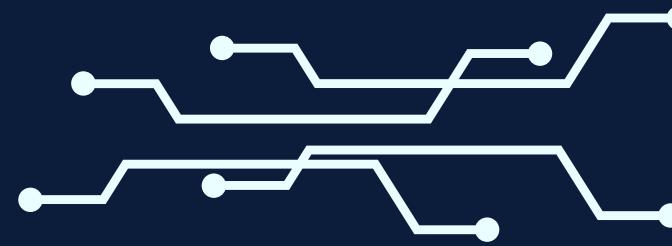
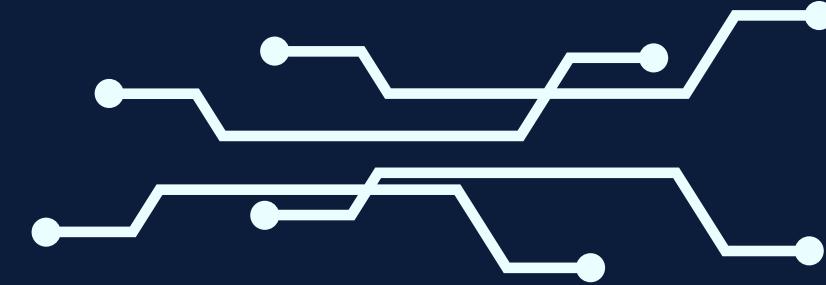


Esercitazione WHDH

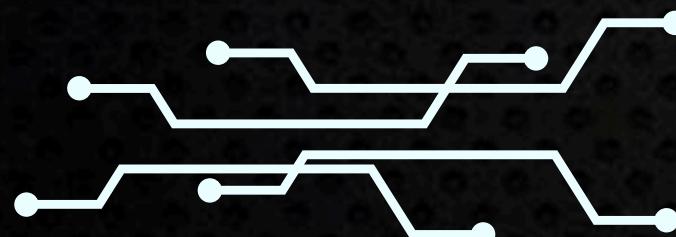
RICCARDO LEPORE



ESERCITAZIONE COMPRENSIVA DI TUTTE LE COMPETENZE
ACQUISITE



1. IMPOSTARE SERVIZI INETSIM



COME PRIMA COSA ANDIAMO A SETTARE INETSIM



kalilinux@kalilinux-20211:~

File Actions Edit View Help

GNU nano 5.4

/etc/inetsim/inetsim.conf

```
#####
# INetSim configuration file
#
#####

#####
# Main configuration
#####

#####
# 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
start_service https
#start_service smtp
#start_service smtps
#start_service pop3
#start_service pop3s
#start_service ftp
#start_service ftps
#start_service tftp
#start_service irc
#start_service ntp
```

[line 1/1999 (0%), col 1/62 (1%), char 0/41704 (0%)]

^G Help
^X Exit

^O Write Out
^R Read File

^W Where Is
^\\ Replace

^K Cut
^U Paste

^T Execute
^J Justify

^C Location
^_ Go To Line

M-U Undo
M-F Redo

M-A Set Mark
M-C Copy



kalilinux@kalilinux-20211:~

File Actions Edit View Help

GNU nano 5.4

```
#start_service quotd_tcp
#start_service quotd_udp
#start_service chargen_tcp
#start_service chargen_udp
#start_service dummy_tcp
#start_service dummy_udp
```

```
#####
# 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
```

```
#####
# 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
#
# Syntax: service_max_childs [1..30]
#
```

[line 91/1999 (4%), col 1/2 (50%), char 1980/41704 (4%)]

^{^G} Help
^{^X} Exit

^{^O} Write Out
^{^R} Read File

^{^W} Where Is
^{^V} Replace

^{^K} Cut

^{^U} Paste

^{^T} Execute
^{^J} Justify

^{^C} Location
^{^L} Go To Line
M-U Undo
M-P Redo

M-A Set Mark
M-S Copy

kalilinux@kalilinux-20211:~

File Actions Edit View Help

GNU nano 5.4 /etc/inetsim/inetsim.conf

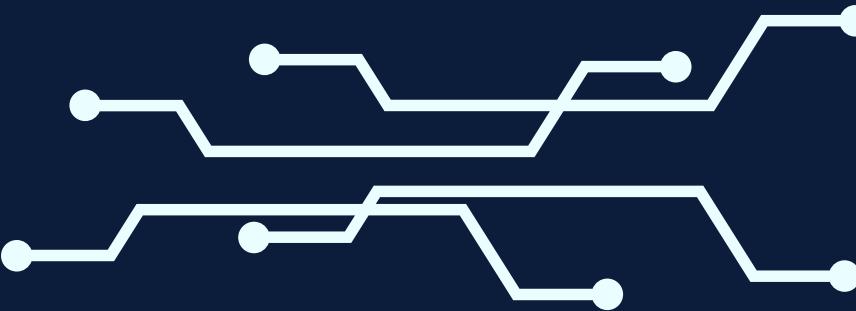
```
# Default: www
#
#dns_default_hostname somehost

#####
# dns_default_domainname
#
# Default domain name to return with DNS replies
#
# Syntax: dns_default_domainname <domain name>
#
# Default: inetsim.org
#
#dns_default_domainname some.domain

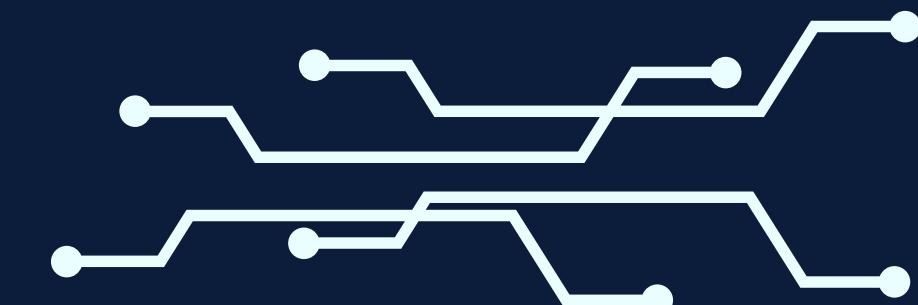
#####
# dns_static
#
# Static mappings for DNS
#
# Syntax: dns_static <fqdn hostname> <IP address>
#
# Default: none
#
#dns_static www.foo.com 10.10.10.10
#dns_static ns1.foo.com 10.70.50.30
#dns_static ftp.bar.net 10.10.20.30
dns_static epicode.internal 192.168.32.100

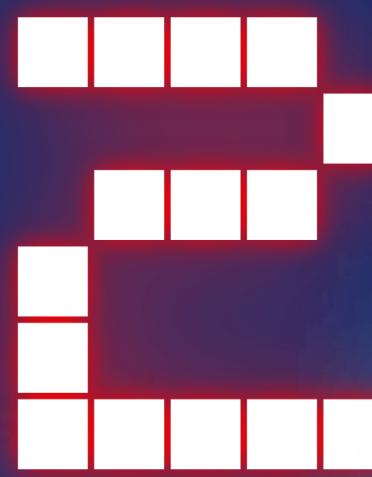
#####
# dns_version
#
# DNS version
#
# Syntax: dns_version <version>
#
# Default: "INetSim DNS Server"
#
[ line 217/1999 (10%), col 1/15 (6%), char 4545/41704 (10%) ]
```

