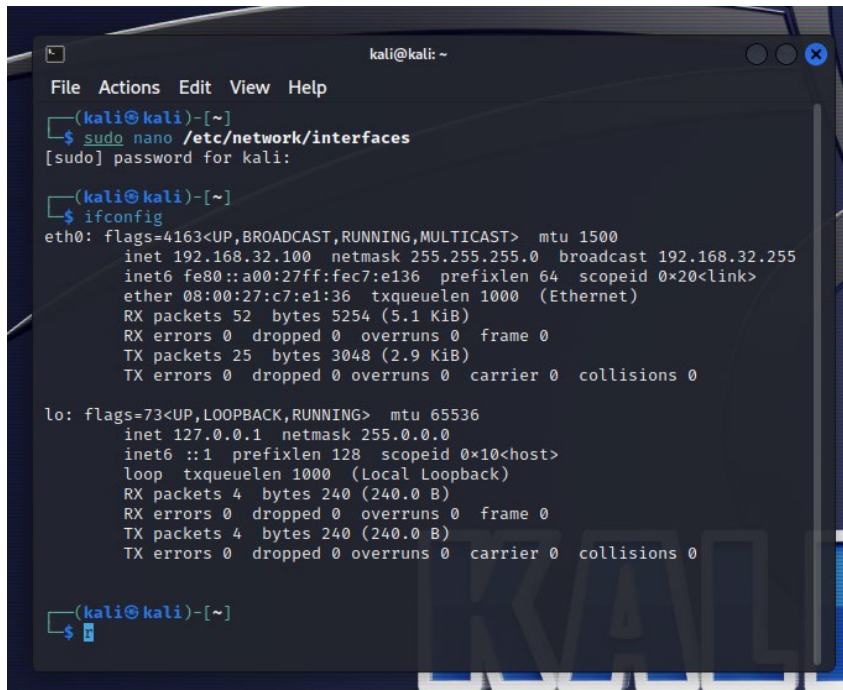


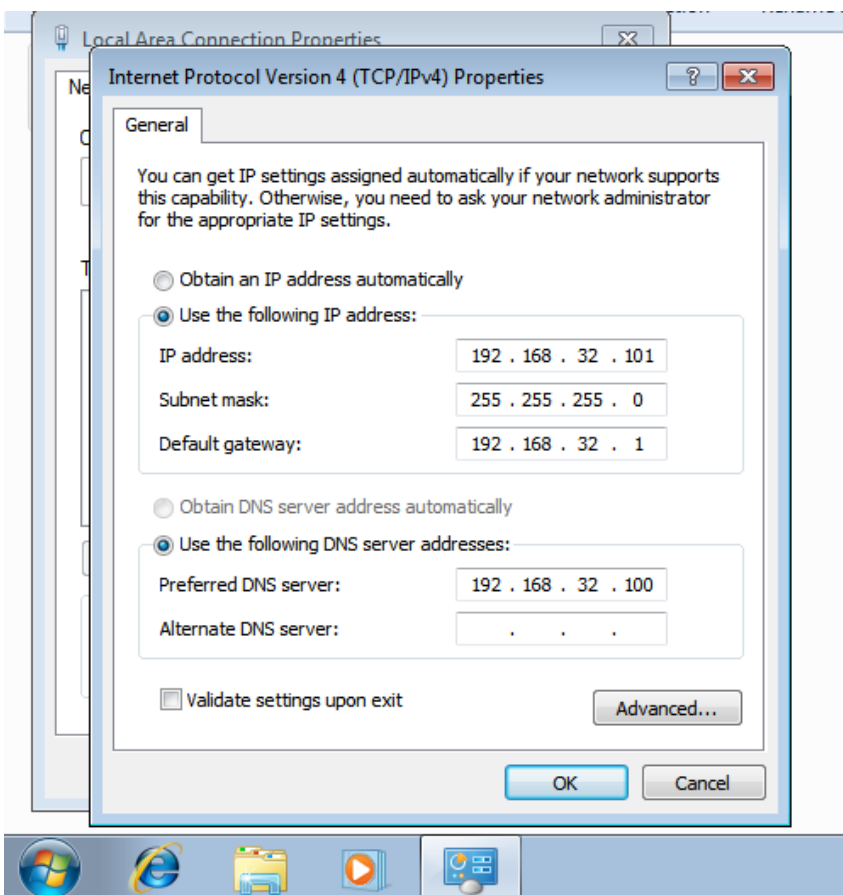
Report: “Simulazione rete complessa”



```
kali@kali: ~  
File Actions Edit View Help  
~  
$ sudo nano /etc/network/interfaces  
[sudo] password for kali:  
~  
$ 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:fec7:e136 prefixlen 64 scopeid 0x20<link>  
    ether 08:00:27:c7:e1:36 txqueuelen 1000 (Ethernet)  
    RX packets 52 bytes 5254 (5.1 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 25 bytes 3048 (2.9 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 4 bytes 240 (240.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 4 bytes 240 (240.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
~  
$
```

