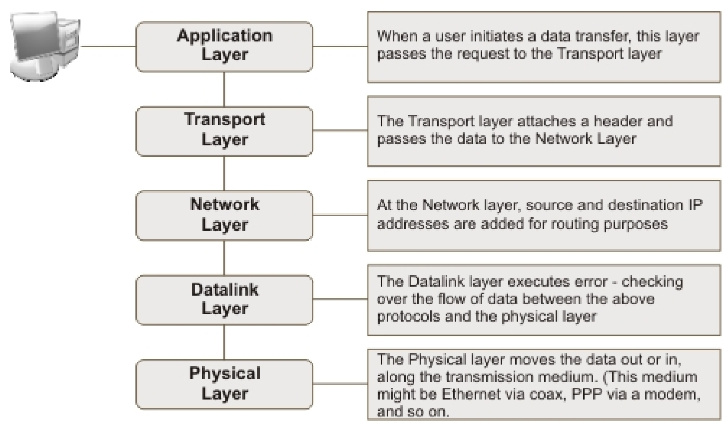
CSE 265: System and Network Administration

- TCP/IP Networking
  - We will cover just some of the practical issues
  - Highly recommend taking a networking course
- What is TCP/IP?
- Layers, addresses, NAT
- Protocols: ARP, DHCP

TCP/IP

- Most common networking protocol suite
- Foundation of the Internet
  - 2.8B+ users online worldwide (Dec 2013)
  - 1.01B+ hosts online (Jan 2014)
- Network applications typically use one of two transport protocols:
  - TCP – Transmission Control Protocol
  - UDP – User Datagram Protocol
- All traffic carried by IP – Internet Protocol

TCP/IP network stack



Protocols

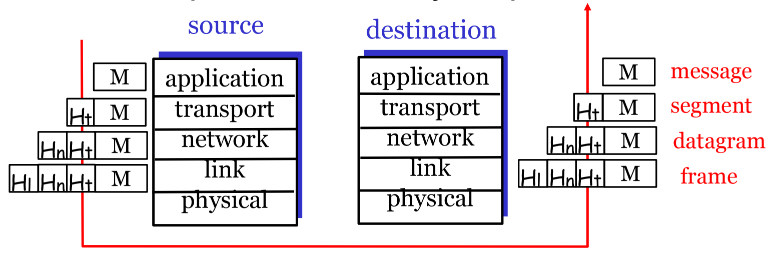
- IP
  - Packet-oriented (routers don't care what is in packets or what came before)
- TCP
  - Connection-oriented, two-way, reliable, in-order transport of stream of bytes
  - Congestion control – slow down when congestion is noticed, speed up when resources available
  - Flow control – don't overwhelm receiver
- UDP
  - Unreliable but quick/easy transport of individual packets

Address types

- IP layer and link layer have multiple address types
  - Unicast – single host (network interface)
  - Broadcast – addresses that include all hosts on a particular network
    - All bits in host part of address are ones
  - Multicast – addresses that identify a group of hosts
    - IPv4 addresses with first byte in 224-239

Layers + Encapsulation

- As data is sent downward through the stack, it is encapsulated with layer-specific headers



- App sends 100 bytes
- UDP segment adds 8 bytes of header
- IP datagram adds 20 bytes
- Ethernet frame adds 18 bytes

IP Addresses

- IPv4 address has four bytes
  - Split into network and host portions
  - Internet originally used classes of IP addresses

Class	1st byte	Format	Comments
A	1-126	N.H.H.H.	Very early networks, DoD
B	128-191	N.N.H.H.	Large sites, usually subnetted
C	192-223	N.N.N.H.	Smaller sites
D	224-239		Multicast addresses
E	240-255		Experimental

  - www.lehigh.edu = 128.180.2.57
    - Class B (128.180); host portion is .2.57

Addressing

- Different layers use different addressing
  - App. layer (usu.) allows people to use hostnames
  - IP (network) layer requires IP addresses
    - Link layer requires MAC (a.k.a. LAN) addresses
      - e.g., Ethernet (48 bits)
        - First 3 bytes are manufacturer ID
        - Last 3 bytes are serial number
- Ports identify process or service on a host
  - List of well-known ports in /etc/services
  - Ports <= 1024 are privileged ports (req. root)

