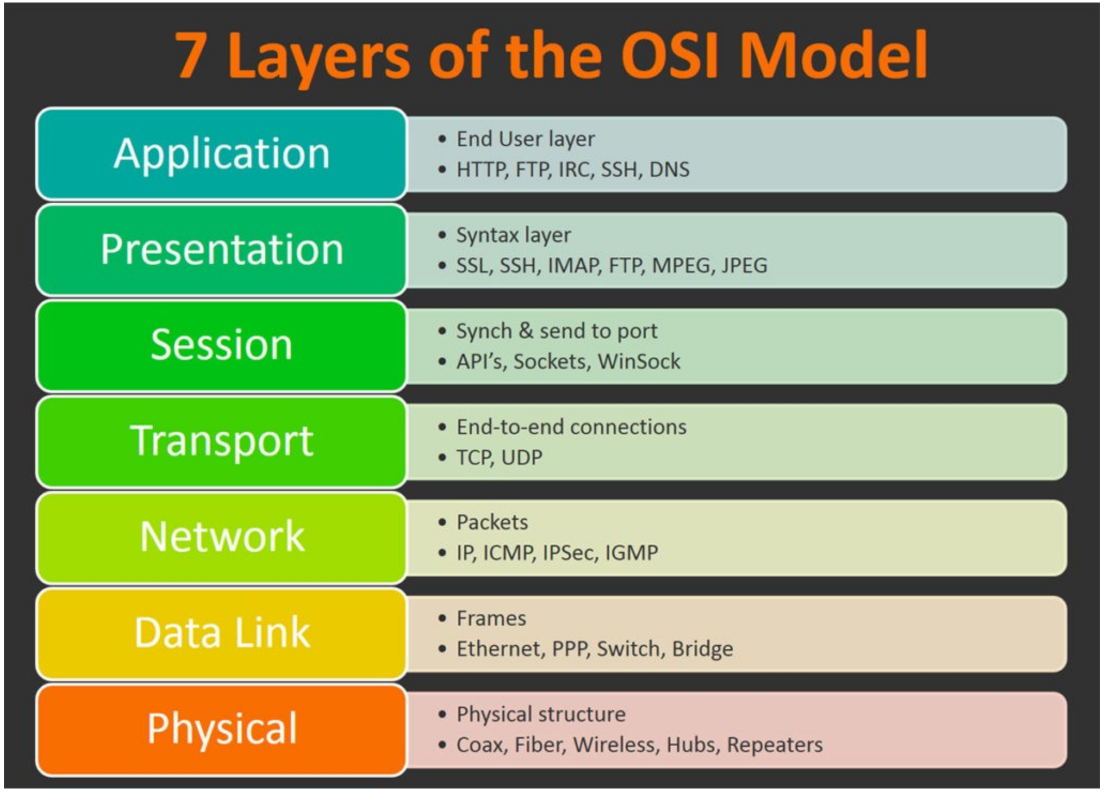
Networking Primer

How does your computer get to
http://www.google.com?

Outline

- OSI Model
- TCP/IP Stack
- Link Layer
- Internet Layer (IPv4 vs IPv6)
    - ipconfig / ifconfig
    - ARP
    - DNS
- Transport Layer
    - TCP vs UDP
- Application Identification
    - Common Ports
- Encapsulation
- LAN vs WAN
- Wireshark