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line M-F Redo M-S Copy



- 1 ATTIVIAMO SOLO I SERVIZI CHE CI SERVONO PER L'ESERCITAZIONE
- 2 SETTIAMO IL BIND ADDRESS SU 0.0.0.0 IN MODO CHE SIA IN ASCOLTO SU TUTTI GLI IP
- 3 SETTIAMO L'INDIRIZZO DI DNS STATICO E L'IP A CUI ASSOCIALO
- 4 DOPO SALVIAMO LA CONFIGURAZIONE CON CTRL O+INVIO

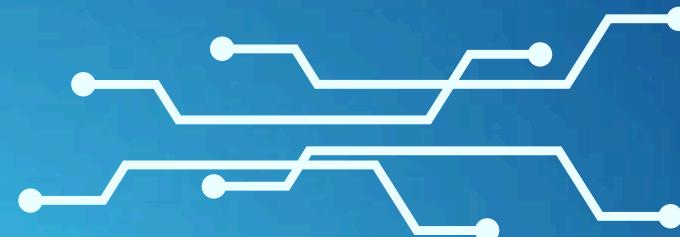




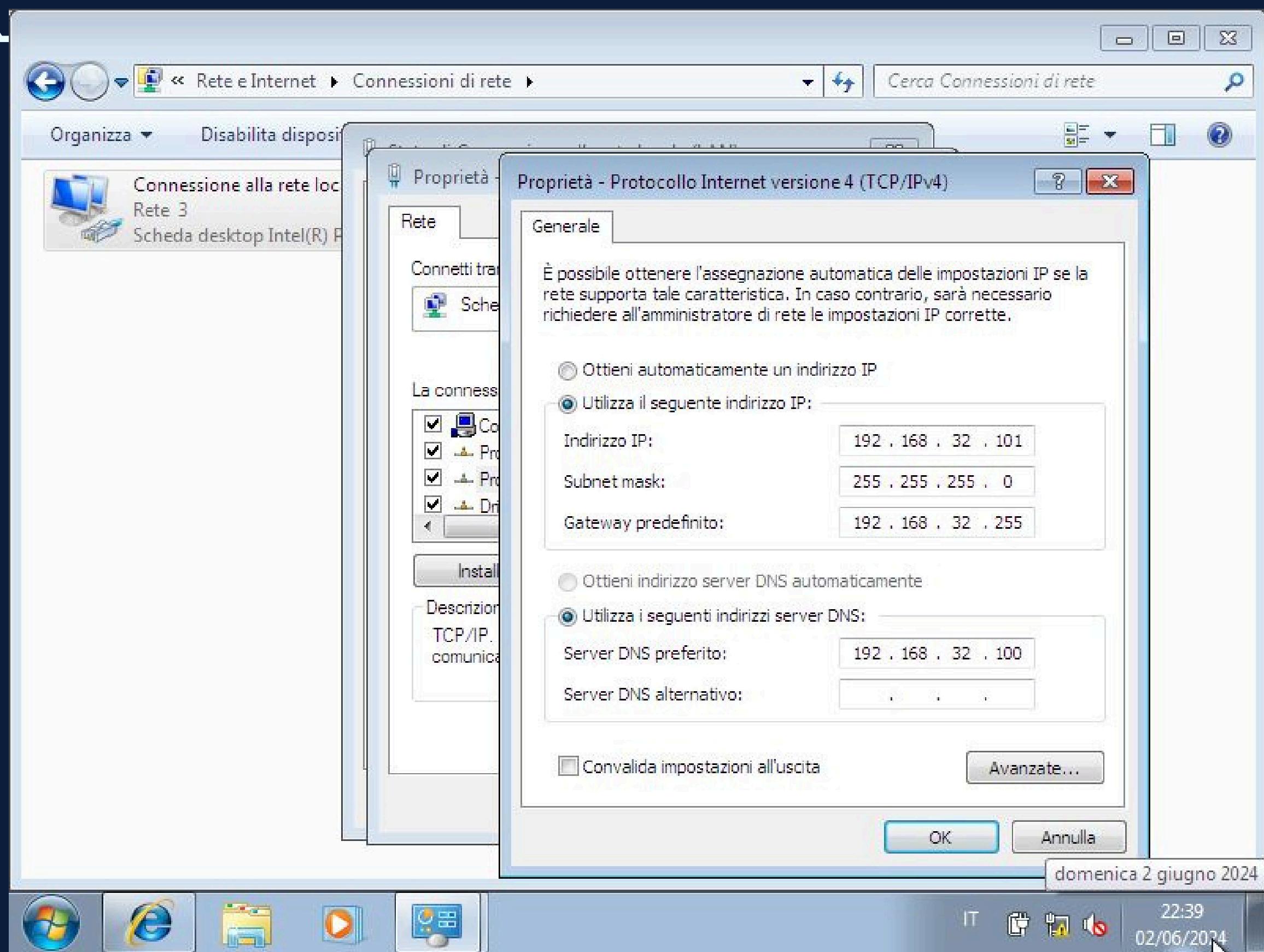
SETTING
WINDOWS 7

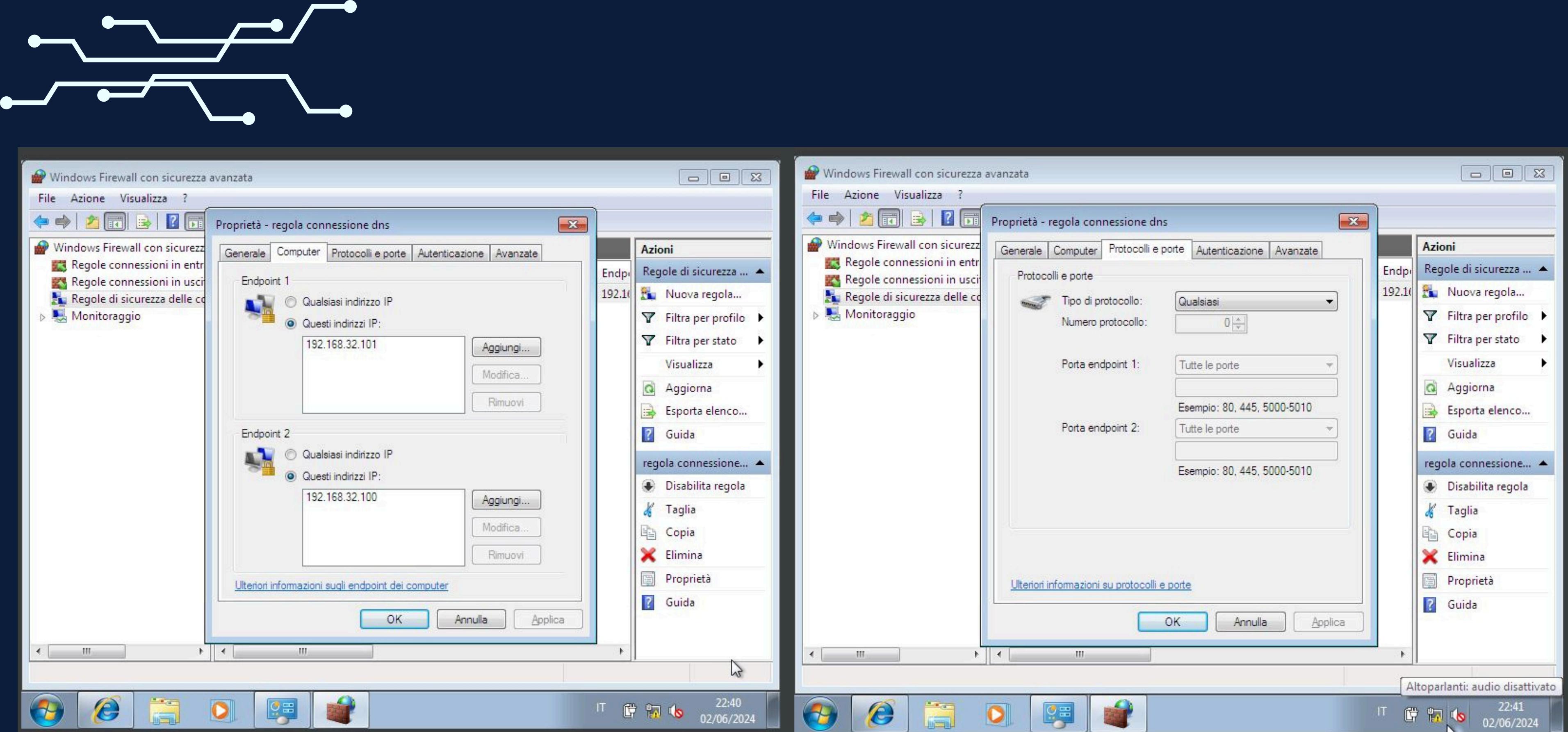


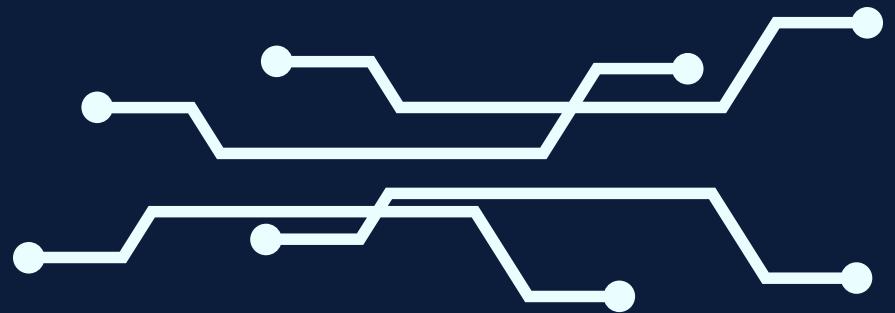
Windows 7™



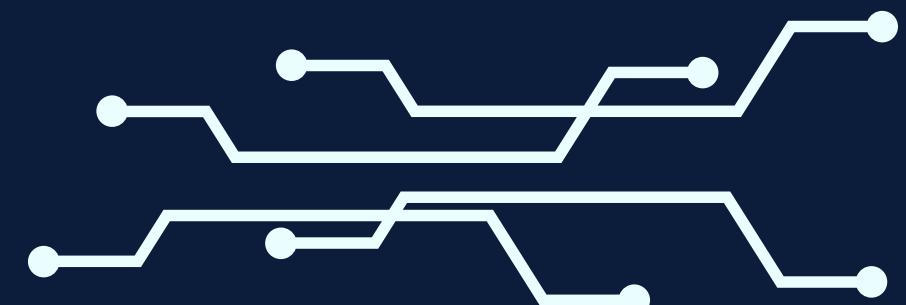
ORA ANDIAMO A SETTARE IL CLIENT



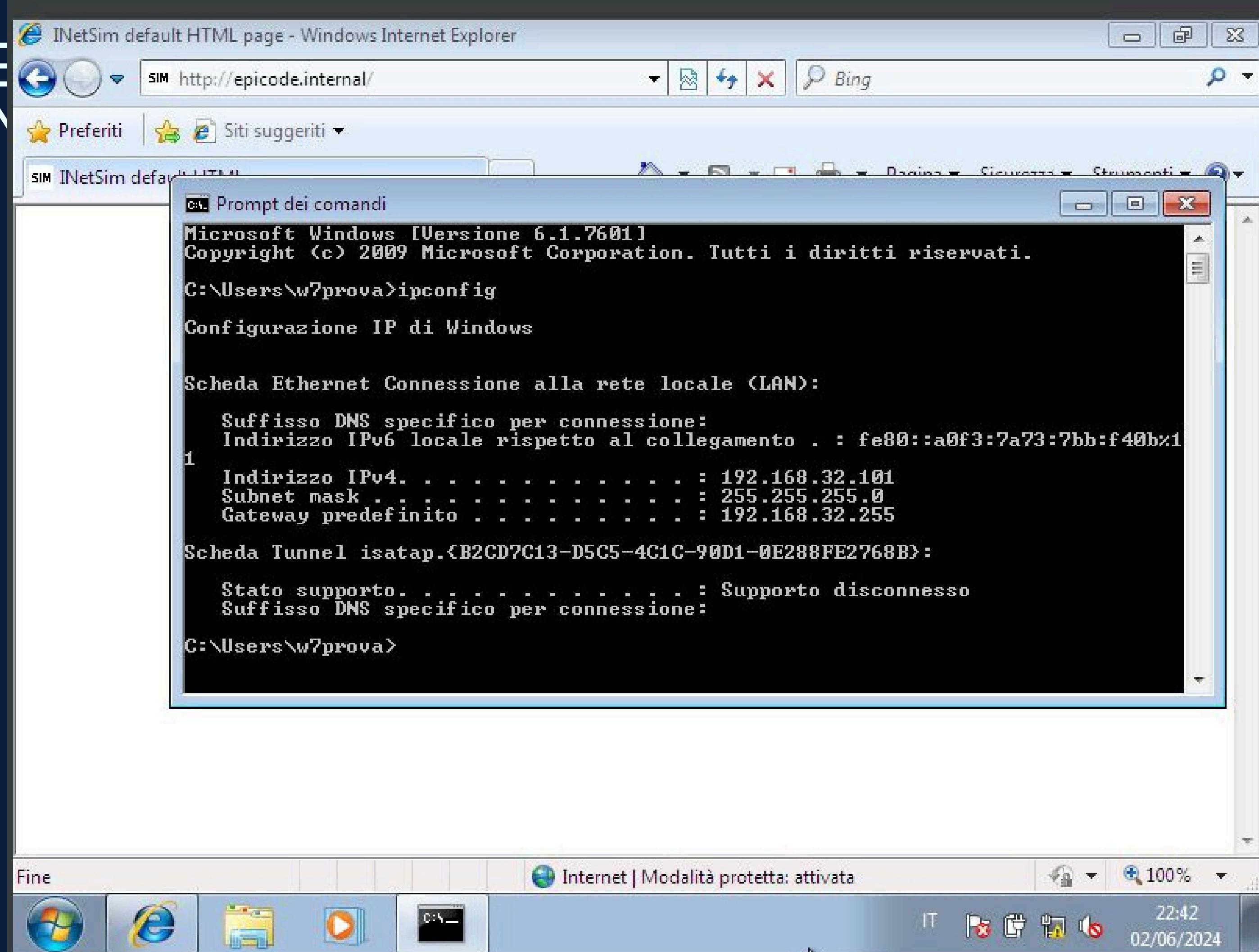




ANDIAMO A CREARE UNA NUOVA POLICY NEL FIREWALL DI
WINDOWS 7 PER CONSENTIRE LA CONNESSIONE TRA LE 2
MACCHINE VIRTUALI



DIGITAL
DATABASE
DATA GENERATION
DATA
SERVER DEBUGGING
WWW TECHNOLOGY
SEARCH
GLOBAL
INNOVATION
FEATURE
FOR
IP
ADDRESS
ADDRESS
WEBSITE
DYNAMIC
MONITOR
ACCESS
ID
PERSONAL
SERVICE
ONLINE
ADDRESS
NETWORKING
ADMINISTRATOR
ADMINISTRATION
ANONYMOUS
ENCRYPTION
BROADBAND
PROTOCOL
LAPTOP
GLOBE
HACKING
FIREWALL
NAVIGATION
PC
WIRELESS
NETWORK
PROXY
SCREEN
SECURITY
TRACKING
CONNECTION
NETWORK
WEB
OR A
ANDIAMO
A VEDERE
COME CAMBIANO I MAC
