

Before starting to capture the packets, use the command “sudo arp -d” to flush the arp cache. Besides, ipconfig/flushdns must be used to flush the DNS cache.

An ICMP packet broadcasting the address of the router is received.

1	0.000000	fe80::d0e1:9aff:f... ff02::1	ICMPv6	174 Router Advertisement from fc:51:a4:16:bf:64
▶	Frame 1: 174 bytes on wire (1392 bits), 174 bytes captured (1392 bits) on interface 0			
▲	Ethernet II, Src: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64), Dst: IPv6mcast_01 (33:33:00:00:00:01)			
▶	Destination: IPv6mcast_01 (33:33:00:00:00:01)			
▶	Source: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64)			
▶	Type: IPv6 (0x86dd)			
▶	Internet Protocol Version 6, Src: fe80::d0e1:9aff:feb7:2705, Dst: ff02::1			
▶	Internet Control Message Protocol v6			

## ICMP packet

The source is ArrisGro\_16:bf:64, and destination is a multicast interface IPv6mcast\_01. In IP packet, this source is replaced by fe80::d0e1:9aff:feb7:2705 (shown in the figure below), which means the default gateway of the network. Host discovers this broadcast message and use the information given to report errors when network problems occur. The destination is replaced by ff02::1, which means all the hosts open for IPv6 communication. ICMP works at networking layer.

连接特定的 DNS 后缀	hsd1.ma.comcast.net
IPv6 地址	2601:197:800:dcc2:2815:6bba:ca00:9b92
临时 IPv6 地址	2601:197:800:dcc2:2c04:cc6f:ee37:30e4
本地链接 IPv6 地址	fe80::2815:6bba:ca00:9b92%11
IPv4 地址	10.0.0.125
子网掩码	255.255.255.0
默认网关	fe80::d0e1:9aff:feb7:2705%11
	10.0.0.1

## ipconfig

Then the DHCP informs and confirms the IP address of our host.

10	6.536362	10.0.0.125	255.255.255.255	DHCP	342 DHCP Inform	- Transaction ID 0xe5028e8c
11	6.544203	10.0.0.1	10.0.0.125	DHCP	342 DHCP ACK	- Transaction ID 0xe5028e8c
▶	Frame 10: 342 bytes on wire (2736 bits), 342 bytes captured (2736 bits)					
▶	Ethernet II, Src: 70:18:8b:2e:57:df (70:18:8b:2e:57:df), Dst: Broadcast (ff:ff:ff:ff:ff:ff)					
▶	Internet Protocol Version 4, Src: 10.0.0.125 (10.0.0.125), Dst: 255.255.255.255 (255.255.255.255)					
▼	User Datagram Protocol, Src Port: bootpc (68), Dst Port: bootps (67)					
▶	Source port: bootpc (68)					
▶	Destination port: bootps (67)					
▶	Length: 308					
▶	Checksum: 0x670e [validation disabled]					
▶	Bootstrap Protocol					
▶	Frame 11: 342 bytes on wire (2736 bits), 342 bytes captured (2736 bits)					
▶	Ethernet II, Src: fc:51:a4:16:bf:64 (fc:51:a4:16:bf:64), Dst: 70:18:8b:2e:57:df (70:18:8b:2e:57:df)					
▶	Internet Protocol Version 4, Src: 10.0.0.1 (10.0.0.1), Dst: 10.0.0.125 (10.0.0.125)					
▼	User Datagram Protocol, Src Port: bootps (67), Dst Port: bootpc (68)					
▶	Source port: bootps (67)					
▶	Destination port: bootpc (68)					
▶	Length: 308					
▶	Checksum: 0x20c6 [validation disabled]					
▶	Bootstrap Protocol					

## DHCP Inform and ACK

The default gateway broadcast message to give the IP address to the our host. Our host gives back a confirm message. DHCP is working on port 67 on gateway and 68 on client, which is defined in RFC. DHCP is based on UDP, working at application layer.

The router needs to know the host's MAC address to deliver the packets. So it broadcasts the message below to find our host 10.0.0.125.

```
2 0.087284  ArrisGro_16:bf:64  HonHaiPr_2e:57:df  ARP      56 Who has 10.0.0.125? Tell 10.0.0.1
> Frame 2: 56 bytes on wire (448 bits), 56 bytes captured (448 bits) on interface 0
> Ethernet II, Src: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64), Dst: HonHaiPr_2e:57:df (70:18:8b:2e:57:df)
  Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64)
    Sender IP address: 10.0.0.1
    Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
    Target IP address: 10.0.0.125
```

#### ARP packet

The sender MAC address is ArrisGro\_16:bf:64(same as the ICMP packet sender), and its IP address is 10.0.0.1. This is a broadcast message to find the MAC address for the target IP address.

