

M1-W4D4- PRATICA

17/11/2023

Cybersecurity Analyst

Studente:

Andrea Scarmagnani

Docente:

Giuseppe Placanica

Traccia e requisiti

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 (Windows 7) richiede tramite web browser una risorsa all'hostname epicode.internal che risponde all'indirizzo 192.168.32.100 (Kali).

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 eventuali differenze tra il traffico appena catturato in HTTP ed il traffico precedente in HTTPS. Spiegare, motivandole, le principali differenze se presenti.

Configurazione macchina KALI


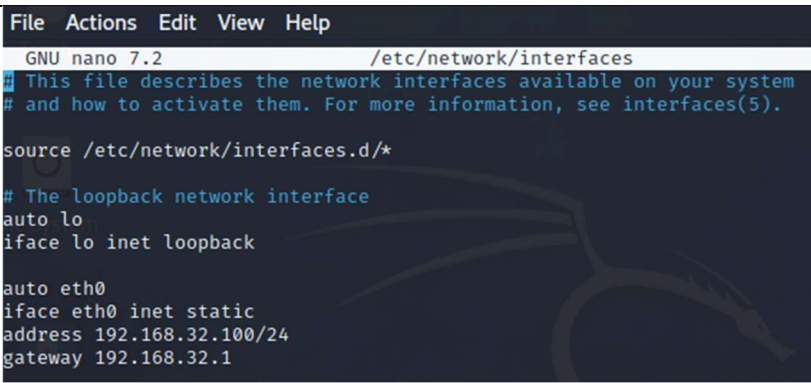

Di seguito la procedura su come è stato cambiato l'indirizzo IP della macchina **KALI** ed eventuali problemi riscontrati.

- Con il comando  si controlla l'indirizzo il risultato è simile a questo:

```
➥$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.32.100 netmask 255.255.255.0 broadcast 192.168.32.255
    inet6 fe80::a00:27ff:fe13:f17f prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:13:f1:7f txqueuelen 1000 (Ethernet)
    RX packets 192 bytes 15628 (15.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 116 bytes 12504 (12.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

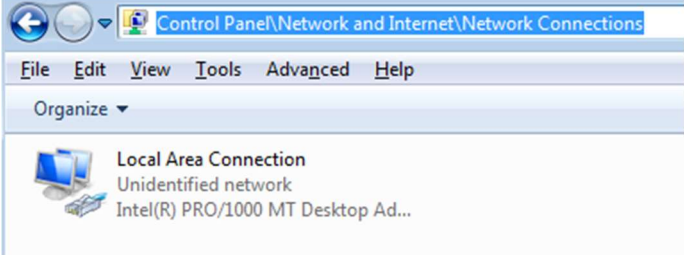
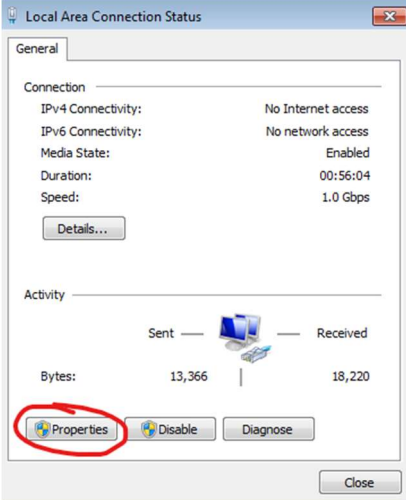
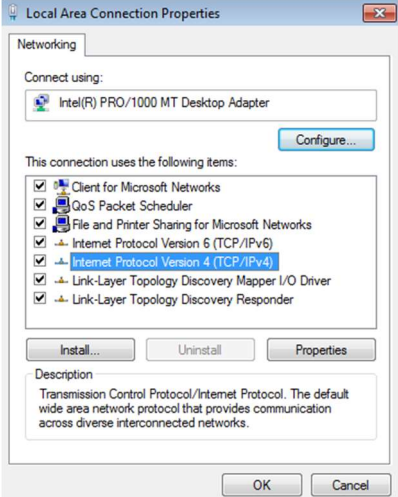
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
```