ADDRESS NELLE 2 MACCHINE VIRTUALI



C:\ Prompt dei comandi

Scheda Ethernet Connessione alla rete locale (LAN):

Suffisso DNS specifico per connessione:
Descrizione : Scheda desktop Intel(R) PRO/1000 MT
Indirizzo fisico : 08-00-27-3B-59-9F
DHCP abilitato : No
Configurazione automatica abilitata : Sì
Indirizzo IPv6 locale rispetto al collegamento : fe80::a0f3:7a73:7bb:f40b%1
1<Preferenziale>
Indirizzo IPv4 : 192.168.32.101<Preferenziale>
Subnet mask : 255.255.255.0
Gateway predefinito : 192.168.32.255
IAID DHCPv6 : 235405351
DUID Client DHCPv6 : 00-01-00-01-2D-AC-90-24-08-00-27-3B-59-9F

Server DNS : 192.168.32.100
NetBIOS su TCP/IP : Attivato

Scheda Tunnel isatap.{B2CD7C13-D5C5-4C1C-90D1-0E288FE2768B}:

Stato supporto : Supporto disconnesso
Suffisso DNS specifico per connessione:
Descrizione : Microsoft ISATAP Adapter
Indirizzo fisico : 00-00-00-00-00-00-E0
DHCP abilitato : No
Configurazione automatica abilitata : Sì

C:\Users\w7prova>

```
kalilinux@kalilinux-20211:~
```

File Actions Edit View Help

Parsing configuration file.

Configuration file parsed successfully.

== INetSim main process started (PID 1337) ==

Session ID: 1337

Listening on: 0.0.0.0

Real Date/Time: 2024-06-02 15:36:51

