

C Y B E R S E C U R I T Y

DHCP STARVATION AND SPOOFING

- Md Miraj Hasan (2005084)
- Wahid Al Azad Navid (2005089)



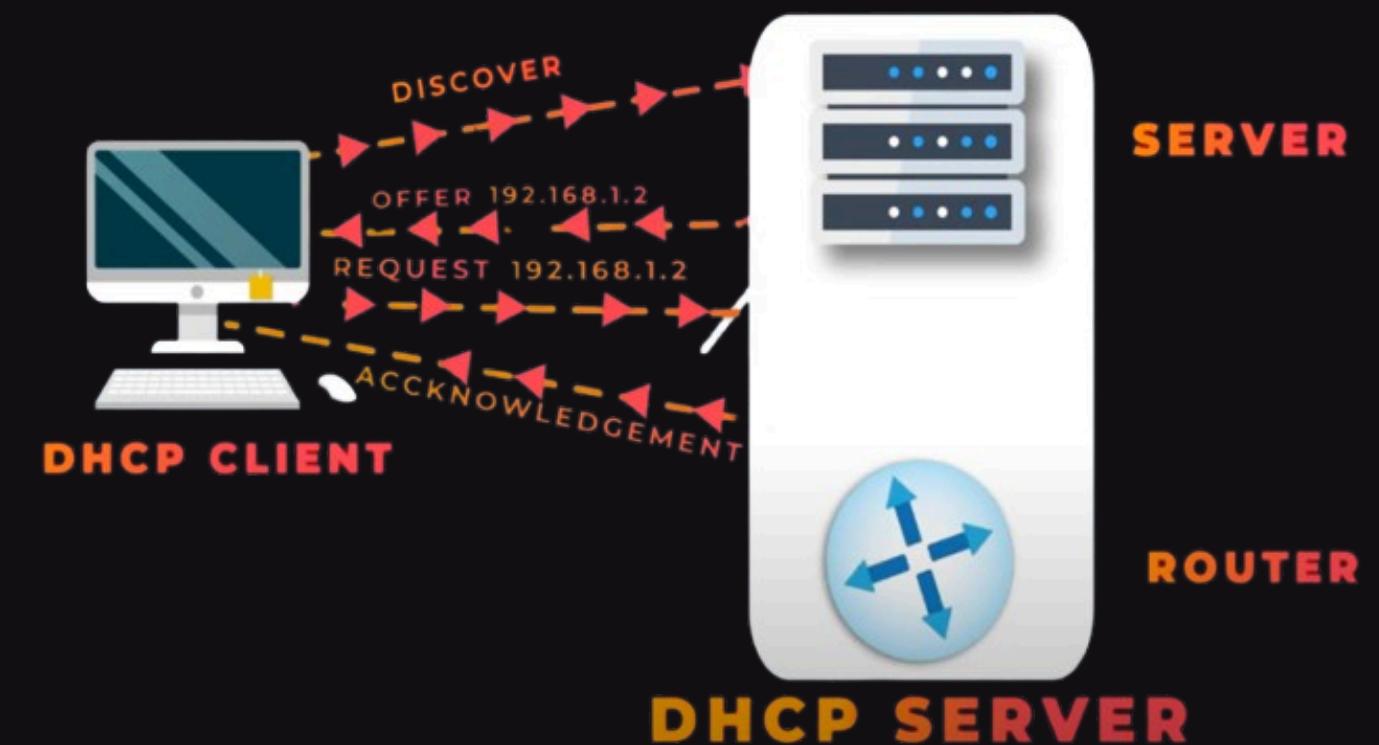
INTRODUCTION

- What is DHCP?

DHCP (Dynamic Host Configuration Protocol) is a network protocol that automatically assigns IP addresses and other network settings to devices when they join a network.

- DHCP – Dynamic Host Configuration Protocol

It ensures each device gets a unique IP without user involvement.



