Computer Systems & Network Administration

Lecture 13. TCP/IP Networking & PKI

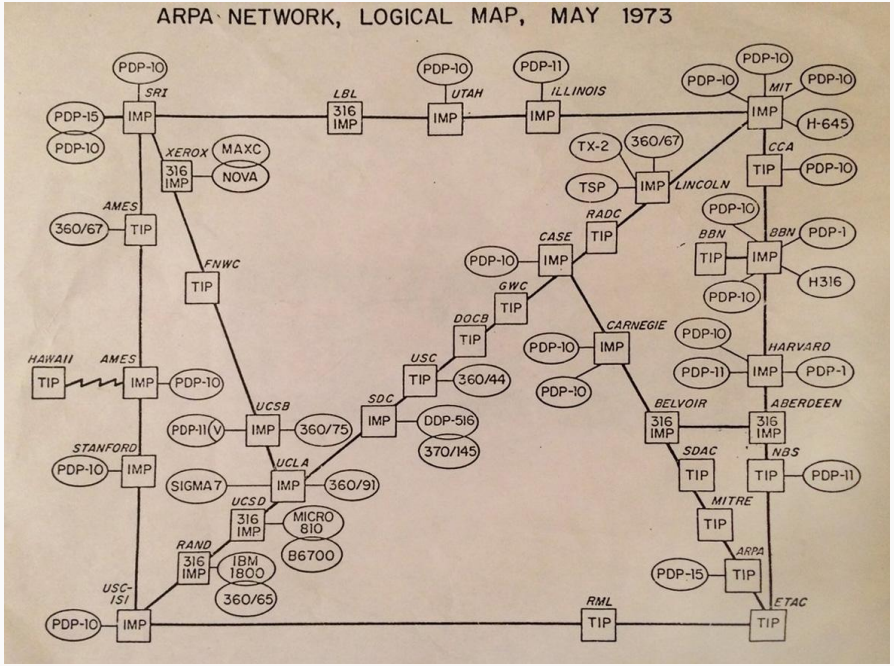
Reference: NYCU CSCC NASA

Outline

- Network Introduction
- TCP/IP suite
  - Link Layer
  - Network Layer
  - Transport Layer
- Network Setup
- PKI
- SSL/TLS
  - OpenSSL
- PGP

Intro - ARPANET

- Advanced Research Projects Agency NETwork
  - Network Control Program (NCP)
    - Provided connections and flow control between processes running on different ARPANET host computers



Intro - ARPANET (cont.)

- Design goal of ARPANET
  - No one point more critical than any other
  - Redundant routes to any destination
  - On-the-fly rerouting of data
  - Ability to connect different types of computers over different types of networks
  - Not controlled by a single corporation
- NCP got replaced by TCP/IP
  - NCP isn't evolved enough to handle growing clients

TCP/IP

- Transmission Control Protocol / Internet Protocol
- Gap between applications and Network
  - Network (IEEE 802)
    - 802.1 Higher Layer LAN Protocols Working Group
    - 802.3 Ethernet
    - 802.11 Wireless LAN & Mesh (Wi-Fi certification)
    - 802.15.1 Bluetooth certification
  - Application
    - Reliable & Performance

TCP/IP - Design Goal

- Hardware independence
- Software independence
- Failure recovery and the ability to handle high error rates
- Efficient protocol with low overhead
- Ability to add new networks to the internetwork without service disruption
- Routable Data

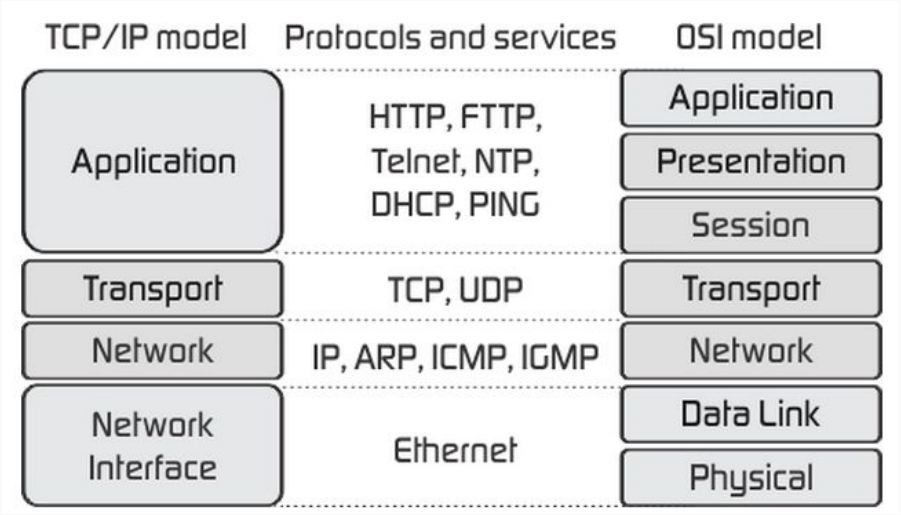
TCP/IP - Layers

- A suite of networking protocols
  - 4-layer architecture
    - Link Layer (Data-Link Layer)
    - Network Layer (IP)
    - Transport Layer (Port)
    - Application Layer

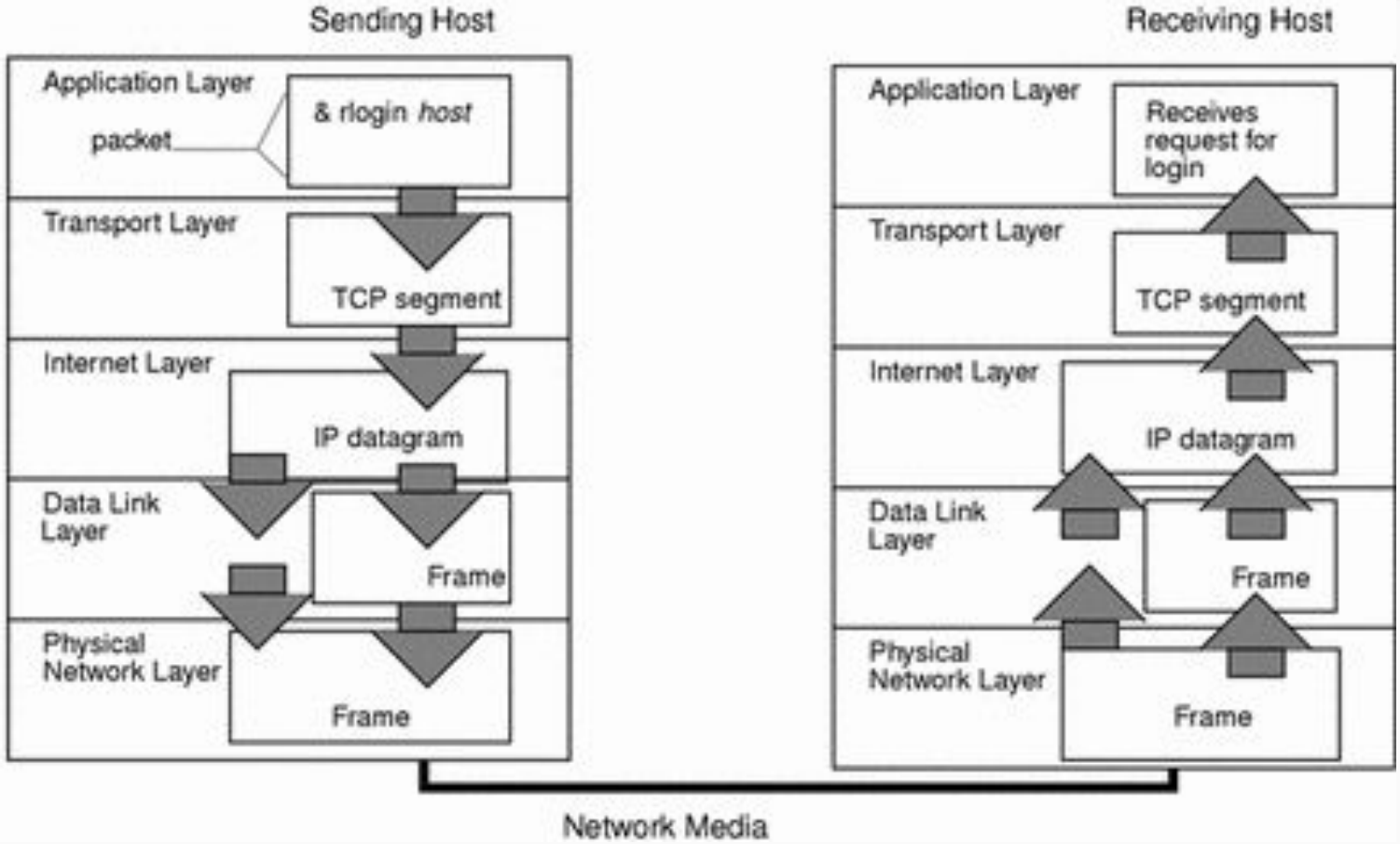
Application	SSH / Telnet / HTTP...
Transport	TCP / UDP
Network	IP / ICMP / IGMP
Link	Device Driver / Interface