Fake Date/Time: 2024-06-02 15:36:51 (Delta: 0 seconds)

Forking services ...

- * dns_53_tcp_udp - started (PID 1341)
- * http_80_tcp - started (PID 1342)
- * https_443_tcp - started (PID 1343)

done.

Simulation running.

^{^C} * https_443_tcp - stopped (PID 1343)

- * http_80_tcp - stopped (PID 1342)
- * dns_53_tcp_udp - stopped (PID 1341)

Simulation stopped.

Report written to '/var/log/inetsim/report/report.1337.txt' (21 lines)

== INetSim main process stopped (PID 1337) ==

.

```
(kalilinux@kalilinux-20211)-[~]
```

```
$ /sbin/ifconfig
```

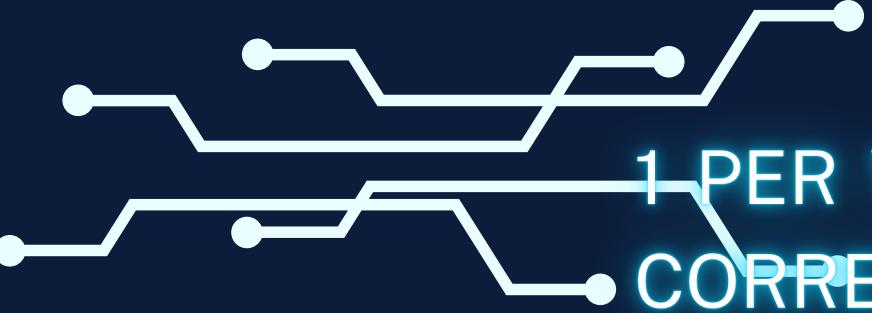
```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.32.100 netmask 255.255.255.0 broadcast 192.168.32.255
      inet6 fe80::e71d:693:8a05:8ecd prefixlen 64 scopeid 0x20<link>
        ether 08:00:27:f9:c9:21 txqueuelen 1000 (Ethernet)
          RX packets 86 bytes 10965 (10.7 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 36 bytes 3412 (3.3 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 400 (400.0 B)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 8 bytes 400 (400.0 B)
          TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
(kalilinux@kalilinux-20211)-[~]
```

```
$
```