DHCP STARVATION

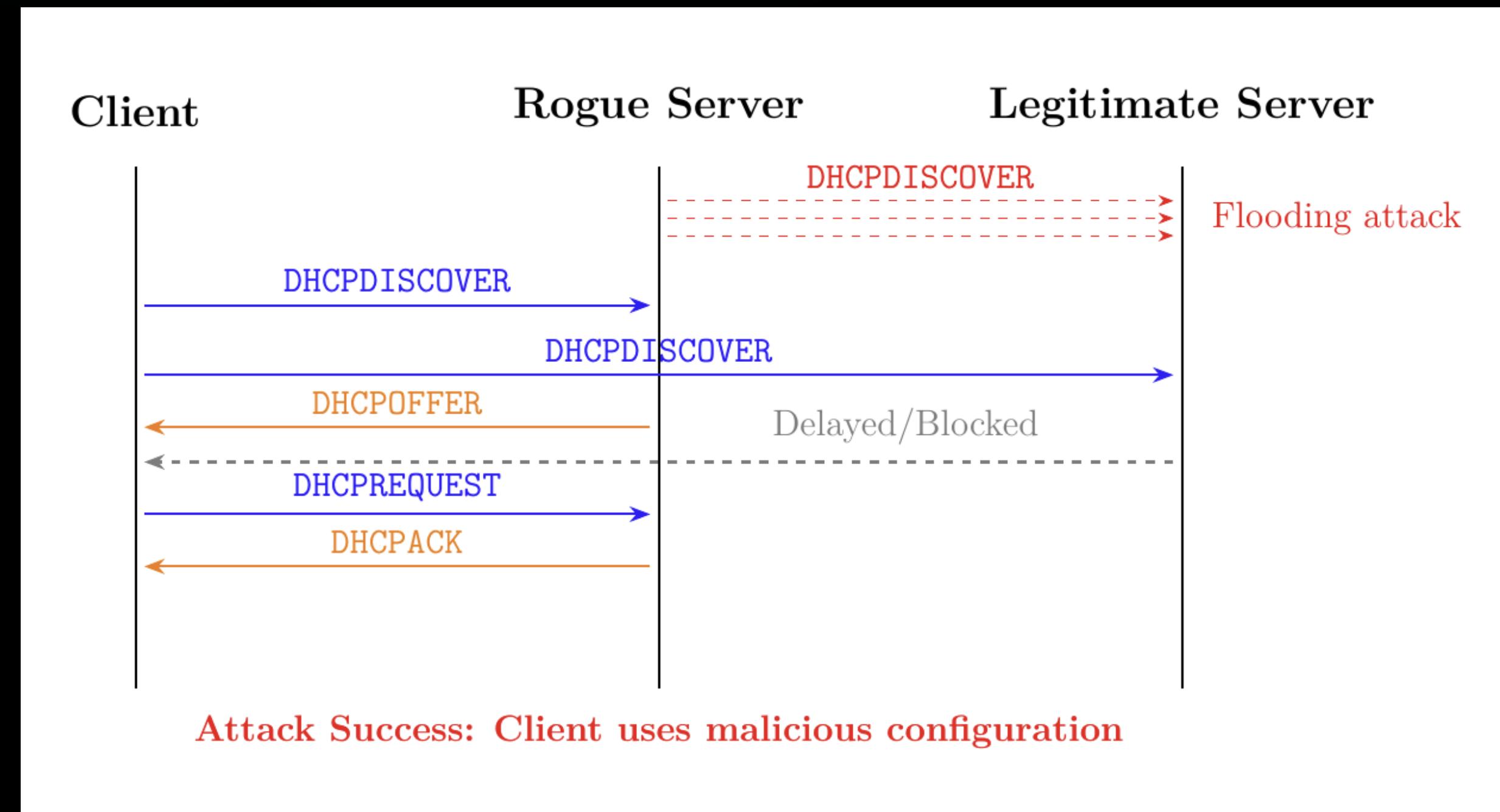
- An attacker floods the DHCP server with fake DHCPDISCOVER requests using spoofed MAC addresses, quickly exhausting its IP pool. As a result, legitimate clients are denied IP addresses.

DHCP SPOOFING

- After starving the legitimate server, the attacker runs a rogue DHCP server that responds to clients with malicious IP configurations, gaining control over their traffic.



