Assignment Series #A1 Spoofer

Compile your first Software in Linux (caida spoofer tool)

1. Task description

The receipt of compiling a software with automake is mostly the same:

1. `gzip -d` project.tar.gz
2. `tar -xvf` project.tar
3. cd project
4. `configure --prefix=/opt/applic/project`
5. `make`
6. `make install`

After you've downloaded, built and run the spoofer tool from scratch, please start wireshark and answer the following questions:

1. Explain the idea of ip spoofing
2. Is this a realistic attack over the internet with TCP/IP - explain
3. Is this a realistic attack over the internet with UDP/IP - explain
4. What are the destination addresses of the spoofer tool (see in Wireshark)
5. What protocol is the spoofer tool trying to spoof?
6. Do you find your own results on https://spoofer.caida.org/recent_tests.php

2. Answers

1. The idea behind IP spoofing is the creation of Internet Protocol (IP) packets which have a modified source address in order to either hide the identity of the sender, to impersonate another computer system, or both. It is a technique often used by bad actors to invoke DDoS attacks against a target device or the surrounding infrastructure.

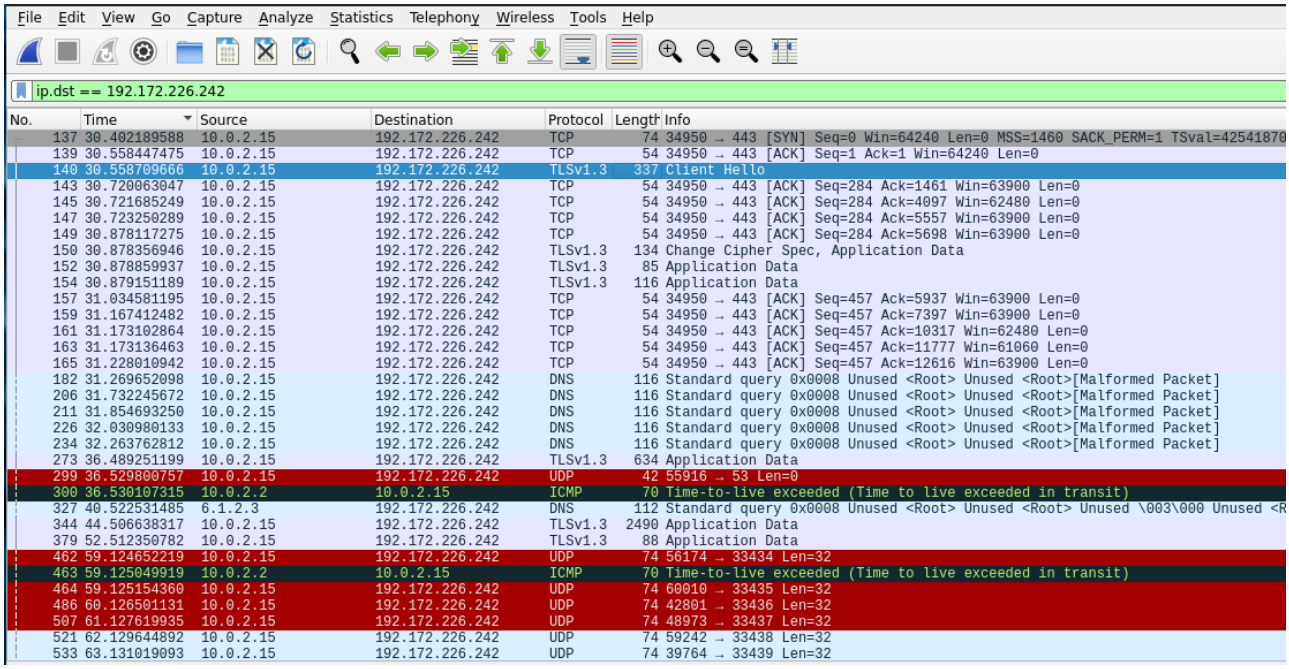
2. Not really. Direct access to traffic is now much more difficult if the intruder's computer is not on the same subnet. This is due to the fact that intercepting a data packet is only possible with the help of the corresponding packet sequence number - a task that is almost impossible today compared to earlier days from outside. In the past, operating systems and network devices still generated these process numbers entered in the TCP header according to a pattern that was always the same. As today's systems output the sequence numbers randomly, these so-called TCP sequence prediction attacks (also called blind spoofing) have basically become ineffective - however, older devices are still at risk.