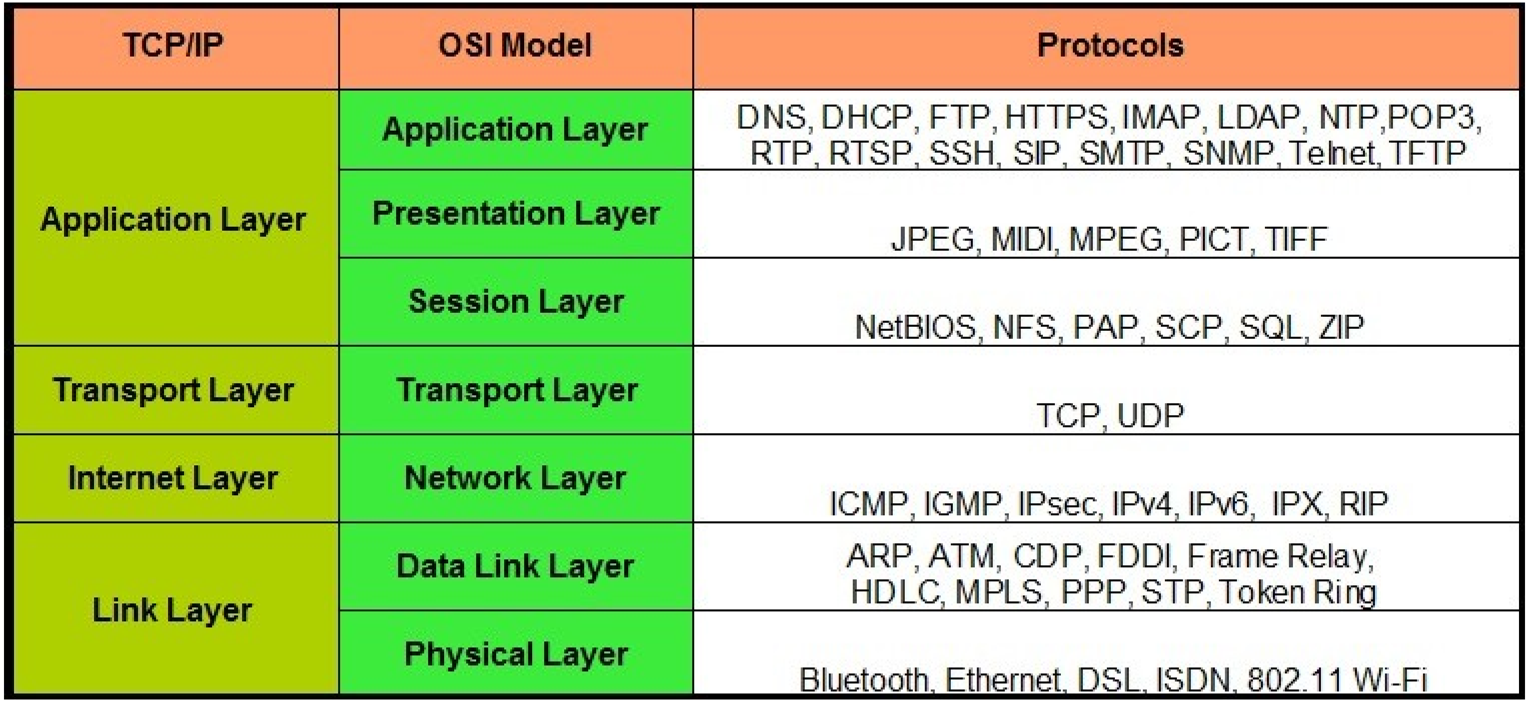
OSI



7 Layers of the OSI Model

Application	- End User layer - HTTP, FTP, IRC, SSH, DNS
Presentation	- Syntax layer - SSL, SSH, IMAP, FTP, MPEG, JPEG
Session	- Synch & send to port - API's, Sockets, WinSock
Transport	- End-to-end connections - TCP, UDP
Network	- Packets - IP, ICMP, IPSec, IGMP
Data Link	- Frames - Ethernet, PPP, Switch, Bridge
Physical	- Physical structure - Coax, Fiber, Wireless, Hubs, Repeaters

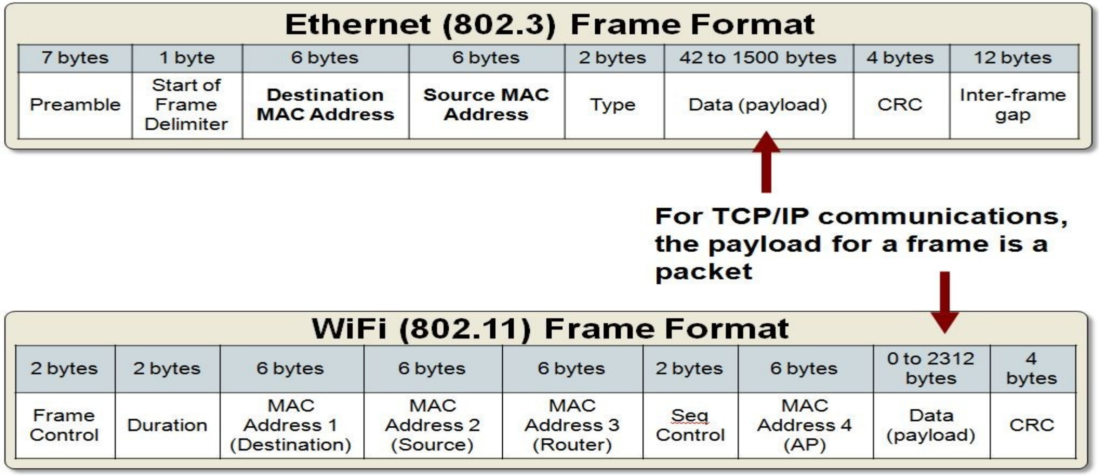
TCP/IP



TCP/IP	OSI Model	Protocols
Application Layer	Application Layer	DNS, DHCP, FTP, HTTPS, IMAP, LDAP, NTP,POP3, RTP, RTSP, SSH, SIP, SMTP, SNMP, Telnet, TFTP
	Presentation Layer	JPEG, MIDI, MPEG, PICT, TIFF
	Session Layer	NetBIOS, NFS, PAP, SCP, SQL, ZIP
Transport Layer	Transport Layer	TCP, UDP
Internet Layer	Network Layer	ICMP, IGMP, IPsec, IPv4, IPv6, IPX, RIP
Link Layer	Data Link Layer	ARP, ATM, CDP, FDDI, Frame Relay, HDLC, MPLS, PPP, STP, Token Ring
	Physical Layer	Bluetooth, Ethernet, DSL, ISDN, 802.11 Wi-Fi

