

Assignment #2

Implementation of Simple ARP Program

데이터통신

컴퓨터학과

2015130741 원혜진

Assignment #2 – Implementation of simple ARP program

1. 프로젝트 개요	3
1. 배경 및 목적	3
2. 프로그램 요약	3
2. 프로그램 구현	3
0. Configuration	3
1. 프로그램 작동 과정	3
2. 프로그램 구조	4
3. 결과물	5
1. 프로그램 작동 과정	5
2. wireshark 스크린샷	6

1. 프로젝트 개요

1. 배경 및 목적

- 본 프로젝트의 목적은 Application layer에서 Ethernet을 직접적으로 사용하는 방법을 확인하는 것이다. 이를 위해, ARP request / reply를 구현하고자 한다. ARP란, IP module이 DLL SAP를 통해 IP packet을 전송할 때, next hop의 MAC 주소를 알아내기 위한 프로토콜이다.

2. 프로그램 요약

- 가상머신 VM1과 VM2를 유선 연결된 상태로 설정한다. VM1->VM2로 데이터를 보내고자 하며, VM2는 VM1이 전송한 패킷에 대해 답장을 보낸다.(VM2가 선제적으로 VM1에 패킷을 보내는 경우는 없다고 가정)

- VM1->VM2로 처음 패킷을 전송할 시, VM2의 MAC주소를 알지 못하므로 ARP가 선행되어야 한다. ARP를 통해 VM2의 MAC주소를 알아낸 후, VM2로 보내고자 했던 데이터를 전송한다.

- VM2로 데이터를 추가 전송한다. 이전 단계에서 VM1는 VM2의 MAC 주소를 알아냈으며, cache에 저장했으므로 ARP를 다시 실행하지 않고 데이터를 전송할 수 있다.

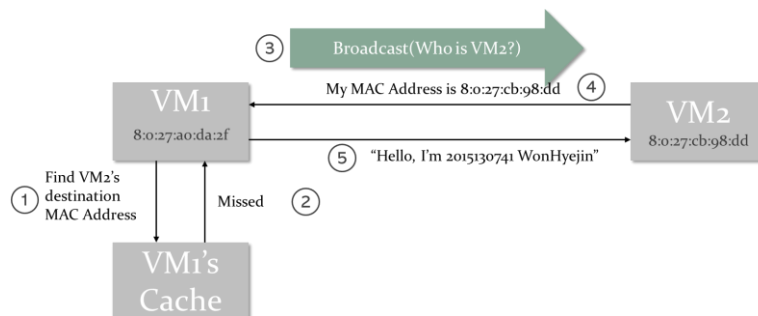
2. 프로그램 구현

0. Configuration

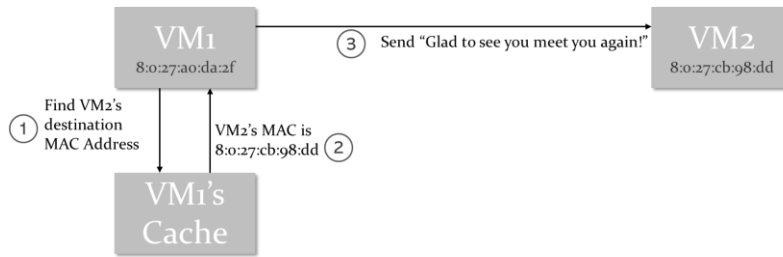
- interface: enp0s8
- VM1 MAC ADDRESS: 8:0:27:a0:da:2f
- VM2 MAC ADDRESS: 8:0:27:cb:98:dd
- Ethertype: 0xFFFD(data), 0xFFFE(ARP)

1. 프로그램 작동 과정

1-1. ARP (Ethertype: 0xFFFE)



1-2. Data(Ethertype: 0xFFFD)



2. 프로그램 구조

2-1. main 함수

VM1	VM2
1. socket 생성 2. ARP 후 data 보내기 2-1. VM2의 MAC주소가 없으므로, ARP request 패킷(0xFFFE) 전송 2-2. VM2의 ARP reply 패킷을 통해 cache에 VM2 MAC 주소 업데이트 2-3. data sending 패킷(0xFFFD) 전송 3. VM2에 data 패킷(0xFFFD) 전송 3-1. cache에서 VM2의 주소를 찾아 VM2에 data 패킷(0xFFFD) 전송	1. socket 생성 2. VM1의 패킷을 받음 2-1. ARP 패킷일 경우, 자신의 MAC 주소 정보를 담아 reply 2-2. Data 패킷일 경우, 답장 메시지를 담아 reply

