

IT PROJECT DOCUMENTATION

DHCP Discover Flood Attack –
Detection, Analysis & Recovery

MORGAN BURERA

JULY 2025

Google IT Support
Professional Certificate

Ubuntu, Terminal

Screenshots of the first part



Key Takeaways

```
morgan@morganclient:~$ sudo ip link set ens33 up
morgan@morganclient:~$ sudo dhclient -v ens33
Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
Corrupt lease file - possible data loss!
Listening on LPF/ens33/00:0c:29:82:f2:2c
Sending on  LPF/ens33/00:0c:29:82:f2:2c
Sending on  Socket/fallback
xid: warning: no netdev with useable HWADDR found for seed's uniqueness enforcement
xid: rand init seed (0x68112ef2) built using gethostid
DHCPREQUEST for 192.168.251.135 on ens33 to 255.255.255.255 port 67 (xid=0x4521ef0a)
DHCPACK of 192.168.251.135 from 192.168.251.254 (xid=0xaef2145)
bound to 192.168.251.135 -- renewal in 879 seconds.
morgan@morganclient:~$
```

I started by setting up the environment correctly:

- Put the network interface down, then up
- Renew the IP address

```
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:82:f2:2c brd ff:ff:ff:ff:ff:ff
    altname enp2s1
        inet 192.168.251.136/24 brd 192.168.251.255 scope global dynamic noprefixroute ens33
            valid_lft 1762sec preferred_lft 1762sec
            inet 192.168.251.135/24 brd 192.168.251.255 scope global secondary dynamic ens33
                valid_lft 1770sec preferred_lft 1770sec
                inet6 fe80::20c:29ff:fe82:f22c/64 scope link
                    valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:82:f2:36 brd ff:ff:ff:ff:ff:ff
    altname enp2s5
        inet 172.16.199.129/24 brd 172.16.199.255 scope global dynamic ens37
            valid_lft 1583sec preferred_lft 1583sec
            inet6 fe80::20c:29ff:fe82:f236/64 scope link
                valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
        inet 10.8.0.1/24 scope global tun0
            valid_lft forever preferred_lft forever
            inet6 fe80::a483:6d2d:bfed:31a5/64 scope link stable-privacy
                valid_lft forever preferred_lft forever
morgan@morganclient:~$
```

Verifying the network interface.

```
arcady@arcadyserver:~$ sudo yersinia -D -I dhcp -i ens33 a 1  
[sudo] password for arcady:
```

This command was done on the server vm to start the dhcp rogue attack on 'yersinia' but i had to do it manually on it's interface.

Testing of the attack on yersinia

I repeated the attack :

- 1 : Sending Discover packet



```
morgan@morganclient:~$ sudo tcpdump -i ens33 -n port 67 or port 68 -w dhcp_flood
1.pcap
tcpdump: listening on ens33, link-type EN10MB (Ethernet), snapshot length 262144
bytes
```

Start of the capture on the right ports using tcpdump and saving the capture on : dhcp_flood1.pcap (for auditing)

```
morgan@morganclient:~$ sudo tcpdump -i ens33 -n port 67 or port 68 -w dhcp_flood
1.pcap
tcpdump: listening on ens33, link-type EN10MB (Ethernet), snapshot length 262144
_bytes
```



```
morgan@morganclient:~$ sudo dhclient -v ens33
[sudo] password for morgan:
Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
Listening on LPF/ens33/00:0c:29:82:f2:c
Sending on LPF/ens33/00:0c:29:82:f2:c
Sending on Socket/fallback
xid: warning: no netdev with useable HWADDR found for seed's uniqueness enforcement
xid: rand init seed (0x68113059) built using gethostid
DHCPREQUEST for 192.168.251.135 on ens33 to 255.255.255.255 port 67 (xid=0x1880f51c)
DHCPREQUEST for 192.168.251.135 on ens33 to 255.255.255.255 port 67 (xid=0x1880f51c)
DHCPDISCOVER on ens33 to 255.255.255.255 port 67 interval 3 (xid=0xd2c7156)
DHCPDISCOVER on ens33 to 255.255.255.255 port 67 interval 3 (xid=0xd2c7156)
DHCPDISCOVER on ens33 to 255.255.255.255 port 67 interval 8 (xid=0xd2c7156)
DHCPDISCOVER on ens33 to 255.255.255.255 port 67 interval 8 (xid=0xd2c7156)
DHCPDISCOVER on ens33 to 255.255.255.255 port 67 interval 17 (xid=0xd2c7156)
```