Link Layer

- Dependent on the network hardware
- Physically identify network card



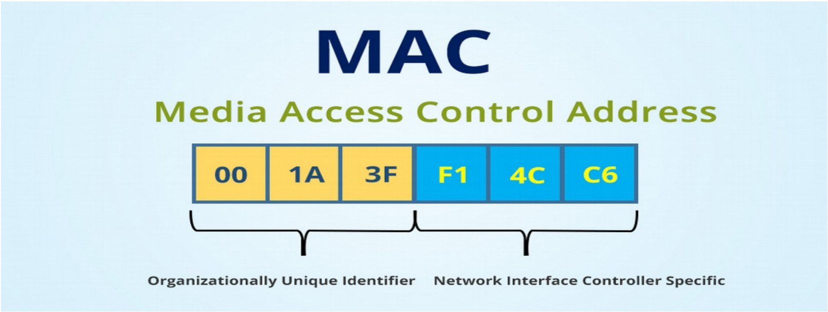
Ethernet (802.3) Frame Format

7 bytes	1 byte	6 bytes	6 bytes	2 bytes	42 to 1500 bytes	4 bytes	12 bytes
Preamble	Start of Frame Delimiter	**Destination MAC Address**	**Source MAC Address**	Type	Data (payload)	CRC	Inter-frame gap