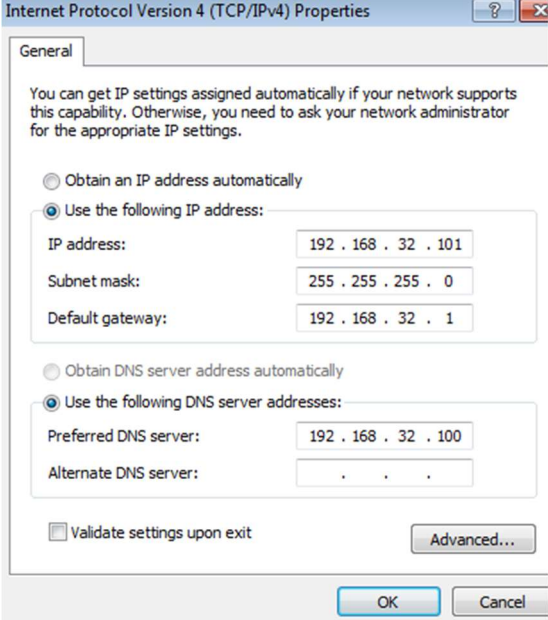
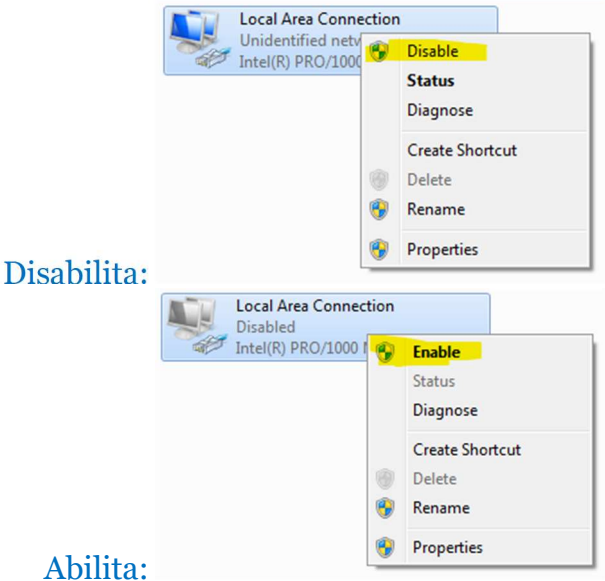
(In questo caso l'indirizzo è già stato modifica con quello richiesto dall'esercitazione.)

Con il comando: <code>sudo nano /etc/network/interfaces</code> si entra nella configurazione di rete:	
Questa è la schermata do viene fatta la configurazione. -Imposta l'indirizzo ip statico: <i>iface etho inet static</i> -Indirizzo ip della macchina: <i>address 192.168.32.100/24</i> -Indirizzo gateway della rete: <i>gateway 192.168.32.1</i> (con CTRL+X si esce, poi si deve confermare con Y e poi INVIO .)	
Problemi riscontrati, la macchina non inizializzava le nuove impostazioni: Con questo comando si riavvia i servizi della scheda di rete per rendere effettive le modifiche. Questo evita il riavvio dell'intera macchina.	

Configurazione macchina Windows7

Di seguito la procedura su come modificare l'indirizzo IP della macchina Windows7 ed eventuali problemi riscontrati.

<p>1: Nella cartella: (Control Panel\Network and Internet\Network Connections)</p> <p>Da qui si vedono le schede di rete installate nel computer. In questo caso una sola.</p>	
<p>2: Facendo clic su " Properties", è possibile accedere alle impostazioni di configurazione dalla schermata visualizzata.</p>	
<p>3: Selezionare il protocollo TCP/IPv4 e poi properties.</p>	

<p>4: Settare gli indirizzi come sei vede in figura</p> <p>L'indirizzo IP: Quello della macchina in funzione.</p> <p>Subnet mask: Aiuta a organizzare e gestire le reti.</p> <p>Default gateway: Punto di uscita predefinito per i dispositivi in una rete, consentendo loro di comunicare con altre reti o risorse esterne</p> <p>DNS: Va messo l'indirizzo della macchina Kali che in questo caso tramite il servizio InetSim ci darà la possibilità di avere un server DNS.</p>	
<p>Problemi riscontrati:</p> <p>Potrebbe succedere come nella macchina kali che le impostazioni di rete non si siano applicate, in questo caso senza dover riavviare l'intera macchina è possibile riavviare la sola scheda di rete disabilitandola e riabilitandola.</p> <p>Una volta impostato tutto è possibile dare Ok e chiudere tutte le finestre.</p>	 <p>Disabilita:</p> <p>Abilita:</p>

Ora non resta che avviare i servizi sulla macchina Kali, vediamo come...

