

Consegna Modulo 1

Configurazione IP statici per le machine kali e windows

Kali:

dal terminale facciamo `sudo nano /etc/network/interfaces` per impostare l'ip statico:



The screenshot shows a terminal window titled 'kali@kali: ~'. The nano editor is open, editing the file '/etc/network/interfaces'. The menu bar includes 'File', 'Actions', 'Edit', 'View', and 'Help'. The status bar at the top indicates 'GNU nano 6.0' and the file path. The content of the file is as follows:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.32.100
netmask 255.255.255.0
network 192.168.32.0
broadcast 192.168.32.255
gateway 192.168.32.1
```

At the bottom of the terminal, there is a status bar with the following information:

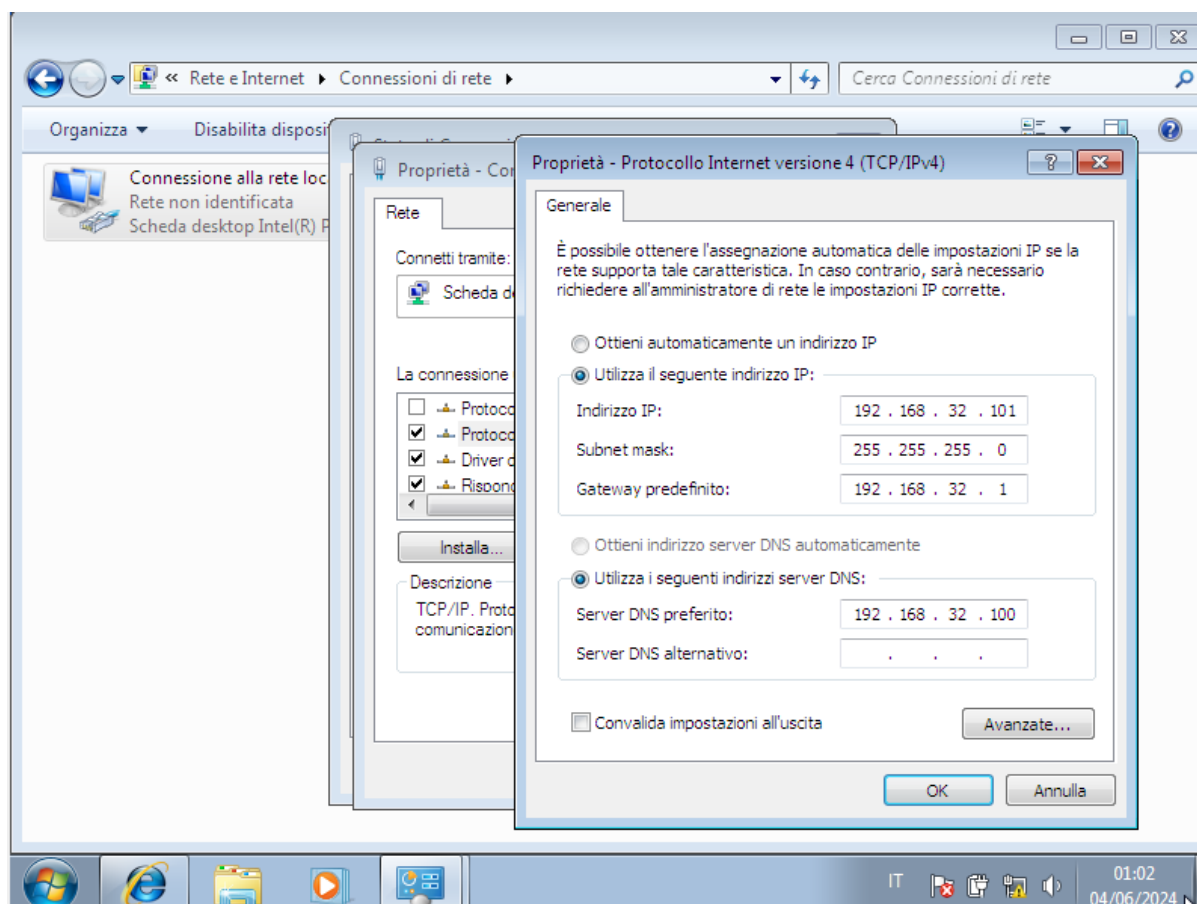
- Left side: `^G Help`, `^X Exit`, `^O Write Out`, `^R Read File`