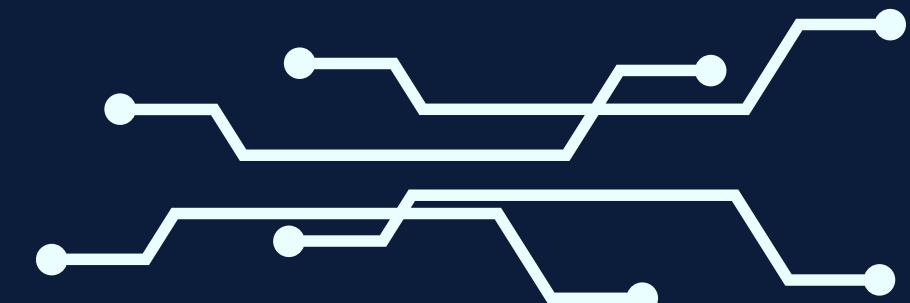


1 PER VERIFICARE DI AVER INSERITO TUTTI GLI INDIRIZZI CORRETTAMENTE SU WINDOWS 7 DIGITIAMO IL COMANDO IPCONFIG SUL PROMPT DEI COMANDI SUCCESSIVAMENTE PER VERIFICARE IL MAC ADDRESS (L'INDIRIZZO UNIVOCO DELLA SCHEDA DI RETE DELLA MACCHINA VIRTUALE) ANDREMO A DIGITARE IPCONFIG/ALL E CI MOSTRERA' ANCHE IL NOSTRO INDIRIZZO MAC CHE CORRISPONDE A:

08--00--27--3B--59--9F

SU KALI INVECE ANDREMO SUL TERMINALE E DIGITEREMO /sbin/ifconfig E CI MOSTRERA' ANCHE IL MAC ADDRESS CHE CORRISPONDE A:

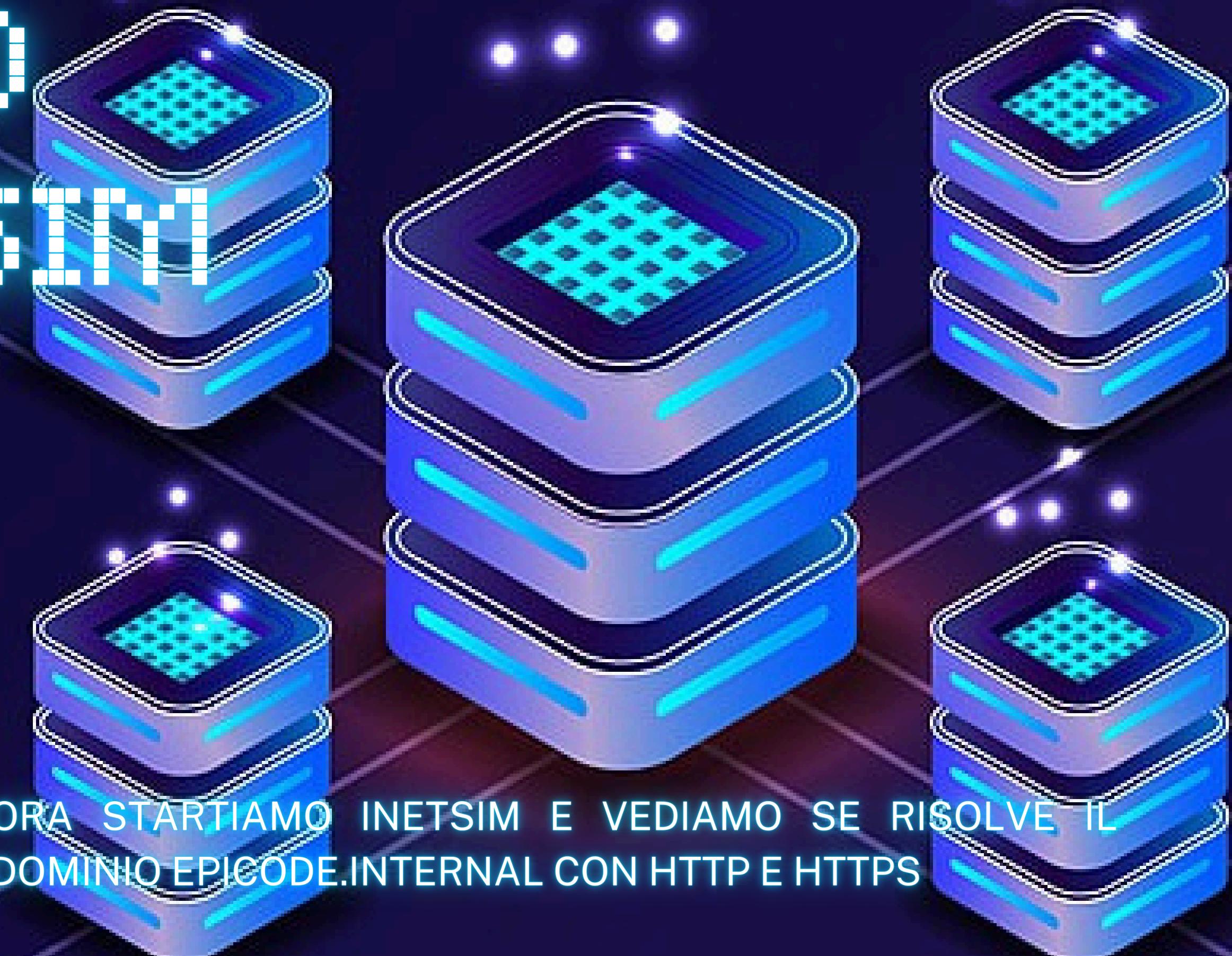
08--00--27--F9--C9:21



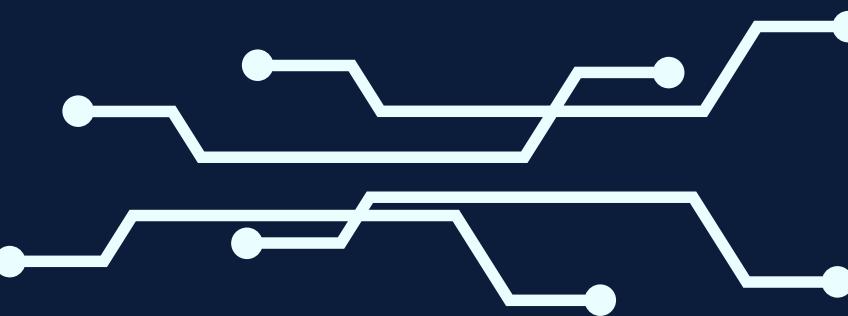
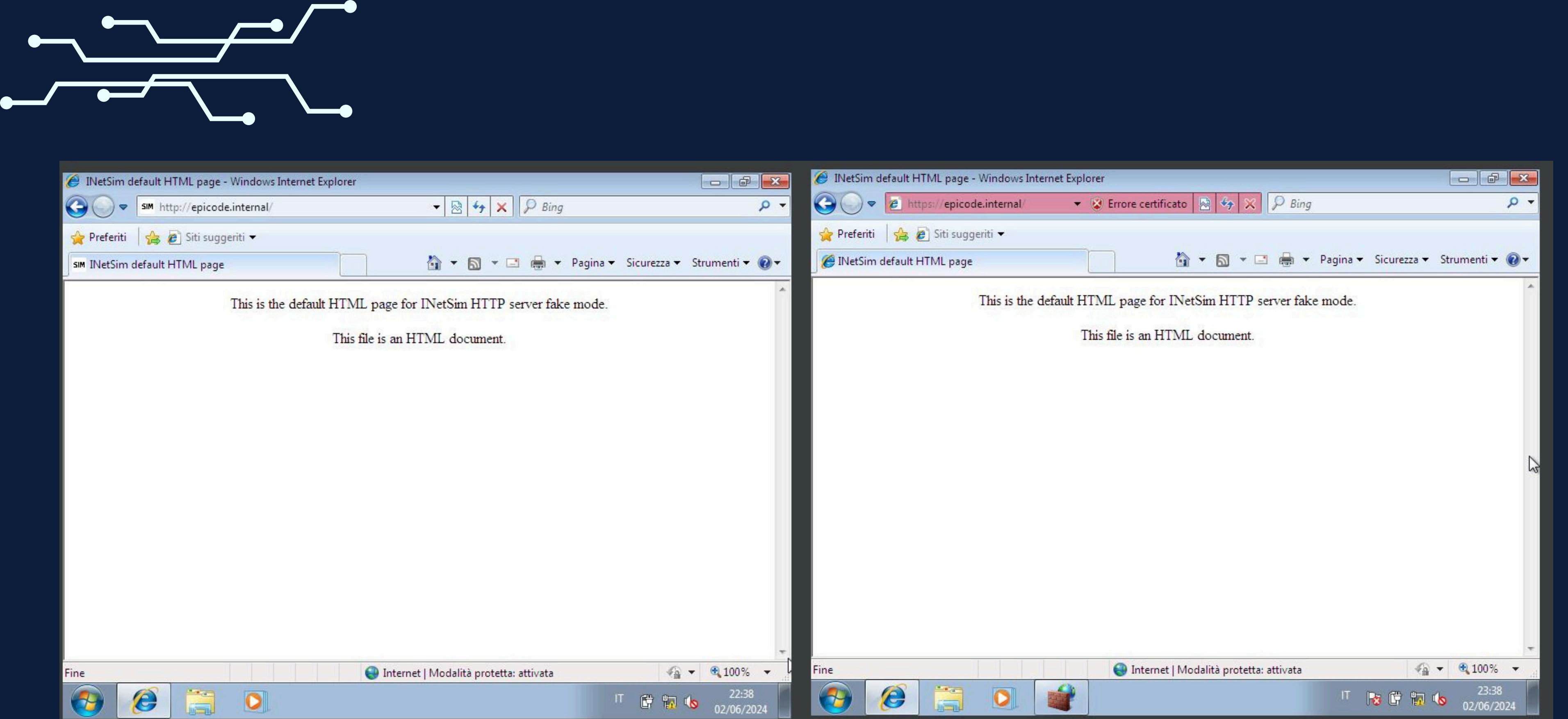
AVVICO INETSIM