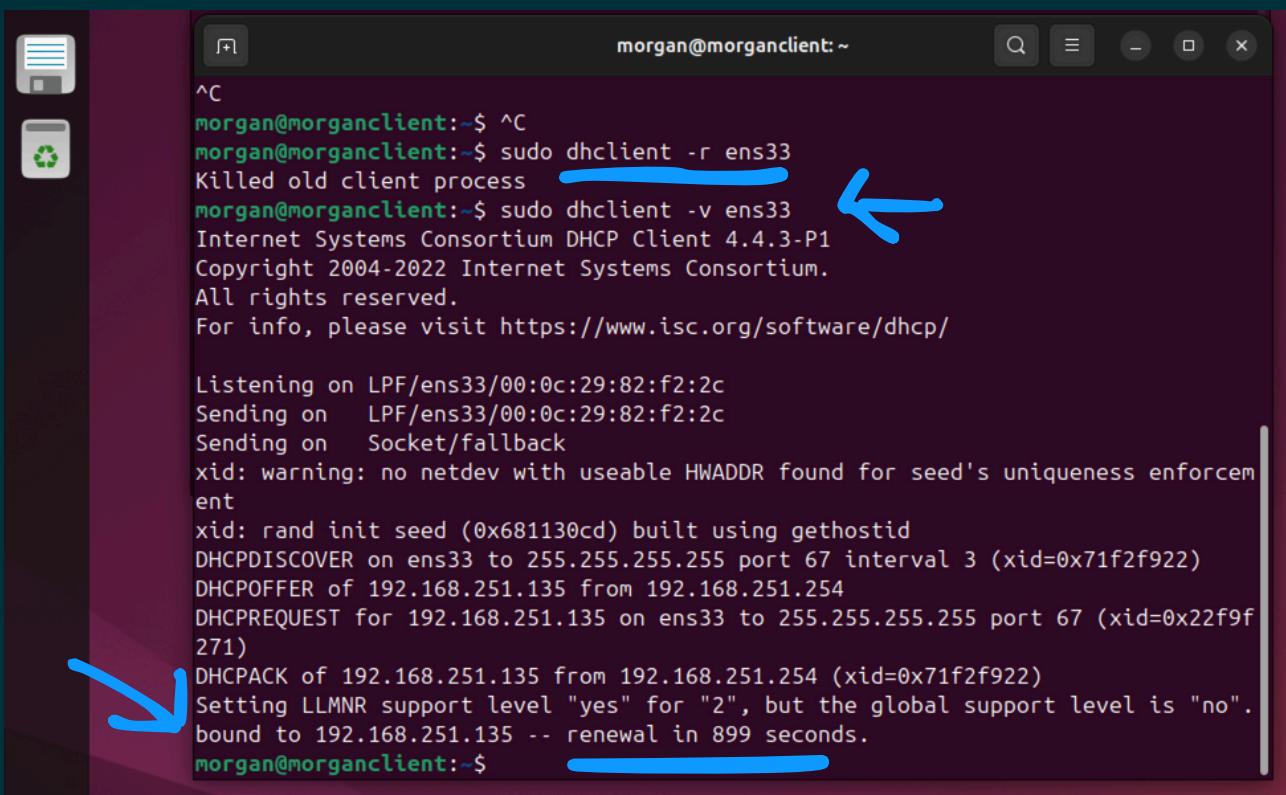
DHCP Fields	Total Packets: 21235
Source MAC	02:48:33:60:00:00
SIP	000.000.000.0000
Op	01
Htype	01
HLEN	00
CI	000.000.000.0000
YI	000.000.000.0000
CH	02:48:33:60:02:51



Start of the issue, due to the dhcp rogue attack every IPs addresses are taken so using 'sudo dhclient -v ens33' to assign an IP address to my client is failing.