- Center: `[Read 17 lines]`, `^W Where Is`, `^N Replace`
- Right side: `^K Cut`, `^U Paste`, `^T Execute`, `^J Justify`

Verifica della corretta configurazione con `ifconfig`

```
kali@kali: ~  
File Actions Edit View Help  
$ 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:fe1e:364a prefixlen 64 scopeid 0x20<link>  
    ether 08:00:27:1e:36:4a txqueuelen 1000 (Ethernet)  
    RX packets 0 bytes 0 (0.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 18 bytes 3143 (3.0 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
```

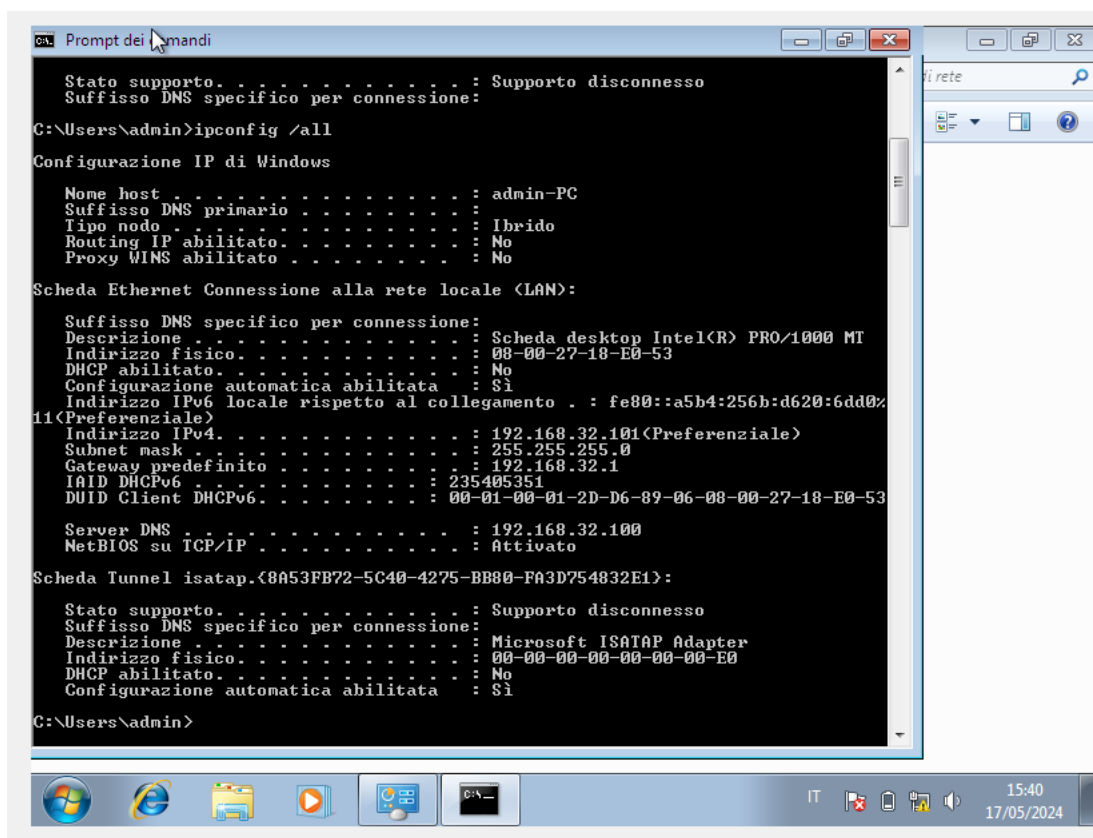
Windows

Seguendo il percorso visibile nell'immagine, si imposta l'ip statico come richiesto.

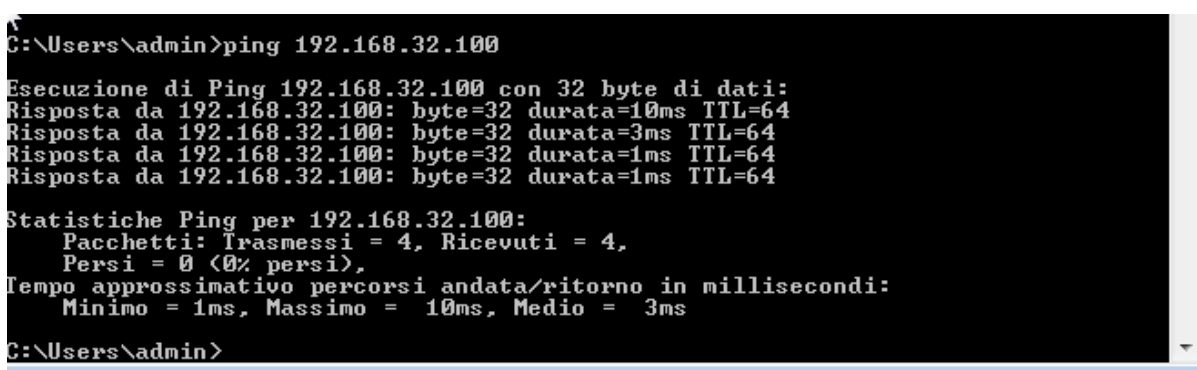


Verifica configurazione dal prompt dei comandi con il comando ifconfig.