For TCP/IP communications, the payload for a frame is a packet

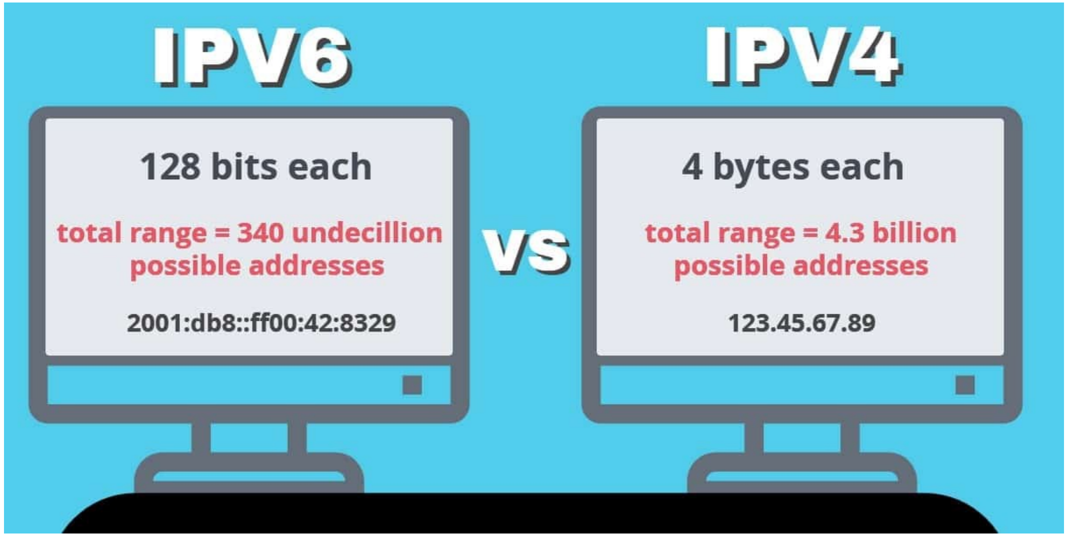
WiFi (802.11) Frame Format

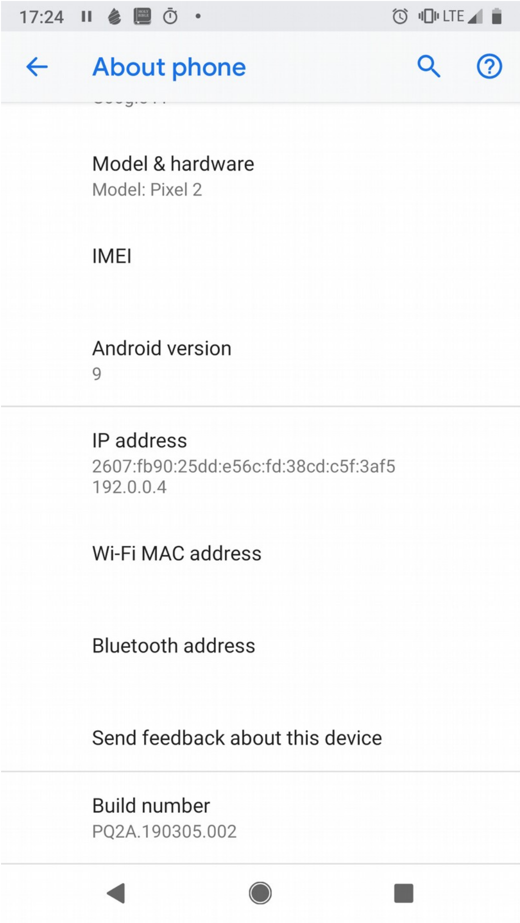
2 bytes	2 bytes	6 bytes	6 bytes	6 bytes	2 bytes	6 bytes	0 to 2312 bytes	4 bytes
Frame Control	Duration	MAC Address 1 (Destination)	MAC Address 2 (Source)	MAC Address 3 (Router)	Seq Control	MAC Address 4 (AP)	Data (payload)	CRC


MAC
Media Access Control Address
00
1A
3F
F1
4C
C6
Organizationally Unique Identifier
Network Interface Controller Specific

Internet Layer

- IP is the most prevalent protocol
- Address is a virtual address


IPV6
128 bits each
total range = 340 undecillion possible addresses
2001:db8::ff00:42:8329
VS
IPV4
4 bytes each
total range = 4.3 billion possible addresses
123.45.67.89


17:24 LTE

About phone

Model & hardware
Model: Pixel 2

IMEI

Android version
9

IP address
2607:fb90:25dd:e56c:fd:38cd:c5f:3af5
192.0.0.4

Wi-Fi MAC address

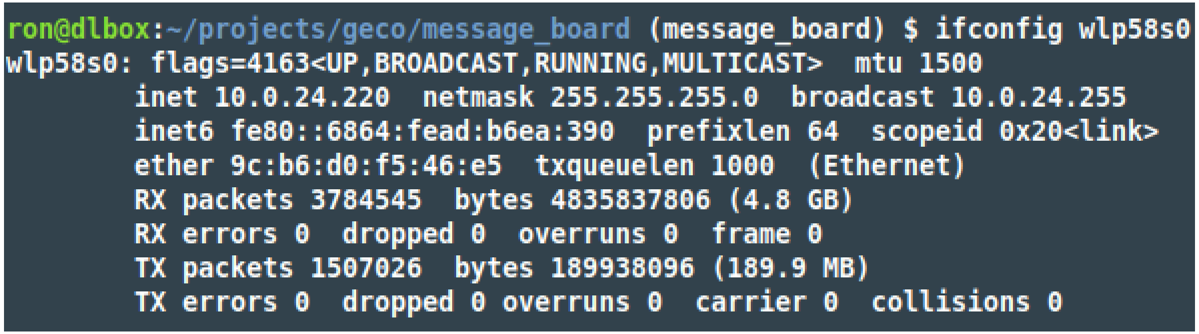
Bluetooth address

Send feedback about this device

Build number
PQ2A.190305.002

ipconfig / ifconfig

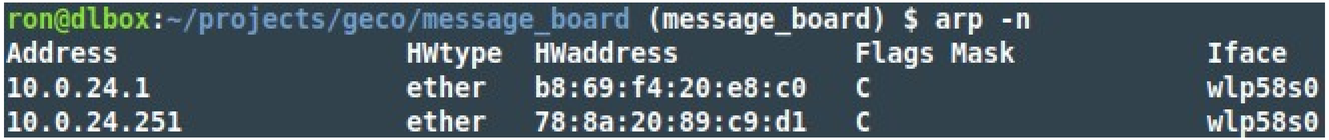
- ***ipconfig*** – Windows
- ***ifconfig*** – Unix like systems
- Being replaced by ***ip address***