Stoping the dhcp flood on yersinia using:

- sudo pkill Yersinia
or
 - q (from its interactive interface)



```
morgan@morganclient:~$ ^C
morgan@morganclient:~$ sudo dhclient -r ens33
Killed old client process
morgan@morganclient:~$ sudo dhclient -v ens33
Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/ens33/00:0c:29:82:f2:2c
Sending on  LPF/ens33/00:0c:29:82:f2:2c
Sending on  Socket/fallback
xid: warning: no netdev with useable HWADDR found for seed's uniqueness enforcement
xid: rand init seed (0x681130cd) built using gethostid
DHCPDISCOVER on ens33 to 255.255.255.255 port 67 interval 3 (xid=0x71f2f922)
DHCPOFFER of 192.168.251.135 from 192.168.251.254
DHCPREQUEST for 192.168.251.135 on ens33 to 255.255.255.255 port 67 (xid=0x22f9f271)
DHCPACK of 192.168.251.135 from 192.168.251.254 (xid=0x71f2f922)
Setting LLMNR support level "yes" for "2", but the global support level is "no".
bound to 192.168.251.135 -- renewal in 899 seconds.
morgan@morganclient:~$
```

Alright, although my laptop was heavily loaded — running the DHCP flood on the VM server, capturing packets on the client VM, and executing commands on another terminal; the capture process still ran without crashing.

After stopping the DHCP flood attack, the IP addresses that had been held "hostage" by Yersinia were finally released. As a result, my client successfully obtained an IP address: 192.168.251.135.

(This IP was dynamically assigned via DHCPv4. If Netplan had been set to static mode, the assignment would have failed unless I ran 'sudo netplan apply'. Also, every time I bring the ens33 interface down and back up, I have to manually clear the IP address first — otherwise, it won't refresh properly.)

```
morgan@morganclient:~
```

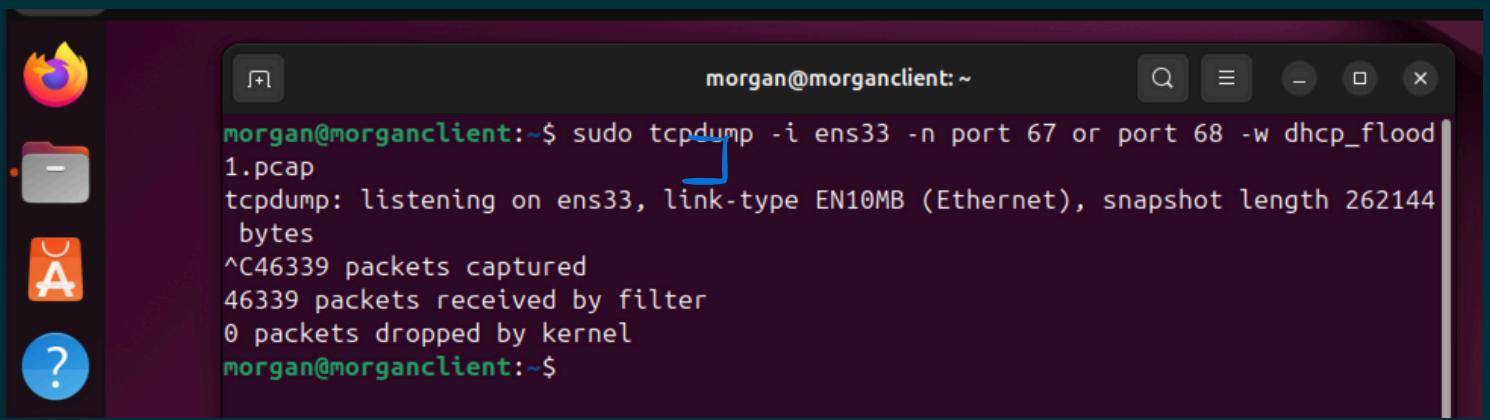
```
        valid_lft 1779sec preferred_lft 1779sec
    inet6 fe80::20c:29ff:fe82:f22c/64 scope link
        valid_lft forever preferred_lft forever
3: ens37: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gro
up default qlen 1000
    link/ether 00:0c:29:82:f2:36 brd ff:ff:ff:ff:ff:ff
    altname enp2s5
    inet 172.16.199.129/24 brd 172.16.199.255 scope global dynamic ens37
        valid_lft 1118sec preferred_lft 1118sec
    inet6 fe80::20c:29ff:fe82:f236/64 scope link
        valid_lft forever preferred_lft forever
4: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UNKNOWN group default qlen 500
    link/none
    inet 10.8.0.1/24 scope global tun0
        valid_lft forever preferred_lft forever
    inet6 fe80::a483:6d2d:bfed:31a5/64 scope link stable-privacy
        valid_lft forever preferred_lft forever
morgan@morganclient:~$ ip route
default via 192.168.251.2 dev ens33
10.8.0.0/24 dev tun0 proto kernel scope link src 10.8.0.1
172.16.199.0/24 dev ens37 proto kernel scope link src 172.16.199.129
192.168.251.0/24 dev ens33 proto kernel scope link src 192.168.251.135
morgan@morganclient:~$
```