CIDR

- Classless Inter-Domain Routing
  - Allows for shorter network address than class-specified – obsoletes network classes
  - Requires length field, e.g., 128.180.0.0/16
  - Aggregates smaller networks into single larger one
    - 192.200.254.0 + 192.200.255.0 = 192.200.254.0/23
  - Can now allocate portions of class A and B addresses
  - Aggregated networks reduces routing table growth

Subnetting

- Individual networks are often **much** smaller than the class sizes
- Subnetting permits breaking up an allocation into multiple smaller networks
- Lehigh breaks up its network into many smaller networks, such as the old EECS nets
  - 128.180.5.*, 128.180.98.*, 128.180.14.*
  - Each can be broken down further

Address Shortage

- Before CIDR, concern for enough addresses
  - Class Bs would be gone by 1995
  - Router tables were exploding (growing beyond router capacities)
- CIDR + NAT + name-based virtual hosting greatly slowed down IP allocations
- IPv6 solves this (16 byte addresses!)

Subnetting Example

- 128.180 under class-full addressing is a Class-B with 65,534 addresses
- Subnetting extends the network address into host portion
- We specify a subnet 128.180.98
  - Using explicit subnet mask 255.255.255.0
  - Alternatively, with network bits specified explicitly
    - 128.180.98.0/24
  - Can also break on non-byte boundaries
    - 128.180.98.128/25
    - 128.180.120.0/22

Private Addresses

- While a NAT can protect your internal addresses from being visible in IP headers, it isn't perfect
  - Some apps will encode addresses in data
  - What if you really want to connect to the external host with an IP address same as an internal host?
- Most use private address space (unroutable)

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

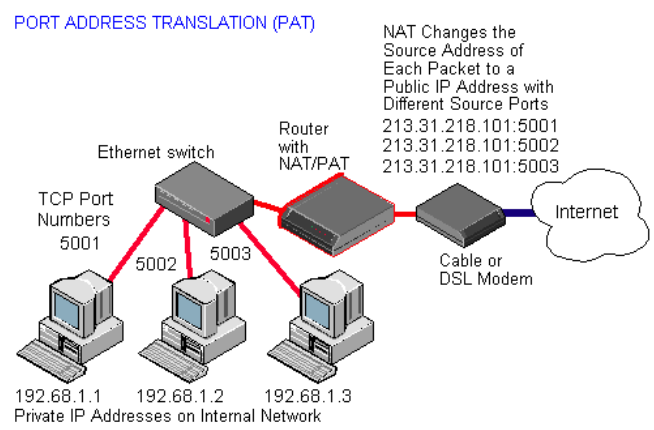
NAT

- Network Address Translation
  - Router intercepts packets, replaces internal network addresses and ports with externally visible addresses and ports
  - Maintains mapping so that external packets are directed to the right internal host
  - Typically uses a single public IP address, many ports, but can (in theory) map arbitrary hosts/ports
  - Capability built into many (cheap) routers, Linux

ARP: Address Resolution Protocol

- Once the routing of a packet has been determined, it must be transmitted to the next gateway or host on the local network
- LAN transmissions use LAN (MAC) addresses
- ARP is used to discover the hardware address of the target IP address
- ARP sends a LAN broadcast asking who has the desired IP address; the owner responds with a unicast message with answer
  - Results cached in a table (also collected via snooping)

NAT: Network Address Translation



Mapping names to IP addresses

- Three choices: /etc/hosts, NIS, DNS
- Simplest: /etc/hosts

```
% more /etc/hosts
#
# Internet host table
#
127.0.0.1       localhost
128.180.120.15  proxima
128.180.120.9   mailhost
128.180.120.103 ariel
```