ATTACK DESIGN



CONSEQUENCES OF THESE ATTACKS

- Network access denied to legitimate users (starvation)
- Users unknowingly connect to rogue server (spoofing)
- Leads to data interception, redirection, or MITM attacks
- Can bring entire networks down



ATTACK IMPLEMENTATION (OUR SIMULATION)

- NS-3 based setup with legitimate and rogue DHCP servers
- Multiple clients, 1 acting as attacker
- Rogue sends spoofed DHCP replies / floods DISCOVERs
- Evaluated both starvation and spoofing independently & together



ATTACK IMPLEMENTATION (OUR SIMULATION)

```
int rogue_pool = 250;
// Rogue DHCP Server (responds fast)
Ptr<DhcpServerApp> rogue = CreateObject<DhcpServerApp>();
rogue->Setup(Ipv4Address("192.168.100.1"), rogue_pool, port, MilliSeconds(1)); // fast
rogue->SetStartTime(Seconds(3.0));
rogueServer.Get(0)->AddApplication(rogue);

// Legitimate DHCP Server (slower)
Ptr<DhcpServerApp> legit = CreateObject<DhcpServerApp>();
legit->Setup(Ipv4Address("10.10.10.1"), 100, port, MilliSeconds(3)); // slow
legit->EnableDefense(enableStarvatingDefense); // Enable defense mechanism
legitServer.Get(0)->AddApplication(legit);
legit->SetStartTime(Seconds(0.0));
```

```
if (m_isAttacker)
{
    for (uint32_t i = 0; i < m_numSpoofed; ++i)
    {
        Simulator::Schedule(Seconds(1.0) + i * m_interval,
                            &DhcpClientApp::SendSpoofedDiscover,
                            this,
                            i);
    }
}
```

```
for (uint32_t i = 0; i < numClients; ++i) {
    Ptr<Node> node = clients.Get(i);
    Ptr<DhcpClientApp> client = CreateObject<DhcpClientApp>();
    client->Setup(broadcastAddr, 67);

    if (i == 0) { // only the first node acts as attacker
        client->SetIsAttacker(true);
    }

    if(enableSpoofingDefense) {
        // Add legitimate DHCP server to whitelist
        client->EnableSpoofingDefense(true);
        client->AddTrustedServer(Ipv4Address("10.1.1.141")); // Legitimate server IP
    } else {
        // No spoofing defense, so add rogue server to whitelist
        client->EnableSpoofingDefense(false);
    }

    double jitter = (rand() % 100) / 1000.0; // 0-0.0995
    client->SetStartTime(Seconds(2.0 + i * 0.2 + jitter));
    client->SetStopTime(Seconds(20.0));
    node->AddApplication(client);
}
```