After our host (MAC address HonHaiPr\_2e:57:df) received this broadcast message, it respond to the default gateway with its MAC address, like saying "I'm here".

```
3 0.087299  HonHaiPr_2e:57:df  ArrisGro_16:bf:64  ARP      42 10.0.0.125 is at 70:18:8b:2e:57:df
> Frame 3: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
> Ethernet II, Src: HonHaiPr_2e:57:df (70:18:8b:2e:57:df), Dst: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64)
  Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: HonHaiPr_2e:57:df (70:18:8b:2e:57:df)
    Sender IP address: 10.0.0.125
    Target MAC address: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64)
    Target IP address: 10.0.0.1
```

#### Responding ARP packet

After sending and receiving the ARP packet, both the host and default gateway has cached the information to go on with further information. ARP works at data link layer

Then, the link <http://www.northeastern.edu> is entered into google chrome. The explorer tries to figure out the IP address of [www.northeastern.edu](http://www.northeastern.edu) by sending a DNS request.

87...	3.080201	10.0.0.125	75.75.75.75	DNS	80 Standard query 0x1a25 A www.northeastern.edu
89...	3.101003	75.75.75.75	10.0.0.125	DNS	96 Standard query response 0x1a25 A www.northeastern.edu A 155.33.17.68
89...	3.101286	10.0.0.125	75.75.75.75	DNS	80 Standard query 0xc440 AAAA www.northeastern.edu
90...	3.120797	75.75.75.75	10.0.0.125	DNS	138 Standard query response 0xc440 AAAA www.northeastern.edu SOA nb4276.neu.edu

### DNS packets

▷	Frame 8748: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface 0
▷	Ethernet II, Src: HonHaiPr_2e:57:df (70:18:8b:2e:57:df), Dst: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64)
▷	Internet Protocol Version 4, Src: 10.0.0.125, Dst: 75.75.75.75
▷	User Datagram Protocol, Src Port: 59801, Dst Port: 53
▲	Domain Name System (query)
	[Response In: 8901]
	Transaction ID: 0x1a25
▷	Flags: 0x0100 Standard query
	Questions: 1
	Answer RRs: 0
	Authority RRs: 0
	Additional RRs: 0
▲	Queries
▷	www.northeastern.edu: type A, class IN

### First DNS request

DNS is based on UDP protocol. Our host 10.0.0.125 sends a request for [www.northeastern.edu](http://www.northeastern.edu) from port 59801 to dns server 75.75.75.75, port 53. After receiving this request packet, DNS server 75.75.75.75 sends the answer back.

·	Frame 8901: 96 bytes on wire (768 bits), 96 bytes captured (768 bits) on interface 0
·	Ethernet II, Src: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64), Dst: HonHaiPr_2e:57:df (70:18:8b:2e:57:df)
·	Internet Protocol Version 4, Src: 75.75.75.75, Dst: 10.0.0.125
·	User Datagram Protocol, Src Port: 53, Dst Port: 59801
·	Domain Name System (response)
▲	Domain Name System (response)
	[Request In: 8748]
	[Time: 0.020802000 seconds]
	Transaction ID: 0x1a25
▷	Flags: 0x8180 Standard query response, No error
	Questions: 1
	Answer RRs: 1
	Authority RRs: 0
	Additional RRs: 0
▲	Queries
▷	www.northeastern.edu: type A, class IN
▲	Answers
▷	www.northeastern.edu: type A, class IN, addr 155.33.17.68

### Answer to First DNS request

The source IP is 75.75.75.75 source port 53. The destination IP is 10.0.0.125 port 59801. These reversed to those of the request packet. The answer 155.33.17.68 is in the answer part of the DNS answering packet. This is the IP for [www.northeastern.edu](http://www.northeastern.edu) . There is one answer to one question.

The second query is asking for IPv6 (TYPE AAAA) address for Northeastern.

```

▶ Frame 8923: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface 0
▶ Ethernet II, Src: HonHaiPr_2e:57:df (70:18:8b:2e:57:df), Dst: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64)
▶ Internet Protocol Version 4, Src: 10.0.0.125, Dst: 75.75.75.75
▶ User Datagram Protocol, Src Port: 52983, Dst Port: 53
▲ Domain Name System (query)
  [Response In: 9079]
  Transaction ID: 0xc440
  ▶ Flags: 0x0100 Standard query
  Questions: 1
  Answer RRs: 0
  Authority RRs: 0
  Additional RRs: 0
  ▲ Queries
    ▲ www.northeastern.edu: type AAAA, class IN
      Name: www.northeastern.edu
      [Name Length: 20]
      [Label Count: 3]
      Type: AAAA (IPv6 Address) (28)
      Class: IN (0x0001)

