

MAYANK SINHA

RA1911003010386 G1

## WEEK-10: ARP IMPLEMENTATION USING UDP

Aim: To study ARP IMPLEMENTATION USING UDP.

Code:

```
#include<sys/types.h>

#include<sys/socket.h>

#include<net/if_arp.h>

#include<sys/ioctl.h>

#include<stdio.h>

#include<string.h>

#include<unistd.h>

#include<math.h>

#include<complex.h>

#include<arpa/inet.h>

#include<netinet/in.h>

#include<netinet/if_ether.h>

#include<net/ethernet.h>

#include<stdlib.h>
```

```
int main()
```

```
{
```

```

struct sockaddr_in sin={0};

struct arpreq myarp={{0}};

unsigned char *ptr;

int sd;

sin.sin_family=AF_INET;

printf("Enter IP address: ");

char ip[20];

scanf("%s", ip);

if(inet_pton(AF_INET,ip,&sin.sin_addr)==0)

{

    printf("IP address Entered '%s' is not valid \n",ip);

    exit(0);

}

memcpy(&myarp.arp_pa,&sin,sizeof(myarp.arp_pa));

strcpy(myarp.arp_dev,"eth0");

sd=socket(AF_INET,SOCK_DGRAM,0);

printf("\nSend ARP request\n");

if(ioctl(sd,SIOCGARP,&myarp)==1)

{

    printf("No Entry in ARP cache for '%s'\n",ip);

    exit(0);

}

ptr=&myarp.arp_pa.sa_data[0];

printf("Received ARP Reply\n");

```

```

        printf("\nMAC Address for '%s' : ",ip);

        printf("%p:%p:%p:%p:%p:%p\n",ptr,(ptr+1),(ptr+2),(ptr+3),(ptr+4),(ptr+5));

        return 0;
}

```

Output:-

```

20 #include <stdio.h>
21 #include <string.h>
22 #include <unistd.h>
23 #include <arpa/inet.h>
24 #include <sys/socket.h>
25 #include <sys/types.h>
26
27 struct arp {
28     char ip[16];
29     char dev[16];
30     struct sockaddr_in pa;
31     struct sockaddr_in data[4];
32 };
33
34 int sd_socket(int domain, int type, int protocol) {
35     int sockfd = socket(domain, type, protocol);
36     if (sockfd < 0) {
37         perror("socket");
38         exit(1);
39     }
40     return sockfd;
41 }
42
43 void send_arp_request(struct arp *arp, int sockfd) {
44     struct sockaddr_in to;
45     to.sin_family = AF_INET;
46     to.sin_port = htons(8080);
47     inet_pton(AF_INET, arp->ip, to.sin_addr.s_addr);
48     send(sockfd, arp, sizeof(arp), 0);
49 }
50
51 void receive_arp_reply(struct arp *arp, int sockfd) {
52     struct sockaddr_in from;
53     from.sin_family = AF_INET;
54     from.sin_port = htons(8080);
55     inet_pton(AF_INET, arp->ip, from.sin_addr.s_addr);
56     recv(sockfd, arp, sizeof(arp), 0);
57 }
58
59 int main() {
60     struct arp arp;
61     int sockfd = sd_socket(AF_INET, SOCK_DGRAM, 0);
62     printf("IP address Entered '%s' is not valid \n",ip);
63     exit(0);
64 }
65
66 memcpy(&myarp.pa, &sin, sizeof(myarp.pa));
67 strcpy(myarp.dev, "eth0");
68 sd_socket(AF_INET, SOCK_DGRAM, 0);
69 printf("\nSend ARP request\n");
70 if (ioctl(sockfd, SIOCGARP, &myarp) == 0) {
71     printf("No Entry in ARP cache for '%s'\n", ip);
72     exit(0);
73 }
74 ptr = &myarp.pa.sa_data[0];
75 printf("Received ARP Reply\n");
76 printf("\nMAC Address for '%s' : ", ip);
77 printf("%p:%p:%p:%p:%p:%p\n", ptr, (ptr+1), (ptr+2), (ptr+3), (ptr+4), (ptr+5));
78 return 0;

```

server386/exp9.c - Stop: x

Run Command: server386/exp9.c Runner: C CWD: ENV

Send ARP request  
Received ARP Reply

MAC Address for '192.168.1.2' : 0x7fffe736bff2:0x7fffe736bff3:0x7fffe736bff4:0x7fffe736bff5:0x7fffe736bff6:0x7fffe736bff7