- Works when NIS or DNS is broken
  - e.g., at boot time

Sample ARP table

```
% /sbin/arp -a

davison.cse.lehigh.edu (128.180.121.225) at 00:11:43:A0:0F:D8 [ether] on eth0
wume2.cse.lehigh.edu (128.180.121.222) at 00:08:54:1E:44:D4 [ether] on eth0
pan.cse.lehigh.edu (128.180.120.90) at 00:14:4F:0F:9C:1A [ether] on eth0
wume1.cse.lehigh.edu (128.180.121.221) at 00:08:54:1E:44:D0 [ether] on eth0
chiron.cse.lehigh.edu (128.180.120.87) at 00:14:4F:21:44:D8 [ether] on eth0
xena.cse.lehigh.edu (128.180.120.86) at 00:14:4F:21:52:E0 [ether] on eth0
hydra.cse.lehigh.edu (128.180.120.89) at 00:14:4F:21:53:F2 [ether] on eth0
kato.eecs.lehigh.edu (128.180.120.6) at 08:00:20:C4:20:08 [ether] on eth0
noon.cse.lehigh.edu (128.180.121.219) at 00:0F:1F:F9:C1:68 [ether] on eth0
wume-lab2.cse.lehigh.edu (128.180.122.153) at 00:18:8B:24:5A:F4 [ether] on eth0
lu-gw.eecs.lehigh.edu (128.180.123.254) at 00:00:0C:07:AC:00 [ether] on eth0
nix.cse.lehigh.edu (128.180.120.88) at 00:14:4F:21:44:C4 [ether] on eth0
ceres.cse.lehigh.edu (128.180.120.91) at 00:14:4F:23:F9:80 [ether] on eth0
rosie.eecs.lehigh.edu (128.180.120.4) at 08:00:20:B1:FC:F3 [ether] on eth0
wume-lab1.cse.lehigh.edu (128.180.122.152) at 00:18:8B:24:5D:E2 [ether] on eth0
morning.cse.lehigh.edu (128.180.120.43) at 00:C0:9F:38:CD:51 [ether] on eth0
wume-lab6.cse.lehigh.edu (128.180.122.157) at 00:0A:E6:5D:48:03 [ether] on eth0
```


ifconfig

- Configure network interfaces with ifconfig
  - ifconfig eth0 128.138.240.1 netmask 255.255.255.0 up
  - shows configuration, e.g., for Solaris:

```
phobos:~% ifconfig -a
eth0      Link encap:Ethernet  HWaddr 88:51:FB:6F:F3:37
          inet addr:128.180.120.85  Bcast:128.180.123.255  Mask:255.255.252.0
          inet6 addr: fe80::8a51:fbff:fe6f:f337/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:82607119 errors:0 dropped:0 overruns:0 frame:0
          TX packets:52787875 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:23578082323 (21.9 GiB)  TX bytes:55411462770 (51.6 GiB)
          Interrupt:20 Memory:ec100000-ec120000
```

  - You've seen the output of ifconfig from your boot logs

Network Configuration

- Adding a machine to a LAN
  - Assign unique IP address and hostname (per interface)
  - Set up host to configure network interfaces at boot time
  - Set up default route
  - Point to DNS name server (resolver)
- Files
  - /etc/sysconfig/network-scripts/ifcfg-eth0
  - Hostname, default route, IP address, netmask, broadcast
- DHCP could do all of this automatically

CentOS/RHEL configuration files

- /etc/sysconfig/network
  - hostname, default route
- /etc/sysconfig/static-routes
  - static routes
- /etc/sysconfig/network-scripts/ifcfg-XXXX
  - IP address, netmask, broadcast address per interface
  - e.g., eth0, eth1, lo
- Use **ifup** and **ifdown** scripts to change interface status, or use /etc/init.d/network for all of networking

DHCP

- Dynamic Host Configuration Protocol
- Clients **lease** network config from server
  - IP addresses and netmasks
  - Gateways (default routes)
  - DNS name servers
  - Syslog hosts
  - X font servers, proxy servers, NTP servers
  - and more

How DHCP works
(at a high level)

- Client broadcasts a “Who am I?” message
- Local DHCP server responds with network configuration lease
- When lease is half over, client renews the lease
  - DHCP server must track lease info (persist through server reboots, etc.)
- DHCP used on essentially all hosts at Lehigh