Assignment 2-Introduction to Network Analysis

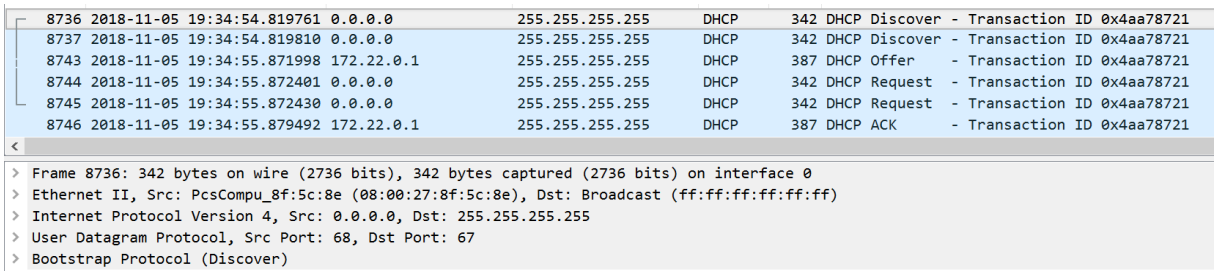
Haotian Yan 1855119

First part - IP address

a) In the terminal of **Virtual Machine**, enter "ip addr", could see its IP address: **172.22.189.200**
b) In the cmd of my **host machine,** enter "ipconfig", could see its IP address: **172.22.2.62**
c) In the WireShark, use filter: **ip.addr == 172.22.189.200**, could see all the activities done in the virtual machine.

Second part - How does the guest virtual machine obtain an IP address?

a) In the terminal of Virtual Machine, enter "**sudo dhclient -r**" and "**sudo dhclient**", to release IP and obtain the fresh IP. Through this process, the IP address of VM has been changed to port 67.
b) In WireShark, use filter "**ip.addr == 255.255.255.255**" , could see the four process that the guest VM obtain an IP address:
   i. DHCP Discover: the client machine request for having a IP address
   ii. DHCP Offer: the server response and offer an IP address
   iii. DHCP Request: the client machine request for choosing this IP
   iv. DHCP ACK: the server acknowledges the client's request and they reach an agreement



```
osboxes@osboxes:~$ sudo dhclient -r
osboxes@osboxes:~$ sudo dhclient -v
Internet Systems Consortium DHCP Client 4.3.5
Copyright 2004-2016 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/enp0s3/08:00:27:8f:5c:8e
Sending on   LPF/enp0s3/08:00:27:8f:5c:8e
Sending on   Socket/fallback
DHCPDISCOVER on enp0s3 to 255.255.255.255 port 67 interval 3 (xid=0x3b968e67)
DHCPREQUEST of 172.22.189.200 on enp0s3 to 255.255.255.255 port 67 (xid=0x678e9
3b)
DHCPOFFER of 172.22.189.200 from 172.22.0.1
DHCPACK of 172.22.189.200 from 172.22.0.1
RTNETLINK answers: File exists
bound to 172.22.189.200 -- renewal in 18983 seconds.
```

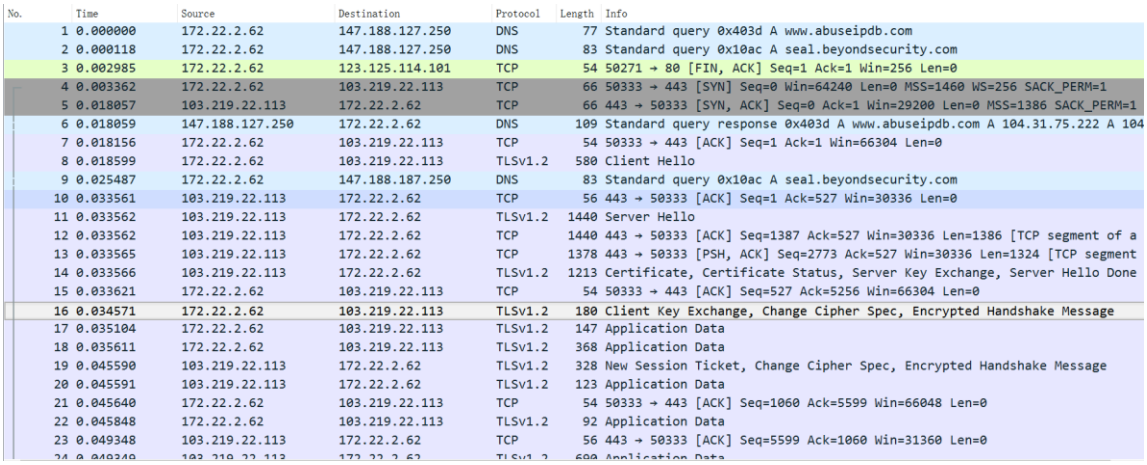
Third part

22	1.012600	10.5.102.43	10.5.0.1	DNS	72 Standard query 0x42ee A neverssl.com
23	1.041073	10.5.0.1	10.5.102.43	DNS	555 Standard query response 0x42ee A neverssl.com A 143.204.194.188 A 14
24	1.041863	10.5.102.43	143.204.194.188	TCP	66 59238 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
25	1.042197	10.5.102.43	143.204.194.188	TCP	66 59239 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
26	1.063737	143.204.194.188	10.5.102.43	TCP	68 80 → 59239 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM
27	1.063738	143.204.194.188	10.5.102.43	TCP	68 80 → 59238 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM
28	1.063830	10.5.102.43	143.204.194.188	TCP	54 59239 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
29	1.063906	10.5.102.43	143.204.194.188	TCP	54 59238 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0