3. I'd say yes, becuase UDP doesn't use a `three way handshake` or any `sequence numbers` to establish a connection.

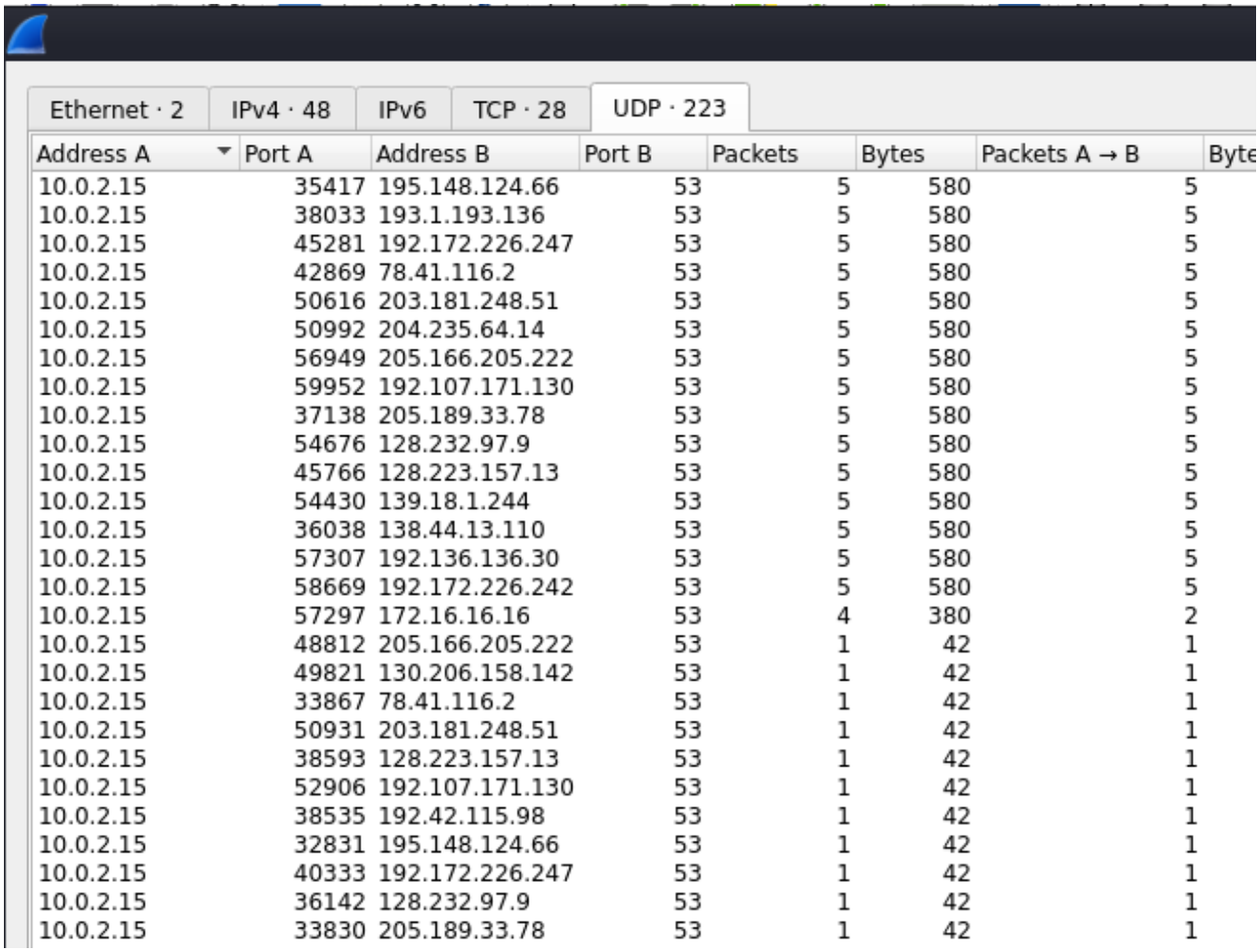
4. The spoofer tool communicates with the ip address `192.172.226.242`

```
root@hlkali:/opt/applic/spoofer/bin# netstat -antp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 10.0.2.15:36498         192.172.226.242:443     ESTABLISHED 32467/spoofer-probe
tcp        0      0 10.0.2.15:45516         34.216.82.69:443        ESTABLISHED 4365/firefox-esr
tcp        0      1 10.0.2.15:56842         136.144.56.255:80       SYN_SENT    32611/curl
root@hlkali:/opt/applic/spoofer/bin#
```