```

#### Second AAAA DNS request

The source and destination IP are the same with the first request, but the port for 10.0.0.125 changes to 52983 and port for 75.75.75.75 remains the same.

```

▲ Authoritative nameservers
  ▲ northeastern.edu: type SOA, class IN, mname nb4276.neu.edu
    Name: northeastern.edu
    Type: SOA (Start Of a zone of Authority) (6)
    Class: IN (0x0001)
    Time to live: 146
    Data length: 46
    Primary name server: nb4276.neu.edu
    Responsible authority's mailbox: postmaster.neu.edu
    Serial Number: 2011031412
    Refresh Interval: 10800 (3 hours)
    Retry Interval: 3600 (1 hour)
    Expire limit: 2592000 (30 days)
    Minimum TTL: 900 (15 minutes)

```

#### Second AAAA DNS answer

The port for client and DNS server remains the same as the request. The answering part for AAAA is different from type A. Instead of an IPv4 address, the DNS server is returning attributes such as name, type, class, TTL, Data Length, Primary name server and so on (shown in the figure above). DNS protocol is located at application layer.

After getting the IP address of <http://www.northeastern.edu>, the host starts the three way handshake with [www.northeastern.edu](http://www.northeastern.edu). The IP address of [www.northeastern.edu](http://www.northeastern.edu) is 155.33.17.68 and the port used by server is 80 (HTTP required). The port used by local host or client is random. In this assignment, the port for local host is 52926.

21	2.130740	10.0.0.125	155.33.17.68	TCP	66	52926→80	[SYN]	Seq=0	Win=8192	Len=0	MSS=1460	WS=4	SACK_PERM=1
23	2.150951	155.33.17.68	10.0.0.125	TCP	62	80→52926	[SYN, ACK]	Seq=0	Ack=1	Win=4380	Len=0	MSS=1460	SACK_PERM=1
24	2.151033	10.0.0.125	155.33.17.68	TCP	54	52926→80	[ACK]	Seq=1	Ack=1	Win=64240	Len=0		

### TCP three-way handshake

The IP address of local host is 10.0.0.125. The local host sends a TCP packet with Seq = 0 to start the three way handshake. The server receives the first handshake message and returns the second handshake with Seq = 0 and ACK = 1 to confirm the first handshake message. Then the local host receives the second handshake message and returns the third handshake with Seq = 1 and Ack = 1 to confirm the second handshake. The TCP connection is established between the local host and the server.

With the TCP connection established, the local host starts the HTTP by sending an HTTP GET packet to get the webpage requested.

25	2.151598	10.0.0.125	155.33.17.68	HTTP	463	GET / HTTP/1.1
▶ Frame 25: 463 bytes on wire (3704 bits), 463 bytes captured (3704 bits) on interface 0 ▶ Ethernet II, Src: HonHaiPr_2e:57:df (70:18:8b:2e:57:df), Dst: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64) ▶ Internet Protocol Version 4, Src: 10.0.0.125, Dst: 155.33.17.68 ▶ Transmission Control Protocol, Src Port: 52926, Dst Port: 80, Seq: 1, Ack: 1, Len: 409 ▶ Hypertext Transfer Protocol ▶ GET / HTTP/1.1\r\n Host: www.northeastern.edu\r\n Connection: keep-alive\r\n Cache-Control: max-age=0\r\n Upgrade-Insecure-Requests: 1\r\n User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/53.0.271						

### HTTP GET packet

After receiving the HTTP get packet from local host, the server starts to send the requested page to the local host. The server transmits the webpage by TCP segments, and the local host returns the Ack message correspondingly to these segment packets. If there are loss of packets, the server will start the fast retransmission or retransmission until all necessary parts are all delivered. Following is the sample data and Ack message during the connection.

48	2.198961	155.33.17.68	10.0.0.125	TCP	1514	[TCP segment of a reassembled PDU]
49	2.198962	155.33.17.68	10.0.0.125	TCP	1514	[TCP segment of a reassembled PDU]
50	2.198963	155.33.17.68	10.0.0.125	TCP	1514	[TCP segment of a reassembled PDU]
54	2.199094	10.0.0.125	155.33.17.68	TCP	54	52926→80 [ACK] Seq=410 Ack=7854 Win=64240 Len=0
55	2.199111	10.0.0.125	155.33.17.68	TCP	54	52926→80 [ACK] Seq=410 Ack=10221 Win=64240 Len=0
56	2.199124	10.0.0.125	155.33.17.68	TCP	54	52926→80 [ACK] Seq=410 Ack=13141 Win=64240 Len=0

