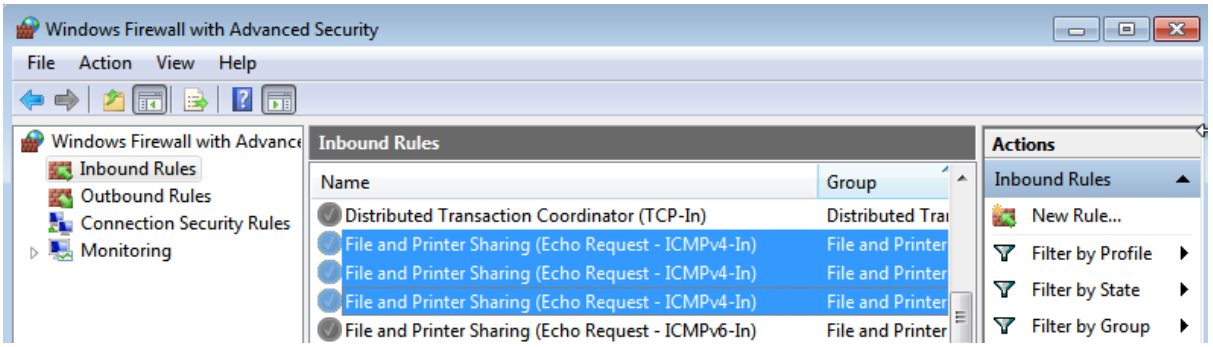
CONFIGURAZIONE FIREWALL WIN7

Dopo aver configurato le 3 macchine virtuali, kali, metasploitable2 e win7 su windows 7 dobbiamo creare una regola per abilitare la regola del ping. Perchè se da Kali proviamo a pingare win7 succede questo

```
└─$ ping 192.168.50.102
PING 192.168.50.102 (192.168.50.102) 56(84) bytes of data.
```

non si riesce a raggiungere la macchina.
Entrando sulle impostazioni del firewall di win7 e andando nelle 'Inbound Rules' possiamo trovare queste 3 regole disabilitate



Abilitandole possiamo ri-pingare da kali a win7 e vediamo che pinga

```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.50.102
PING 192.168.50.102 (192.168.50.102) 56(84) bytes of data.
64 bytes from 192.168.50.102: icmp_seq=1 ttl=128 time=1.38 ms
64 bytes from 192.168.50.102: icmp_seq=2 ttl=128 time=0.939 ms
64 bytes from 192.168.50.102: icmp_seq=3 ttl=128 time=0.868 ms
^C
— 192.168.50.102 ping statistics —
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 0.868/1.061/1.378/0.225 ms
```

CREAZIONE DI UN LOCALHOST CON INETSIM E SNIFFAMENTO DI PACCHETTI CON WIRESHARK

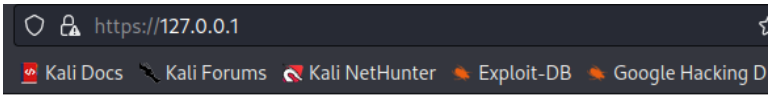
Su Kali abbiamo un tool pre-installato di nome inetsim che ci permette di creare un servizio http in locale, di default i server in locale sono con il seguente ip: 127.0.0.1

Avviamo il servizio con questo comando

```
┌──(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.
```

Una volta eseguito apriamo Wireshark selezioniamo l'opzione any, così andiamo a catturare tutti i pacchetti da tutte le schede di rete.

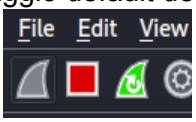
e avviamo la cattura dei pacchetti, poi apriamo il sito di inetsim cioè 127.0.0.1



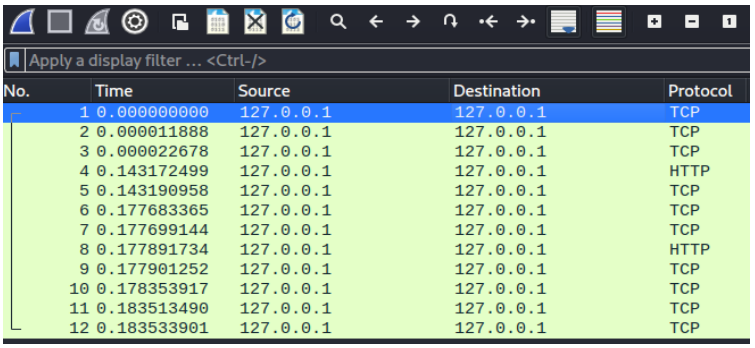
This is the default HTML page for INetSim HTTP server fake mode.