I verified the network interface to see if it was back to a normal state :

- ip a
- ip route

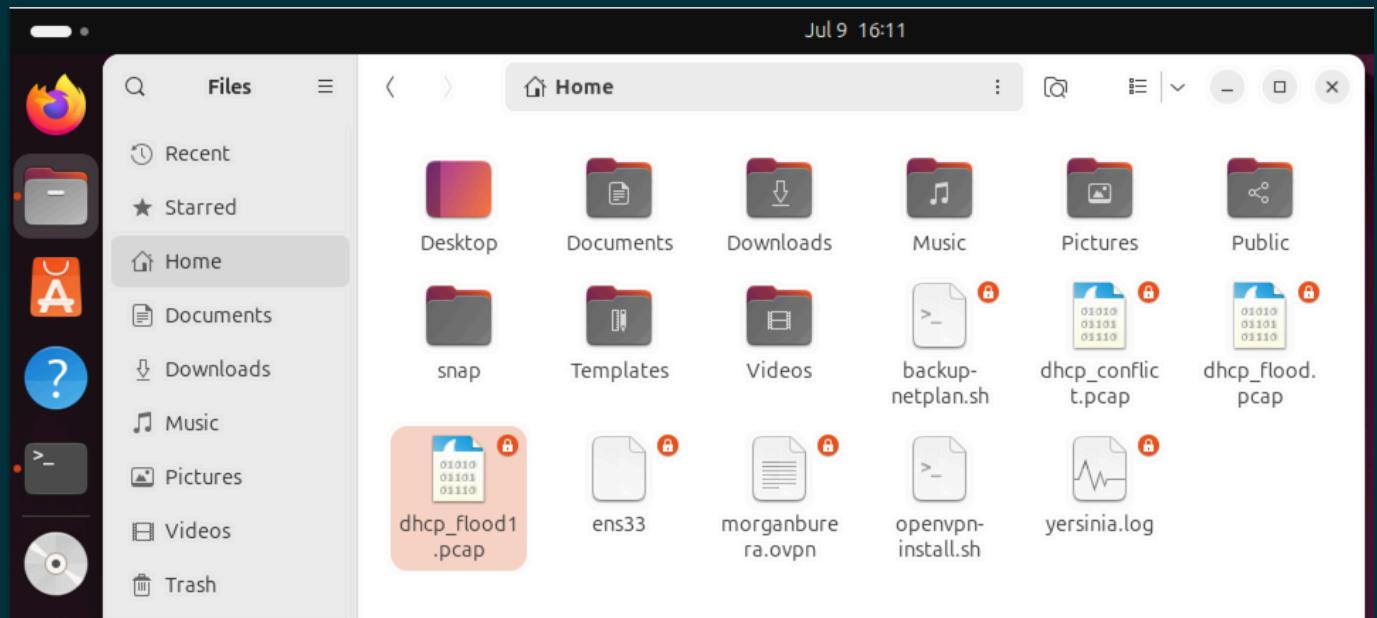
```
morgan@morganclient:~
```

```
10.8.0.0/24 dev tun0 proto kernel scope link src 10.8.0.1
172.16.199.0/24 dev ens37 proto kernel scope link src 172.16.199.129
192.168.251.0/24 dev ens33 proto kernel scope link src 192.168.251.135
morgan@morganclient:~$ sudo dhclient -r ens33
Killed old client process
morgan@morganclient:~$ sudo ip link set ens33 down
morgan@morganclient:~$ sudo ip link set ens33 up
morgan@morganclient:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:82:f2:2c brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.251.136/24 brd 192.168.251.255 scope global dynamic noprefixroute ens33
        valid_lft 1796sec preferred_lft 1796sec
    inet6 fe80::20c:29ff:fe82:f22c/64 scope link
        valid_lft forever preferred_lft forever
```



