Basic Networking Concepts and Tools

Tony Espinoza

am.espinoza@utexas.edu

Networks

- What are some networks you are familiar with?
  - Local Area Network, home network.
  - Office network.
  - University network.

Networks

- Let's go into detail with a common network everyone uses every day.
- The Internet.
- What is the Internet?
  - On a basic level it is just a network of networks.

The Internet

- When going to a website how does your computer know where to go?
  - Type in the Uniform Resource Locator (URL) bar, e.g. google.com, utexas.edu...
- Your computer needs to translate that URL into something the network knows how to use.
  - Internet Protocol (IP) address.
  - utexas.edu $->$ 23.185.0.4

IP address

- Is a 32 bit number represented by a grouping of 4 octets.
    - 192.168.0.1
    - In hex: c0 a8 00 01

DNS[1] resolution

- How do domain names get resolved to IP addresses?
- i.e. How does my browser know how to take me to wikipedia.org
  - A query (IPv4)
  - AAAA query (IPv6)
- How to get IP address of wikipedia.org
  - `nslookup wikipedia.org`

[1] Domain Name System

nslookup output

```
> nslookup wikipedia.org

Server:     128.83.185.40
Address:    128.83.185.40#53

Non-authoritative answer:
Name:   wikipedia.org
Address: 208.80.153.224
Name:   wikipedia.org
Address: 2620:0:860:ed1a::1
```

Server: is the DNS server your computer is querying.

Address: is the DNS server and the port.

Why port 53?[2]

[2] Click

Your Local DNS server

For linux /etc/resolve.conf

```
> cat /etc/resolv.conf

# Generated by resolvconf
domain public.utexas.edu
nameserver 128.83.185.40
nameserver 128.83.185.41
```

Your Local DNS server

- How does your local DNS server know where to go?
- DNS is a distributed hierarchical database
  - Root DNS server
    - 13 labeled A-M
  - Top Level Domain (TLD) server
    - com, org, edu
  - Authoritative DNS server
    - amazon.com, pbs.org, utexas.edu

Example:

Let's look at wikipedia.org while recording a TCP dump which we will open with wireshark.

Tools:

- whois
  - Additional information about the IP address from the whois database
- dig
  - Similar to nslookup
- traceroute
  - Tries to find all the intermediary machines to a host
  - use with -T or -I and run as sudo
- nmap
  - -A Aggressive
  - -O OS detection

Tools:

- Zmap
  - Is a network tool for scanning the entire Internet (or large samples).
  - `wget http://64.106.81.7/blacklist.txt`
  - `sudo zmap --bandwidth=1M --target-port=80 --output-file=results.csv -b blacklist.txt`
- If we were to zmap ece.utexdas.edu how would we go about it?
  - Find out the range of IPs assigned to http://www.ece.utexas.edu/
    - dig or nslookup to get IP
  - Whois acquired IP to get the range of IP's in the network

RFC

- Request for Comments.
- Internet Engineering Task Force (IETF).
- Internet Research Task Force (IRTF).
- Internet Architecture Board (IAB).
- Independent authors.
- Engineers and computer scientists.

CIDR

- Classless Inter-Domain Routing.
- Notation for talking about ranges of IP address.
- Rare to see 192.168.0.0 - 192.168.0.255.
- Instead you would see 192.168.0.0/24.
- Equevalant to matching a netmask of 255.255.255.0.

CIDR

- Value after the / is called the prefix length.
- Number of address is
  - $2^{addressLength - prefixLength}$
- Prefix length is the number of leading 1's in the subnet netmask.

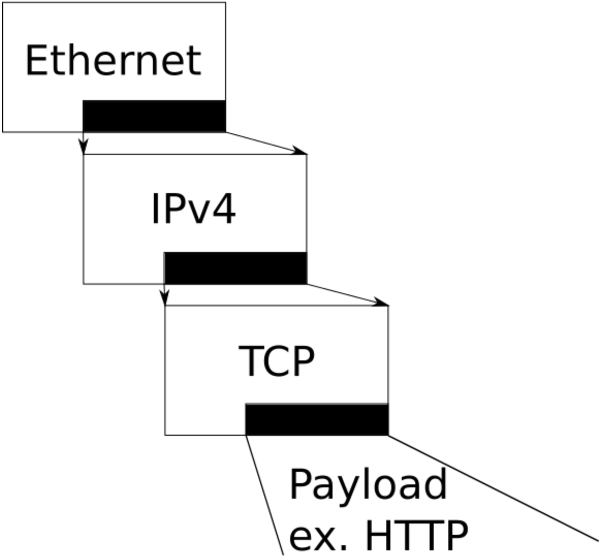
CIDR

- 0.0.0.0/8 = Class A
- 0.0.0.0/16 = Class B
- 0.0.0.0/24 = Class C

CIDR

- $/29$
  - $32 - 29 = 3$
  - $2^3 = 8$
- $/32$
  - size of 1
- $/9$
  - $32 - 9 = 23$
  - $2^{23} = 8388608$

Packets



Ethernet

Preamble	Destination MAC address	Source MAC address	Type	User Data	Frame Check Sequence (FCS)
8	6	6	2	46 - 1500	4

Preamble:Ethernet hardware filters this field so it won't be visible in wireshark

FCS:Often missing from wireshark

IPv4

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IHL: Internet Header Length, number of 32-bit words.