TCP/IP - Layers - OSI model

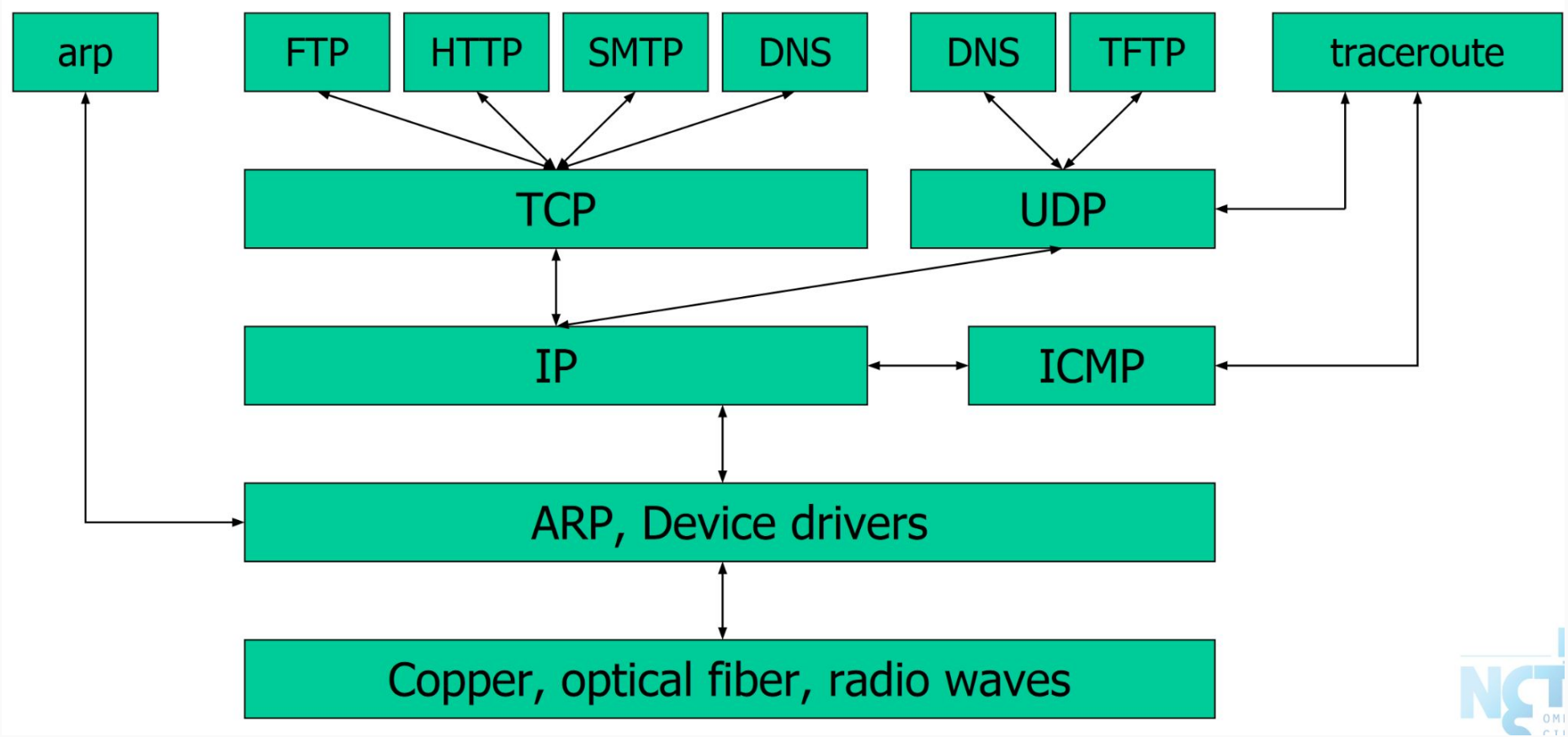
- ISO/OSI Model
  - Internatioal Organization for Standardization / Open System Interconnection Reference Model



TCP/IP - Layers - Encapsulation & Decapsulation

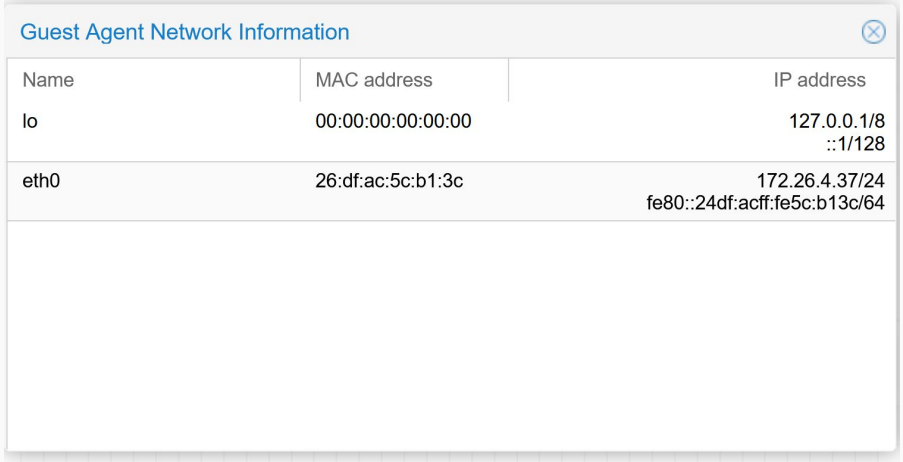


TCP/IP - TCP/IP Family



TCP/IP - Addressing

- IP
  - IPv4 (32-bit)
  - IPv6 (128-bit)
- Port
  - 16-bits (1 ~ 65535)
  - Uniquely Identify Application
- MAC Address
  - Media Access Control Address