Come da traccia ho impostato l'indirizzo Ip della macchina virtuale Kali Linux come "192.168.32.100" con il comando "sudo nano /etc/network/interfaces" per poi resettare la macchina e confermare tramite il comando "ifconfig"(figura in alto). Per impostare l'indirizzo IP della macchina virtuale Window 7 come "192.168.32.101" ho dovuto aprire il control panel/network and internet/network and sharing center/change adapter settings/ local area connection/ Internet protocol Version 4 nelle proprietà, da lì ho settato l'indirizzo IP e ho approfittato per specificare l'indirizzo IP del server DNS che avvierò tramite inetsim (figura in basso).

Dopo aver resettato Window7 e ho finito il cambio di indirizzi IP accertandomi che le due macchine comunicassero tra di loro usando il comando "ping 192.168.32.101" da Kali.



```

28 #
29 start_service dns
30 #start_service http
31 start_service https
32 #start_service smtp
33 #start_service smtps
34 start_service pop3
35 start_service pop3s
36 start_service ftp
37 start_service ftps
38 start_service tftp
39 start_service irc
40 start_service ntp
41 start_service finger
42 start_service ident
43 start_service syslog
44 start_service time_tcp
45 start_service time_udp

```

Una volta installa Inetsim su Kali Linux ho aperto il file di configurazione “inetsim.Conf” in /etc/inetsim/ e per prima cosa ho “chiuso” il servizio http mettendo un cancelletto per far ignorare la riga di testo e con lui altri servizi con rischio di conflitto sotto.

Quando dovrò scambiare il servizio da server https a http mi basterà apportare il cancelletto davanti alla linea del servizio https e levarlo a http.

```

197
198 #####
199 # dns_default_ip
200 #
201 # Default IP address to return with DNS replies
202 #
203 # Syntax: dns_default_ip <IP address>
204 #
205 # Default: 127.0.0.1
206 #
207 dns_default_ip 192.168.32.100
208
209
210 #####
211 # dns_default_hostname
212 #
213 # Default hostname to return with DNS replies
214 #
215 # Syntax: dns_default_hostname <hostname>
216 #
217 # Default: www
218 #
219 dns_default_hostname epicodeinternal
220
221
222 #####
223 # dns_default_domainname
224 #
225 # Default domain name to return with DNS replies
226 #
227 # Syntax: dns_default_domainname <domainname>
228 #
229 # Default: inetsim.org
230 #
231 #dns_default_domainname some.domain
232

