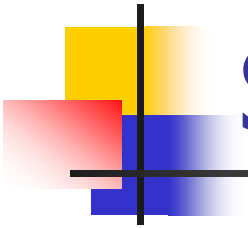
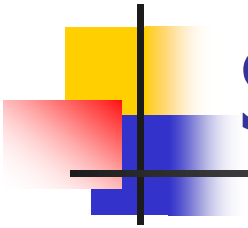
Lecture No 03

# Subnetting, NAT



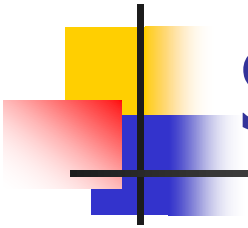
# Subnet Routing

- Conventional routing table entry
  - (network address, next hop address)
    - Network address format is predetermined for a given class (e.g., first 16 bits for class B addresses!)
- With subnetting, routing table entry becomes
  - (subnet mask, network address, next hop address)
  - Then compare with network address field of entries to find next hop address
    - Subnet mask indicates the network address!



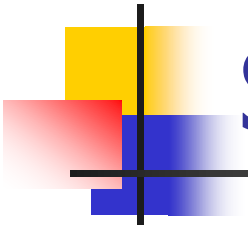
# Subnet Routing

- The use of mask generalizes the subnet routing algorithm to handle all the special cases of the standard algorithm
  - Routes to individual hosts
  - Default route
  - Routes to directly connected networks
  - Routes to conventional networks (that do not use subnet addressing)
- Merely combine the 32-bit mask field with the 32-bit IP address
- Example: To install a route for:
  - Individual host (Mask of all 1's, Host IP address)
  - Default Route (Mask of all 0's, network address all 0's)
  - Class B network address (Mask of two octets of 1's and two of 0's)



# Subnet Routing

- Algorithm
  - Extract destination IP (D) from datagram
  - Compute IP address of destination network N
  - If N matches any directly connected network address
    - Send datagram over that network (obviously encapsulated in a frame)
  - Else
    - For each entry in the routing table, do
    - N* = bitwise-AND of D and subnet mask
    - If N* equals the network address field of the entry, then route the datagram to the specified next hop



# Subnetting: Example

- Consider a corporate network assigned a class C address P.Q.R.00000000
- The company needs 5 subnets:
  - 2 subnets of 16 hosts each
  - 3 subnets with 32, 64, and 128 hosts
- External routers reach the corporate via single routing table entry
  - P.Q.R.0 network and 255.255.255.0 mask (if any)
- What about internal routers?

# Subnetting: Example

| IP addresses | Subnet Mask | Network/Subnet address | Subnet Name |
|---|---|---|---|
| P.Q.R.0000 hhhh | 255.255.255.1111 0000 | P.Q.R.0000 0000 | S1 |
| P.Q.R.0001 hhhh | 255.255.255.1111 0000 | P.Q.R.0001 0000 | S2 |
| P.Q.R.001 0 hhhh | 255.255.255.111 00000 | P.Q.R.001 00000 | S3 |
| P.Q.R.001 1 hhhh | 255.255.255.111 00000 | | |
| P.Q.R.01 00 hhhh | 255.255.255.11 000000 | P.Q.R.01 000000 | S4 |
| P.Q.R.01 01 hhhh | 255.255.255.11 000000 | | |
| P.Q.R.01 10 hhhh | 255.255.255.11 000000 | | |
| P.Q.R.01 11 hhhh | 255.255.255.11 000000 | | |
| P.Q.R.1 000 hhhh | 255.255.255.1 0000000 | P.Q.R.1 0000000 | S5 |
| P.Q.R.1 001 hhhh | 255.255.255.1 0000000 | | |
| P.Q.R.1 010 hhhh | 255.255.255.1 0000000 | | |
| P.Q.R.1 011 hhhh | 255.255.255.1 0000000 | | |
| P.Q.R.1 100 hhhh | 255.255.255.1 0000000 | | |
| P.Q.R.1 101 hhhh | 255.255.255.1 0000000 | | |
| P.Q.R.1 110 hhhh | 255.255.255.1 0000000 | | |
| P.Q.R.1 111 hhhh | 255.255.255.1 0000000 | | |

# Subnetting: Example

| IP addresses | Subnet Mask | Network/Subnet address | Subnet Name |
|---|---|---|---|
| P.Q.R.0000 hhhh | 255.255.255.1111 0000 | P.Q.R.0000 0000 | S1 |
| P.Q.R.0001 hhhh | 255.255.255.1111 0000 | P.Q.R.0001 0000 | S2 |
| P.Q.R.001 hhhhh | 255.255.255.1110 0000 | P.Q.R.001 00000 | S3 |
| P.Q.R.01 hhhhhh | 255.255.255.11 000000 | P.Q.R.01 000000 | S4 |
| P.Q.R.1 hhhhhhh | 255.255.255.1 0000000 | P.Q.R.1 0000000 | S5 |