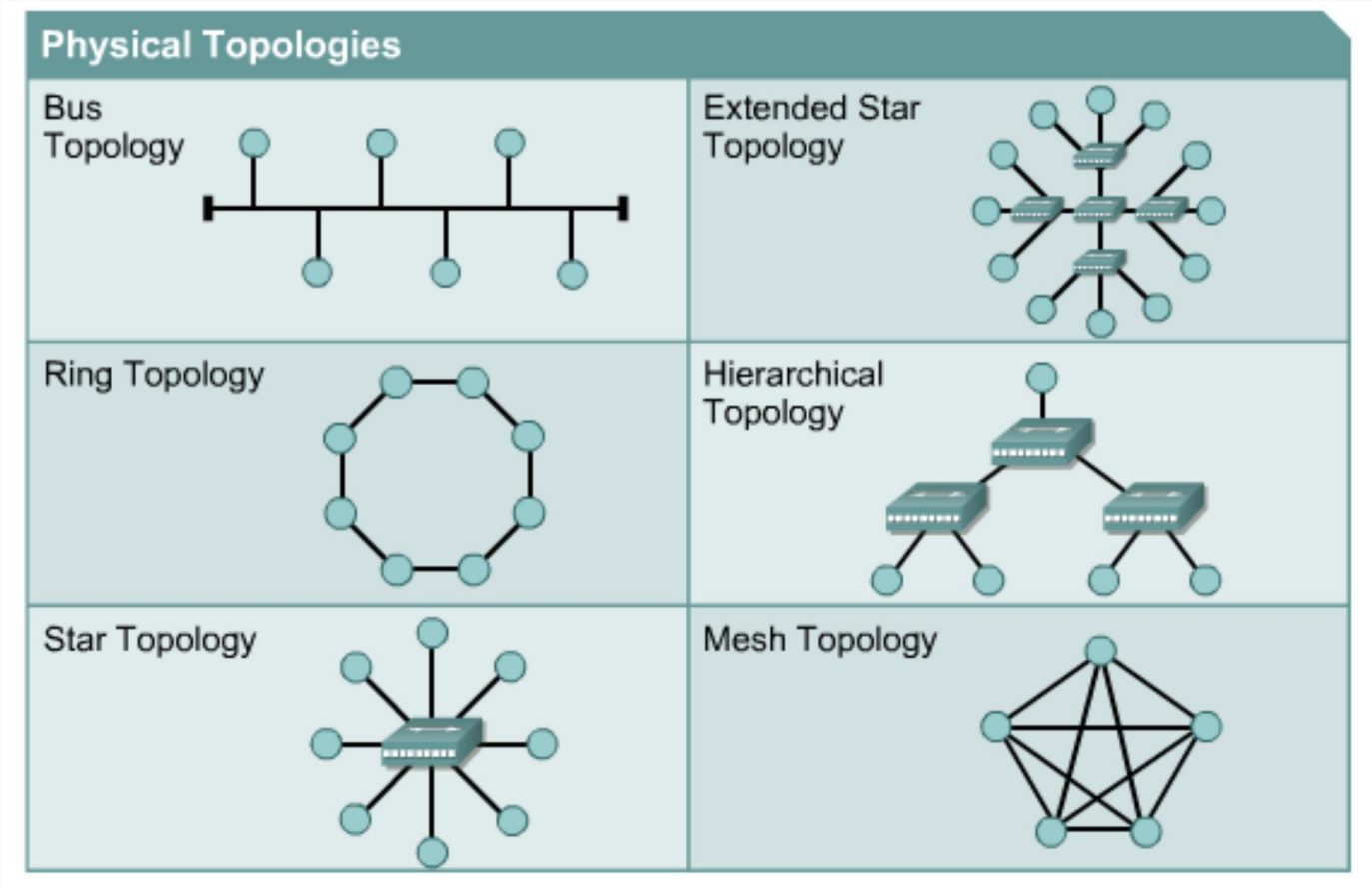


Link Layer

Network Interface & Hardware

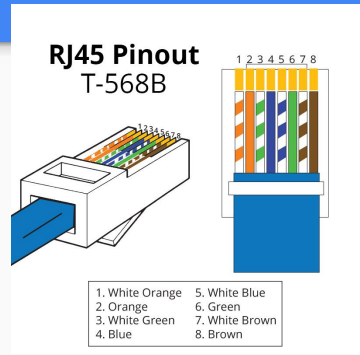
- LAN (Local), WAN (Wide), MAN(Metropolitan)
  - Ethernet, Token-Ring, FDDI
  - PPP, xDSL, ISDN
- Physical Topologies
- Logical Topologies
  - Broadcast, Token-passing
- Common LAN Devices
  - NIC, Repeater, Bridge, Switch, Router
- Common LAN Media
  - UTP, STP, Coaxial, Fiber Optic

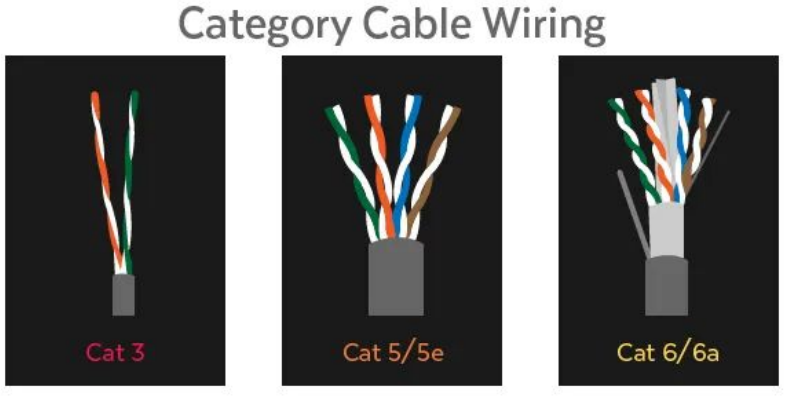
Network Interface & Hardware - Physical Topologies



Network Interfaces & Hardware - Media

- Coaxial Cable
  - Thicknet vs. Thinnet
  - BNC connector
- Twisted Pair Standards
  - T568-A
    - 綠白、綠、橘白、藍、藍白、橘、棕白、棕
  - T568-B
    - 橘白、橘、綠白、藍、藍白、綠、棕白、棕
  - Straight-through vs. Crossover
  - RJ-45 connector





Network Interfaces & Hardware - Media (cont.)

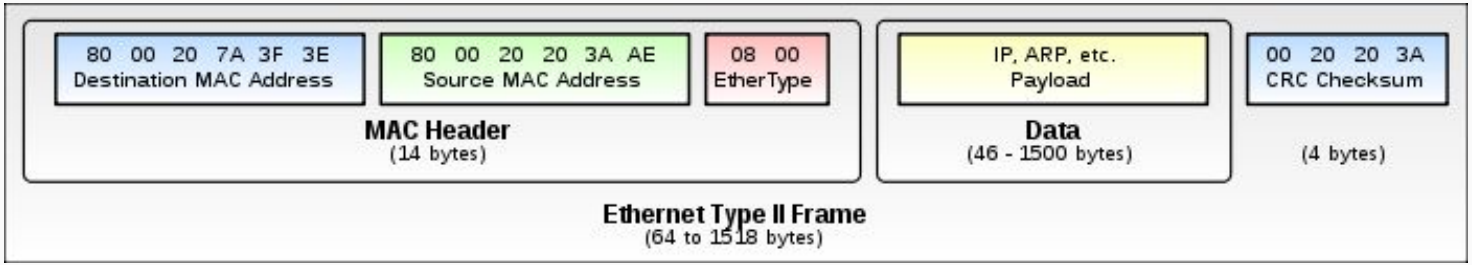
- Fiber Optic Cable
  - Multimode vs. Single mode
- Wireless
  - 802.11 a (5GHz, 54Mbps)
  - 802.11 b (2.4GHz, 11Mbps)
  - 802.11 g (2.4GHz, 54Mbps)
  - 802.11 n (2.4GHz / 5GHz, 288 / 600Mbps)
  - 802.11 ac (5GHz, 1733.2 Mbps with 80MHz)
  - 802.11 ax (2.4GHz / 5GHz / 6GHz, 9608 Mbps with 160MHz)

Link Layer

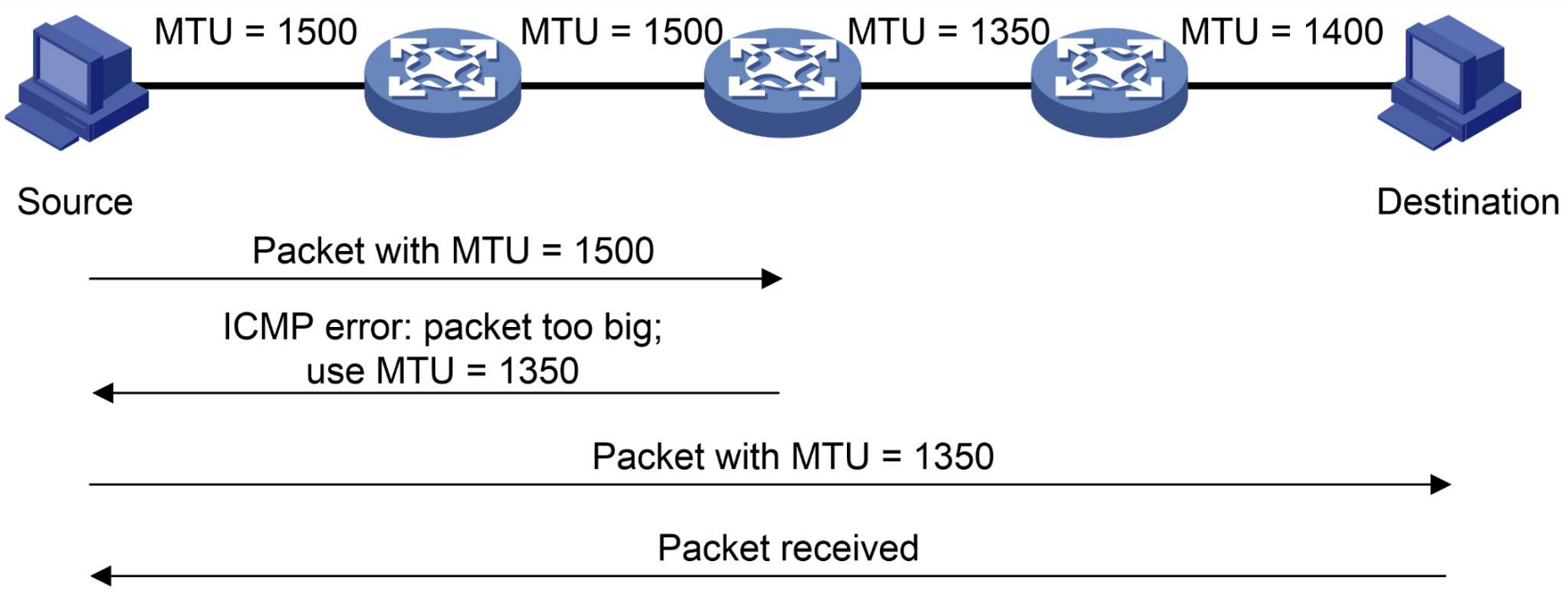
- Ethernet
  - 10Mbps -> 100Mbps -> 1Gbps -> 10Gbps
  - Cat3 -> Cat5 -> Cat5e -> Cat6a
  - 802.3 -> 802.3u -> 802.3z/802.3ab -> 802.3ae/...
  - CSMA/CD
- MAC address (48bit)
  - OUI vendor information
  - Use first 24bit to identify vendor

Link Layer (cont.)

