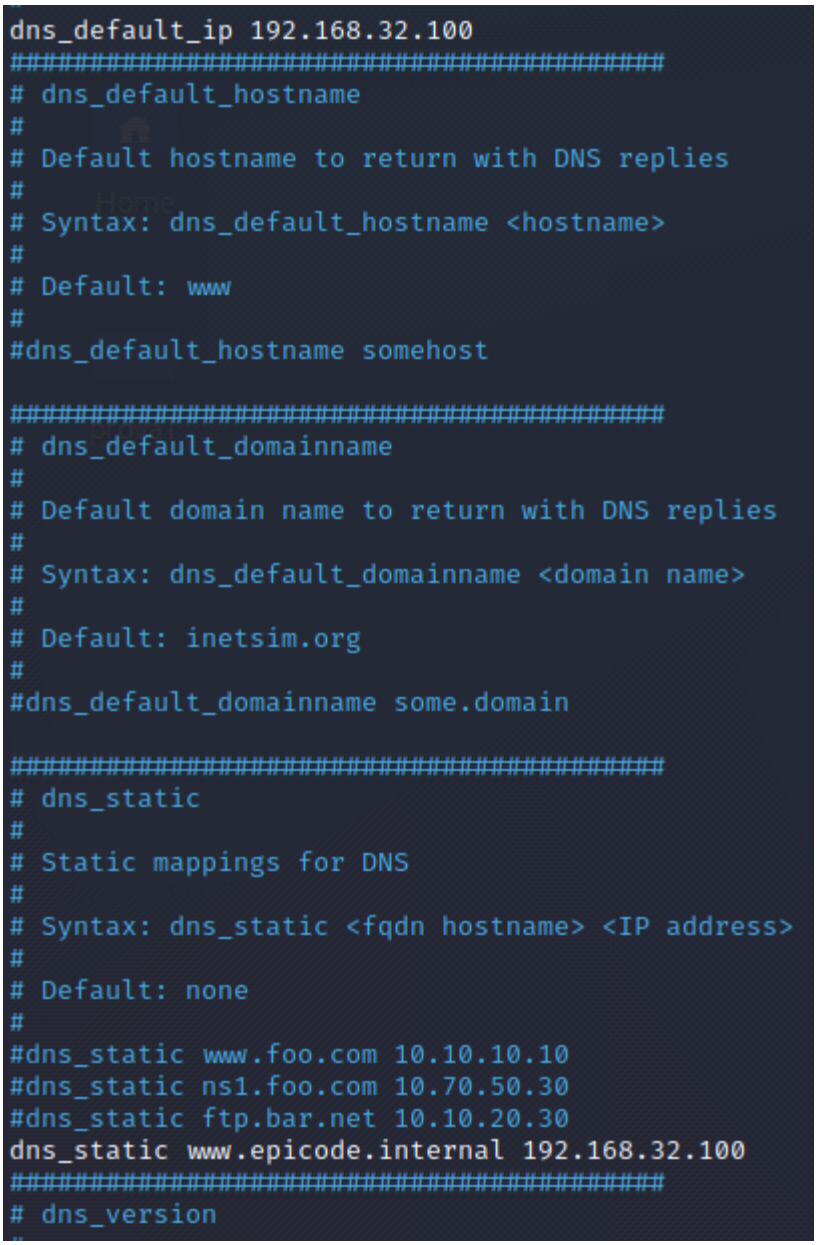
- 1. Disabilitazione dei servizi non richiesti

- 2. Impostare come IP di default il nostro IP macchina
- 3. Assegnare il DNS IP che avevamo utilizzato su windows (Indirizzo IP di kali ndr.)
- 4. Assegnare un hostname per la chiamata DNS

E' stato possibile modificare tutte queste parti, accedendo alla shell di linux come super user e modificare il file .conf , richiamando il comando nano /etc/inetsim/inetsim.conf

Di seguito riportate le immagini di configurazione:

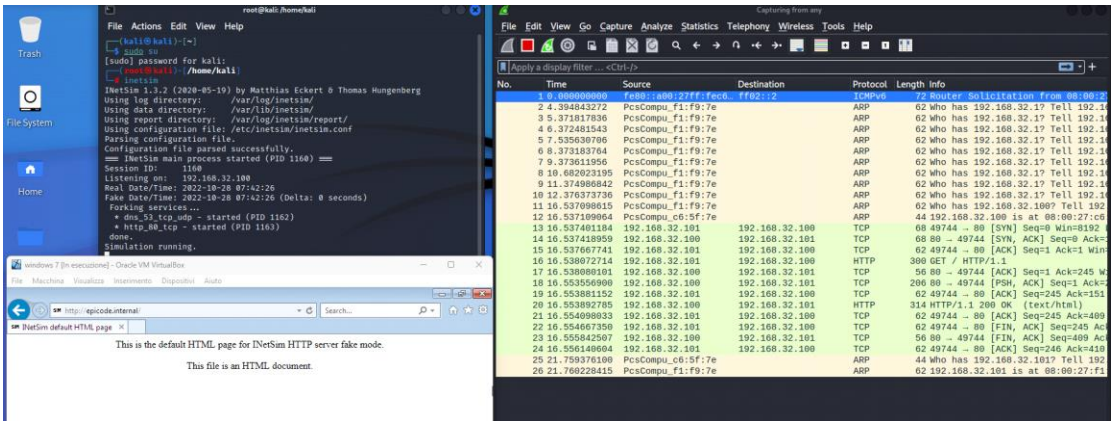
```
# start_service
#
# The services to start
#
# Syntax: start_service <service name>
#
# Default: none
#
# Available service names are:
# dns, http, smtp, pop3, tftp, ftp, ntp, time_tcp,
# time_udp, daytime_tcp, daytime_udp, echo_tcp,
# echo_udp, discard_tcp, discard_udp, quotd_tcp,
# quotd_udp, chargen_tcp, chargen_udp, finger,
# ident, syslog, dummy_tcp, dummy_udp, smtps, pop3s,
# ftps, irc, https
#
start_service dns
start_service http


##############################################
# service_bind_address
#
# IP address to bind services to
#
# Syntax: service_bind_address <IP address>
#
# Default: 127.0.0.1
#
#service_bind_address 10.10.10.1
service_bind_address 192.168.32.100

##############################################
# service_run_as_user
#
# User to run services
#
# Syntax: service_run_as_user <username>
#
# Default: inetsim
#
#service_run_as_user nobody

##############################################
# service_max_childs
#
# Maximum number of child processes (parallel connections)
# for each service
#
```



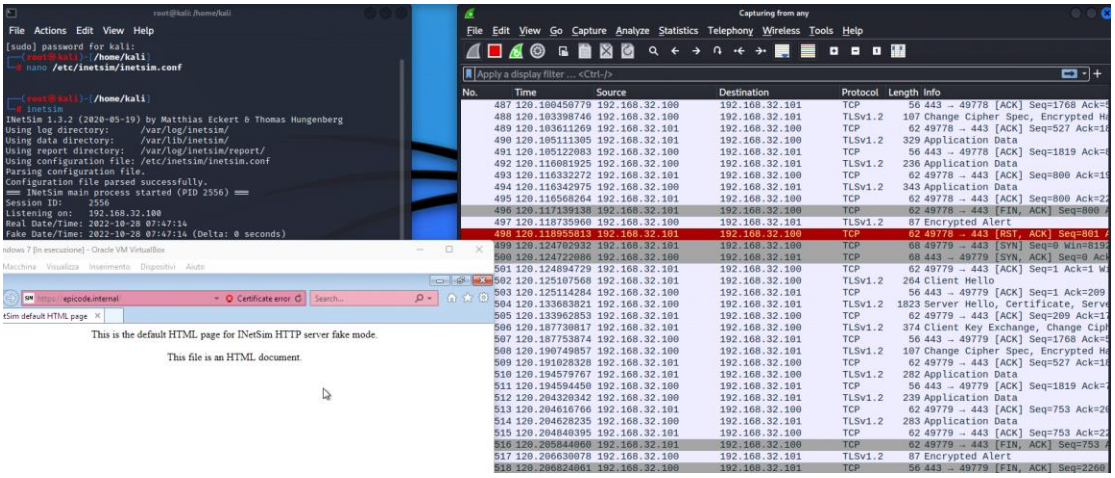
TEST SU WINDOWS E SNIFFING DI PACCHETTI:

Ultimata la configurazione del nostro file .conf è stato necessario richiamare il programma nella shell di comando per avviare il server e collegarci attraverso Windows 7.

SNIFFING PACCHETTI HTTP:

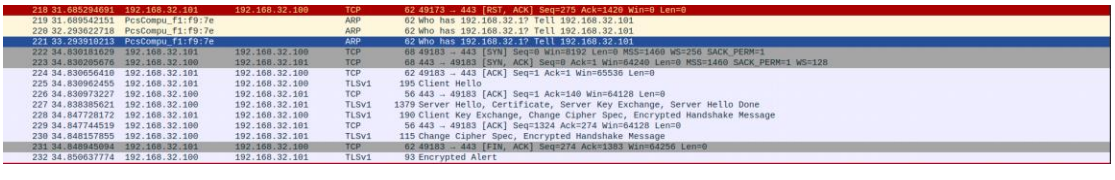


SNIFFING PACCHETTI HTTPS:



DIFFERENZA TRA HTTP E HTTPS

Le differenze che ho riscontrato nella fase di collegamento HTTP e HTTPS sono principalmente due, la prima riguarda il Flusso di pacchetti, infatti possiamo notare come la struttura di chiamata web cambia, Nella connessione in HTTP vediamo un three-way HandShake con GET diretto alla pagina HTTP, mentre se analizziamo lo sniffing di pacchetti HTTPS, possiamo notare anche i pacchetti di cifratura del protocollo TLS, con la relativa creazione della chiave di sessione, e relativo crypt del messaggio che possiamo analizzare più nel dettaglio nello screen qui sotto:



Altra differenza che possiamo notare è legata alla "Non autenticità del certificato TLS", ovvero il nostro certificato TLS non disponendo di una firma autentica viene riconosciuto dal nostro

Browser come "Sito non sicuro", e per tale ragione il browser cercherà di bloccarci nell'accesso.