Utilizzeremo il programma **InetSim**. Programma che verrà eseguito dalla macchina Kali, questo ci permette di avviare dei servizi e fare dei test in tutta sicurezza.

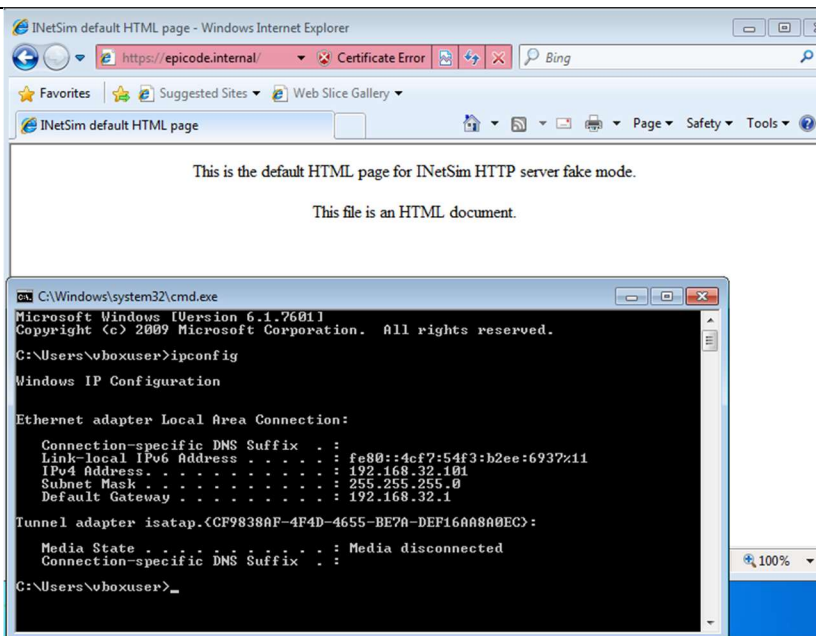
***InetSim** è un software progettato per emulare servizi di rete come HTTP, DNS, e molti altri, al fine di creare un ambiente di laboratorio sicuro per testare applicazioni e strumenti di sicurezza. In sostanza, InetSim crea un ambiente simulato di servizi di rete, consentendo agli sviluppatori e agli esperti di sicurezza di testare le applicazioni in modo controllato, senza connettersi effettivamente a Internet. Questo strumento può essere utilizzato per simulare una vasta gamma di scenari di rete e analizzare il comportamento di applicazioni e dispositivi in condizioni controllate.*

Configurazione InetSim

<p>Con questo comando entriamo nella configurazione di InetSim, entrando come super user (sudo) chiederà la password di root(in questo caso password di sistema).</p>	
<p>Ci troveremo di fronte a una schermata come quella a fianco, eseguito il comando sopra si apre tramite Nano (Editor di testo) il file di configurazione.</p> <p>Mettendo il carattere #(hashtag) prima del testo, viene abilitato oppure disabilitato il servizio.</p> <p>Esempio: <i>Disabilitato:</i> #start-service dns <i>Abilitato:</i> start-service dns</p> <p>Nel nostro caso va abilitato il dns per poter completare la nostra esercitazione.</p>	
<p>Scendendo lungo il file va attivato il <i>service_bind_address</i> con indirizzo della macchina server (Macchina Kali) oppure con 0.0.0.0 per mettere in ascolto la macchina su ogni chiamata.</p>	
<p>Scendendo ancora va aggiunto questa riga: <i>dns_static epicode.internal 192.168.32.100</i></p> <p>questo permette di simulare il dns epicode.internal</p> <p>Quindi dalla macchina Windows7 da un browser scrivendo <i>epicode.internal</i> il risultato è la risposta del Server Web Kali</p> <p>Anche in questo caso con (con <i>CTRL+X</i>; <i>Y</i> e poi <i>INVIO</i>) si chiude il file, si accettano i cambiamenti con (<i>Y</i>) e <i>INVIO</i> per confermare tutto.</p>	
<p>Con il comando <i>sudo inetSim</i>, si avvia il web server, così facendo siamo pronti.</p>	
<p>Ad avvio eseguito avremo una schermata come quella affianco:</p>	

Adesso si dimostra la corretta esecuzione e la corretta chiama dalla macchina Windwos7 (192.168.32.101) verso il server web KALI (192.168.32.100). Chiamata su HTTPS

Dalla macchina Windows7 si può vedere come chiamando il dominio <https://epicode.internal/>, il Server Web (Kali) risponda correttamente.



