

CUSTOMIZED DHCP SERVER IN PYTHON USING SCAPY

After a successful installation of Python and Scapy in either Windows or Linux, copy and paste the code below in a file called scapy_dhcp.py and save it in your Scapy directory.

```
#input info
server_ip="12.154.254.33"
client_ip="10.154.254.15"
server_mac="00:0B:CD:AE:9F:C6"
client_mac="00:02:a5:ea:54:20"
subnet_mask="255.255.255.192"
gateway="12.154.254.10"

#DHCP leases
def detect_dhcp(pkt):
    #If DHCP Discover then DHCP Offer
    if pkt[DHCP] and pkt[DHCP].options[0][1] == 1:
        print "\nDHCP Discover packet detected"
        sendp(
            Ether(src=server_mac,dst="ff:ff:ff:ff:ff:ff")/
            IP(src=server_ip,dst="255.255.255.255")/
            UDP(sport=67,dport=68)/
            BOOTP(
                op=2,
                yiaddr=client_ip,
                siaddr=server_ip,
                giaddr=gateway,
                chaddr=client_mac,
                xid=pkt[BOOTP].xid
            )
            DHCP(options=[('message-type','offer')])/
            DHCP(options=[('subnet_mask',subnet_mask)])/
            DHCP(options=[('server_id',server_ip),('end')])
        )
        print "DHCP Offer packet sent\n."
    #If DHCP Request then DHCP Ack
    if pkt[DHCP] and pkt[DHCP].options[0][1] == 3:
        print "DHCP Request packet detected"
        sendp(
            Ether(src=server_mac,dst="ff:ff:ff:ff:ff:ff")/
            IP(src=server_ip,dst="255.255.255.255")/
            UDP(sport=67,dport=68)/
            BOOTP(
                op=2,
                yiaddr=client_ip,
```

```

siaddr=server_ip,
giaddr=gateway,
chaddr=client_mac,
xid=pkt[BOOTP].xid
)/
DHCP(options=[('message-type','ack')])/
DHCP(options=[('subnet_mask',subnet_mask)])/
DHCP(options=[('server_id',server_ip),('end')])

```

```

print "DHCP Ack packet sent\n\nCtrl+C to exit\n"
#exit when lease has been granted
#sys.exit(0)

```

#sniff DHCP requests

```

def start():
    sniff(filter="arp or (udp and (port 67 or 68))", prn=detect_dhcp, store=0)

```

Open Scapy and type the following:

```

import scapy_dhcp
scapy_dhcp.start()

```

The result can be seen using Wireshark

The screenshot shows a Wireshark capture of network traffic. The packet list pane displays several DHCP-related packets:

No.	Source	Destination	Protocol	Info
80291	NortelNe_d9:a0:09	Bay-Networks-(Synopti	SONMP	SONMP - FlatNet Hello
806452	12.154.254.15	12.154.254.33	DHCP	DHCP Release - Transaction ID 0xaf8eb72b
808130	NortelNe_d9:a0:61	Spanning-tree-(for-br	STP	Conf. Root = 32768/00:0e:40:d9:a0:01 Cost = 0 Port = 0x8061
8087418	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0xf9349e0d
8086475	12.154.254.33	255.255.255.255	DHCP	DHCP Offer - Transaction ID 0xf9349e0d
8087361	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0xf9349e0d
80893847	12.154.254.33	255.255.255.255	DHCP	DHCP ACK - Transaction ID 0xf9349e0d
8091112	HewlettP_ea:54:20	Broadcast	ARP	Gratuitous ARP for 12.154.254.15 (Request)
8098058	NortelNe_d9:a0:61	Spanning-tree-(for-br	STP	Conf. Root = 32768/00:0e:40:d9:a0:01 Cost = 0 Port = 0x8061
80985204	HewlettP_ea:54:20	Broadcast	ARP	Gratuitous ARP for 12.154.254.15 (Request)

The packet details pane for the selected DHCP Discover packet (No. 8087418) shows:

- Message type: DHCP Request packet detected
- Hardware type: Ethernet II, Src: NortelNe_d9:a0:09, Dst: 01:00:00:00:00:00
- Hardware address: 01:00:00:00:00:00
- Hops: 0
- Transaction ID: 0xf9349e0d
- Seconds elapsed since last message: 0
- Bootp flags: 0x00000000
- Client IP address: 0.0.0.0 (0.0.0.0)
- Your (client) IP address: 0.0.0.0 (0.0.0.0)

The terminal window shows the execution of Scapy:

```

Welcome to Scapy (1.2.0.2-win)
>>> import scapy_dhcp
>>> scapy_dhcp.start()
DHCP Discover packet detected
Sent 1 packets.
DHCP Offer packet sent
DHCP Request packet detected
Sent 1 packets.
DHCP Ack packet sent
Ctrl+C to exit
>>>

```

