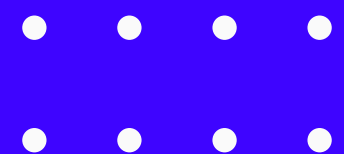


Simulazione architettura client-server



ERNESTO ROBLES



Dispositivi coinvolti:

Kali Linux (dns server) - IP 192.168.32.100/24

Il Dns server Kali Linux ha come servizi attivi DNS, Http e Https.

Windows pc (client) - IP 192.168.32.101/24

Software WireShark utilizza lo stesso canale di comunicazione condiviso tra più utenti consentendo di esaminare il contenuto di tutti i pacchetti dati in transito.

Scopo della simulazione:

Data una richiesta di un specifico servizio internet es: sito web (www.epicode.internal.com), dal client ad un server quest'ultimo risponderà al client mediante l'uso del servizio DNS il quale associa ad un nome ovvero un dominio un indirizzo IP che risponde al servizio richiesto, in questo caso la pagina web.

LABORATORIO VM

NEL LABORATORIO VM

ATTIVIAMO ALL'INTERNO DELLA MACCHINA KALI I SERVIZI DNS, HTTP E HTTPS AVVALENDOCI DI INETSIM CHE È UN SOFTWARE CHE PERMETTE DI EMULARE SERVIZI INTERNET.

CON IL COMANDO “SUDO NANO /ETC/INETSIM/INETSIM.CONF”

ACCEDIAMO ALL'INTERFACCIA DEI SERVIZI NELLA QUALE ATTIVIAMO I SERVIZI DESIDERATI SOPRACITATI NELLE IMPOSTAZIONI DEL SERVIZIO DNS MODIFICHIAMO LE VOCI:

DNS DEFAULTIP INSERENDO 192.168.32.100

DNS DEFAULT INSERENDONAME

WWW.EPICODE.INTERNAL

DNS STATIC INSERENDO

WWW.EPICODE.INTERNAL.COM 192.162.32.100

```
# 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
```

```
# Default: 192.168.32.100
#
dns_default_ip 192.168.32.100

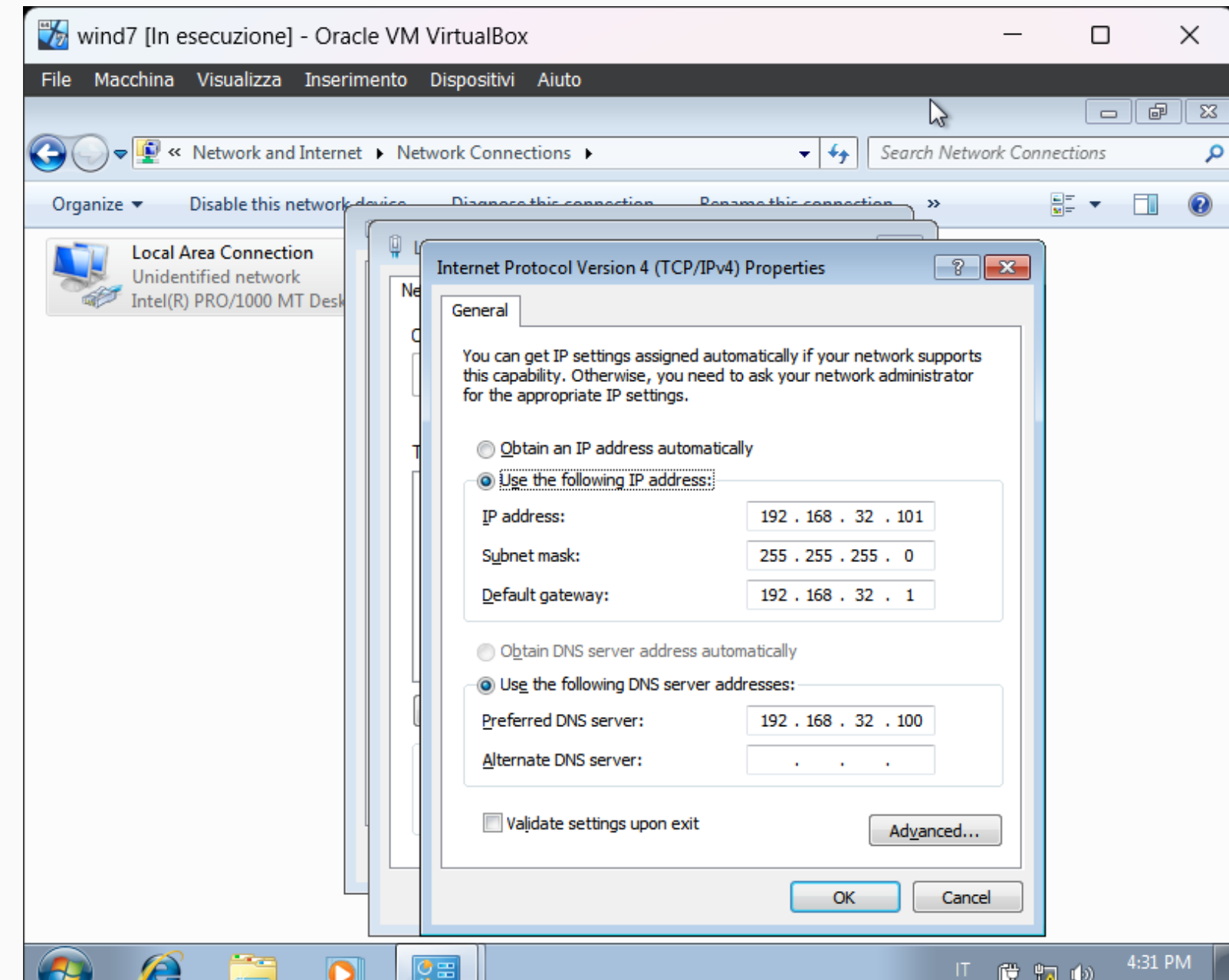
#####
# dns_default_hostname
#
# Default hostname to return with DNS replies
#
# Syntax: dns_default_hostname <hostname>
#
# Default: www.epicode.internal.com
#
#dns_default_hostname www.epicode.internal.com

#####
# dns_default_domainname
#
# Default domain name to return with DNS replies
#
# Syntax: dns_default_domainname <domain name>
#
# Default: epicode.internal
#
dns_default_domainname www.epicode.internal.com

#####
# dns_static
#
# Static mappings for DNS
#
# Syntax: dns_static <fqdn hostname> <IP address>
#
# Default: none
#
dns_static www.epicode.internal.com 192.168.32.100
#dns_static ns1.foo.com 10.70.50.30
#dns_static ftp.bar.net 10.10.20.20
```