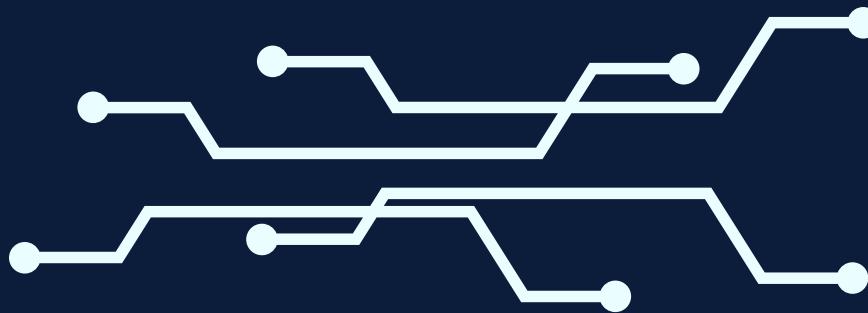


ORA STARTIAMO INETSIM E VEDIAMO SE RISOLVE IL
DOMINIO EPICODE.INTERNAL CON HTTP E HTTPS

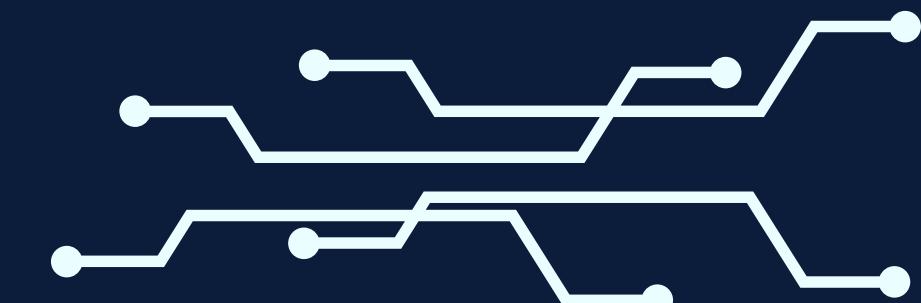


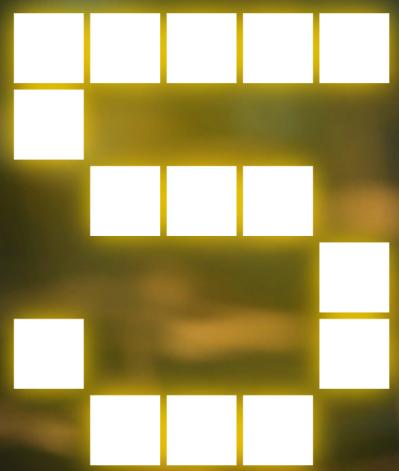
```
File Actions Edit View Help  
kalilinux@kalilinux-20211:~  
└─(kalilinux@kalilinux-20211)-[~]  
└─$ sudo nano -c /etc/inetsim/inetsim.conf  
└─(kalilinux@kalilinux-20211)-[~]  
└─$ sudo nano -c /etc/inetsim/inetsim.conf  
└─(kalilinux@kalilinux-20211)-[~]  
└─$ sudo inetsim  
INetSim 1.3.2 (2020-05-19) by Matthias Eckert & Thomas Hungenberg  
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 1337) ≡≡  
Session ID:      1337  
Listening on:    0.0.0.0  
Real Date/Time: 2024-06-02 15:36:51  
Fake Date/Time: 2024-06-02 15:36:51 (Delta: 0 seconds)  
Forking services ...  
* dns_53_tcp_udp - started (PID 1341)  
* http_80_tcp - started (PID 1342)  
* https_443_tcp - started (PID 1343)  
done.  
Simulation running.
```





SU KALI LANCIAMO IL COMANDO “SUDO INETSIM” PER AVVIARE LA SIMULAZIONE .
ORA APRIAMO IL BROWSER DI WINDOWS 7 E NELLA BARRA DI RICERCA DIGITIAMO HTTP/HTTPS “EPICODE.INTERNAL.
NOTEREMO CHE DIGITANDO HTTPS IL BROWSER CI AVVERTIRA’ CHE C’E’ UN CERTIFICATO DI SICUREZZA MANCANTE





WIRESHARK

WIRESHARK



ORA ANDIAMO A TRACCIARE I PACCHETTI HTTPS E HTTP
CON WIRESHARK

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.32.101	192.168.32.100	TCP	68	49186 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
2	0.000026386	192.168.32.100	192.168.32.101	TCP	68	80 → 49186 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
3	0.000038534	192.168.32.101	192.168.32.100	TCP	62	49186 → 80 [ACK] Seq=1 Ack=1 Win=65700 Len=0
4	0.000038588	192.168.32.101	192.168.32.100	HTTP	356	GET / HTTP/1.1
5	0.0000901332	192.168.32.100	192.168.32.101	TCP	56	80 → 49186 [ACK] Seq=1 Ack=301 Win=64128 Len=0
6	0.010615421	192.168.32.100	192.168.32.101	TCP	206	80 → 49186 [PSH, ACK] Seq=1 Ack=301 Win=64128 Len=150 [TCP segment of a reassembled PDU]
7	0.011794545	192.168.32.100	192.168.32.101	HTTP	314	HTTP/1.1 200 OK (text/html)
8	0.012527389	192.168.32.101	192.168.32.100	TCP	62	49186 → 80 [ACK] Seq=301 Ack=410 Win=65292 Len=0
9	0.012527421	192.168.32.101	192.168.32.100	TCP	62	49186 → 80 [FIN, ACK] Seq=301 Ack=410 Win=65292 Len=0
10	0.012547198	192.168.32.100	192.168.32.101	TCP	56	80 → 49186 [ACK] Seq=410 Ack=302 Win=64128 Len=0

```

Frame 7: 314 bytes on wire (2512 bits), 314 bytes captured (2512 bits) on interface any, id 0
Linux cooked capture v1
Internet Protocol Version 4, Src: 192.168.32.100, Dst: 192.168.32.101
Transmission Control Protocol, Src Port: 80, Dst Port: 49186, Seq: 151, Ack: 301, Len: 258
[2 Reassembled TCP Segments (408 bytes): #6(150), #7(258)]
Hypertext Transfer Protocol
Line-based text data: text/html (10 lines)

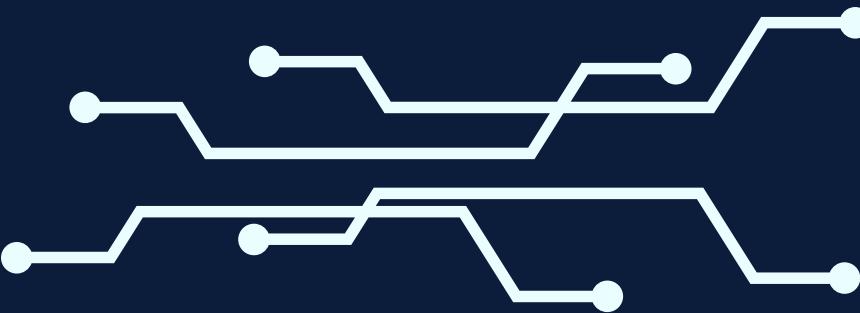
0000  00 04 00 01 00 06 08 00  27 f9 45 9e 00 00 08 00  .... ' E ...
0010  45 00 01 2a 38 c5 40 00  40 06 3e ef c0 a8 20 64  E .. *8 @ @ > . d
0020  c0 a8 20 65 00 50 c0 22  50 91 e9 ee 1e 7c dd c5  .. e P " P . . | ..
0030  50 19 01 f5 c3 36 00 00  3c 68 74 6d 6c 3e 0a 20  P . . 6 . <html>
0040  20 3c 68 65 61 64 3e 0a  20 20 20 20 3c 74 69 74  <head> . <tit
0050  6c 65 3e 49 4e 65 74 53  69 6d 20 64 65 66 61 75  le>INetS im defau
0060  6c 74 20 48 54 4d 4c 20  70 61 67 65 3c 2f 74 69  lt HTML page</ti
0070  74 6c 65 3e 0a 20 20 3c  2f 68 65 61 64 3e 0a 20  tle> . </head>
0080  20 3c 62 6f 64 79 3e 0a  20 20 20 20 3c 70 3e 3c  <body> . <p><
0090  2f 70 3e 0a 20 20 20 20  3c 70 20 61 6c 69 67 6e  /p> . <p align
00a0  3d 22 63 65 6e 74 65 72  22 3e 54 68 69 73 20 69  = "center " > This i
00b0  73 20 74 68 65 20 64 65  66 61 75 6c 74 20 48 54  s the de fault HT

```

Frame (314 bytes) Reassembled TCP (408 bytes)