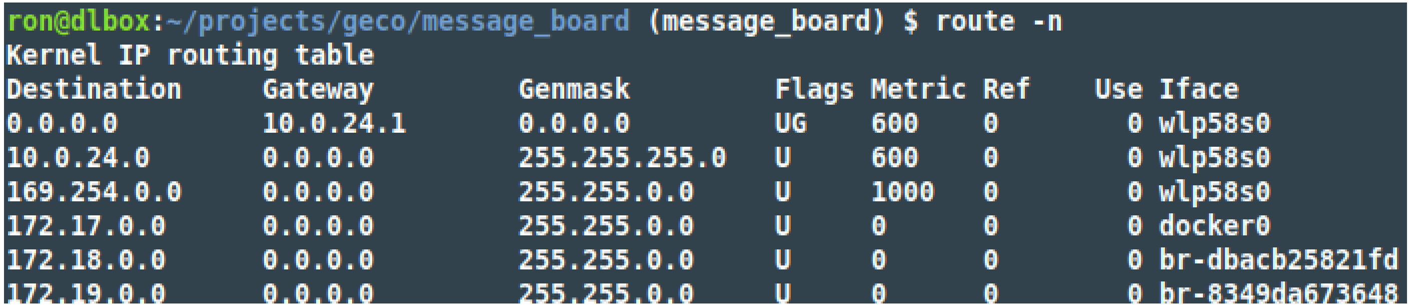
```
ron@dlbox:~/projects/geco/message_board (message_board) $ ifconfig wlp58s0
wlp58s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.24.220  netmask 255.255.255.0  broadcast 10.0.24.255
        inet6 fe80::6864:fead:b6ea:390  prefixlen 64  scopeid 0x20<link>
        ether 9c:b6:d0:f5:46:e5  txqueuelen 1000  (Ethernet)
        RX packets 3784545  bytes 4835837806 (4.8 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1507026  bytes 189938096 (189.9 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

ARP

- Address Resolution Protocol



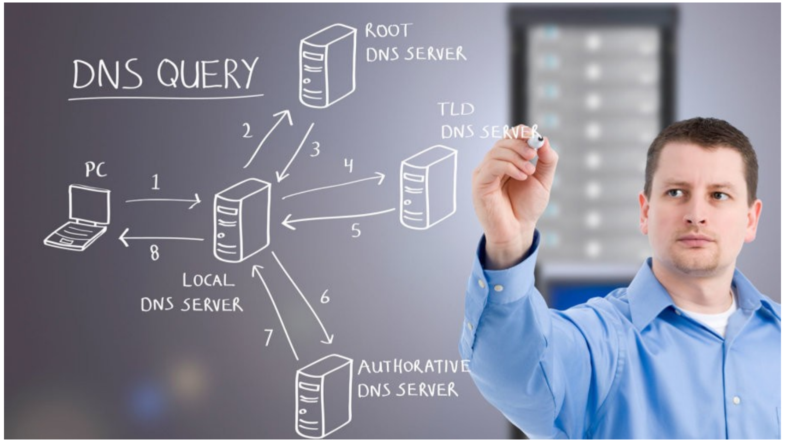
```
ron@dlbox:~/projects/geco/message_board (message_board) $ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
10.0.24.1                ether   b8:69:f4:20:e8:c0   C                     wlp58s0
10.0.24.251              ether   78:8a:20:89:c9:d1   C                     wlp58s0
```

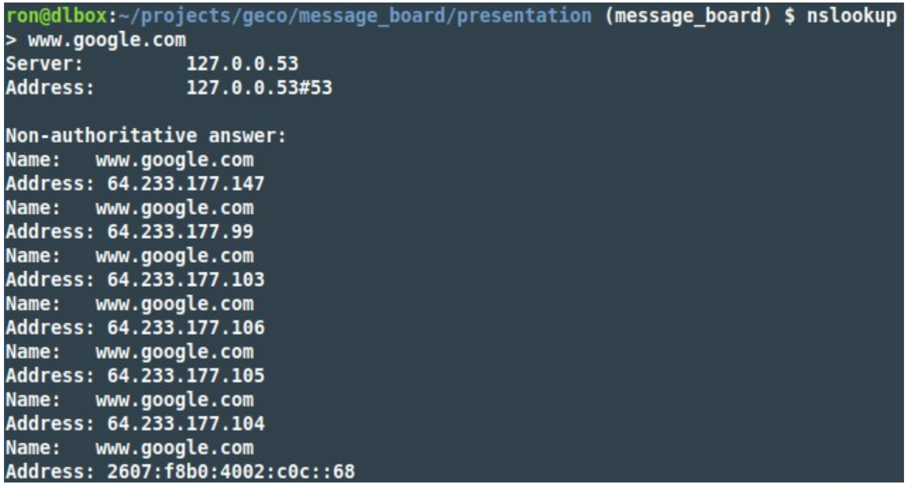