LABORATORIO VM

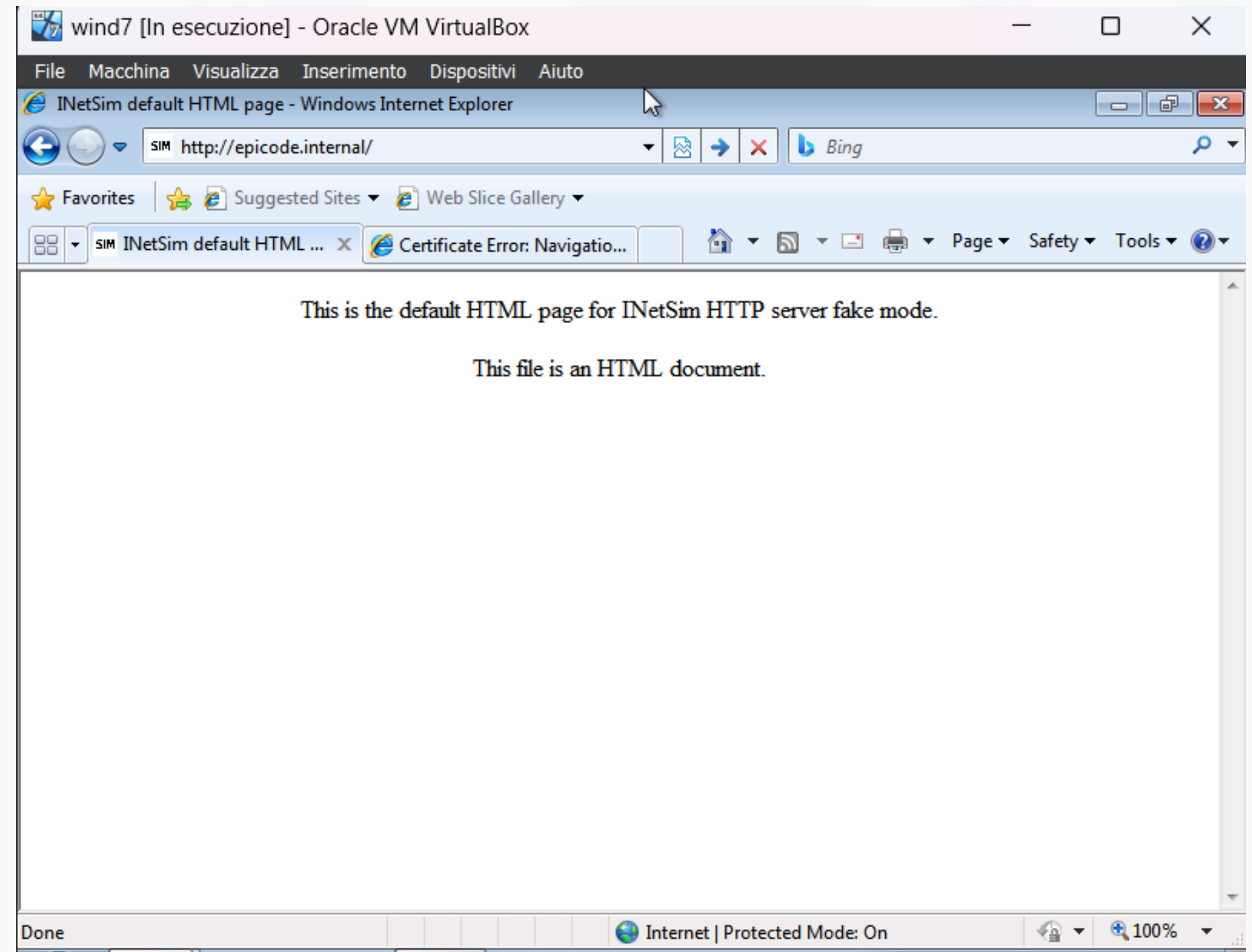
INSERIAMO ALL'INTERNO DELLA
MACCHINA WINDOWS L'IP DNS
192.168.32.100 IL QUALE
CORRISPONDE AL SERVER DNS KALI