- Etherenet Frame
  - Etherenet MTU - Usually set as 1500
    - Jumbo Frame - Usually means MTU 9000 or up
  - IP fragmentation
    - RFC 791, RFC 815
  - Path MTU
    - MTU of various physical device



Link Layer - Path MTU Discovery



Network Layer

Network Layer

- IP address
- ARP
- Subnet / Netmask
- Address types
- Routing

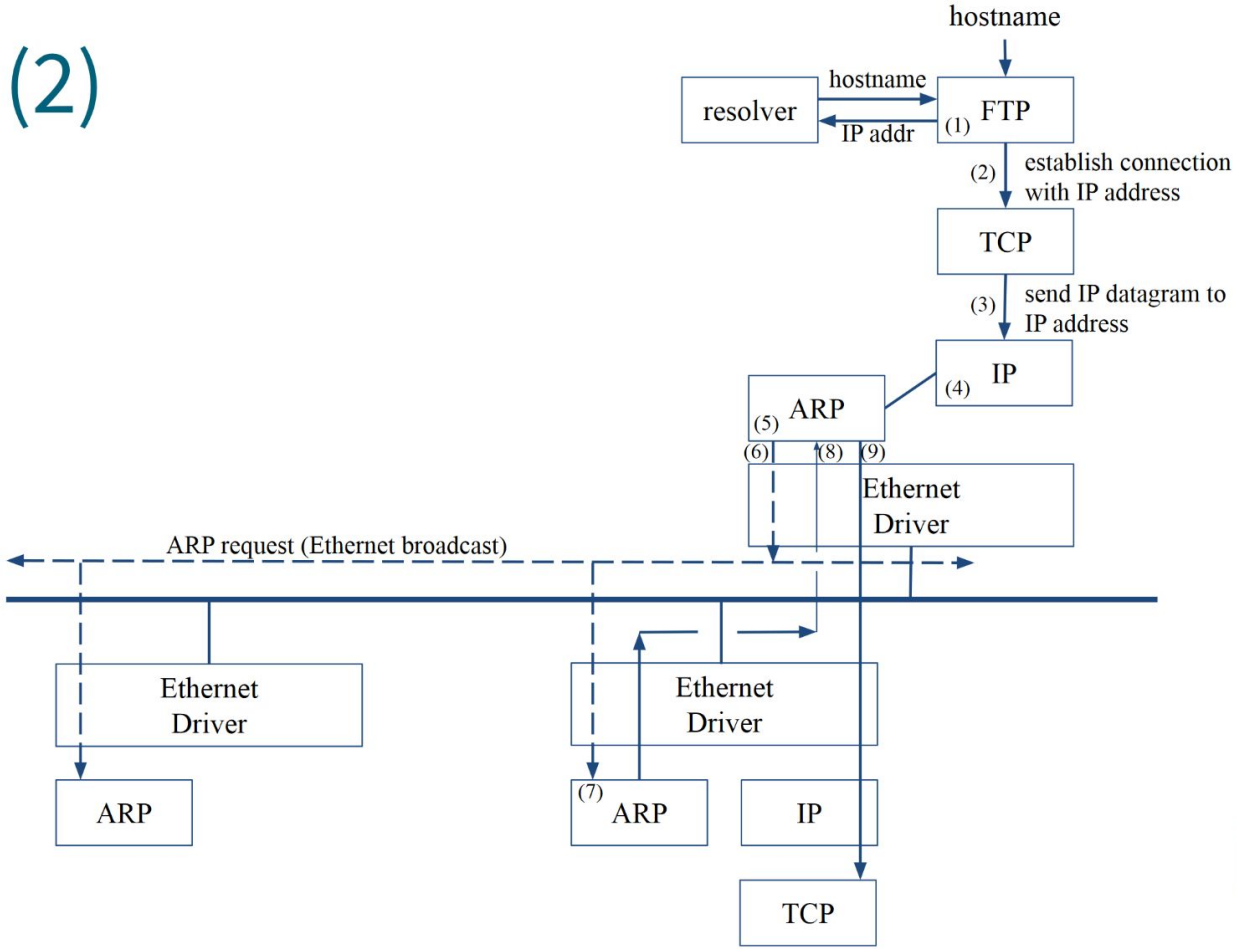
Network Layer - IP address (IPv4)

- 32-bit long
  - Network Part
    - Identify a logical network
  - Host Part
    - Identify a machine on a certain network
- NCKU
  - Class B: 140.116.0.0
  - Network Part: 140.116
  - Number of hosts: 256 * 256 = 65536

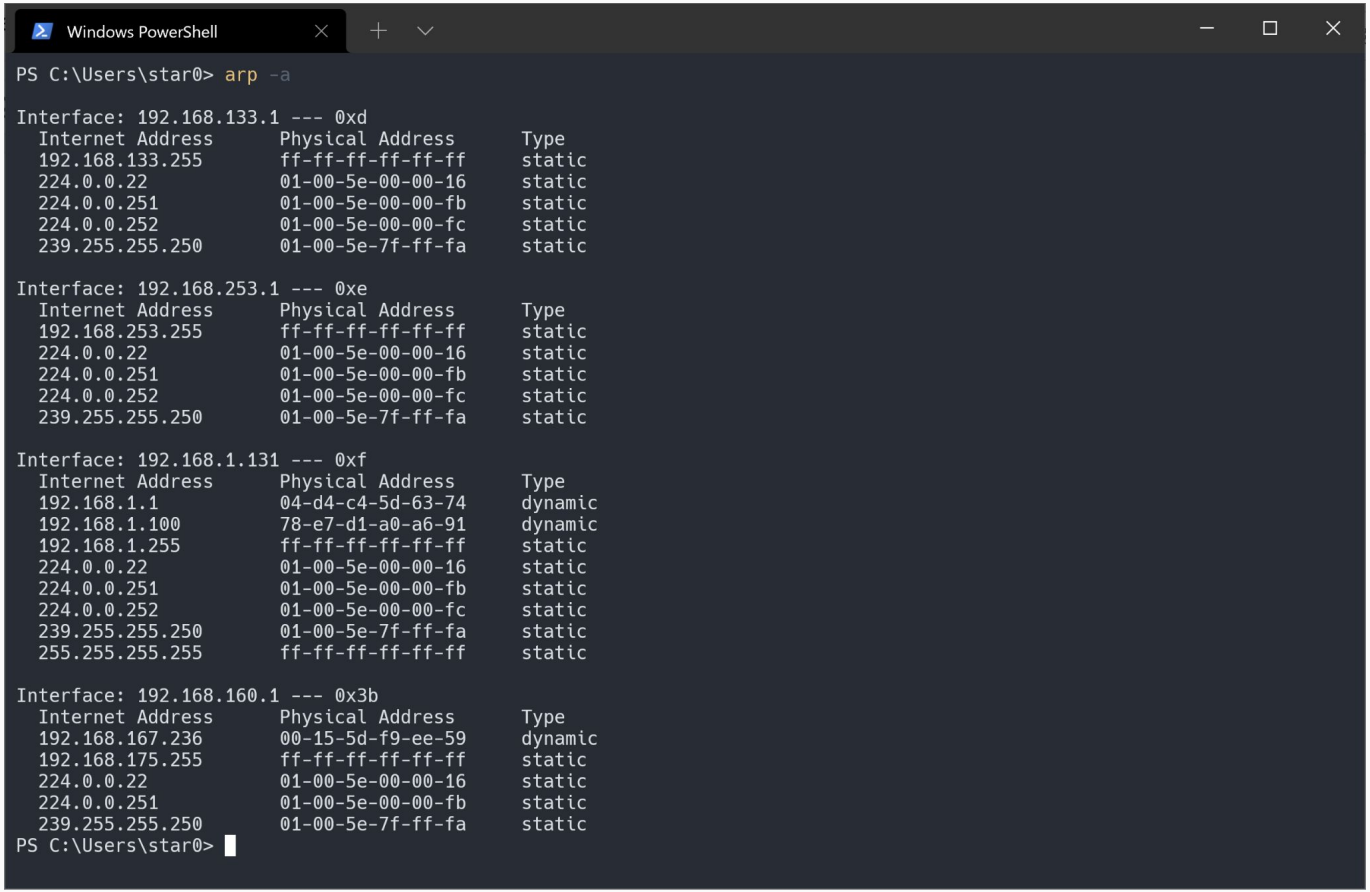
ARP

- Address Resolution Protocol
  - Ask MAC address of certain IP
  - Broadcast
  - Anyone receiving ARP packet and having this IP will reply to the sender
  - When the host owing this IP is not on the same network, sender will use the MAC address of next-hop router to send the packet
  - What is a ROUTER? // FREE CCNA // EP 2
    - Watch how ARP works

ARP (2)


hostname
resolver
hostname
IP addr
FTP
(1)
(2)
establish connection
with IP address
TCP
(3)
send IP datagram to
IP address
IP
(4)
ARP
(5)
(6)
(8)
(9)
Ethernet
Driver
ARP request (Ethernet broadcast)
Ethernet
Driver
Ethernet
Driver
ARP
(7)
ARP
IP
TCP

ARP (cont.)