```
ron@dlbox:~/projects/geco/message_board (message_board) $ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.0.24.1       0.0.0.0         UG    600    0        0 wlp58s0
10.0.24.0       0.0.0.0         255.255.255.0   U     600    0        0 wlp58s0
169.254.0.0     0.0.0.0         255.255.0.0     U     1000   0        0 wlp58s0
172.17.0.0      0.0.0.0         255.255.0.0     U     0      0        0 docker0
172.18.0.0      0.0.0.0         255.255.0.0     U     0      0        0 br-dbacb25821fd
172.19.0.0      0.0.0.0         255.255.0.0     U     0      0        0 br-8349da673648
```

DNS

- Domain Name System
- Domain name → IP address



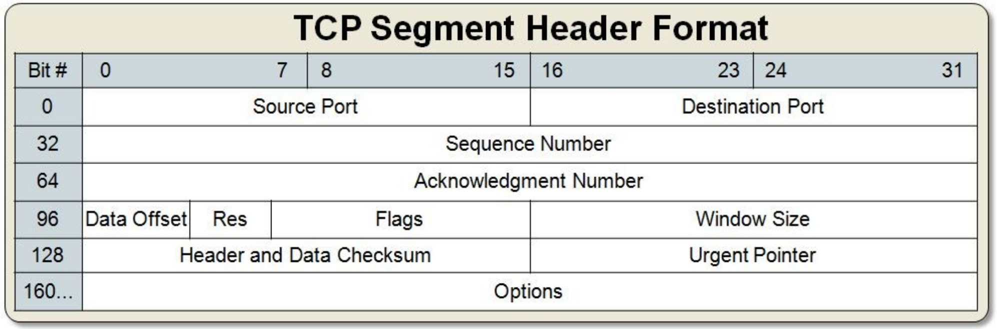