Indirizzo MAC (Indirizzo fisico) 08-00-27-18-E0-53



Test ping verso la macchina kali (192.168.32.100)



Simulazione della rete

Per il Progetto si è deciso di usare **InetSim** che è una suite di software open source che simula servizi Internet comuni in un ambiente di laboratorio. È utilizzato per analizzare il comportamento di rete di campioni di malware sconosciuti.

Configurazione file Inetsim.conf nella cartella /etc/inetsim.

Vengono scommentati solo i servizi che ci servono (dns, http,https)



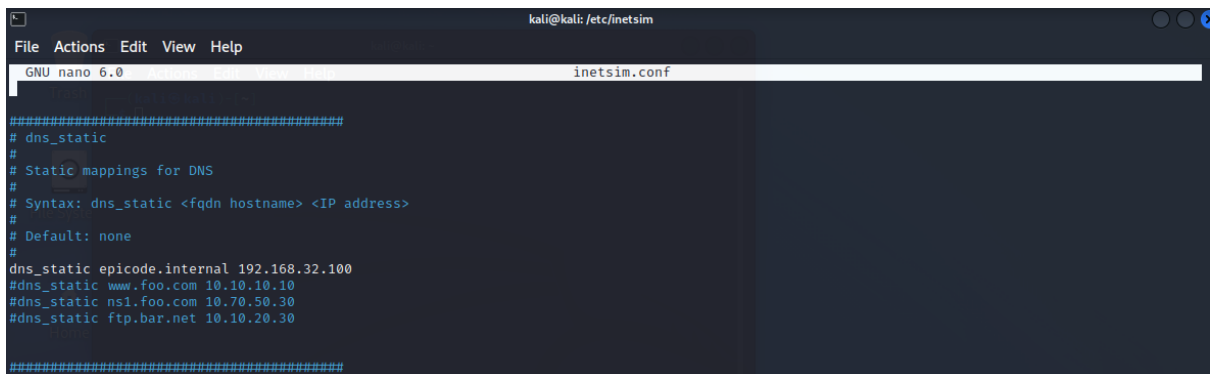
```
kali@kali: /etc/inetsim
File Actions Edit View Help
GNU nano 6.0 inetsim.conf
# quotd_udp, chargen_tcp, chargen_udp, finger,
# ident, syslog, dummy_tcp, dummy_udp, smtps, pop3s,
# ftps, irc, https
#
start_service dns