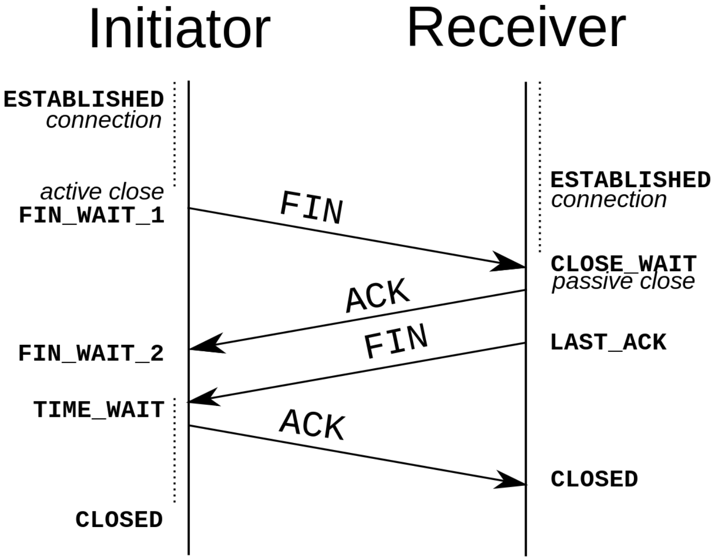
TCP

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Source Port          |       Destination Port        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Acknowledgment Number                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Data |       |C|E|U|A|P|R|S|F|                               |
| Offset|Resrvd |W|C|R|C|S|S|Y|I|            Window             |
|       |       |R|E|G|K|H|T|N|N|                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Checksum            |         Urgent Pointer        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Three way hand shake

- Client Sends SYN packet.
  - Client chooses a random sequence number.
- Server Sends SYN/ACK packet.
  - The acknowledgment number is set to one more than the received sequence number.
  - Server chooses a random sequence number.
- Client sends ACK packet.
  - The sequence number is set to the received acknowledgement value.
  - The acknowledgement number is set to one more than the received sequence number.

Terminate connection



But what if we don’t finish the handshake?

We end up with a half open connection.

- What is a half open connection?
- Two ways to store half open connections.
  - TCP backlog.
    - size: `sysctl net.ipv4.tcp_max_syn_backlog`
  - SYN cookies.
    - Stateless, require no system resources.
    - Limited in entropy.
    - Stored in the sequence number.

SYN cookies

Return a special sequence number where they encode the following:

- Top 5 bits: t mod 32, where t is a 32-bit time counter that increases every 64 seconds;
- Next 3 bits: an encoding of an MSS selected by the server in response to the client's MSS;
- Bottom 24 bits: a server-selected secret function of the client IP address and port number, the server IP address and port number, and t.

Why SYN cookies

- Pro
  - Defend against DOS/DDOS attacks
  - Stays up when SYN cache is exhausted
- Con
  - Loss of entropy
  - Attacks that require the attacker to know the initial sequence number are easier to execute with a decress of entropy.
  - Attacks: blind RST, blind injection, blind connection.

Sequence and Acknowledgment number

- Reliable transmission of data.
  - If a packet is not received, the protocol retransmits the data.
- Other uses of sequence numbers?
  - Out of order packets.

Windows

- Each endpoint has a receive buffer size.
- There are many ways to send data...
  - However sending one packet at a time can be wasteful.
- Windows are solution.
  - The receiver has a window of packets for which it will accept sequence numbers.
  - The sender has a window as well..
- Two common methods to implementing windows.
  - Go-Back-N [3]
  - Selective Repeat Protocol(SRP) [4]

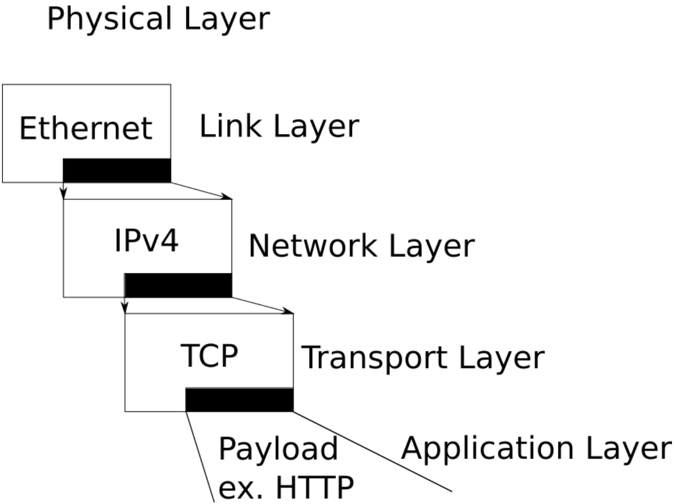
[3] Click the link

[4] Click the link

OSI stack

- Traditionally had 7 layers:
  - Application layer, presentation layer, session layer, transport layer, network layer, data link layer, and physical layer.
  - Antiquated as the OSI model was invented during the Internet's infancy.
- More common model is 5 layered.
  - Application
  - Transport
  - Network
  - Link
  - Physical

OSI stack


Physical Layer
Ethernet
Link Layer
IPv4
Network Layer
TCP
Transport Layer
Payload
ex. HTTP
Application Layer

Scapy

- Must use as sudo if you want to send packets.
- Can import the scapy library into python.
- Can use scapy to make send and receive packets.
- `IP()`
- `IP()/TCP()`
- `IP(dst="slashdot.org")/TCP()`
- `IP(dst="slashdot.org")/TCP(dport=80)`
- `IP(dst="slashdot.org")/TCP(dport=[80,443])`
- `z = IP(dst="slashdot.org")/TCP(dport=80)`
- r = sr(z)

Scapy

- `p = IP(dst="slashdot.org")/TCP(dport=80)`
- `p[1]` = TCP section
- In python `import scapy.all` give you everything but you need to use scapy.all.SCAPYFUNC
- `from scapy.all import IP, TCP, sr`
- use \ to compose e.g. `a = IP(dst="slashdot.org")/TCP(dport=80)/"GET / HTTP/1.0\r\n\r\n"`