```

Subito dopo Ho modificato l’opzione “dns_default_ip” con lo stesso ip assegnato a Kali Linux e specificato precedentemente come Ip del sever DNS a cui Window 7 farà riferimento, inoltre alla voce “dns_default_hostname” ho specificato il nome “epicode.internal” come richiesto dalla traccia.

```

58
59
60 #####
61 # service_bind_address
62 #
63 # IP address to bind services to
64 #
65 # Syntax: service_bind_address <IP address>
66 #
67 # Default: 127.0.0.1
68 #
69 service_bind_address 192.168.32.100
70
71
72 #####

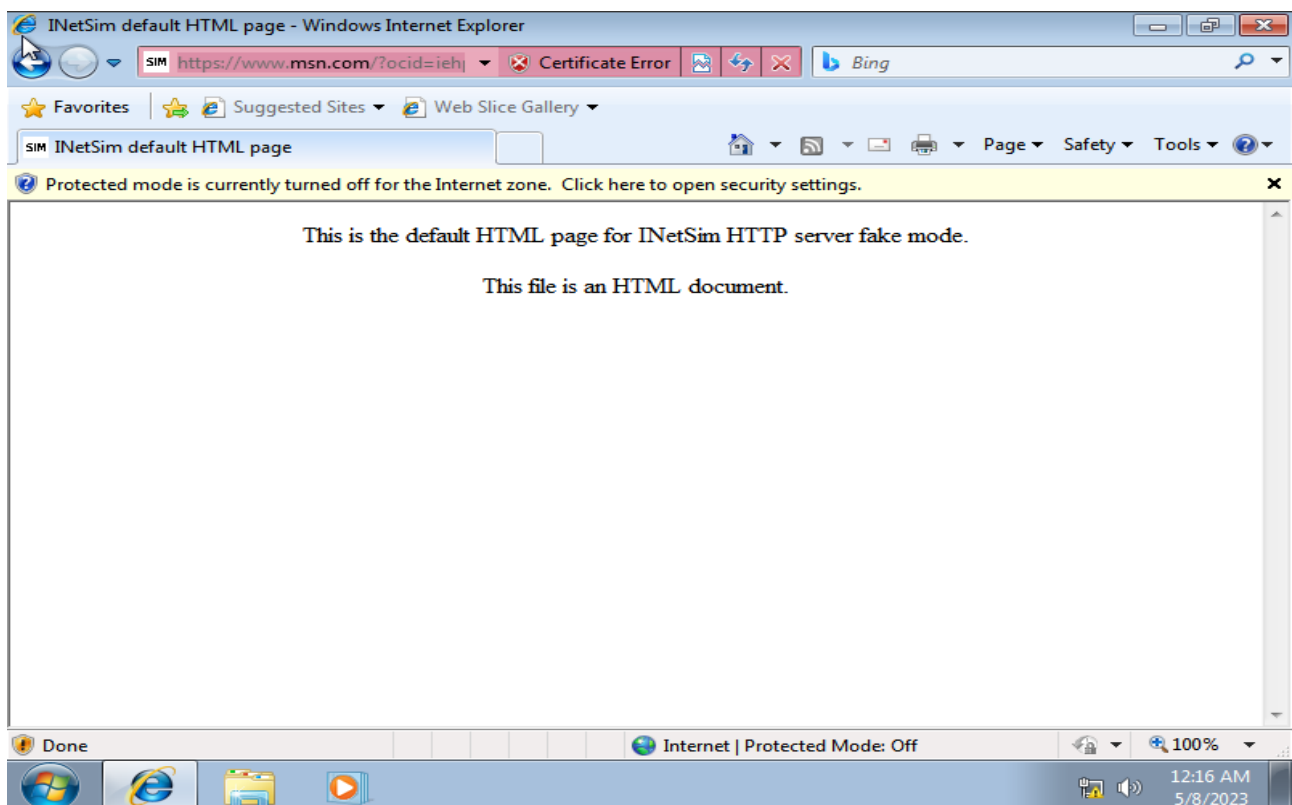
```

Per rendere consistente l’indirizzo Ip dei vari servizi offerti da inetsim ho modificato l’opzione “service_bind_address” con lo stesso indirizzo Ip di Kali Linux.

```
kali@kali: ~  
File Actions Edit View Help  
[kali@kali]~  
$ sudo inetsim  
[sudo] password for kali:  
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 2000) ==  
Session ID: 2000  
Listening on: 192.168.32.100  
Real Date/Time: 2023-05-07 18:15:29  
Fake Date/Time: 2023-05-07 18:15:29 (Delta: 0 seconds)  
Forking services ...  
* dns_53_tcp_udp - started (PID 2010)  
* ntp_123_udp - started (PID 2018)  
* ident_113_tcp - started (PID 2020)  
* irc_6667_tcp - started (PID 2017)  
* daytime_13_tcp - started (PID 2024)  
* time_37_tcp - started (PID 2022)  
* time_37_udp - started (PID 2023)  
* syslog_514_udp - started (PID 2021)  
* finger_79_tcp - started (PID 2019)  
* daytime_13_udp - started (PID 2025)  
* echo_7_tcp - started (PID 2026)
```

Una volta configurato Inetsim in modo da avviare un server DNS e Https attivo il software computando nel terminal di Kali “sudo inetsim” che mi conferma che la simulazione della rete internet e dei servizi è iniziata.