LABORATORIO VM

**UNA VOLTA AVVIATO IL SERVIZIO
INETSIM NELLA MACCHINA KALI
TRAMITE IL COMANDO “SUDO
INETSIM”**

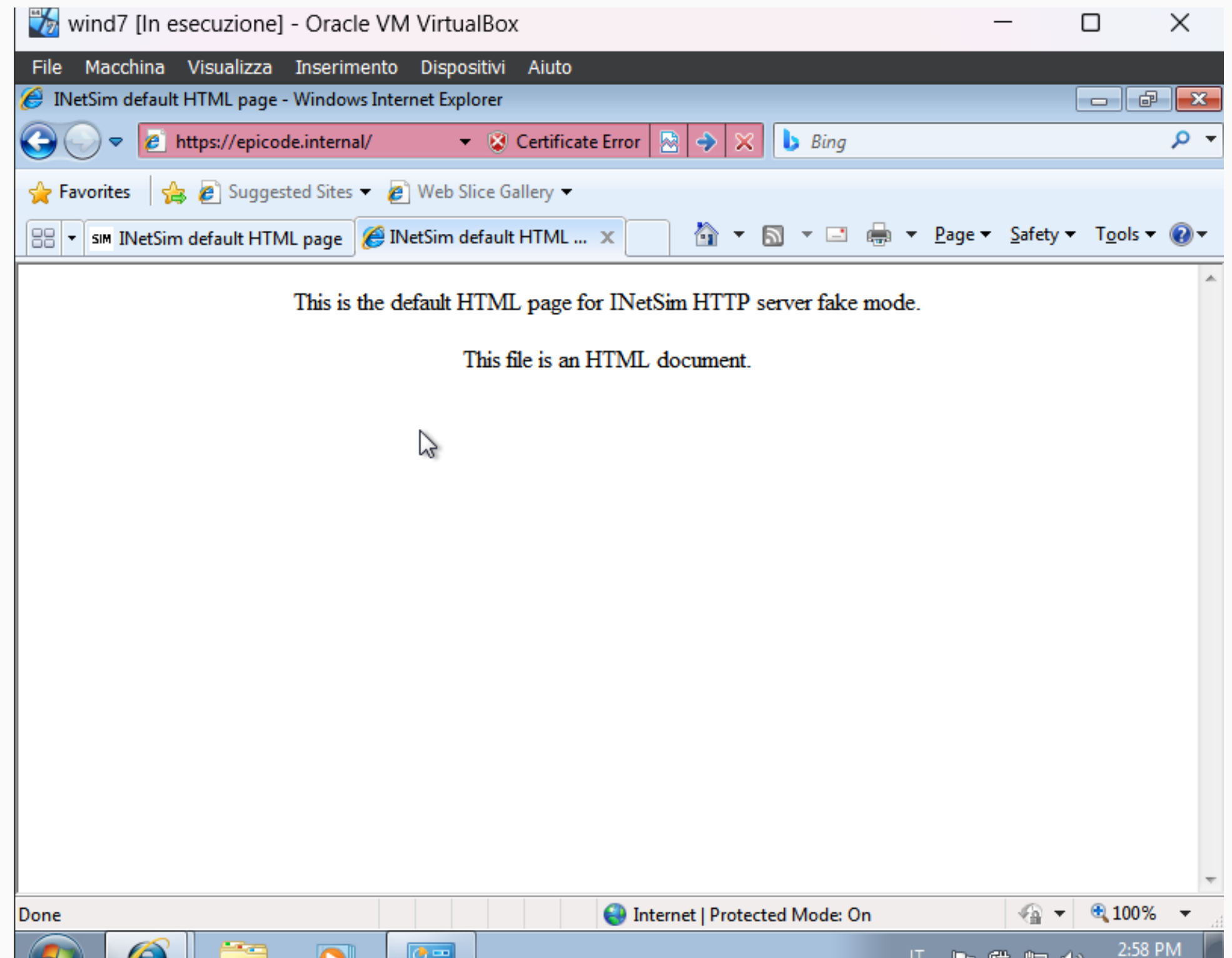
**AVVALENDOCI DEL BROWSER DELLA
MACCHINA WINDOWS CERCHIAMO IL
SITO D'INTERESSE INSERENDO IL SUO
DOMINIO NELLA BARRA DELL'URL
TRAMITE UNA RICHIESTA HTTP,
VISUALIZZANDO COSÌ LA RISPOSTA
DEL DNS SERVER.**