start_service http
start_service https
#start_service smtp
#start_service smtps
#start_service pop3
```

Service bind address impostata a 0.0.0.0



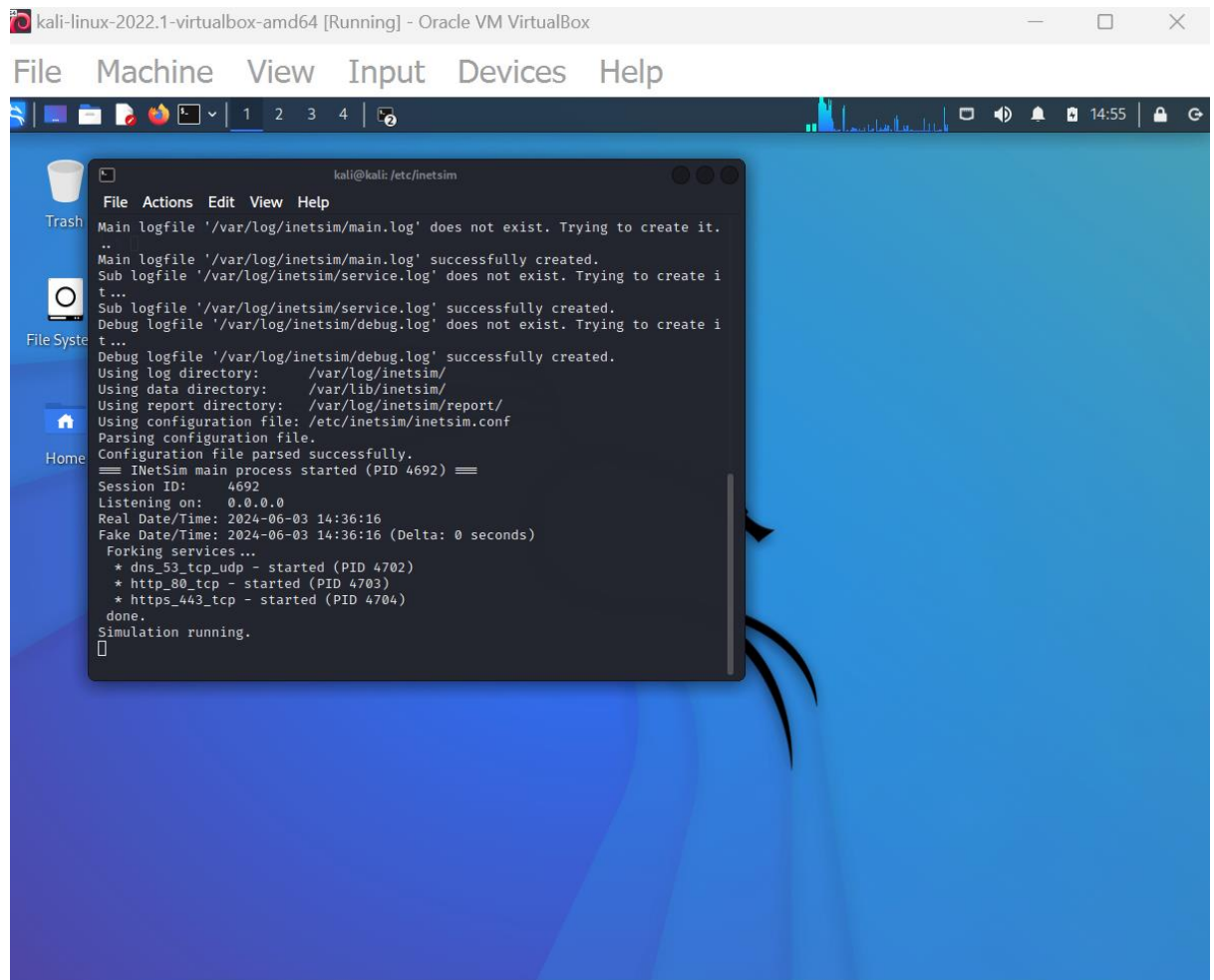
```
#####
# service_bind_address
#
# IP address to bind services to
#
# Syntax: service_bind_address <IP address>
#
# Default: 127.0.0.1
#
service_bind_address 0.0.0.0
|
```

Inserita la riga dns_static epicode.internal



```
kali@kali: /etc/inetsim
File Actions Edit View Help
GNU nano 6.0 inetsim.conf
#####
# dns_static
#
# Static mappings for DNS
#
# Syntax: dns_static <fqdn hostname> <IP address>
#
# Default: none
#
dns_static epicode.internal 192.168.32.100
#dns_static www.foo.com 10.10.10.10