2-2. main 외 주요 함수 정의

함수	설명
int init_socket(unsigned short ethr_type, int* sock_ll_ptr);	모든 ethernet 패킷을 송신하는 socket을 생성한다.(ETH_P_ALL) STATION_ADDR에 프로그램을 실행 중인 기기의 MAC 주소를 저장한다.
struct registered_dst* waitARPReply(char* vm_id);	ARP Request를 보낸 후, ARP reply를 받아 cache에 (id, MAC주소)를 저장하는 함수이다. cache는 linked list 형태로, 새로운 (id, MAC주소)는 cache의 맨 마지막에 추가된다.
void sendARPRequest(char* id);	id에 해당하는 기기의 MAC 주소를 알기 위해 실행하는 함수로, BROADCAST 주소에 frame을 전송한다. Frame은 (len, addr, id)로 구성되며, addr은 request를 보내는 기기의 MAC주소이다. 이는 상대방에서 reply할

	수 있도록, 자신의 MAC 주소를 알려주기 위함이다.
void sendFrameTo(char* id, unsigned short ethr_type, unsigned char* data, int len);	frame을 전송하기 전, id에 대응되는 주소값이 cache에 있는지 확인 후 frame을 전송하는 함수이다. Cache에 주소가 없다면, ARP를 실행한다.
void dispatchReceivedFrame(unsigned char* buff_ptr);	받은 frame의 Ethertype을 확인 후, 이에 따라 다음 행동을 결정하는 함수이다. 0xFFFF라면 "receiveARPFrame"을, 0xFFFD라면 "receiveDataFrame"을 실행한다.
void receiveARPFrame(unsigned char* dst_mac, unsigned char* arp_packet);	arp_packet이 자신의 MAC 주소를 요청하는 것인지 확인한다. 자신에게 온 요청이 아니라면 무시, 맞다면 ARP reply를 보낸다(Ethertype : 0xFFFE). (len, addr, name)으로 frame이 구성되며, 이 때 addr은 reply를 보내는 기기(VM2)의 MAC주소이다.
void receiveDataFrame(unsigned char* dst_mac, unsigned char* data_packet);	VM1: 받은 데이터를 콘솔에 출력한다. VM2: 답장 data frame을 상대방(VM1)에 전송한다.

3. 결과물

1. 프로그램 작동 과정

1-1. VM1

```
hyejin@hyejin-VirtualBox:~/share/assignment2$ sudo ./temp1 enp0s8 VM1 VM2
Failed to find entry.
Sending Message.. : Hello, I'm 2015130741 WonHyejin
Send ARP..
ARP Received. Target MAC Address is 8:0:27:cb:98:dd
Received Data : Nice to meet you!

Finding Entry Success
Sending Message.. : I'm Glad to meet you again.
Received Data : Nice to meet you!
```

1-2. VM2

```
hyejin@hyejin-VirtualBox:~/share/assignment2$ sudo ./temp2 enp0s8 VM2 VM1
ARP Received from 8:0:27:a0:da:2f.
Received Target name is mine. send ARP Reply..
Received Data : Hello, I'm 2015130741 WonHyejin
Received Data : I'm Glad to meet you again.
```

2. wireshark 스크린샷

No.	Time	Source	Destination	Protocol	Length	Info
12	5.114875845	fe80::fcb2:bcd9:510...	ff02::16	ICMPv6	110	Multicast Listener Report Mes
13	5.710787570	fe80::fcb2:bcd9:510...	ff02::16	ICMPv6	110	Multicast Listener Report Mes
14	5.806928079	fe80::fcb2:bcd9:510...	ff02::16	ICMPv6	110	Multicast Listener Report Mes
15	5.877650897	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	213	Standard query response 0x000
16	6.675980359	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction I
17	7.067313949	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	125	Standard query 0x0000 PTR _pg
18	7.958695775	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	213	Standard query response 0x000
19	8.001766411	fe80::ba15:19f1:3e9...	ff02::fb	MDNS	107	Standard query 0x0000 PTR _ip
20	8.869953404	fe80::fcb2:bcd9:510...	ff02::2	ICMPv6	62	Router Solicitation
21	11.069369958	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	125	Standard query 0x0000 PTR _pg
22	12.869327686	fe80::fcb2:bcd9:510...	ff02::2	ICMPv6	62	Router Solicitation
23	13.530737311	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction I
24	13.878904751	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction I
25	15.809348968	PcsCompu_a0:da:2f	Broadcast	0xffffe	29	Ethernet II
26	15.809819537	PcsCompu_cb:98:dd	PcsCompu_a0:da:2f	0xffffe	60	Ethernet II
27	15.809923582	PcsCompu_a0:da:2f	PcsCompu_cb:98:dd	0xffffd	45	Ethernet II
28	15.810332893	PcsCompu_cb:98:dd	PcsCompu_a0:da:2f	0xffffd	60	Ethernet II
29	15.810656077	PcsCompu_a0:da:2f	PcsCompu_cb:98:dd	0xffffd	41	Ethernet II
30	15.811034878	PcsCompu_cb:98:dd	PcsCompu_a0:da:2f	0xffffd	60	Ethernet II