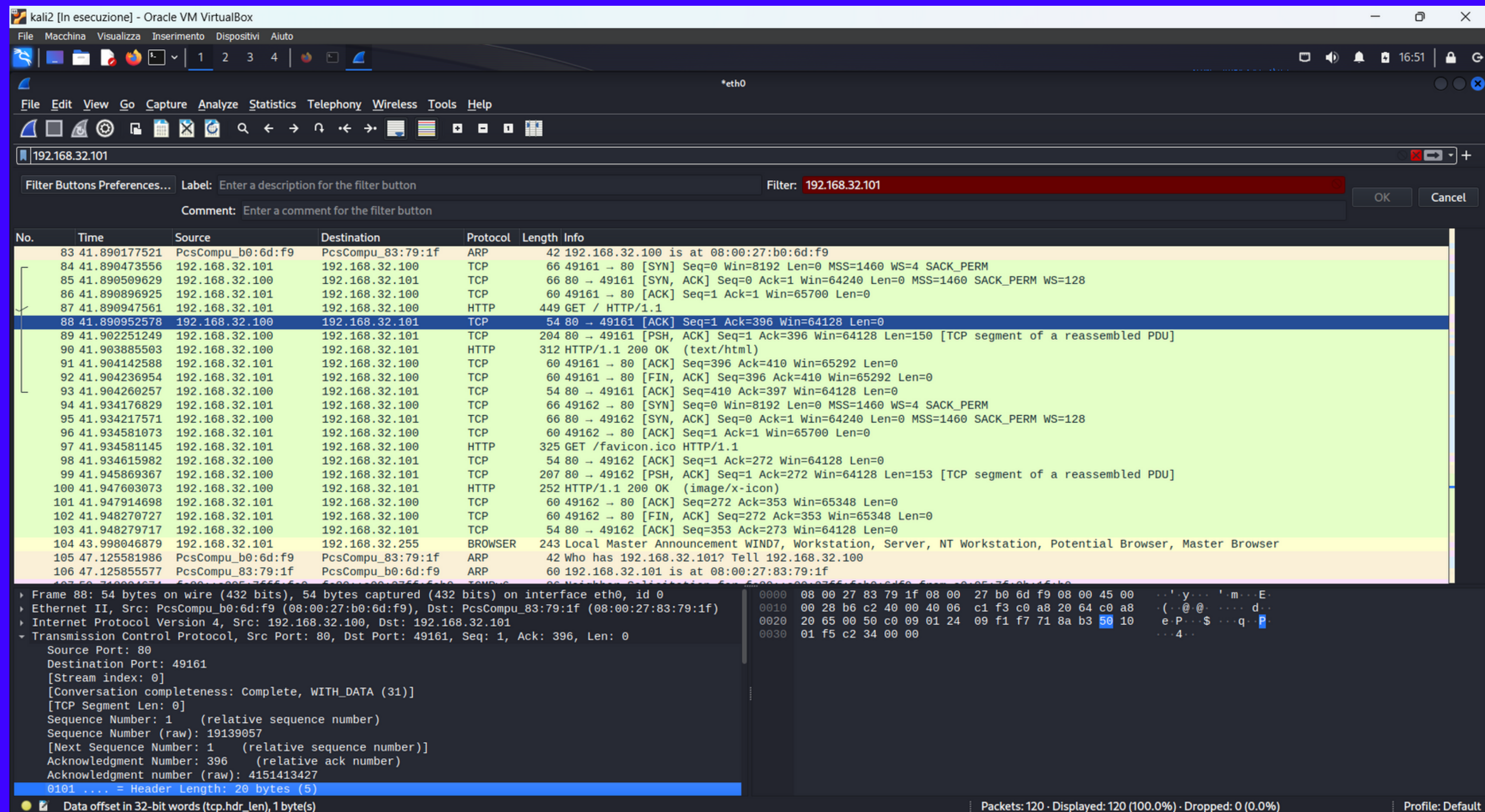
LABORATORIO VM

UNA VOLTA AVVIATO IL SERVIZIO
INETSIM NELLA MACCHINA KALI
TRAMITE IL COMANDO “SUDO
INETSIM”

AVVALENDOCI DEL BROWSER DELLA
MACCHINA WINDOWS CERCHIAMO IL
SITO D'INTERESSE INSERENDO IL SUO
DOMINIO NELLA BARRA DELL'URL
TRAMITE UNA RICHIESTA HTTPS,
VISUALIZZANDO COSÌ LA RISPOSTA
DEL DNS SERVER



**TRAMITE L'USO DI WIRESHARK
SI EVIDENZIA COME CON IL SOLO HTTP SI EFFETTUA SOLO LA
CONNESSIONE POICHE' TCP/ DI INTERNET NON PREVEDE
FUNZIONALITÀ DI SICUREZZA ALL'USO PRINCIPALE DELLA RETE**



TRAMITE L'USO DI WIRESHARK

MENTRE L'USO DI HTTPS PERMETTE UNA COMUNICAZIONE SICURA DALLA SORGENTE AL DESTINATARIO (END-TO-END) FORNENDO AUTENTICAZIONE, CONFIDENZIALITÀ E INTEGRITÀ DEI DATI POICHÈ LAVORA AL DI SOPRA DEL LIVELLO DI TRASPORTO

Wireshark capture of a network packet. The filter is set to 192.168.32.101. The packet list shows a TCP RST packet (No. 56) and a TLSv1 record (No. 62). The packet details pane shows the TLSv1 record structure, including the Handshake Protocol: Server Hello.