#dns_static ns1.foo.com 10.70.50.30
#dns_static ftp.bar.net 10.10.20.30
#####
```

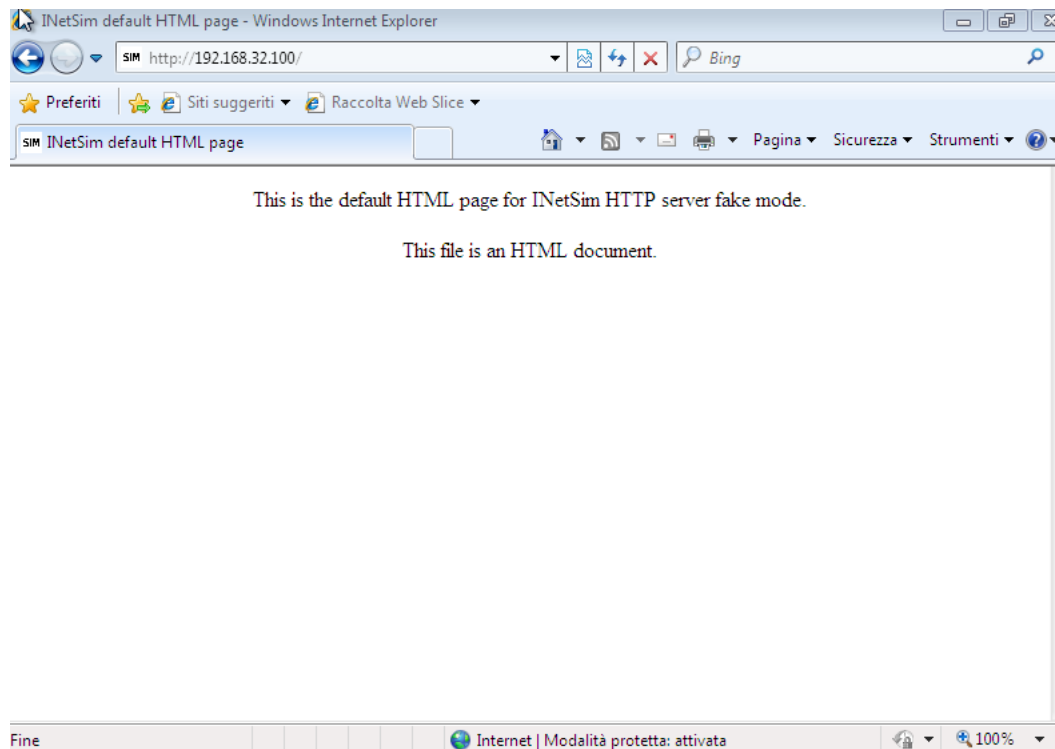
Avvio della simulazione con inetsim da terminale con comando sudo inetsim



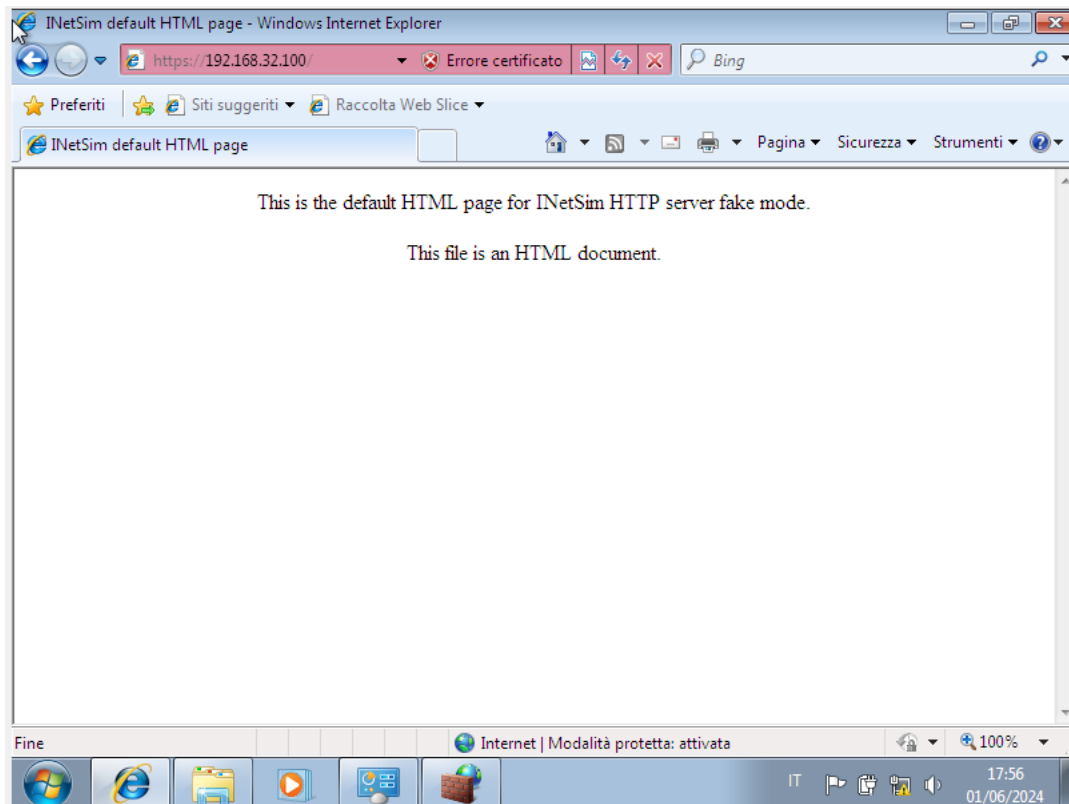
```
kali@kali: /etc/inetsim
File Actions Edit View Help
Main logfile '/var/log/inetsim/main.log' does not exist. Trying to create it.
..
Main logfile '/var/log/inetsim/main.log' successfully created.
Sub logfile '/var/log/inetsim/service.log' does not exist. Trying to create i
t...
Sub logfile '/var/log/inetsim/service.log' successfully created.
Debug logfile '/var/log/inetsim/debug.log' does not exist. Trying to create i
t...
Debug logfile '/var/log/inetsim/debug.log' successfully created.