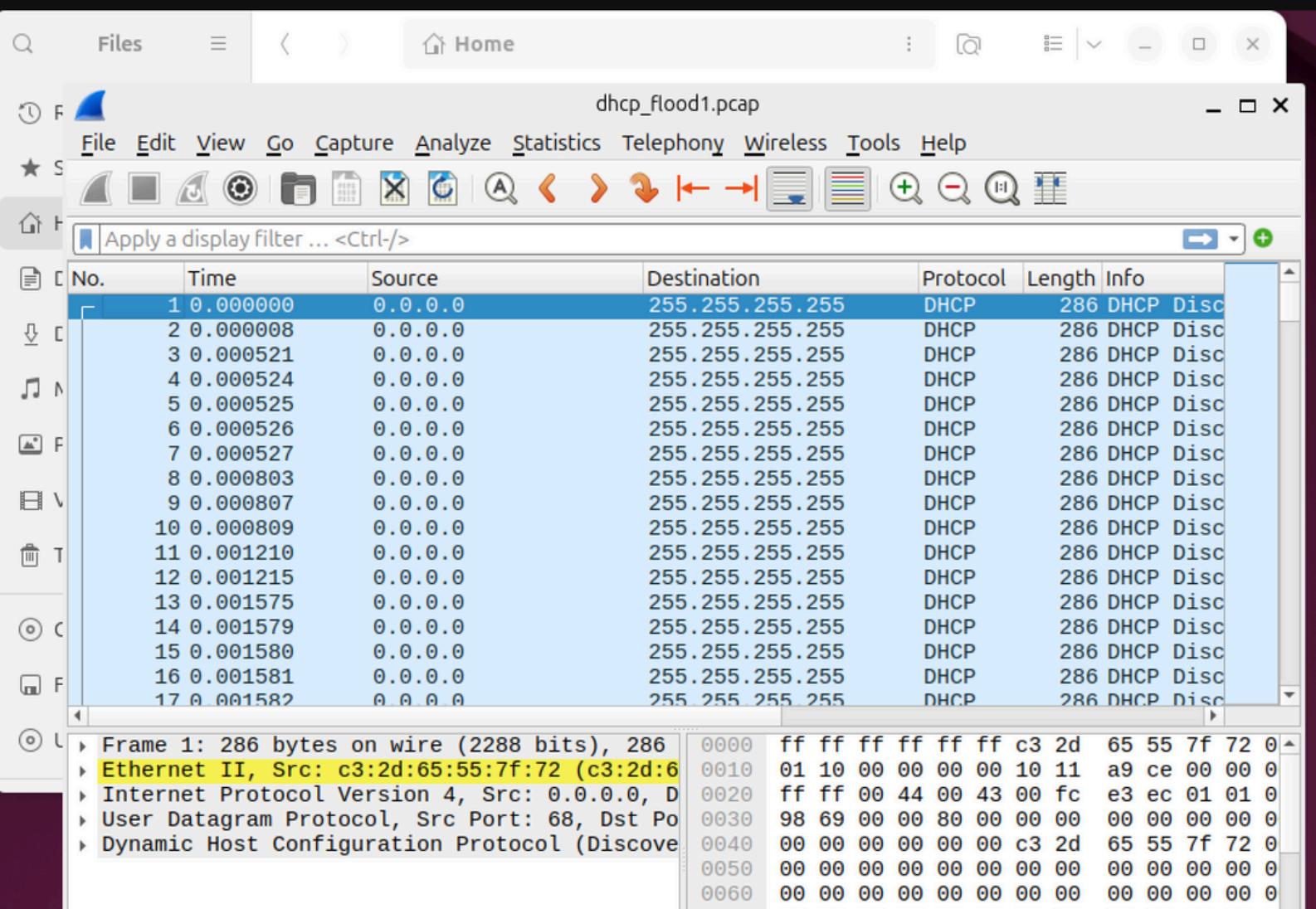
```
morgan@morganclient:~$ sudo tcpdump -i ens33 -n port 67 or port 68 -w dhcp_flood1.pcap
tcpdump: listening on ens33, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C46339 packets captured
46339 packets received by filter
0 packets dropped by kernel
morgan@morganclient:~$
```

Time to stop the capture, poor machine.