Il passaggio successivo è aprire il Web Browser da Window7 per confermare che esso prende il ruolo di client per i server messi a disposizione da inetsim.



Da qui avviando Wireshark da Kali possiamo osservare tutti i passaggi di pacchetti tra il server e il client: La prima immagine in alto è il passaggio di pacchetti con il servizio del server https e possiamo notare che la quantità di passaggi è di numero molto più alto rispetto al servizio http (immagine in basso) sicuramente dovuto al fatto che il protocollo di comunicazione https è criptato quindi necessità di “key exchanges” per

decriptare i pacchetti scambiati, questo si può vedere anche dal protocollo “TLSv1” usato solo da https che è un protocollo criptato usato proprio garantire la sicurezza delle informazioni.

kali-linux-2023.1-virtualbox-amd64 [In esecuzione] - Oracle VM VirtualBox

File Macchina Visualizza Inserimento Dispositivi Aiuto

Capturing from any

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

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

No.	Time	Source	Destination	Protocol	Length	Info
18	5.230324084	192.168.32.101	192.168.32.100	TCP	68	49181 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM
19	5.230367896	192.168.32.100	192.168.32.101	TCP	56	80 → 49181 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
20	5.253184211	PcsCompu.c7:e1:36	192.168.32.101	ARP	44	Who has 192.168.32.101? Tell 192.168.32.100
21	5.253559901	PcsCompu.63:3c:2d	192.168.32.101	ARP	62	192.168.32.101 is at 08:00:27:63:3c:2d
22	5.743911247	192.168.32.101	192.168.32.100	TCP	68	[TCP Retransmission] [TCP Port numbers reused] 49181 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM
23	5.743944468	192.168.32.100	192.168.32.101	TCP	56	80 → 49181 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
24	6.244296983	192.168.32.101	192.168.32.100	TCP	64	[TCP Retransmission] [TCP Port numbers reused] 49181 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM
25	6.244425416	192.168.32.100	192.168.32.101	TCP	56	80 → 49181 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
26	6.248132937	192.168.32.101	192.168.32.100	TCP	62	49180 → 443 [FIN, ACK] Seq=259 Ack=1379 Win=64320 Len=0
27	6.248245435	192.168.32.100	192.168.32.101	TLSv1	93	Encrypted Alert
28	6.248433602	192.168.32.101	192.168.32.100	TCP	62	49180 → 443 [RST, ACK] Seq=260 Ack=1410 Win=0 Len=0
29	6.278978375	192.168.32.101	192.168.32.100	TCP	68	49182 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
30	8.219013332	192.168.32.100	192.168.32.101	TCP	68	443 → 49182 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
31	8.279218569	192.168.32.101	192.168.32.100	TCP	62	49182 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