Using log directory: /var/log/inetsim/
Using data directory: /var/lib/inetsim/
Using report directory: /var/log/inetsim/report/
Using configuration file: /etc/inetsim/inetsim.conf
Parsing configuration file.
Configuration file parsed successfully.
== INetSim main process started (PID 4692) ==
Session ID: 4692
Listening on: 0.0.0.0
Real Date/Time: 2024-06-03 14:36:16
Fake Date/Time: 2024-06-03 14:36:16 (Delta: 0 seconds)
Forking services ...
* dns_53_tcp_udp - started (PID 4702)
* http_80_tcp - started (PID 4703)
* https_443_tcp - started (PID 4704)
done.
Simulation running.
```

Sulla macchina virtuale windows, in internet explorer facciamo il test per http, https e dns

Isabelle Adjetej

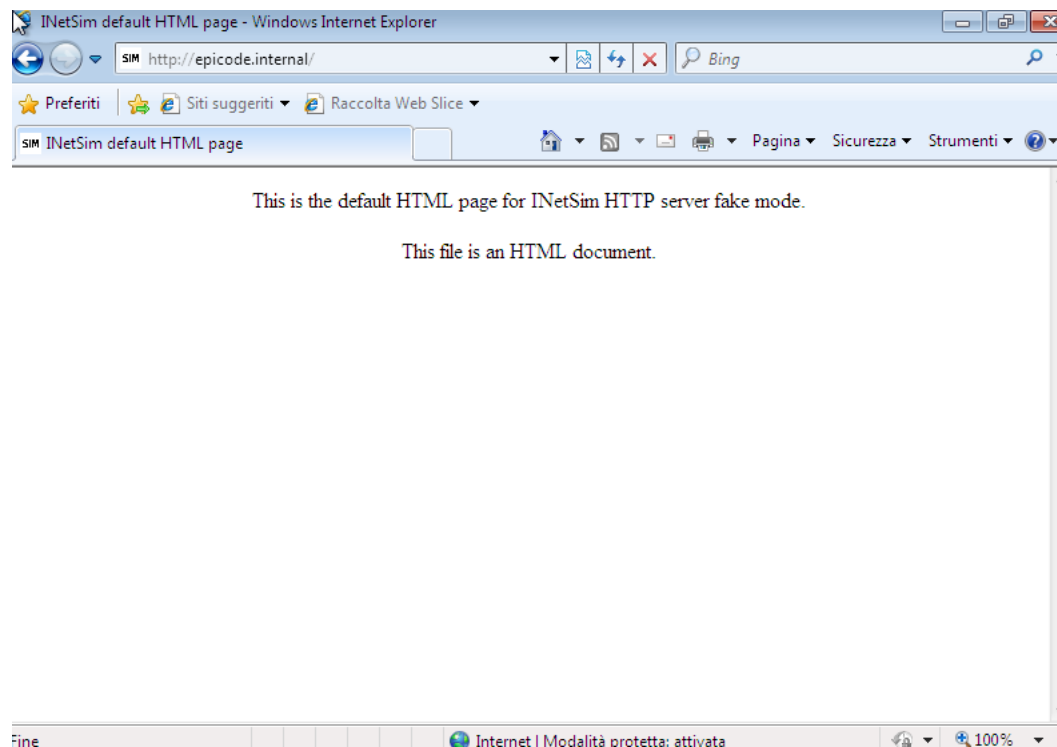


https



Isabelle Adjetey

DNS



Analisi della rete con Wireshark

HTTP

Le macchine iniziano a comunicare tramite l'indirizzo fisico. Viene ricercata la corrispondenza tra ip e mac address (08-00-27-18-E0-53 per la macchina windows 08:00:27:95:BD:54 per la KALI)

Isabelle Adjetejy

Apply a display filter ... <Ctrl-F>						
No.	Time	Source	Destination	Protocol	Length	Info
64	215.77595395	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
65	216.775516321	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
66	218.336117721	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.1007 Tell 192.168.32.101
67	218.336140727	PcsCompu_18:e0:53	Broadcast	ARP	42	192.168.32.100 is at 08:06:27:95:bd:54
68	218.344143324	192.168.32.101	192.168.32.100	DNS	76	Standard query 0x7c78 A go.microsoft.com
69	218.388477597	192.168.32.100	192.168.32.101	DNS	92	Standard query response 0x7c78 A go.microsoft.com A 127.0.0.1
70	218.422852690	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
71	218.753619187	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
72	219.277481649	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
73	220.279392444	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
74	221.571914788	192.168.32.101	192.168.32.100	DNS	72	Standard query 0x1491 A www.bing.com
75	221.575490845	192.168.32.100	192.168.32.101	DNS	88	Standard query response 0x1491 A www.bing.com A 127.0.0.1
76	223.394967866	PcsCompu_95:bd:54	PcsCompu_18:e0:53	ARP	42	Who has 192.168.32.1017 Tell 192.168.32.100
77	223.397558894	PcsCompu_18:e0:53	PcsCompu_95:bd:54	ARP	60	192.168.32.101 is at 08:06:27:18:e0:53
78	226.445468316	192.168.32.101	192.168.32.100	DNS	76	Standard query 0x5212 A epicode.internal
79	226.450572415	192.168.32.100	192.168.32.101	DNS	92	Standard query response 0x5212 A epicode.internal A 192.168.32.100
80	231.966507853	192.168.32.101	192.168.32.100	TCP	66	49223 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
81	231.966678428	192.168.32.100	192.168.32.101	TCP	66	80 → 49223 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
82	231.971191637	192.168.32.101	192.168.32.100	TCP	60	49223 → 80 [ACK] Seq=1 Ack=1 Win=65700 Len=0
83	231.975721508	192.168.32.101	192.168.32.100	HTTP	465	GET / HTTP/1.1
84	231.975771979	192.168.32.100	192.168.32.101	TCP	54	80 → 49223 [ACK] Seq=1 Ack=412 Win=64128 Len=0