- #25: VM1에서 ARP Request를 실행한다. ID가 "VM2"인 기기의 주소를 모르는 상태이므로, Broadcast 주소로 ARP Request를 보낸다.
- #26: VM2->VM1 ARP reply하는 frame이다. Frame의 data영역에는, VM2의 MAC주소가 기재되어 있다.
- #27: ARP request를 통해 VM2의 MAC 주소를 알아낸 VM1에서 data frame을 전송한다.

No.	Time	Source	Destination	Protocol	Length	Info
12	5.114875845	fe80::fcb2:bcd9:510...	ff02::16	ICMPv6	110	Multicast Listener Report Message v2
13	5.710787570	fe80::fcb2:bcd9:510...	ff02::16	ICMPv6	110	Multicast Listener Report Message v2
14	5.806928079	fe80::fcb2:bcd9:510...	ff02::16	ICMPv6	110	Multicast Listener Report Message v2
15	5.877650897	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	213	Standard query response 0x0000 PTR, cache fl
16	6.675980359	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x49db214e
17	7.067313949	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	125	Standard query 0x0000 PTR _pgkey-hkp._tcp.l
18	7.958695775	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	213	Standard query response 0x0000 PTR, cache fl
19	8.001766411	fe80::ba15:19f1:3e9...	ff02::fb	MDNS	107	Standard query 0x0000 PTR _ipps._tcp.local,
20	8.869953404	fe80::fcb2:bcd9:510...	ff02::2	ICMPv6	62	Router Solicitation
21	11.069369958	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	125	Standard query 0x0000 PTR _pgkey-hkp._tcp.l
22	12.869327686	fe80::fcb2:bcd9:510...	ff02::2	ICMPv6	62	Router Solicitation
23	13.530737311	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x49db214e
24	13.878904751	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0xaafc8e48
25	15.809348968	PcsCompu_a0:da:2f	Broadcast	0xffffe	29	Ethernet II
26	15.809819537	PcsCompu_cb:98:dd	PcsCompu_a0:da:2f	0xffffe	60	Ethernet II
27	15.809923582	PcsCompu_a0:da:2f	PcsCompu_cb:98:dd	0xffffd	45	Ethernet II
28	15.810332893	PcsCompu_cb:98:dd	PcsCompu_a0:da:2f	0xffffd	60	Ethernet II
29	15.810656077	PcsCompu_a0:da:2f	PcsCompu_cb:98:dd	0xffffd	41	Ethernet II
30	15.811034878	PcsCompu_cb:98:dd	PcsCompu_a0:da:2f	0xffffd	60	Ethernet II
31	19.071719786	fe80::fcb2:bcd9:510...	ff02::fb	MDNS	125	Standard query 0x0000 PTR _pgkey-hkp._tcp.l

Frame 27: 45 bytes on wire (360 bits), 45 bytes captured (360 bits) on interface 0	
Ethernet II, Src: PcsCompu_a0:da:2f (08:00:27:a0:da:2f), Dst: PcsCompu_cb:98:dd (08:00:27:cb:98:dd)	
Destination: PcsCompu_cb:98:dd (08:00:27:cb:98:dd)	
Source: PcsCompu_a0:da:2f (08:00:27:a0:da:2f)	
Type: Unknown (0xffffd)	
Data (31 bytes)	
Data: 48656c6c6f2c2049276d203230313531333037343120576f...	
[Length: 31]	

0000	08 00 27 cb 98 dd 08 00 27 a0 da 2f ff fd 48 65	...	He
0010	6c 6c 6f 2c 20 49 27 6d 20 32 30 31 35 31 33 30	...	110, I'm 2015130
0020	37 34 31 20 57 6f 6e 48 79 65 6a 69 6e	...	741 WonH yejin

- #28: VM2->VM1, VM1의 데이터를 잘 수신했음을 답장으로 보낸다.
- #29~#30: VM1->VM2, VM2->VM1 data frame을 주고 받는 패킷이다. VM1에서는 #26을 통해 VM2의 MAC 주소를 알아냈으므로, 추가적인 ARP 과정 없이 VM2에 data frame을 전송하였다.