```
ron@dlbox:~/projects/geco/message_board/presentation (message_board) $ nslookup
> www.google.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   www.google.com
Address: 64.233.177.147
Name:   www.google.com
Address: 64.233.177.99
Name:   www.google.com
Address: 64.233.177.103
Name:   www.google.com
Address: 64.233.177.106
Name:   www.google.com
Address: 64.233.177.105
Name:   www.google.com
Address: 64.233.177.104
Name:   www.google.com
Address: 2607:f8b0:4002:c0c::68
```

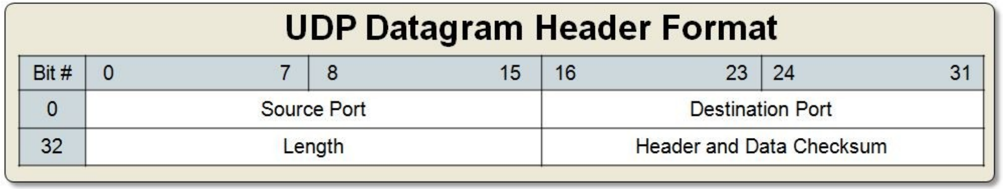
Transport Layer

- Two main protocols in use:
  - TCP – Transport Control Protocol
    - Reliable
  - UDP – User Datagram Protocol
    - Unreliable but fast



TCP Segment Header Format

Bit #	0 7	8 15	16 23	24 31
0	Source Port		Destination Port	
32	Sequence Number			
64	Acknowledgment Number			
96	Data Offset Res	Flags	Window Size	
128	Header and Data Checksum		Urgent Pointer	
160...	Options			

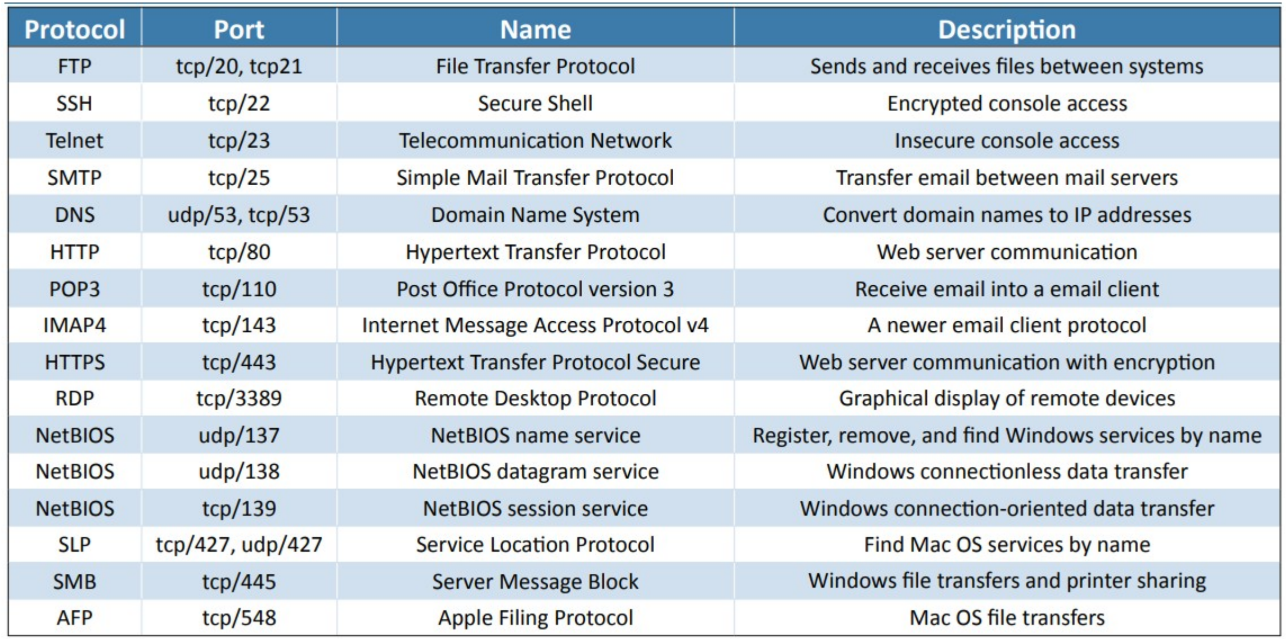


UDP Datagram Header Format

Bit #	0 7	8 15	16 23	24 31
0	Source Port		Destination Port	
32	Length		Header and Data Checksum	

Application Identification

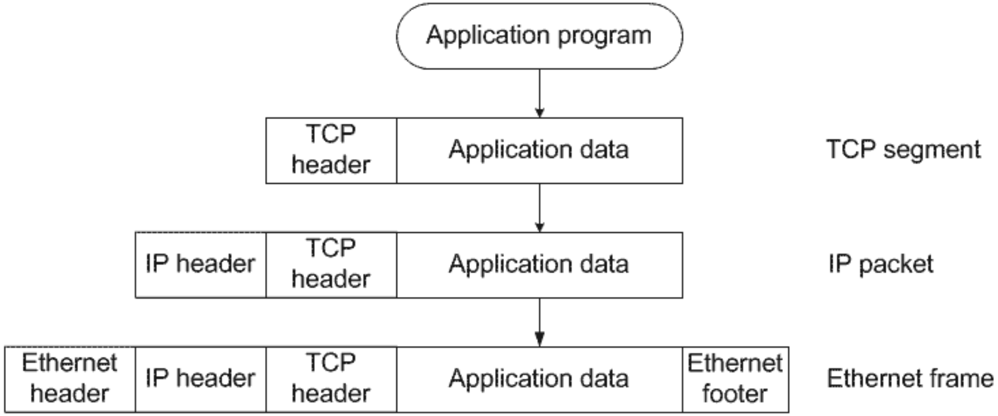
- Port numbers typically identify the application



Protocol	Port	Name	Description
FTP	tcp/20, tcp21	File Transfer Protocol	Sends and receives files between systems
SSH	tcp/22	Secure Shell	Encrypted console access
Telnet	tcp/23	Telecommunication Network	Insecure console access
SMTP	tcp/25	Simple Mail Transfer Protocol	Transfer email between mail servers
DNS	udp/53, tcp/53	Domain Name System	Convert domain names to IP addresses
HTTP	tcp/80	Hypertext Transfer Protocol	Web server communication
POP3	tcp/110	Post Office Protocol version 3	Receive email into a email client
IMAP4	tcp/143	Internet Message Access Protocol v4	A newer email client protocol
HTTPS	tcp/443	Hypertext Transfer Protocol Secure	Web server communication with encryption
RDP	tcp/3389	Remote Desktop Protocol	Graphical display of remote devices
NetBIOS	udp/137	NetBIOS name service	Register, remove, and find Windows services by name
NetBIOS	udp/138	NetBIOS datagram service	Windows connectionless data transfer
NetBIOS	tcp/139	NetBIOS session service	Windows connection-oriented data transfer
SLP	tcp/427, udp/427	Service Location Protocol	Find Mac OS services by name
SMB	tcp/445	Server Message Block	Windows file transfers and printer sharing
AFP	tcp/548	Apple Filing Protocol	Mac OS file transfers

Encapsulation

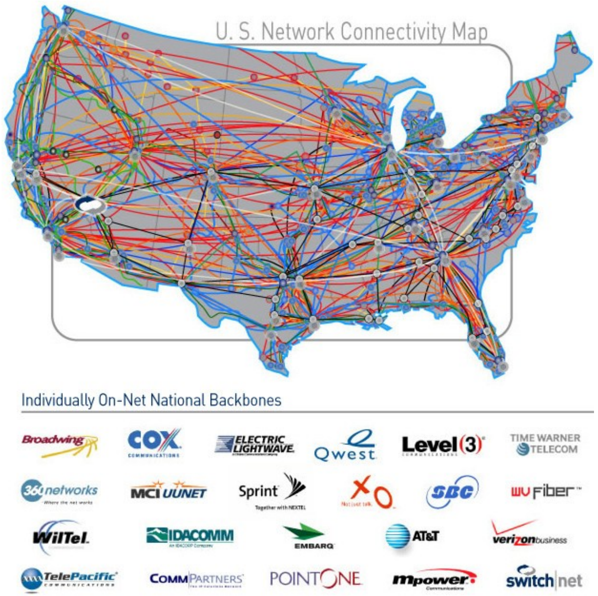
- Application Data → Segment / Datagram → Packet → Frame
  - TCP header adds sequence numbers, source port, destination port, etc
  - IP header adds source IP, destination IP, etc
  - Ethernet header adds source MAC, destination MAC, FCS, etc.



LAN vs WAN

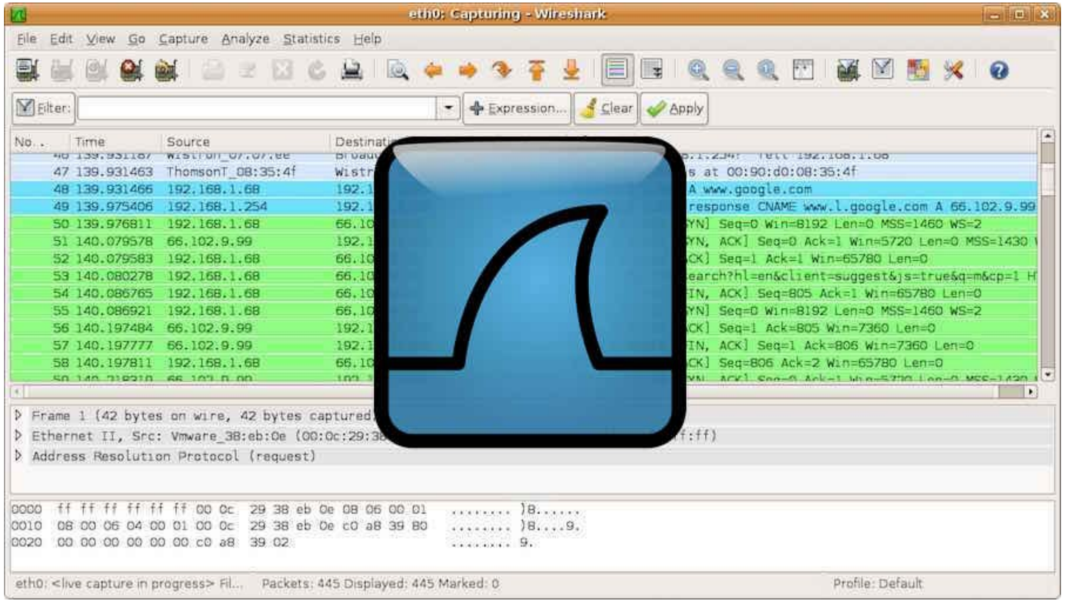
- LAN – Local Area Network
- WAN – Wide Area Network

http://www.
ROUTER



Wireshark

- Allows for the easy capture and exploration of network traffic


eth0: Capturing - Wireshark
File Edit View Go Capture Analyze Statistics Help
Filter:
Expression...
Clear
Apply