32	8.280913437	192.168.32.101	192.168.32.100	TLSv1	212	Client Hello
33	8.280932043	192.168.32.100	192.168.32.101	TCP	56	443 → 49182 [ACK] Seq=1 Ack=157 Win=64128 Len=0
34	8.336614307	192.168.32.100	192.168.32.101	TLSv1	1375	Server Hello, Certificate, Server Key Exchange, Server Hello Done
35	8.344768316	192.168.32.101	192.168.32.100	TLSv1	190	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
36	8.344799580	192.168.32.100	192.168.32.101	TCP	56	443 → 49182 [ACK] Seq=1320 Ack=291 Win=64128 Len=0
37	8.345482929	192.168.32.100	192.168.32.101	TLSv1	115	Change Cipher Spec, Encrypted Handshake Message
38	8.350142906	192.168.32.101	192.168.32.100	TCP	62	49182 → 443 [FIN, ACK] Seq=291 Ack=1379 Win=64320 Len=0
39	8.350670988	192.168.32.101	192.168.32.100	TCP	68	49183 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
40	8.350695250	192.168.32.100	192.168.32.101	TCP	68	443 → 49183 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
41	8.351048994	192.168.32.101	192.168.32.100	TCP	62	49183 → 443 [ACK] Seq=1 Ack=1 Win=65700 Len=0
42	8.351278352	192.168.32.100	192.168.32.101	TLSv1	93	Encrypted Alert
43	8.351767336	192.168.32.101	192.168.32.100	TLSv1	212	Client Hello
44	8.351767519	192.168.32.101	192.168.32.100	TCP	62	49182 → 443 [RST, ACK] Seq=292 Ack=1410 Win=0 Len=0
45	8.351779873	192.168.32.100	192.168.32.101	TCP	56	443 → 49183 [ACK] Seq=1 Ack=157 Win=64128 Len=0
46	8.408095250	192.168.32.100	192.168.32.101	TLSv1	1375	Server Hello, Certificate, Server Key Exchange, Server Hello Done
47	8.418406165	192.168.32.101	192.168.32.100	TLSv1	190	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
48	8.418450491	192.168.32.100	192.168.32.101	TCP	56	443 → 49183 [ACK] Seq=1320 Ack=291 Win=64128 Len=0
49	8.419203874	192.168.32.100	192.168.32.101	TLSv1	115	Change Cipher Spec, Encrypted Handshake Message
50	8.426565664	192.168.32.101	192.168.32.100	TLSv1	509	Application Data
51	8.443454563	192.168.32.100	192.168.32.101	TLSv1	237	Application Data
52	8.446084837	192.168.32.100	192.168.32.101	TLSv1	386	Application Data, Encrypted Alert
53	8.446963417	192.168.32.101	192.168.32.100	TCP	62	49183 → 443 [ACK] Seq=744 Ack=1891 Win=65700 Len=0
54	8.447128213	192.168.32.100	192.168.32.101	TCP	62	49183 → 443 [FIN, ACK] Seq=744 Ack=1891 Win=65700 Len=0
55	8.447150277	192.168.32.100	192.168.32.101	TCP	56	443 → 49183 [ACK] Seq=1891 Ack=745 Win=64128 Len=0
56	36.385889927	192.168.32.101	192.168.32.255	BROWSER	250	Domain/Workgroup Announcement WORKGROUP, NT Workstation, Domain Enum
57	64.539426784	192.168.32.101	192.168.32.100	NBSS	62	NBSS Continuation Message
58	64.539453940	192.168.32.100	192.168.32.101	TCP	80	52048 → 139 [ACK] Seq=1 Ack=2 Win=501 Len=0 TSval=3180966568 TSecr=60818 SLE=1 SFE=?