Windows PowerShell
PS C:\Users\star0> arp -a
Interface: 192.168.133.1 --- 0xd
Internet Address Physical Address Type
192.168.133.255 ff-ff-ff-ff-ff-ff static
224.0.0.22 01-00-5e-00-00-16 static
224.0.0.251 01-00-5e-00-00-fb static
224.0.0.252 01-00-5e-00-00-fc static
239.255.255.250 01-00-5e-7f-ff-fa static
Interface: 192.168.253.1 --- 0xe
Internet Address Physical Address Type
192.168.253.255 ff-ff-ff-ff-ff-ff static
224.0.0.22 01-00-5e-00-00-16 static
224.0.0.251 01-00-5e-00-00-fb static
224.0.0.252 01-00-5e-00-00-fc static
239.255.255.250 01-00-5e-7f-ff-fa static
Interface: 192.168.1.131 --- 0xf
Internet Address Physical Address Type
192.168.1.1 04-d4-c4-5d-63-74 dynamic
192.168.1.100 78-e7-d1-a0-a6-91 dynamic
192.168.1.255 ff-ff-ff-ff-ff-ff static
224.0.0.22 01-00-5e-00-00-16 static
224.0.0.251 01-00-5e-00-00-fb static
224.0.0.252 01-00-5e-00-00-fc static
239.255.255.250 01-00-5e-7f-ff-fa static
255.255.255.255 ff-ff-ff-ff-ff-ff static
Interface: 192.168.160.1 --- 0x3b
Internet Address Physical Address Type
192.168.167.236 00-15-5d-f9-ee-59 dynamic
192.168.175.255 ff-ff-ff-ff-ff-ff static
224.0.0.22 01-00-5e-00-00-16 static
224.0.0.251 01-00-5e-00-00-fb static
239.255.255.250 01-00-5e-7f-ff-fa static
PS C:\Users\star0>

Network Layer - IP address (IPv4) - Class

Class	1st byte	Format	Comments
A	1 ~ 126	N.H.H.H	Very early networks, or reserved for DOD
B	128 ~ 191	N.N.H.H	Large sites, usually subnetted
C	192 ~ 223	N.N.N.H	Easy to get, often obtained in sets
D	224 ~ 239	-	Multicast addresses, not permanently assigned
E	240 ~ 254	-	Experimental addresses

Network Layer - Subnet & Netmask

- Subnet
  - Borrow some bits from host ID to extends network ID
  - For example
    - 140.116.0.0 is class B subnet
      - which contains 256 class C-like subnet
    - 140.116.246.0 is class C-like class B subnet
- Netmask
  - Specify how many bits of network ID are used for network ID
  - 140.116.0.0/16
    - First 16 bits are network ID
    - Last 16 bits are host ID
  - 140.116.246.0/24

Network Layer - Subnet & Netmask (cont.)

- To determine network ID
  - Bitwise-AND IP and netmask
    - 140.116.246.189 & 255.255.255.0 => 140.116.246.0
    - 10.20.30.40 & 255.255.255.240 => 10.20.30.32
    - IP Calculator / IP Subnetting

Network Layer - CIDR Notation

- A compact representation of an IP address and its associated network mask

- 140.116.246.189/24 means
  - Network ID: 140.116.246.0
  - Host ID: 189
  - Subnet: 24
  - 1 ~ 254, total 254 IPs are usable

Network Layer - Classless Inter-Domain Routing

IPv4 CIDR blocks [edit]

Address format	Difference to last address	Mask	Addresses		Relative to class A, B, C	Restrictions on *a*, *b*, *c* and *d* (0..255 unless noted)	Typical use
			Decimal	2^n			
a.b.c.d/32	+0.0.0.0	255.255.255.255	1	2^0	1/256 C		Host route
a.b.c.d/31	+0.0.0.1	255.255.255.254	2	2^1	1/128 C	*d* = 0 ... (2*n*) ... 254	Point-to-point links (RFC 3021)
a.b.c.d/30	+0.0.0.3	255.255.255.252	4	2^2	1/64 C	*d* = 0 ... (4*n*) ... 252	Point-to-point links (glue network)
a.b.c.d/29	+0.0.0.7	255.255.255.248	8	2^3	1/32 C	*d* = 0 ... (8*n*) ... 248	Smallest multi-host network
a.b.c.d/28	+0.0.0.15	255.255.255.240	16	2^4	1/16 C	*d* = 0 ... (16*n*) ... 240	Small LAN
a.b.c.d/27	+0.0.0.31	255.255.255.224	32	2^5	1/8 C	*d* = 0 ... (32*n*) ... 224	
a.b.c.d/26	+0.0.0.63	255.255.255.192	64	2^6	1/4 C	*d* = 0, 64, 128, 192	
a.b.c.d/25	+0.0.0.127	255.255.255.128	128	2^7	1/2 C	*d* = 0, 128	Large LAN
a.b.c.0/24	+0.0.0.255	255.255.255.0	256	2^8	1 C		
a.b.c.0/23	+0.0.1.255	255.255.254.0	512	2^9	2 C	*c* = 0 ... (2*n*) ... 254	
a.b.c.0/22	+0.0.3.255	255.255.252.0	1,024	2^{10}	4 C	*c* = 0 ... (4*n*) ... 252	Small business
a.b.c.0/21	+0.0.7.255	255.255.248.0	2,048	2^{11}	8 C	*c* = 0 ... (8*n*) ... 248	Small ISP/ large business
a.b.c.0/20	+0.0.15.255	255.255.240.0	4,096	2^{12}	16 C	*c* = 0 ... (16*n*) ... 240	
a.b.c.0/19	+0.0.31.255	255.255.224.0	8,192	2^{13}	32 C	*c* = 0 ... (32*n*) ... 224	ISP/ large business
a.b.c.0/18	+0.0.63.255	255.255.192.0	16,384	2^{14}	64 C	*c* = 0, 64, 128, 192	
a.b.c.0/17	+0.0.127.255	255.255.128.0	32,768	2^{15}	128 C	*c* = 0, 128	
a.b.0.0/16	+0.0.255.255	255.255.0.0	65,536	2^{16}	256 C = B		
a.b.0.0/15	+0.1.255.255	255.254.0.0	131,072	2^{17}	2 B	*b* = 0 ... (2*n*) ... 254	
a.b.0.0/14	+0.3.255.255	255.252.0.0	262,144	2^{18}	4 B	*b* = 0 ... (4*n*) ... 252	

Network Layer - IPv4 crisis

- IPv4 addresses are running out
  - No more IPv4 addresses to assign to organizations
  - IP addresses were being allocated on a FCFS basis
- Solutions
  - Short term
    - Subnetting / CIDR
    - NAT
  - Long term
    - IPv6