This file is an HTML document.

Vediamo che esce il messaggio default del localhost e su wireshark stoppiamo subito la cattura con il tasto rosso

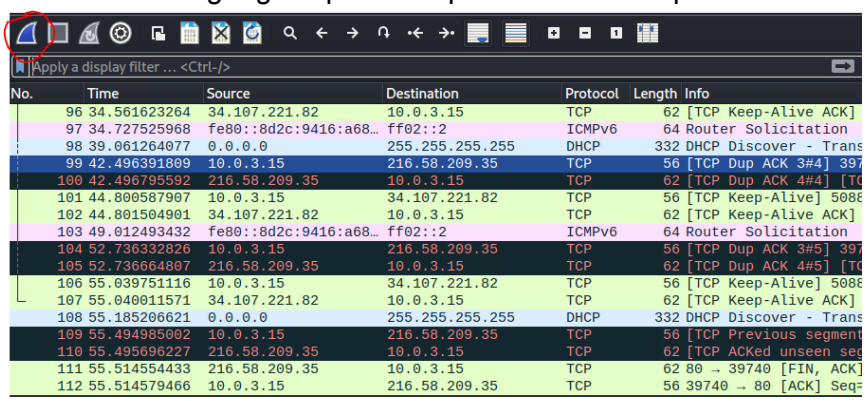


Di seguito possiamo vedere tutti i pacchetti catturati



No.	Time	Source	Destination	Protocol
1	0.000000000	127.0.0.1	127.0.0.1	TCP
2	0.000011888	127.0.0.1	127.0.0.1	TCP
3	0.000022678	127.0.0.1	127.0.0.1	TCP
4	0.143172499	127.0.0.1	127.0.0.1	HTTP
5	0.143190958	127.0.0.1	127.0.0.1	TCP
6	0.177683365	127.0.0.1	127.0.0.1	TCP
7	0.177699144	127.0.0.1	127.0.0.1	TCP
8	0.177891734	127.0.0.1	127.0.0.1	HTTP
9	0.177901252	127.0.0.1	127.0.0.1	TCP
10	0.178353917	127.0.0.1	127.0.0.1	TCP
11	0.183513490	127.0.0.1	127.0.0.1	TCP
12	0.183533901	127.0.0.1	127.0.0.1	TCP

Se andiamo su google.it per esempio vedremo altri pacchetti



No.	Time	Source	Destination	Protocol	Length	Info
96	34.561623264	34.107.221.82	10.0.3.15	TCP	62	[TCP Keep-Alive ACK]
97	34.727525968	fe80::8d2c:9416:a68…	ff02::2	ICMPv6	64	Router Solicitation
98	39.061264077	0.0.0.0	255.255.255.255	DHCP	332	DHCP Discover - Trans
99	42.496391809	10.0.3.15	216.58.209.35	TCP	56	[TCP Dup ACK 3#4] 397
100	42.496795592	216.58.209.35	10.0.3.15	TCP	62	[TCP Dup ACK 4#4] [TC
101	44.800587907	10.0.3.15	34.107.221.82	TCP	56	[TCP Keep-Alive] 5088
102	44.801504901	34.107.221.82	10.0.3.15	TCP	62	[TCP Keep-Alive ACK]
103	49.012493432	fe80::8d2c:9416:a68…	ff02::2	ICMPv6	64	Router Solicitation
104	52.736332826	10.0.3.15	216.58.209.35	TCP	56	[TCP Dup ACK 3#5] 397
105	52.736664807	216.58.209.35	10.0.3.15	TCP	62	[TCP Dup ACK 4#5] [TC
106	55.039751116	10.0.3.15	34.107.221.82	TCP	56	[TCP Keep-Alive] 5088
107	55.040011571	34.107.221.82	10.0.3.15	TCP	62	[TCP Keep-Alive ACK]
108	55.185206621	0.0.0.0	255.255.255.255	DHCP	332	DHCP Discover - Trans
109	55.494985002	10.0.3.15	216.58.209.35	TCP	56	[TCP Previous segment
110	55.495696227	216.58.209.35	10.0.3.15	TCP	62	[TCP ACKed unseen seg
111	55.514554433	216.58.209.35	10.0.3.15	TCP	62	80 → 39740 [FIN, ACK]
112	55.514579466	10.0.3.15	216.58.209.35	TCP	56	39740 → 80 [ACK] Seq=