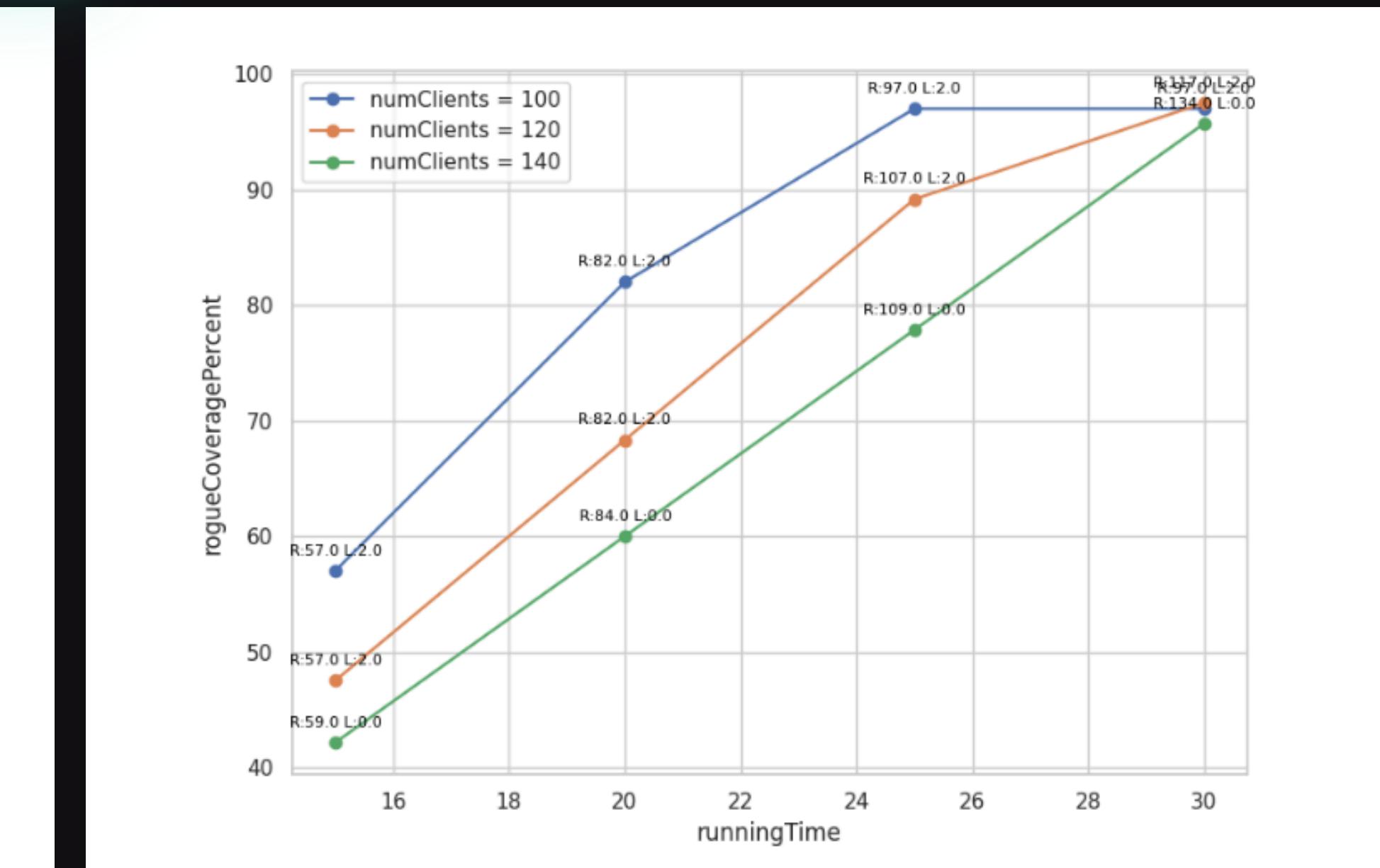
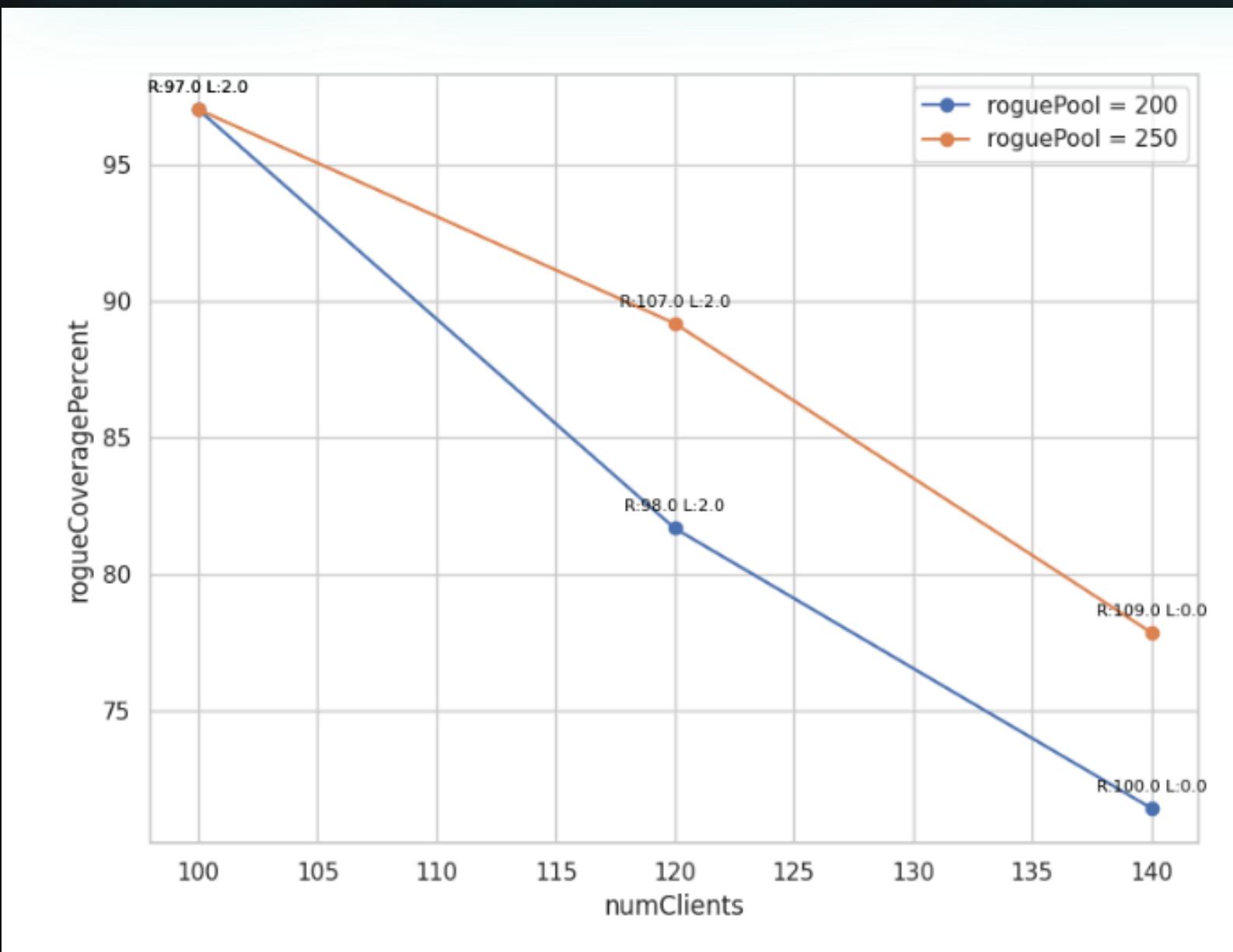


