

ASSIGNMENT 7

Name: Imon Raj
Class: BCSE III
Roll: 002010501098
Section: A3
Subject: Computer Networks Lab Report

PROBLEM STATEMENT: Implement any two protocols using TCP/UDP Socket as suitable.

1. ARP 2. BOOTP 3. DHCP

DESIGN: This assignment is based on upper layers' protocols. Among the three mentioned, I have implemented ARP(Address resolution protocol) and DHCP(Dynamic host configuration protocol). I have used Java as my programming language and an Object-oriented approach.

- ARP has the work to translate the logical address(IP address) in a network to the corresponding device's MAC address. Here, if one host wants to know the MAC address of another host whose IP is known, it will send a broadcast to all the hosts requesting the MAC. The target host will respond to the requesting machine with its MAC address. We will have a static table that contains IP addresses and their corresponding MAC addresses.
- DHCP is used to assign an IP address to a host in the network, dynamically. In the case of DHCP, no host has a fixed(static) IP address in a network. Also if a host is inactive for a while its IP is revoked, such that other new hosts can accommodate in the network. For obvious reasons, a host must renew its already assigned IP address regularly according to the given rules.

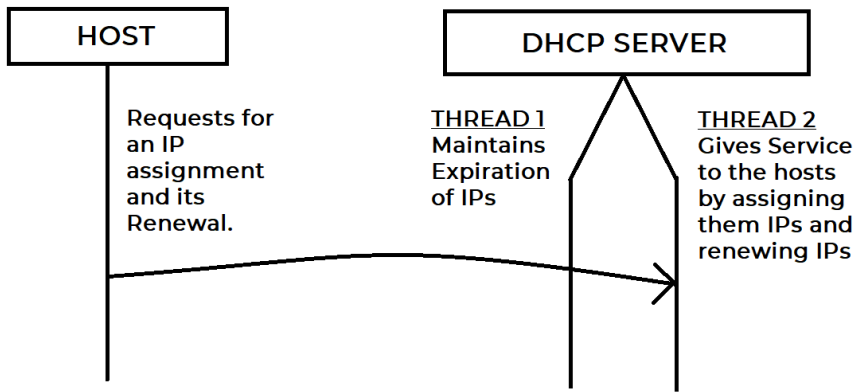
For our implementation purpose, we will have a DHCP server. It will run two concurrent threads internally. One thread will just check the expiration of IP addresses, which are not renewed after a certain duration, and must be freed. And another thread will communicate directly with hosts(Clients). Any Client request for DHCPDISCOVER or DHCPRENEW will be served by this thread. For each IP, we will have a Boolean indicating its availability, one timestamp to indicate its last renewal and an Integer to indicate the host which has been assigned this address.

IMPLEMENTATION:

All Java codes are uploaded in the shared Drive folder.

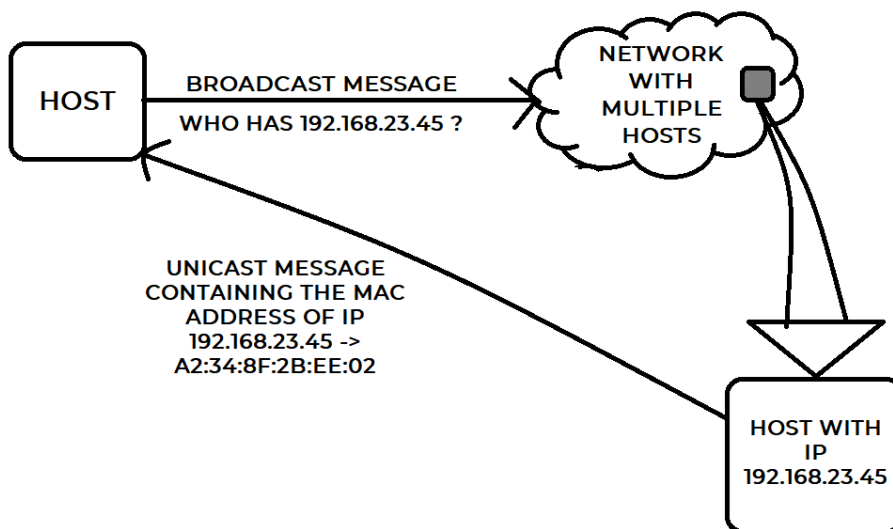
I have used TCP socket of Java for the implementations. I have used library classes – **ServerSocket** and **Socket** for creating Client-Server system. For reading data from socket and writing data into socket, **DataInputStream**, **DataOutputStream** classes are used. To keep track of time in case of DHCP, **TimeStamp** class is used.

To show the implementation model, I have attached a figure below for DHCP.



I have attached a figure for ARP implementation as well.

ARP IMPLEMENTATION



TEST CASES:

For ARP:

The program should be able to assure -

- Each IP in the network should have a corresponding MAC address.
- In case of a wrong(not present) IP, the program should show an error message.

For DHCP:

The program should be able to assure -

- If at least one IP is free, then a requesting host should be assigned.
- The DHCP server should continuously check the expirations of IPs.
- No host should be allowed to renew an IP which is already expired, or assigned to another host.

RESULTS & ANALYSIS:

DHCP OUTPUT:

```
-----  
1. DISCOVER NEW IP  
2. RENEW IP  
0. EXIT  
ENTER CHOICE: 1  
YOUR CLIENT_NO(AS IF PORT): 10  
CLIENT 10 IS ASSIGNED 120.120.30.0
```

```
-----  
1. DISCOVER NEW IP  
2. RENEW IP  
0. EXIT  
ENTER CHOICE: 1  
YOUR CLIENT_NO(AS IF PORT): 20  
CLIENT 20 IS ASSIGNED 120.120.30.1
```

```
-----  
1. DISCOVER NEW IP  
2. RENEW IP  
0. EXIT  
ENTER CHOICE: 2  
YOUR CLIENT_NO(AS IF PORT): 10  
SORRY, IP EXPIRED..
```

```
IP 120.120.30.0 EXPIRED  
IP 120.120.30.1 EXPIRED  
120.120.30.0
```

ARP OUTPUT:

```
Enter the Logical address(IP):  
165.165.80.80  
The Physical Address is: 6A:08:AA:C2:A3:23
```

This program is so far able to provide all the functionalities which are mentioned in the test-cases section.

COMMENTS:

This program was interesting to implement. There is chance of improvement in case of ARP implementation. We can have different approach rather than using a static table. More or less both the protocols' implementations are complete.