Network Layer - NAT

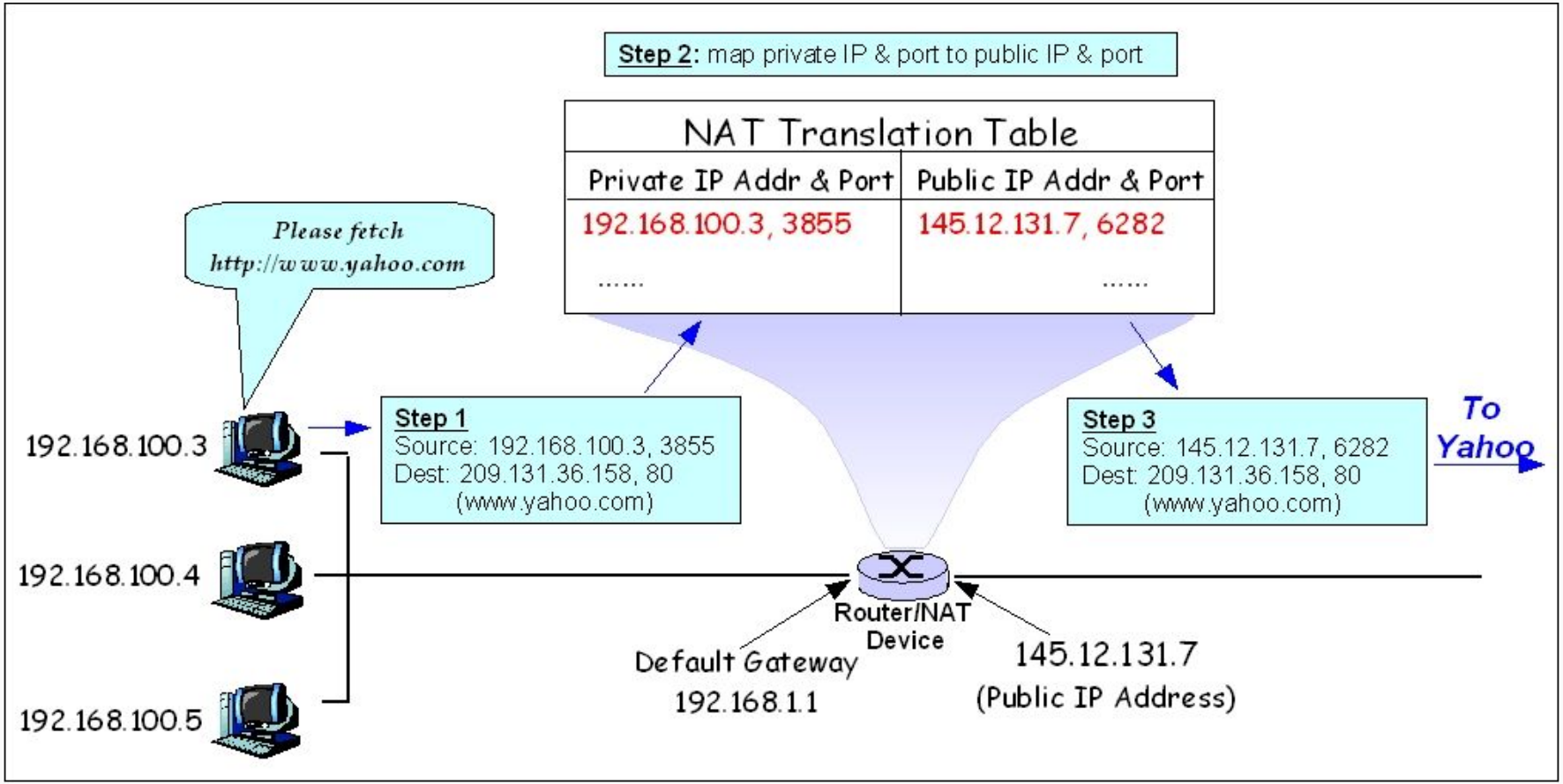
- Private Address
  - Packets with private address will not go out to the Internet
  - 3 private address ranges

IP Class	From	To	CIDR notation
Class A	10.0.0.0	10.255.255.255	10.0.0.0/8
Class B	172.16.0.0	172.31.255.255	172.16.0.0/12
Class C	192.168.0.0	192.168.255.255	192.168.0.0/16

Network Layer - NAT

- Network Address Translation
- Allow hosts using private address to talk with outside

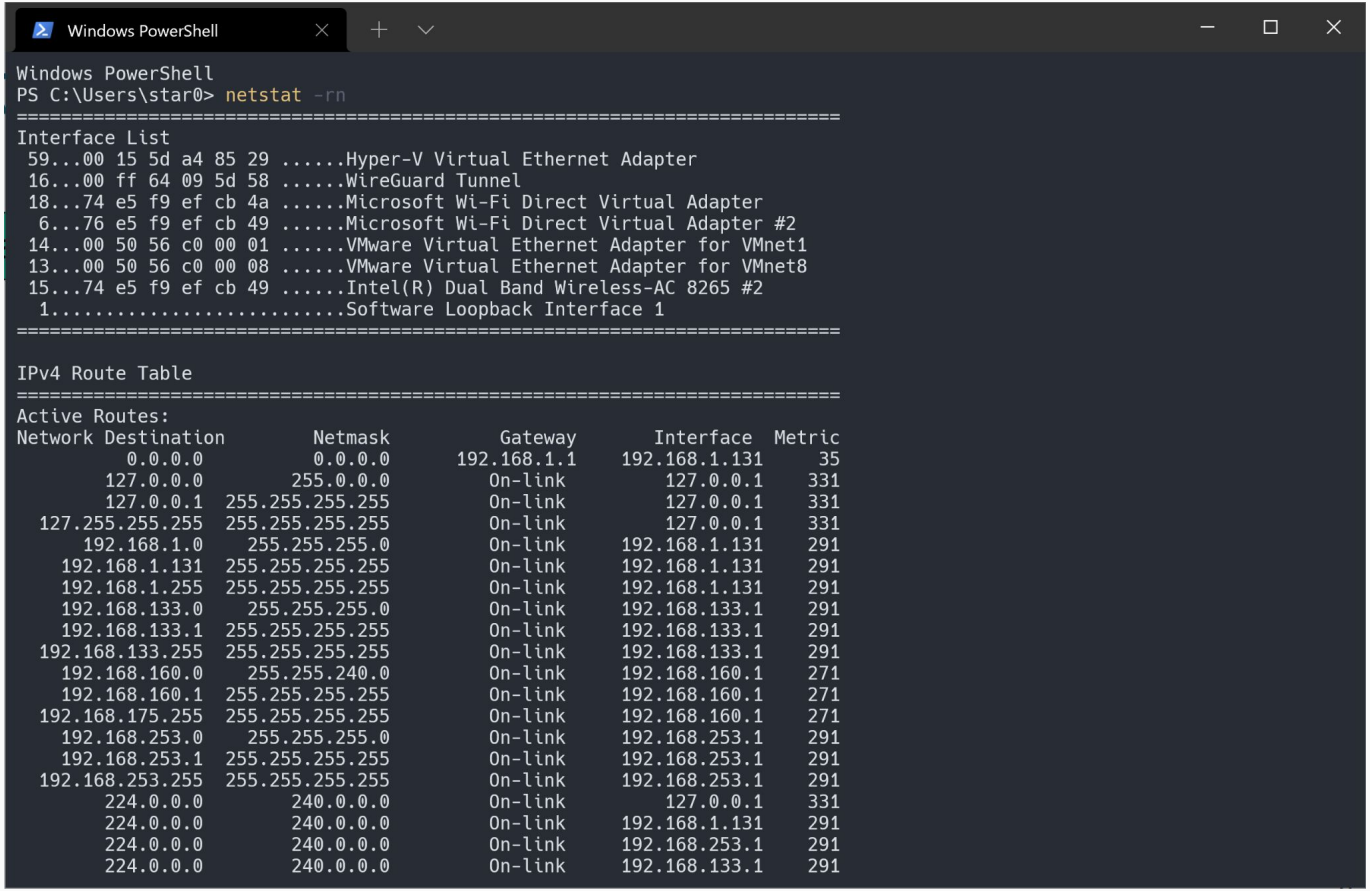
Network Layer - NAT (cont.)



Network Layer - Routing

- Direct a packer closer to the destination
- Flat vs. Hierarchical
- Routing Table
  - Routing Information
  - Rule-based Information
  - Kernel will pick the most suitable way to route the packets

Network Layer - Routing (cont.)