ATTACK IMPLEMENTATION (OUR SIMULATION)

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0x7dff9d09
4	0.004055	10.1.1.141	10.1.1.1	DHCP	290	DHCP Offer - Transaction ID 0x7dff9d09
5	0.010000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0x69e7f3e5
6	0.013030	10.1.1.141	10.1.1.1	DHCP	290	DHCP Offer - Transaction ID 0x69e7f3e5
7	0.020000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0x1816f8c4
8	0.024030	10.1.1.141	10.1.1.1	DHCP	290	DHCP Offer - Transaction ID 0x1816f8c4
9	0.030000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0x7ab49daf
10	0.034030	10.1.1.141	10.1.1.1	DHCP	290	DHCP Offer - Transaction ID 0x7ab49daf
11	0.040000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0x61e74ea3
12	0.043030	10.1.1.141	10.1.1.1	DHCP	290	DHCP Offer - Transaction ID 0x61e74ea3
13	0.050000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0xf819e7f
14	0.054030	10.1.1.141	10.1.1.1	DHCP	290	DHCP Offer - Transaction ID 0xf819e7f
15	0.060000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0x312167ad
16	0.063030	10.1.1.141	10.1.1.1	DHCP	290	DHCP Offer - Transaction ID 0x312167ad
17	0.070000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0x78b5e776
18	0.074030	10.1.1.141	10.1.1.1	DHCP	290	DHCP Offer - Transaction ID 0x78b5e776
19	0.080000	10.1.1.1	255.255.255.255	DHCP	290	DHCP Discover - Transaction ID 0x706b674e
						Frame 14: 290 bytes on wire (2320 bits), 290 bytes captured (2320 bits)
						Ethernet II, Src: 00:00:00_00:00:8d (00:00:00:00:00:8d), Dst: 00:00:00_00:00:01 (00:00:00:00:00:01)
						Internet Protocol Version 4, Src: 10.1.1.141, Dst: 10.1.1.1
						User Datagram Protocol, Src Port: 67, Dst Port: 49153
						Dynamic Host Configuration Protocol (Offer)
						Frame 1: 290 bytes on wire (2320 bits), 290 bytes captured (2320 bits)
						Ethernet II, Src: 00:00:00_00:00:01 (00:00:00:00:00:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
						Internet Protocol Version 4, Src: 10.1.1.1, Dst: 255.255.255.255
						User Datagram Protocol, Src Port: 49153, Dst Port: 67
						Dynamic Host Configuration Protocol (Discover)