Because after configuration, my virtual machine still only could see the incoming traffic, so I used my host machine instead. (I did this part at home, so the IP address has changed.) From the capture, we could see that:

- Use protocol: TCP
- Traffic:
  - First, the client machine tries to **find the IP address corresponding to the domain name of neverssl.com** through **DNS**
  - Then, the server **responded** the request and **returned** some information, including the **IP address** needed.
  - **Three handshakes** of TCP/IP
    - the client sent SYN to the server
    - the server sent SYN ACK to the client
    - the client sent ACK to the server

Fourth part



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	172.22.2.62	147.188.127.250	DNS	77	Standard query 0x403d A www.abuseipdb.com
2	0.000118	172.22.2.62	147.188.127.250	DNS	83	Standard query 0x10ac A seal.beyondsecurity.com
3	0.002985	172.22.2.62	123.125.114.101	TCP	54	50271 → 80 [FIN, ACK] Seq=1 Ack=1 Win=256 Len=0
4	0.003362	172.22.2.62	103.219.22.113	TCP	66	50333 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
5	0.018057	103.219.22.113	172.22.2.62	TCP	66	443 → 50333 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1386 SACK_PERM=1
6	0.018059	147.188.127.250	172.22.2.62	DNS	109	Standard query response 0x403d A www.abuseipdb.com A 104.31.75.222 A 104
7	0.018156	172.22.2.62	103.219.22.113	TCP	54	50333 → 443 [ACK] Seq=1 Ack=1 Win=66304 Len=0
8	0.018599	172.22.2.62	103.219.22.113	TLSv1.2	580	Client Hello
9	0.025487	172.22.2.62	147.188.187.250	DNS	83	Standard query 0x10ac A seal.beyondsecurity.com
10	0.033561	103.219.22.113	172.22.2.62	TCP	56	443 → 50333 [ACK] Seq=1 Ack=527 Win=30336 Len=0
11	0.033562	103.219.22.113	172.22.2.62	TLSv1.2	1440	Server Hello
12	0.033562	103.219.22.113	172.22.2.62	TCP	1440	443 → 50333 [ACK] Seq=1387 Ack=527 Win=30336 Len=1386 [TCP segment of a
13	0.033565	103.219.22.113	172.22.2.62	TCP	1378	443 → 50333 [PSH, ACK] Seq=2773 Ack=527 Win=30336 Len=1324 [TCP segment
14	0.033566	103.219.22.113	172.22.2.62	TLSv1.2	1213	Certificate, Certificate Status, Server Key Exchange, Server Hello Done
15	0.033621	172.22.2.62	103.219.22.113	TCP	54	50333 → 443 [ACK] Seq=527 Ack=5256 Win=66304 Len=0
16	0.034571	172.22.2.62	103.219.22.113	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
17	0.035104	172.22.2.62	103.219.22.113	TLSv1.2	147	Application Data
18	0.035611	172.22.2.62	103.219.22.113	TLSv1.2	368	Application Data
19	0.045590	103.219.22.113	172.22.2.62	TLSv1.2	328	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
20	0.045591	103.219.22.113	172.22.2.62	TLSv1.2	123	Application Data
21	0.045640	172.22.2.62	103.219.22.113	TCP	54	50333 → 443 [ACK] Seq=1060 Ack=5599 Win=66048 Len=0
22	0.045848	172.22.2.62	103.219.22.113	TLSv1.2	92	Application Data
23	0.049348	103.219.22.113	172.22.2.62	TCP	56	443 → 50333 [ACK] Seq=5599 Ack=1060 Win=31360 Len=0

- Use protocol: TCP and TLSv1.2
- Traffic:
  - First, the client machine tries to **find the IP address corresponding to the domain name** through **DNS**
  - Then, the server **responded** the request and **returned** some information, including the **IP address** needed.
  - Then the **three handshakes of TCP/IP**
    - the client sent SYN to the server
    - the server sent SYN ACK to the client
    - the client sent ACK to the server
  - Then the TLS part:
    - The client sent **Client Hello** to the server
    - The server sent **Server Hello** to the client
    - The client sent **Client Key Exchange, Change Cipher Spec, Encrypted handshake message** to the server
    - The server sent **New Session ticket, Change Cipher Spec**, **Encrypted handshake message** to the client
    - The client sent **Application Data** to the server
    - The server sent **Application Data** to the client