Frame 1: 68 bytes on wire (544 bits), 68 bytes captured (544 bits) on interface any, id 0

any: alive capture in progress

Packets: 70 - Displayed: 70 (100.0%)

Profile: Default

kali-linux-2023.1-virtualbox-amd64 [In esecuzione] - Oracle VM VirtualBox

File Macchina Visualizza Inserimento Dispositivi Aiuto

Capturing from any

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	DNS	87	Standard query 0x5e16 A teredo.ipv6.microsoft.com
2	0.034350710	192.168.32.100	192.168.32.101	DNS	103	Standard query response 0x5e16 A teredo.ipv6.microsoft.com A 192.168.32.100
3	2.030668525	fe80::a00:27ff:fec7::f602:2	ff02::2	ICMPv6	72	Router Solicitation from 08:00:27:c7:e1:36
4	4.575872436	PcsCompu.63:3c:2d	192.168.32.101	ARP	62	Who has 192.168.32.100? Tell 192.168.32.101
5	4.575891227	PcsCompu.c7:e1:36	192.168.32.100	ARP	44	192.168.32.100 is at 08:00:27:c7:e1:36
6	6.144009061	192.168.32.101	192.168.32.100	DNS	73	Standard query 0x6009 A www.msn.com
7	6.153874931	192.168.32.100	192.168.32.101	DNS	89	Standard query response 0x6009 A www.msn.com A 192.168.32.100
8	6.157355320	192.168.32.101	192.168.32.100	TCP	68	49158 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
9	6.157373147	192.168.32.100	192.168.32.101	TCP	56	443 → 49158 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
10	6.608716485	192.168.32.101	192.168.32.100	TCP	68	[TCP Retransmission] [TCP Port numbers reused] 49158 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
11	6.608774088	192.168.32.100	192.168.32.101	TCP	56	443 → 49158 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
12	7.183504450	192.168.32.101	192.168.32.100	TCP	64	[TCP Retransmission] [TCP Port numbers reused] 49158 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 SACK_PERM
13	7.183533559	192.168.32.100	192.168.32.101	TCP	56	443 → 49158 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
14	11.095286402	192.168.32.101	192.168.32.100	TCP	68	49159 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
15	11.095387298	192.168.32.100	192.168.32.101	TCP	68	80 → 49159 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
16	11.095582646	192.168.32.101	192.168.32.100	TCP	62	49159 → 80 [ACK] Seq=1 Ack=1 Win=65700 Len=0
17	11.095712232	192.168.32.101	192.168.32.100	HTTP	470	GET /favicon.ico HTTP/1.1
18	11.095751284	192.168.32.100	192.168.32.101	TCP	56	80 → 49159 [ACK] Seq=1 Ack=424 Win=64128 Len=0
19	11.122497912	192.168.32.100	192.168.32.101	TCP	204	80 → 49159 [PSH, ACK] Seq=1 Ack=424 Win=64128 Len=148 [TCP segment of a reassembled PDU]
20	11.125235729	192.168.32.100	192.168.32.101	HTTP	314	HTTP/1.1 200 OK (text/html)
21	11.125791224	192.168.32.101	192.168.32.100	TCP	62	49159 → 80 [ACK] Seq=424 Ack=408 Win=65292 Len=0
22	11.125945612	192.168.32.101	192.168.32.100	TCP	62	49159 → 80 [FIN, ACK] Seq=424 Ack=408 Win=65292 Len=0
23	11.125965140	192.168.32.100	192.168.32.101	TCP	56	80 → 49159 [ACK] Seq=408 Ack=425 Win=64128 Len=0
24	11.177395792	192.168.32.101	192.168.32.100	TCP	68	49160 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
25	11.177430978	192.168.32.100	192.168.32.101	TCP	68	80 → 49160 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM WS=128
26	11.178991908	192.168.32.101	192.168.32.100	TCP	62	49160 → 80 [ACK] Seq=1 Ack=1 Win=65700 Len=0
27	11.179936920	192.168.32.101	192.168.32.100	HTTP	345	GET /favicon.ico HTTP/1.1
28	11.179961753	192.168.32.100	192.168.32.101	TCP	56	80 → 49160 [ACK] Seq=1 Ack=290 Win=64128 Len=0
29	11.203126302	192.168.32.100	192.168.32.101	TCP	207	80 → 49160 [PSH, ACK] Seq=1 Ack=290 Win=64128 Len=151 [TCP segment of a reassembled PDU]
30	11.205936510	192.168.32.100	192.168.32.101	HTTP	254	HTTP/1.1 200 OK (image/x-icon)
31	11.206123036	192.168.32.101	192.168.32.100	TCP	62	49160 → 80 [ACK] Seq=290 Ack=351 Win=65348 Len=0
32	11.206233930	192.168.32.101	192.168.32.100	TCP	62	49160 → 80 [FIN, ACK] Seq=290 Ack=351 Win=65348 Len=0
33	11.206249028	192.168.32.100	192.168.32.101	TCP	56	80 → 49160 [ACK] Seq=351 Ack=291 Win=64128 Len=0
34	37.982706550	192.168.32.101	192.168.32.255	BROWSER	250	Domain/Workgroup Announcement WORKGROUP, NT Workstation, Domain Enum
35	67.566683259	fe80::a00:27ff:fec7::f602:2	ff02::2	ICMPv6	72	Router Solicitation from 08:00:27:c7:e1:36
36	97.950918159	192.168.32.101	192.168.32.255	BROWSER	250	Domain/Workgroup Announcement WORKGROUP, NT Workstation, Domain Enum
37	108.169646494	192.168.32.101	192.168.32.255	BROWSER	245	Local Master Announcement RN-PC, Workstation, Server, NT Workstation, Potential Browser, Master Browser

Frame 1: 87 bytes on wire (696 bits), 87 bytes captured (696 bits) on interface any, id 0

any: alive capture in progress

Packets: 37 - Displayed: 37 (100.0%)

Profile: Default