85	232.023573815	192.168.32.100	192.168.32.101	TCP	284	80 → 49223 [PSH, ACK] Seq=1 Ack=412 Win=64128 Len=150 [TCP segment of a reassembled PDU]
86	232.031167185	192.168.32.100	192.168.32.101	HTTP	312	HTTP/1.1 200 OK (text/html)
87	232.033957344	192.168.32.101	192.168.32.100	TCP	60	49223 → 80 [ACK] Seq=412 Ack=410 Win=65292 Len=0
88	232.050807473	192.168.32.101	192.168.32.100	TCP	60	49223 → 80 [FIN, ACK] Seq=412 Ack=410 Win=65292 Len=0
89	232.0509523891	192.168.32.100	192.168.32.101	TCP	54	80 → 49223 [ACK] Seq=410 Ack=413 Win=64128 Len=0
90	232.178895661	192.168.32.101	192.168.32.100	TCP	66	49224 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
91	232.179682742	192.168.32.100	192.168.32.101	TCP	66	80 → 49224 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
92	232.181906409	192.168.32.101	192.168.32.100	TCP	60	49224 → 80 [ACK] Seq=1 Ack=1 Win=65700 Len=0
93	232.183715681	192.168.32.101	192.168.32.100	HTTP	341	GET /favicon.ico HTTP/1.1
94	232.183747889	192.168.32.100	192.168.32.101	TCP	54	80 → 49224 [ACK] Seq=1 Ack=288 Win=64128 Len=0
95	232.218918732	192.168.32.100	192.168.32.101	TCP	287	80 → 49224 [PSH, ACK] Seq=1 Ack=288 Win=64128 Len=153 [TCP segment of a reassembled PDU]
96	232.226726819	192.168.32.100	192.168.32.101	HTTP	252	HTTP/1.1 200 OK (image/x-icon)
97	232.230457660	192.168.32.101	192.168.32.100	TCP	60	49224 → 80 [ACK] Seq=288 Ack=353 Win=65348 Len=0
98	232.231377640	192.168.32.101	192.168.32.100	TCP	60	49224 → 80 [FIN, ACK] Seq=288 Ack=353 Win=65348 Len=0
99	232.231499418	192.168.32.100	192.168.32.101	TCP	54	80 → 49224 [ACK] Seq=353 Ack=289 Win=64128 Len=0

HTTPS

Apply a display filter ... <Ctrl-F>						
No.	Time	Source	Destination	Protocol	Length	Info
1	309.356377782	192.168.32.101	192.168.32.100	TLsv1	180	Client Hello
2	309.356464531	192.168.32.100	192.168.32.101	TCP	54	443 → 49163 [ACK] Seq=1 Ack=137 Win=64128 Len=0
173	309.361841863	192.168.32.100	192.168.32.101	TLsv1	1373	Server Hello, Certificate, Server Key Exchange, Server Hello Done
174	309.367974279	192.168.32.101	192.168.32.100	TLsv1	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
175	309.368086611	192.168.32.100	192.168.32.101	TCP	54	443 → 49163 [ACK] Seq=1320 Ack=271 Win=64128 Len=0
176	309.368892289	192.168.32.100	192.168.32.101	TLsv1	113	Change Cipher Spec, Encrypted Handshake Message
177	309.400331998	192.168.32.101	192.168.32.100	TCP	60	49163 → 443 [FIN, ACK] Seq=272 Ack=1370 Win=64128 Len=0
178	309.400445578	192.168.32.100	192.168.32.101	TLsv1	91	Encrypted Alert
179	309.400594592	192.168.32.100	192.168.32.101	TCP	54	443 → 49163 [FIN, ACK] Seq=272 Ack=1370 Win=64128 Len=0
180	309.407674395	192.168.32.101	192.168.32.100	TCP	60	49163 → 443 [RST, ACK] Seq=272 Ack=1416 Win=0 Len=0
181	309.408093163	192.168.32.101	192.168.32.100	TCP	60	49164 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
182	309.408093764	192.168.32.100	192.168.32.101	TCP	66	443 → 49164 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
183	309.410624324	192.168.32.101	192.168.32.100	TCP	60	49164 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