In Wireshark it looks like this (filter `ip.dst == 192.172.226.242`)



To get a list of all destination addresses I use the `Statistics` --> `Coversations` menu:



Ethernet · 2 | IPv4 · 48 | IPv6 | TCP · 28 | UDP · 223

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Byte
10.0.2.15	35417	195.148.124.66	53	5	580	5	
10.0.2.15	38033	193.1.193.136	53	5	580	5	
10.0.2.15	45281	192.172.226.247	53	5	580	5	
10.0.2.15	42869	78.41.116.2	53	5	580	5	
10.0.2.15	50616	203.181.248.51	53	5	580	5	
10.0.2.15	50992	204.235.64.14	53	5	580	5	
10.0.2.15	56949	205.166.205.222	53	5	580	5	
10.0.2.15	59952	192.107.171.130	53	5	580	5	
10.0.2.15	37138	205.189.33.78	53	5	580	5	
10.0.2.15	54676	128.232.97.9	53	5	580	5	
10.0.2.15	45766	128.223.157.13	53	5	580	5	
10.0.2.15	54430	139.18.1.244	53	5	580	5	
10.0.2.15	36038	138.44.13.110	53	5	580	5	
10.0.2.15	57307	192.136.136.30	53	5	580	5	
10.0.2.15	58669	192.172.226.242	53	5	580	5	
10.0.2.15	57297	172.16.16.16	53	4	380	2	
10.0.2.15	48812	205.166.205.222	53	1	42	1	
10.0.2.15	49821	130.206.158.142	53	1	42	1	
10.0.2.15	33867	78.41.116.2	53	1	42	1	
10.0.2.15	50931	203.181.248.51	53	1	42	1	
10.0.2.15	38593	128.223.157.13	53	1	42	1	
10.0.2.15	52906	192.107.171.130	53	1	42	1	
10.0.2.15	38535	192.42.115.98	53	1	42	1	
10.0.2.15	32831	195.148.124.66	53	1	42	1	
10.0.2.15	40333	192.172.226.247	53	1	42	1	
10.0.2.15	36142	128.232.97.9	53	1	42	1	
10.0.2.15	33830	205.189.33.78	53	1	42	1	

5. It's the UDP Protocol. The methodology says:

> The spoofer program attempts to send a series of spoofed UDP packets to servers distributed throughout the world. These packets are designed to test:
>
> - Different classes of spoofed IPv4 and IPv6 addresses, including private and routable
> - Ability to spoof neighboring, adjacent addresses
> - Ability to spoof inbound (towards the client) and outbound (from the client)
> - Where along the path filtering is observed
> - Presence of a NAT device along the path

6. Yes. https://spoofer.caida.org/report.php?sessionid=1014179

Session	Timestamp (UTC)	Client IP Block	ASN	Country	NAT	Outbound Private Status	Outbound Routable Status	Adj Spoof Prefix Len	Results
1014179	2020-10-25 11:22:05	109.106.59.x/24	15600 (FINECOM)	che (Switzerland)	yes	unknown	rewritten	none	Report
1014126	2020-10-25 09:01:05	85.195.251.x/24	13030 (INIT7)	che (Switzerland)	yes	rewritten	rewritten	none	Report
1013902	2020-10-24 22:24:49	109.106.59.x/24	15600 (FINECOM)	che (Switzerland)	yes	unknown	rewritten	none	Report
1013870	2020-10-24 21:02:40	85.195.251.x/24	13030 (INIT7)	che (Switzerland)	yes	rewritten	rewritten	none	Report
1013865	2020-10-24 20:53:26	109.106.59.x/24	15600 (FINECOM)	che (Switzerland)	yes	unknown	rewritten	none	Report
1013376	2020-10-23 21:52:45	77.56.24.x/24	6830 (LGI-UPC)	che (Switzerland)	yes	unknown	rewritten	none	Report
1013369	2020-10-23 21:33:01	77.56.24.x/24	6830 (LGI-UPC)	che (Switzerland)	yes	unknown	rewritten	none	Report
1013365	2020-10-23 21:30:08	77.56.24.x/24	6830 (LGI-UPC)	che (Switzerland)	yes	unknown	rewritten	none	Report
1013362	2020-10-23 21:23:14	77.56.24.x/24	6830 (LGI-UPC)	che (Switzerland)	yes	unknown	rewritten	none	Report