### Sample data and Acks

Then, the server sends a 200 OK packet as the answer of the HTTP GET message for each HTTP GET message. Each HTTP GET packet would have a response. Following is a sample of HTTP response

made by the server.

```

318 2.311161 155.33.17.68 10.0.0.125 HTTP 639 HTTP/1.1 200 OK (text/javascript)
▶ Frame 318: 639 bytes on wire (5112 bits), 639 bytes captured (5112 bits) on interface 0
▶ Ethernet II, Src: ArrisGro_16:bf:64 (fc:51:a4:16:bf:64), Dst: HonHaiPr_2e:57:df (70:18:8b:2e:57:df)
▶ Internet Protocol Version 4, Src: 155.33.17.68, Dst: 10.0.0.125
▶ Transmission Control Protocol, Src Port: 80, Dst Port: 52926, Seq: 79915, Ack: 1090, Len: 585
▲ Hypertext Transfer Protocol
  ▶ HTTP/1.1 200 OK\r\n
    Date: Tue, 17 Jan 2017 19:48:12 GMT\r\n
    Server: Apache/2.2.15 (Red Hat)\r\n
    Last-Modified: Tue, 17 Jan 2017 15:03:06 GMT\r\n
    ETag: "422749-11d-5464b98f79d22"\r\n
    Accept-Ranges: bytes\r\n
    ▶ Content-Length: 285\r\n

```

## Sample HTTP 200 response

By the end of the whole connection, the server sends a FIN packet as the ending signal of the whole process. The following figure is the ending process of this connection.

58...	32.846207	155.33.17.68	10.0.0.125	TCP	56	80→52926	[FIN, ACK]	Seq=535794	Ack=3202	Win=7581	Len=0
58...	32.846267	10.0.0.125	155.33.17.68	TCP	54	52926→80	[ACK]	Seq=3202	Ack=535795	Win=64240	Len=0
58...	33.238708	10.0.0.125	155.33.17.68	TCP	54	52926→80	[FIN, ACK]	Seq=3202	Ack=535795	Win=64240	Len=0
58...	33.272281	155.33.17.68	10.0.0.125	TCP	56	80→52926	[ACK]	Seq=535795	Ack=3203	Win=7581	Len=0

## Ending process of connection

As shown in the figure given above, the server first send a FIN packet to show that this is the end of transmission. PSH flag can be set in this ending packet to show that there is still some payload in this packet. After receiving this FIN packet, the client or local host send the ACK packet for this FIN packet and another FIN ACK packet to end this connection. Finally, the server returns the final ACK for the last FIN ACK sent by the client. The whole process is ended. Following is the figure of part of flow for this connection. HTTP is based on TCP. It is located at application layer.

```

GET / HTTP/1.1
Host: www.northeastern.edu
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/53.0.2785.101 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Encoding: gzip, deflate, sdch
Accept-Language: zh-CN,zh;q=0.8

HTTP/1.1 200 OK
Date: Tue, 17 Jan 2017 19:48:12 GMT
Server: Apache/2.2.15 (Red Hat)
Last-Modified: Tue, 17 Jan 2017 15:03:04 GMT
ETag: "280230-1320a-5464b98dae191"
Accept-Ranges: bytes
Content-Length: 78346
Keep-Alive: timeout=30, max=500
Connection: Keep-Alive
Content-Type: text/html

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" lang="en-us">
<head>
<meta charset="utf-8"/>
<title>Northeastern University: a leader in global experiential learning in Boston, MA</title>
<meta content="Northeastern is a global, experiential, research university built on a tradition of engagement with the world, creating a distinctive approach to education" name="description"/>
<meta content="index,follow" name="robots"/>
<meta content="width=device-width, minimum-scale=1.0, maximum-scale=1.0" name="viewport"/>
<meta content="IE=Edge" http-equiv="X-UA-Compatible"/>
<!--[if lt IE 9]>
<script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
<![endif]>-->
<link href="https://www.northeastern.edu/css/html5styles.css" media="screen" rel="stylesheet" type="text/css"/><link href="https://fast.fonts.com/cssapi/cac43e8c-6965-44df-b8ca-9784607a3b53.css" rel="stylesheet" type="text/css"/><link href="css/mosaic.css" media="screen" rel="stylesheet" type="text/css"/><link href="css/styles.css" rel="stylesheet" type="text/css"/><link href="css/menu.css" media="screen" rel="stylesheet" type="text/css"/><!--[if gte IE 7]>
<link href="css/menu-ie.css" media="screen" rel="stylesheet" type="text/css" />
<![endif]>--><!--[if gte IE 9]>
<link href="css/video-ie.css" media="screen" rel="stylesheet" type="text/css" />
<![endif]>--><link href="css/flexslider.css" media="screen" rel="stylesheet" type="text/css"/><link href="css/mobile-nav.css" media="screen" rel="stylesheet" type="text/css"/>
<script src="scripts/modernizr.js" type="text/javascript">
// <![CDATA[
// ]]>
</script>
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js" type="text/javascript">

```