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

Kali (Server Web): IP :192.168.32.100 MAC: 08:00:27:13:F1:7F

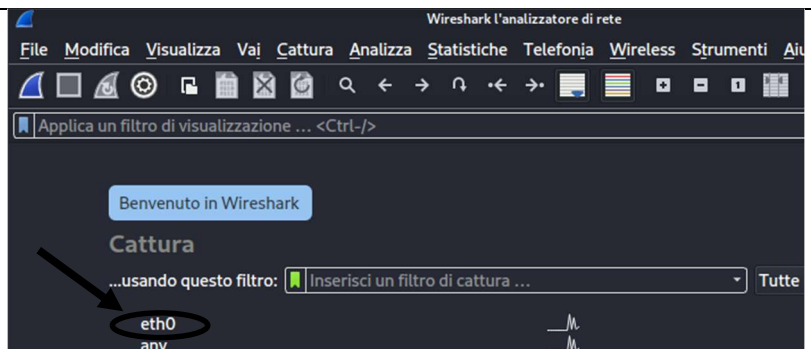
```
(kali@kali)~$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.32.100 netmask 255.255.255.0 broadcast 192.168.32.255
    inet6 fe80::a00:27ff:fe13:f17f prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:13:f1:7f txqueuelen 1000 (Ethernet)
```

Windows7 (Client): IP: 192.168.32.101 MAC: 08:00:27:AA:6B:B6

```
Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix  . : 
Description . . . . . : Intel(R) PRO/1000 MT Desktop Adapter
Physical Address. . . . . : 08-00-27-AA-6B-B6
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::4cf7:54f3:b2ee:6937%11(Preferred)
IPv4 Address. . . . . : 192.168.32.101(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.32.1
```

Da Wireshark si deve analizzare la scheda eth0.



Estratto Wireshark:

ip.addr == 192.168.32.101						
No.	Time	Source	Destination	Protocol	Length	Info
3	0.001728289	192.168.32.101	192.168.32.100	TCP	66	49169 → 443 [SYN] Seq=0 Win=8192
4	0.001751628	192.168.32.100	192.168.32.101	TCP	66	443 → 49169 [SYN, ACK] Seq=0 Ack=
5	0.002544318	192.168.32.101	192.168.32.100	TCP	60	49169 → 443 [ACK] Seq=1 Ack=1 Win
6	0.003438059	192.168.32.101	192.168.32.100	TLSv1	215	Client Hello
7	0.003449524	192.168.32.100	192.168.32.101	TCP	54	443 → 49169 [ACK] Seq=1 Ack=162 W
8	0.037774593	192.168.32.100	192.168.32.101	TLSv1	1373	Server Hello, Certificate, Server
9	0.049102191	192.168.32.101	192.168.32.100	TLSv1	188	Client Key Exchange, Change Cipe
10	0.049769693	192.168.32.100	192.168.32.101	TLSv1	113	Change Cipher Spec, Encrypted Han
12	0.250183743	192.168.32.101	192.168.32.100	TCP	60	49169 → 443 [ACK] Seq=296 Ack=137
16	3.217044874	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0x08fd A wpad
18	3.314916866	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0x08fd A wpad
19	3.516414931	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
20	4.266317240	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
21	5.016342567	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
28	8.913285789	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0xd0c4 A wpad
30	9.016541750	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0xd0c4 A wpad
31	9.219540359	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
32	9.969184782	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
33	10.719750133	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
34	12.473280900	192.168.32.101	192.168.32.100	TLSv1	363	Application Data
35	12.483815975	192.168.32.100	192.168.32.101	TLSv1	235	Application Data
36	12.485664130	192.168.32.100	192.168.32.101	TLSv1	384	Application Data, Encrypted Alert
37	12.486118369	192.168.32.101	192.168.32.100	TCP	60	49169 → 443 [ACK] Seq=605 Ack=189
38	12.486704807	192.168.32.101	192.168.32.100	TCP	60	49169 → 443 [FIN, ACK] Seq=605 Ac

- 1) **Riga No.3** - Si nota come l'ip sorgente **192.168.32.101** sorgente (Win7) fa una chiamata a tramite il protocollo TCP di tipo (SYN: avvia una connessione ti tipo TCP),
- 2) **Riga No.4** mentre l'indirizzo IP 192.168.32.100(Kali) riceve un pacchetto di tipo SYN sucssivamente risponde con un pacchetto (SYN, ACK: conferma la ricezione del pacchetto).
- 3) **Riga No.5** successivamente Wind7 conferma.
- 4) **Riga No.6** Adesso con il pacchetto TLSv1 con il messaggio "Client Hello" da conferma che la trasmissione è sicura. Adesso la pagina https è sicura e può essere trasmessa.