184	309.410759196	192.168.32.101	192.168.32.100	TLsv1	180	Client Hello
185	309.410790340	192.168.32.100	192.168.32.101	TCP	54	443 → 49164 [ACK] Seq=1 Ack=137 Win=64128 Len=0
186	309.420139185	192.168.32.100	192.168.32.101	TLsv1	1373	Server Hello, Certificate, Server Key Exchange, Server Hello Done
187	309.440755880	192.168.32.101	192.168.32.100	TLsv1	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
188	309.446793789	192.168.32.100	192.168.32.101	TCP	54	443 → 49164 [ACK] Seq=1320 Ack=271 Win=64128 Len=0
189	309.448181699	192.168.32.100	192.168.32.101	TLsv1	113	Change Cipher Spec, Encrypted Handshake Message
190	309.462338527	192.168.32.101	192.168.32.100	TLsv1	491	Application Data
191	309.462391841	192.168.32.100	192.168.32.101	TCP	54	443 → 49164 [ACK] Seq=1370 Ack=708 Win=64128 Len=0
192	309.501057182	192.168.32.100	192.168.32.101	TLsv1	295	Application Data
193	309.506558869	192.168.32.100	192.168.32.101	TLsv1	384	Application Data, Encrypted Alert
194	309.509917212	192.168.32.101	192.168.32.100	TCP	60	49164 → 443 [ACK] Seq=708 Ack=1891 Win=65700 Len=0
195	309.509917632	192.168.32.101	192.168.32.100	TCP	66	49164 → 443 [FIN, ACK] Seq=708 Ack=1891 Win=65700 Len=0
196	309.509972919	192.168.32.100	192.168.32.101	TCP	54	443 → 49164 [ACK] Seq=1891 Ack=709 Win=64128 Len=0
197	309.509982458	192.168.32.100	192.168.32.101	TCP	60	49165 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
198	309.509735467	192.168.32.100	192.168.32.101	TCP	66	443 → 49165 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
199	309.506722074	192.168.32.101	192.168.32.100	TCP	60	49165 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
200	309.601360602	192.168.32.101	192.168.32.100	TLsv1	156	Client Hello
201	309.601342220	192.168.32.100	192.168.32.101	TCP	54	443 → 49165 [ACK] Seq=1 Ack=185 Win=64256 Len=0
202	309.605338907	192.168.32.100	192.168.32.101	TLsv1	1373	Server Hello, Certificate, Server Key Exchange, Server Hello Done
203	309.619510246	192.168.32.101	192.168.32.100	TLsv1	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
204	309.619586888	192.168.32.100	192.168.32.101	TCP	54	443 → 49165 [ACK] Seq=1320 Ack=239 Win=64128 Len=0
205	309.620158966	192.168.32.100	192.168.32.101	TLsv1	113	Change Cipher Spec, Encrypted Handshake Message
206	309.639655293	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
207	309.822147196	192.168.32.101	192.168.32.100	TCP	60	49165 → 443 [ACK] Seq=239 Ack=1379 Win=64128 Len=0
208	310.399753729	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
209	311.998623728	PcsCompu_18:e0:53	Broadcast	ARP	60	Who has 192.168.32.17 Tell 192.168.32.101
210	312.745050768	192.168.32.101	224.0.0.252	LLMNR	64	Standard query 0x4406 A www

Differenze http e https

Avviene uno scambio 3 way handshake e i pacchetti scambiati sono leggibili mentre con HTTPS in più dello scambio 3 way handshake c'è il criptaggio con TLS.

Lo scambio avviene sulla porta 80 per http e 443 per https.

Il messaggio con http è intercettabile mentre non lo è con https.