I wanted to inspect the capture from the saved file

Jul 9 16:12



We can clearly see the interpretation of the DHCP rogue attack, numerous attacks (0.0.0.0) overwhelming the server with DHCP Discoveries sent by broadcast.

Jul 9 16:12

The screenshot shows the Wireshark interface with a capture file named "dhcp_flood1.pcap". The main pane displays a list of network frames, mostly DHCP discover messages from various clients. A red bracket highlights the first few frames. The details and bytes panes below show the structure of a selected frame, which is identified as an Ethernet II frame (Src: c3:2d:65:55:7f:72) containing an IP header and a DHCP discovery message.

No.	Time	Source	Destination	Protocol	Length	Info
46324	256.594336	0.0.0.0	255.255.255.255	DHCP	286	DHCP Disc
46325	256.594337	0.0.0.0	255.255.255.255	DHCP	286	DHCP Disc
46326	256.594338	0.0.0.0	255.255.255.255	DHCP	286	DHCP Disc
46327	256.594339	0.0.0.0	255.255.255.255	DHCP	286	DHCP Disc
46328	256.594340	0.0.0.0	255.255.255.255	DHCP	286	DHCP Disc
46329	256.594341	0.0.0.0	255.255.255.255	DHCP	286	DHCP Disc
46330	256.594342	0.0.0.0	255.255.255.255	DHCP	286	DHCP Disc
46331	256.594342	0.0.0.0	255.255.255.255	DHCP	286	DHCP Disc
46332	261.141773	192.168.251.136	192.168.251.254	DHCP	342	DHCP Rele
46333	269.494662	0.0.0.0	255.255.255.255	DHCP	342	DHCP Disc
46334	270.495730	192.168.251.254	192.168.251.135	DHCP	342	DHCP Offe
46335	270.496199	0.0.0.0	255.255.255.255	DHCP	342	DHCP Requ
46336	270.502243	192.168.251.254	192.168.251.135	DHCP	342	DHCP ACK
46337	320.192308	192.168.251.135	192.168.251.254	DHCP	342	DHCP Rele
46338	338.005072	0.0.0.0	255.255.255.255	DHCP	338	DHCP Requ
46339	338.011172	192.168.251.254	192.168.251.136	DHCP	342	DHCP ACK

Frame 1: 286 bytes on wire (2288 bits), 286 bytes captured (2288 bits) on interface
Ethernet II, Src: c3:2d:65:55:7f:72 (c3:2d:6
Internet Protocol Version 4, Src: 0.0.0.0, D
User Datagram Protocol, Src Port: 68, Dst Po
Dynamic Host Configuration Protocol (Discove
0000 ff ff ff ff ff c3 2d 65 55 7f 72 0
0010 01 10 00 00 00 00 10 11 a9 ce 00 00 0
0020 ff ff 00 44 00 43 00 fc e3 ec 01 01 0
0030 98 69 00 00 80 00 00 00 00 00 00 00 0
0040 00 00 00 00 00 00 c3 2d 65 55 7f 72 0
0050 00 00 00 00 00 00 00 00 00 00 00 00 0

This is the end of the capture, just after stop Yersinia DHCP attack. We can observe the client was assigned a final IP address : 192.168.251.135.

The communication was terminated by the server's ACK (acknowledgement).

Also The difference in the packet lenght can make us understand that the machine requesting the IP address was different from the last one.



Key Takeaways

- DHCP Floods can paralyze a network by exhausting available IP addresses.
- Packet inspection tools (Wireshark, tcpdump) are essential to analyze and verify network behavior during and after an attack.
- Client-side testing on limited hardware (single laptop) is viable but demanding – separating roles across multiple machines improves accuracy and performance.
- Proper interface control (netplan apply, ifdown/ifup, clearing leases) is crucial to reinitialize DHCP communication reliably.