No.	Time	Source	Destination	Protocol	Length	Info
49	27.499851596	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
50	28.500433286	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
51	29.942053052	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
52	30.500629236	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
53	31.499424341	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
54	33.095603673	192.168.32.101	192.168.32.100	TCP	60	49164 → 443 [FIN, ACK] Seq=271 Ack=1379 Win=64320 Len=0
55	33.095694772	192.168.32.100	192.168.32.101	TLSv1	91	Encrypted Alert
56	33.096031607	192.168.32.101	192.168.32.100	TCP	60	49164 → 443 [RST, ACK] Seq=272 Ack=1416 Win=0 Len=0
57	33.096347863	192.168.32.101	192.168.32.100	TCP	66	49165 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
58	33.096360707	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 WS=128
59	33.096563705	192.168.32.101	192.168.32.100	TCP	60	49165 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
60	33.096912335	192.168.32.101	192.168.32.100	TLSv1	190	Client Hello
61	33.096917250	192.168.32.100	192.168.32.101	TCP	54	443 → 49165 [ACK] Seq=1 Ack=137 Win=64128 Len=0
62	33.127189048	192.168.32.100	192.168.32.101	TLSv1	1373	Server Hello, Certificate, Server Key Exchange, Server Hello Done
63	33.131920520	192.168.32.101	192.168.32.100	TLSv1	188	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
64	33.132224694	192.168.32.100	192.168.32.101	TLSv1	113	Change Cipher Spec, Encrypted Handshake Message
65	33.142468710	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
66	33.340134499	192.168.32.100	192.168.32.101	TCP	113	[TCP Retransmission] 443 → 49165 [PSH, ACK] Seq=1320 Ack=271 Win=64128 Len=59
67	33.340475924	192.168.32.101	192.168.32.100	TCP	66	49165 → 443 [ACK] Seq=271 Ack=1379 Win=64320 Len=0 SLE=1320 SRE=1379
68	33.526735008	SernetSu_0b:1f:b0	Broadcast	ARP	60	Who has 192.168.1.55? Tell 192.168.1.254
69	33.999334854	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
70	34.999665283	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
71	36.284977359	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101
72	36.999847974	PcsCompu_83:79:1f	Broadcast	ARP	60	Who has 192.168.32.1? Tell 192.168.32.101

Checksum: 0xc75b [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
[iRTT: 0.000215842 seconds]
[Bytes in flight: 1319]
[Bytes sent since last PSH flag: 1319]
TCP payload (1319 bytes)
Transport Layer Security
TLSv1 Record Layer: Handshake Protocol: Server Hello
TLSv1 Record Layer: Handshake Protocol: Certificate
TLSv1 Record Layer: Handshake Protocol: Server Key Exchange
TLSv1 Record Layer: Handshake Protocol: Server Hello Done

0020 20 65 01 bb c0 0d 9e ce a5 a4 4c f6 b2 8f 50 18 e L . . . P .
0030 01 f5 c7 5b 00 00 16 03 01 00 59 02 00 00 55 03 . . . [. Y . . . U .
0040 01 ac 8d d0 1c 05 3f a3 cc 1d 5a db ab e7 4c 24 ? . . . Z . . . L \$
0050 8a 45 55 a3 02 79 e6 a8 93 44 4f 57 4e 47 52 44 EU . y . . . DOWNGRD
0060 00 20 25 56 5d 80 0e 73 5b 2e 7a 71 5c ed d8 7f . %V] . . s [. zq \ . . .
0070 90 9e ee b4 68 5b db a9 66 cb bf 18 02 75 87 a9 . . . h [. . . f . . . u . . .
0080 e4 5a c0 14 00 00 0d ff 01 00 01 00 00 0b 00 04 . Z
0090 03 00 01 02 16 03 01 03 6b 0b 00 03 67 00 03 64 k . . . g . . . d
00a0 00 03 61 30 82 03 5d 30 82 02 45 a0 03 02 01 02 . . a0 . .]0 . . E
00b0 02 14 3c 16 48 25 e3 58 65 c9 f5 ce 05 35 66 a6 . . < . H % X e 5f .
00c0 0a 74 07 74 01 09 30 0d 06 09 2a 86 48 86 f7 0d . t . t . 0 * . H . . .
00d0 01 01 05 05 00 30 3e 31 10 30 0e 06 03 55 04 0a 0 > 1 . 0 . . . U . . .
00e0 0c 07 49 4e 65 74 53 69 6d 31 14 30 12 06 03 55 . . INetSi m1 0 . . . U . . .
00f0 04 0b 0c 0b 44 65 76 65 6c 6f 70 6d 65 6e 74 31 Deve lopment1
0100 14 30 12 06 03 55 04 03 0c 0b 69 6e 65 74 73 69 . 0 . . . U . . . inetsi

Data offset in 32-bit words (tcp.hdr_len), 1 byte(s) Packets: 120 · Displayed: 120 (100.0%) · Dropped: 0 (0.0%) Profile: Default