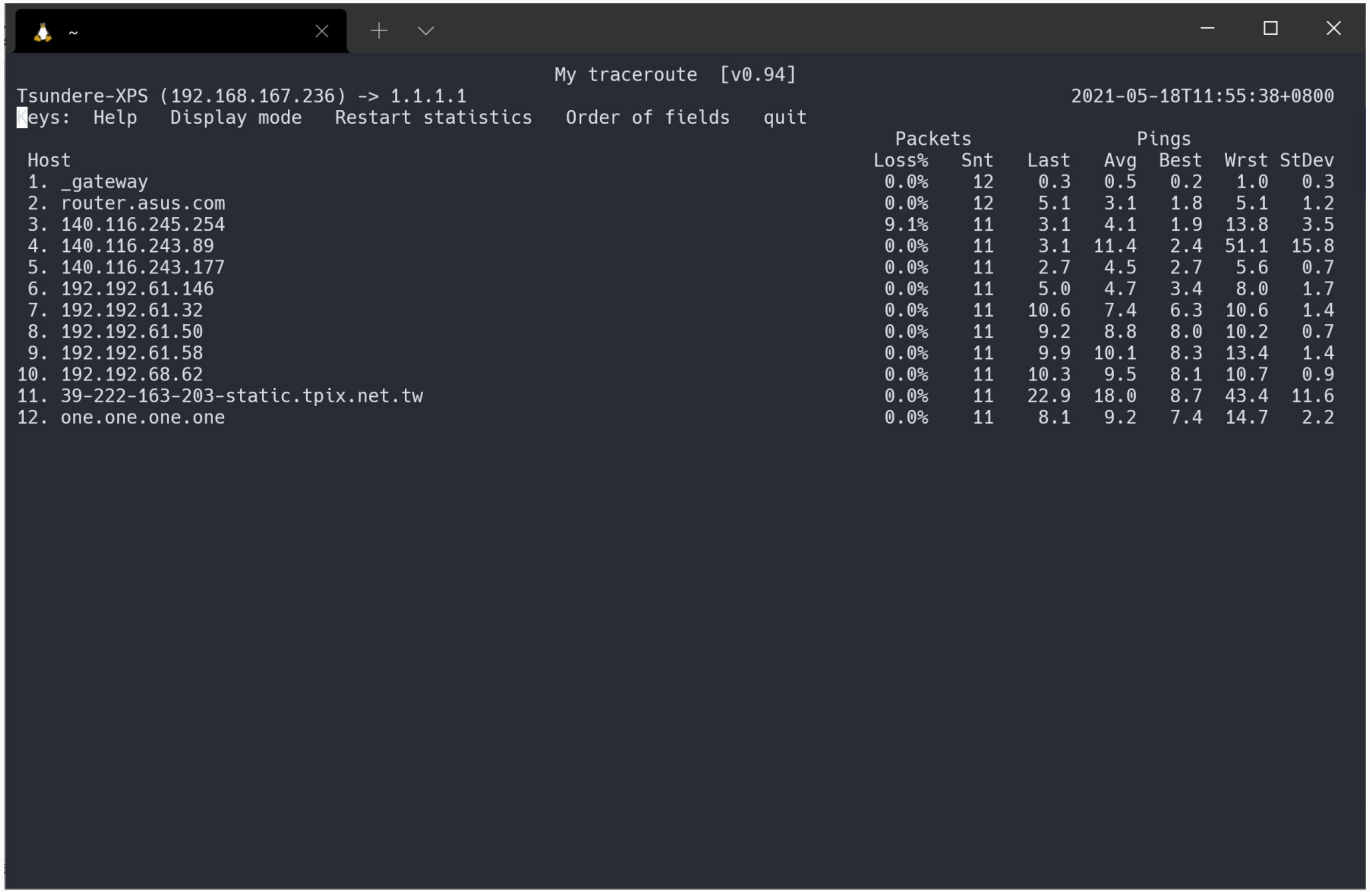
Network Layer - Routing (cont.)

- Static Route
  - Statically configured by “route” command
  - # ip route add 172.10.1.0/24 via 10.0.0.100 dev eth0
- Dynamic Route
  - Configured by some routing protocol

Network Layer - Routing (cont.)

- Trace packet
  - ping -R
  - traceroute
  - mtr

Network Layer - Routing - mtr



Transport Layer

Transport Layer - ports

- 16-bits number
- Preserve ports
  - 1 ~ 1024 (root access only)
- Well-known ports
  - /etc/services

File: /etc/services

```
1   # Full data: /usr/share/iana-etc/port-numbers.iana
2
3   tcpmux              1/tcp
4   tcpmux              1/udp
5   compressnet         2/tcp
6   compressnet         2/udp
7   compressnet         3/tcp
8   compressnet         3/udp
9   rje                 5/tcp
10  rje                 5/udp
11  echo                7/tcp
12  echo                7/udp
13  discard             9/tcp
14  discard             9/udp
```

Transport Layer - TCP vs. UDP

Function	UDP	TCP
Connection-oriented	No	Yes
Message boundaries	Yes	No
Data checksum	Optional	Yes
Positive acknowledgement	No	Yes
Time-out and retransmit	No	Yes
Duplicate detection	No	Yes
Sequencing	No	Yes
Flow control	No	Yes

Transport Layer - useful commands

- tcpdump, sniffit, trafshow, netstat -s
- Wireshark
  - GUI-based network protocol analyzer

Network Setup

Network Setup

- Assign IP address / Hostname
- Default Route
- DNS
- ping / traceroute to make sure network works

Network Setup - IP assignment

- In Ubuntu Linux 20.04, OS use netplan
- /etc/netplan/50-cloud-init.yaml
  - PLEASE DO NOT EDIT THIS FILE IN YOUR VM!!!

Network Setup - IP assignment - netplan

F74076310@F74076310: ~

```
F74076310@F74076310:~$ cat /etc/netplan/50-cloud-init.yaml
```

File: /etc/netplan/50-cloud-init.yaml

```
1   # This file is generated from information provided by the datasource.  Changes
2   # to it will not persist across an instance reboot.  To disable cloud-init's
3   # network configuration capabilities, write a file
4   # /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
5   # network: {config: disabled}
6   network:
7       version: 2
8       ethernets:
9           eth0:
10              addresses:
11              - 172.26.4.37/24
12              gateway4: 172.26.4.254
13              match:
14                  macaddress: 26:df:ac:5c:b1:3c
15              nameservers:
16                  addresses:
17                  - 101.101.101.101
18                  search:
19                  - nasa.imslab.org
20              set-name: eth0
```

```
F74076310@F74076310:~$
```

Network Setup - Hostname

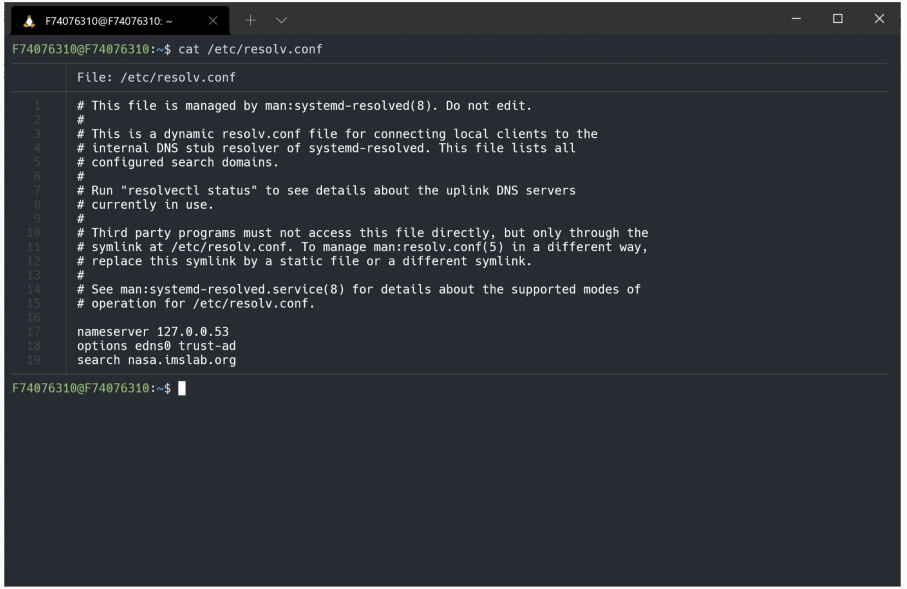
- /etc/hosts
  - Hostname database
- /etc/hostname
  - Hostname for this machine