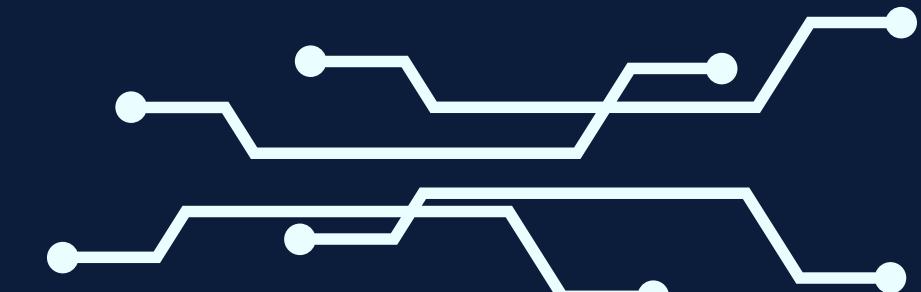
Packets: 10 · Displayed: 10 (100.0%)

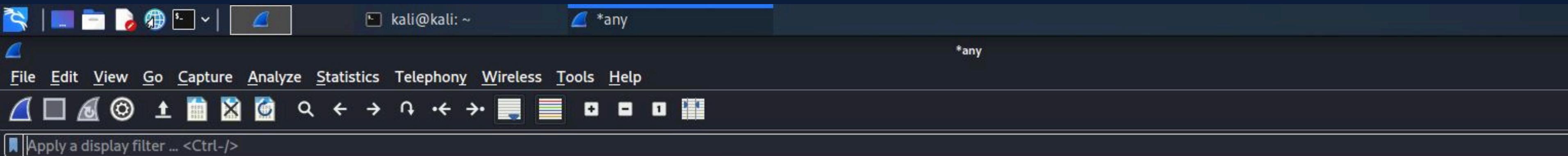
Profile: Default



PRIMA DI EFFETTUARE LA RICERCA EPICODE.INTERNAL
BISOGNERÀ APRERE WIRESHARK ED IMPOSTARLO SU
ANY.

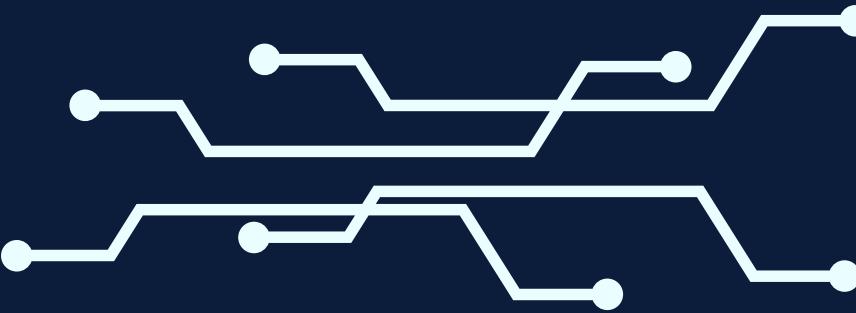
UNA VOLTA FATTO POSSIAMO RICHIEDERE
EPICODE.INTERNAL (HTTP) DAL BROWSER DI WINDOWS 7.
DA WIRESHARK POSSIAMO NOTARE CHE IL DISPOSITIVO
CHE VUOLE RICEVERE IL PACCHETTO è IL 192.168.32.101
MENTRE CHI LO SPEDISCE E' IL 192.168.32.100.
IL SERVER RESTITUISCE LA PAGINA CORRETTAMENTE
MOSTRANDO UN "GET" E UN "200 OK"





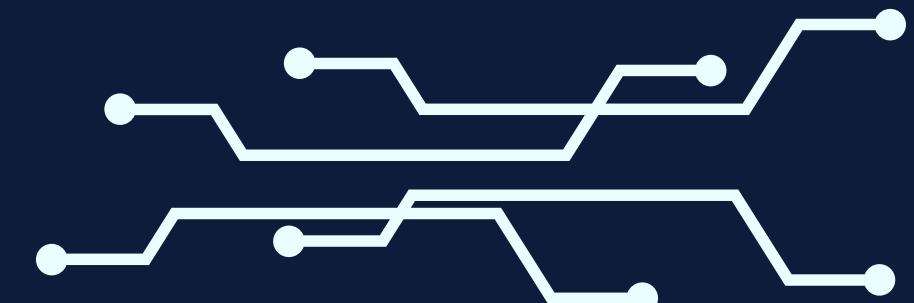
No.	Time	Source	Destination	Protocol	Length	Info
54	13.421249706	192.168.32.100	192.168.32.101	TCP	68	443 → 49166 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
55	13.421704171	192.168.32.101	192.168.32.100	TCP	62	49166 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
56	13.422742989	192.168.32.101	192.168.32.100	TLSv1	185	Client Hello
57	13.422749770	192.168.32.100	192.168.32.101	TCP	56	443 → 49166 [ACK] Seq=1 Ack=130 Win=64128 Len=0
58	13.425559788	192.168.32.100	192.168.32.101	TLSv1	1375	Server Hello, Certificate, Server Key Exchange, Server Hello Done
59	13.432835604	192.168.32.101	192.168.32.100	TLSv1	190	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
60	13.432845965	192.168.32.100	192.168.32.101	TCP	56	443 → 49166 [ACK] Seq=1320 Ack=264 Win=64128 Len=0
61	13.433225604	192.168.32.100	192.168.32.101	TLSv1	115	Change Cipher Spec, Encrypted Handshake Message
62	13.635504362	192.168.32.101	192.168.32.100	TCP	62	49166 → 443 [ACK] Seq=264 Ack=1379 Win=64320 Len=0
63	14.475152679	192.168.32.101	192.168.32.100	TCP	62	49166 → 443 [FIN, ACK] Seq=264 Ack=1379 Win=64320 Len=0
64	14.476258220	192.168.32.100	192.168.32.101	TLSv1	93	Encrypted Alert
65	14.476843700	192.168.32.101	192.168.32.100	TCP	62	49166 → 443 [RST, ACK] Seq=265 Ack=1416 Win=0 Len=0

```
► Frame 1: 87 bytes on wire (696 bits), 87 bytes captured (696 bits) on interface any, id 0
  ► Linux cooked capture v1
  ► Internet Protocol Version 4, Src: 192.168.32.101, Dst: 192.168.32.100
  ► User Datagram Protocol, Src Port: 61066, Dst Port: 53
  ► Domain Name System (query)
```



LA PROCEDURA PER IMPOSTARE WIRESHARK E' LA STESSA DI PRIMA.

IN QUESTO CASO E' POSSIBILE VEDERE CHE LA CHIAMATA E' CRIPTATA POICHE' DOPO IL THREE-WAY-HEADSHAKE VENGONO SCAMBIATE LE CHIAVI DI CRIPTAZIONE E DECRYPTAZIONE, QUESTO CI FA CAPIRE CHE LA RICHIESTA E' DI TIPO HTTPS





FINE PRESENTAZIONE CYBERPUNK

RICCARDO LEPORE