```
No..  Time        Source             Destinati
   47 139.931463  ThomsonT_08:35:4f  Wistr                                s at 00:90:d0:08:35:4f
   48 139.931466  192.168.1.68       192.1                                A www.google.com
   49 139.975406  192.168.1.254      192.1                                response CNAME www.l.google.com A 66.102.9.99
   50 139.976811  192.168.1.68       66.10                                YN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
   51 140.079578  66.102.9.99        192.1                                YN, ACK] Seq=0 Ack=1 Win=5720 Len=0 MSS=1430
   52 140.079583  192.168.1.68       66.10                                CK] Seq=1 Ack=1 Win=65780 Len=0
   53 140.080278  192.168.1.68       66.10                                earch?hl=en&client=suggest&js=true&q=m&cp=1 H
   54 140.086765  192.168.1.68       66.10                                IN, ACK] Seq=805 Ack=1 Win=65780 Len=0
   55 140.086921  192.168.1.68       66.10                                YN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
   56 140.197484  66.102.9.99        192.1                                CK] Seq=1 Ack=805 Win=7360 Len=0
   57 140.197777  66.102.9.99        192.1                                IN, ACK] Seq=1 Ack=806 Win=7360 Len=0
   58 140.197811  192.168.1.68       66.10                                CK] Seq=806 Ack=2 Win=65780 Len=0
```

```
▷ Frame 1 (42 bytes on wire, 42 bytes captured)
▷ Ethernet II, Src: Vmware_38:eb:0e (00:0c:29:38                          f:ff)
▷ Address Resolution Protocol (request)
```

```
0000  ff ff ff ff ff ff 00 0c  29 38 eb 0e 08 06 00 01   ........ )8......
0010  08 00 06 04 00 01 00 0c  29 38 eb 0e c0 a8 39 80   ........ )8....9.
0020  00 00 00 00 00 00 c0 a8  39 02                     ........ 9.
```

eth0: <live capture in progress> Fil... Packets: 445 Displayed: 445 Marked: 0 Profile: Default