Network Setup - Default Route

- # ip route add default via 192.168.1.254 dev eth0

Network Setup - DNS

- /etc/resolv.conf
- 127.0.0.53
  - systemd-resolved
  - systemd DNS resolver



Topics we didn’t cover this time

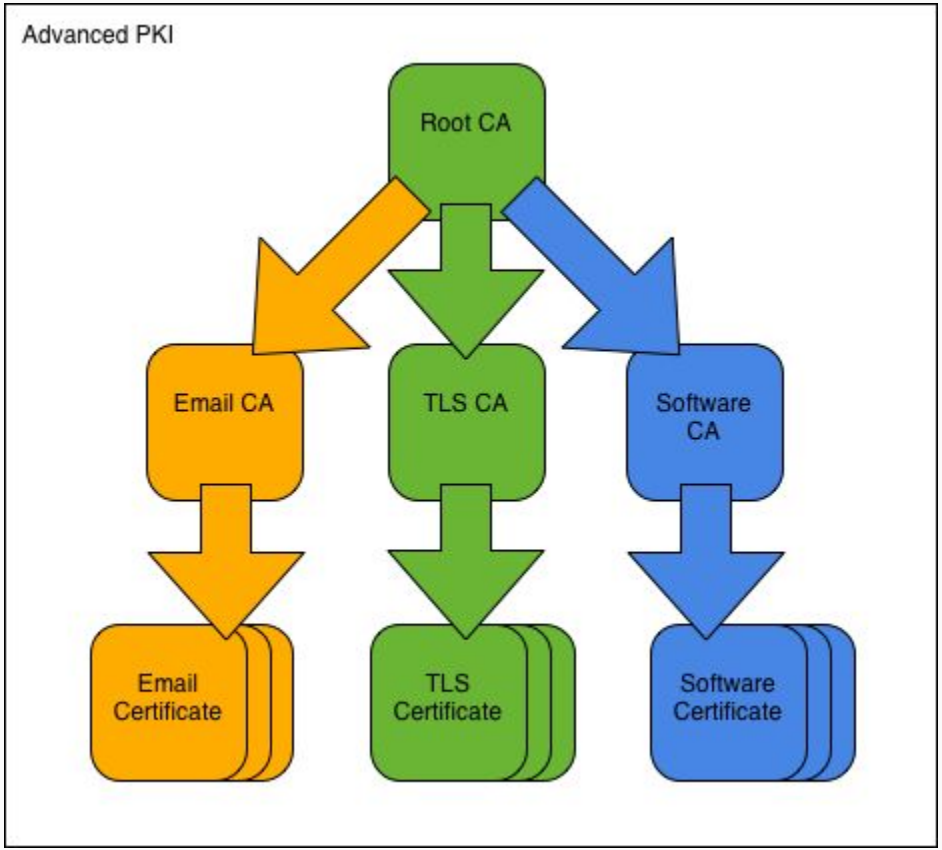
- DHCP
- NAT
- DNS
  - Previous lecture
- Mail
- VPN
- Tunnel
- ...

PKI

Public-Key Infrastructure

- A set of hardware, software, people, policies, and procedures
- To create, manage, distribute, use, store, and revoke digital certificates
- Encryption, authentication, and signature
- Bootstrapping secure communication protocols

Certificate Authority / Tree structure of CA



Certificate

- Contains data of the owner, such as Company Name, Server Name, Name, Email, Address…
- Public key of the owner
- Followed by some digital signatures
  - Signed for the certificate
- X.509
  - A certificate is signed by a CA
  - To verify the authenticity of the certificate, check the signature of CA

Certificate Authority

- Trusted server which signs certificates
- One private key and relative public key
- Tree source of X.509
  - Root CA

Root CA

- Do not sign the certificates for users
- Root CA signs for itself
- To trust Root CA
  - Install the certificate of Root CA via secure channel

Cost of certificate

- Public CA: $78 / per year / per host
- Self-signed: $0 (But no one will trust this certificate except yourself)
- Let's Encrypt: $0

SSL / TLS

SSL / TLS

- Secure Socket Layer
- Transport Layer Security

- Provide communication security over the Internet
  - prevent eavesdropping and tampering
- Encrypt segments over Transport Layer

SSL - History

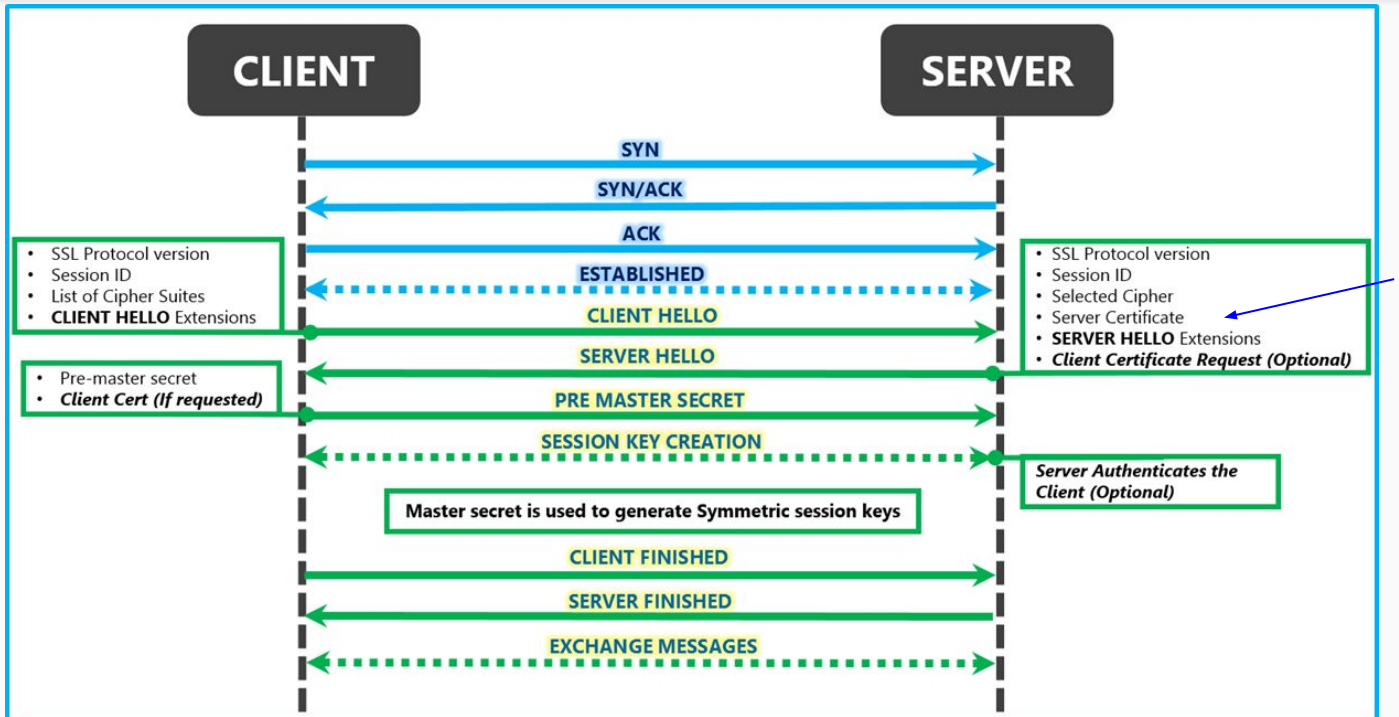
- SSL was developed by Netscape
  - SSL 1.0
    - Was never publicly published
  - SSL 2.0, 1995
    - A number of security flaws
  - SSL 3.0, 1996
    - A complete redesign
    - Newer version of SSL/TLS are based on SSL 3.0
    - POODLE attack

TLS - History

- TLS - IETF RFC
  - TLS 1.0 (SSL 3.1), RFC 2246 in 1999
    - Backward compatible to SSL 3.0
    - CBC vulnerability discovered in 2002
  - TLS 1.1 (SSL 3.2), RFC 4346 in 2006
    - Prevent CBC attacks
  - TLS 1.2 (SSL 3.3), RFC 5246 in 2008
    - Enhance security strength
    - Introduce new cryptographic algorithms
  - TLS 1.3, RFC 8446 in 2018

Source: https://medium.com/@vanrijn/an-overview-of-the-ssl-handshake-3885c37c3e0f

SSL/TLS Negotiation



SSL/TLS Negotiation (cont.)

- (C) Request a secure connection, and present a list of supported ciphers and hash functions
- (S) Select common cipher and hash function, and send back with server's digital certificate
- (C) Confirm the validity of the certificate
- (C) Encrypt a random number (pre-master secret) with server's public key, and send it to the server
- (C/S) Generate session key(s) from the random number

C: Client
S: Server

SSL/TLS Applications

- Implemented on top of Transport Layer protocols
  - TCP
  - UDP (DTLS)
- To protect insecure services
  - HTTP / FTP / SMTP / VPN / VoIP...
- To activate SSL/TLS
  - Use a different port (Like HTTP/HTTPS is 80/443)
  - Use different mechanism (Like STARTTLS)

SSL/TLS Application - Name-based Virtual Server

- All virtual servers belong to the same domain
  - Wildcard certificate (*.example.com)
  - Add all virtual hostnames in subjectAltName
  - Disadvantage
    - Certificate need re-issue whenever adding a new virtual server
- Server Name Indication (SNI)
  - RFC 4366
  - Client browser also need to support SNI

OpenSSL

OpenSSL

- Contains an open-source implementation of the SSL and TLS protocols. The core library, written in the C programming language, implements basic cryptographic functions and provides various utility functions. Wrappers allowing the use of the OpenSSL library in a variety of computer languages are available.

OpenSSL - Heartbleed

- OpenSSL Heartbleed 全球駭客的殺戮祭典，你參與了嗎？ | DEVCORE

PGP

PGP

- Pretty Good Privacy
- Public key system
  - Encryption
  - Signature
- Will not cover in this class, maybe next time