Qui sotto evidenziati la dimostrazione che la macchina Win7 sorgente comunica con la macchina Kali in modalità crittografata. Si vede come dal punto 6 il traffico sia su protocollo **TLS** quindi crittografato e fa capire che è una pagina **HTTPS**. Il contenuto non è leggibile senza la chiave per la decodificare il pacchetto,

Più in basso i Mac address delle due machina.

5	0.002544318	192.168.32.101	192.168.32.100	TCP	60	49169 → 443 [ACK] Seq=1 /
6	0.003438059	192.168.32.101	192.168.32.100	TLSv1	215	Client Hello
7	0.003449524	192.168.32.100	192.168.32.101	TCP	54	443 → 49169 [ACK] Seq=1 /
8	0.037774593	192.168.32.100	192.168.32.101	TLSv1	1373	Server Hello, Certificate
9	0.049102191	192.168.32.101	192.168.32.100	TLSv1	188	Client Key Exchange, Char
10	0.049769693	192.168.32.100	192.168.32.101	TLSv1	113	Change Cipher Spec, Encry
12	0.250183743	192.168.32.101	192.168.32.100	TCP	60	49169 → 443 [ACK] Seq=296
16	3.217044874	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0x08fd A v
18	3.314916866	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0x08fd A v
19	3.516414931	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
20	4.266317240	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
21	5.016342567	192.168.32.101	192.168.32.255	NBNS	92	Name query NB WPAD<00>
28	8.913285789	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0xd0c4 A v
30	9.016541750	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0xd0c4 A v

Frame 6: 215 bytes on wire (1720 bits), 215 bytes captured (1720 bits) on interface eth0, id 0	
Ethernet II, Src: PcsCompu_aa:6b:b6 (08:00:27:aa:6b:b6), Dst: PcsCompu_13:f1:7f (08:00:27:13:f1:7f)	
Destination: PcsCompu_13:f1:7f (08:00:27:13:f1:7f)	
Address: PcsCompu_13:f1:7f (08:00:27:13:f1:7f)	
.... 0. = LG bit: Globally unique address (factory default)	
.... 0. = IG bit: Individual address (unicast)	
Source: PcsCompu_aa:6b:b6 (08:00:27:aa:6b:b6)	
Address: PcsCompu_aa:6b:b6 (08:00:27:aa:6b:b6)	
.... 0. = LG bit: Globally unique address (factory default)	
.... 0. = IG bit: Individual address (unicast)	

Adesso il medesimo processo effettuato su una pagina HTTP rivela come le informazioni **non siano crittografate (HTTPS)**; l'intero traffico transita in chiaro attraverso la rete, consentendo una lettura agevole del contenuto.

La differenza tra i due teste eseguiti si nota come la pagina HTTPS utilizzi i protocolli di crittografia più utilizzati come TLSv1.2 o TLSv1.3 e SSL quest'ultimo meno utilizzato. Mentre HTTP non utilizzi nessun metodo per crittografare i pacchetti.

The image shows a network capture in Wireshark on the left and a web browser window on the right. The Wireshark packet list shows an HTTP GET request from 192.168.32.101 to 192.168.32.100. The packet details pane shows the Hypertext Transfer Protocol section with the following HTML content:

```
<html>\n<head>\n<title>INetSim default HTML page</title>\n</head>\n<body>\n<p>\n<p align="center">This is the default HTML page for INetSim HTTP server fake mode.\n</p>\n<p align="center">This file is an HTML document.</p>\n</body>\n</html>\n
```

The browser window on the right shows the default HTML page for INetSim HTTP server fake mode. A red arrow points from the HTML content in the Wireshark packet details to the text in the browser window.

In conclusione: Per garantire una maggiore sicurezza, è consigliabile evitare l'utilizzo di siti web non crittografati HTTP al fine di ridurre l'esposizione a potenziali rischi. È sempre opportuno verificare che le pagine visitate siano dotate del protocollo HTTPS per un livello aggiuntivo di sicurezza.

Cybersecurity Analyst 2023