ATTACK UNDER DIFFERENT CONFIGURATIONS



DEFENSE MECHANISM

- **Against DHCP Starvation**

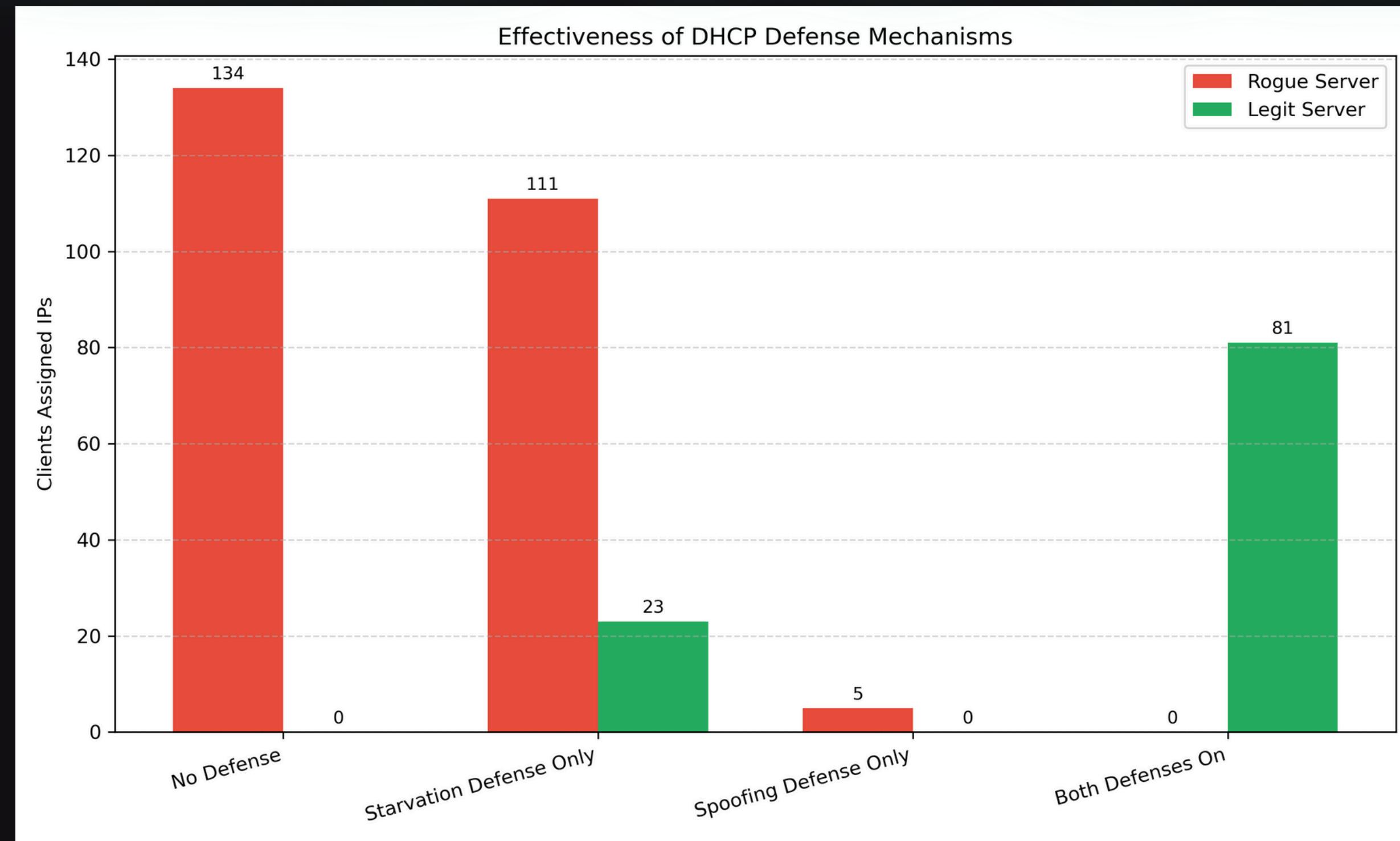
We implemented rate limiting on the server side. This mechanism tracks how frequently DHCP requests arrive from clients and flags any unusual burst of activity, effectively preventing an attacker from overwhelming the IP pool with rapid, spoofed requests.

- **Against DHCP Spoofing**

Our defense is based on IP whitelisting. Each client maintains a trusted list of DHCP server IPs. When a DHCPOFFER or DHCPACK is received, the client checks if the sender is on the whitelist. If not, the response is discarded, ensuring only legitimate configurations are accepted.



IMPROVEMENT AFTER DEFENSE



CYBERSECURITY PRESENTATION :

THANK YOU