# Subnetting: Routing Table

| Subnet Mask | Network/Subnet address | Next Hop/Port |
|---|---|---|
| 255.255.255.1111 0000 | P.Q.R.0000 0000 | P1 |
| 255.255.255.1111 0000 | P.Q.R.0001 0000 | P2 |
| 255.255.255.1110 0000 | P.Q.R.0010 0000 | P3 |
| 255.255.255.11 000000 | P.Q.R.0100 0000 | P4 |
| 255.255.255.1 0000000 | P.Q.R.1000 0000 | P5 |

# Subnetting: Routing Table

| Network/Subnet address | Next Hop/Port |
|---|---|
| P.Q.R.0000 0000 / 28 | P1 |
| P.Q.R.0001 0000 / 28 | P2 |
| P.Q.R.0010 0000 / 27 | P3 |
| P.Q.R.0100 0000 / 26 | P4 |
| P.Q.R.1000 0000 / 25 | P5 |

Number after / indicates number of bits to look at!

# Subnetting: Routing Table

Subnet S4 has 64 hosts. Can we make two subnets? 16+48?

| | | | |
|---|---|---|---|
| P.Q.R.01 hhhhhh | 255.255.255.11 000000 | P.Q.R.01 000000 | S4 |

| | Old mask | Old subnet | New mask |
|---|---|---|---|
| P.Q.R.01 00 hhhh | 255.255.255.11 000000 | P.Q.R.0100 0000 | P.Q.R.01 00 hhhh | 255.255.255.1111 0000 |
| P.Q.R.01 01 hhhh | 255.255.255.11 000000 | | P.Q.R.01 hhhhhh | 255.255.255.11 000000 |
| P.Q.R.01 10 hhhh | 255.255.255.11 000000 | | P.Q.R.01 hhhhhh | 255.255.255.11 000000 |
| P.Q.R.01 11 hhhh | 255.255.255.11 000000 | | P.Q.R.01 hhhhhh | 255.255.255.11 000000 |

| | | | |
|---|---|---|---|
| P.Q.R.0100 hhhh | 255.255.255.1111 0000 | P.Q.R.0100 0000 | S41 |
| P.Q.R.01 hhhhhh | 255.255.255.11 000000 | P.Q.R.01 000000 | S42 |

# Subnetting: Routing Table

| | | | |
|---|---|---|---|
| P.Q.R.0100 hhhh | 255.255.255.1111 0000 | P.Q.R.0100 0000 | S41 |
| P.Q.R.01 hhhhhh | 255.255.255.11 000000 | P.Q.R.01 000000 | S42 |

What if an IP in S42 is received?

It will match on the second entry!

What if an IP in S41 is received?

It will match both entries!

Which entry should be used?

USE LONGEST PREFIX MATCH

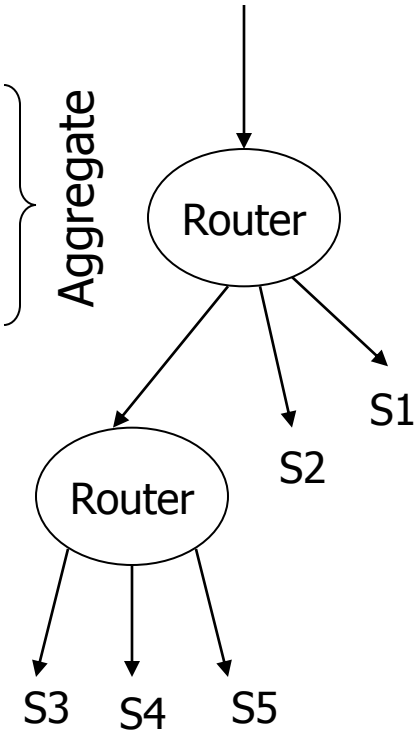
# Subnetting: Routing Table

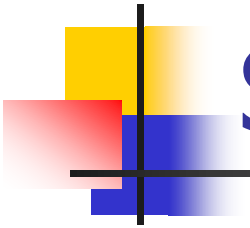
Where else longest prefix match can be used?

| Subnet Mask | Network/Subnet address | Next Hop/Port |
|---|---|---|
| 255.255.255.1111 0000 | P.Q.R.0000 0000 | P1 |
| 255.255.255.1111 0000 | P.Q.R.0001 0000 | P2 |
| 255.255.255.1110 0000 | P.Q.R.0010 0000 | P345 |
| 255.255.255.11 000000 | P.Q.R.0100 0000 | P345 |
| 255.255.255.1 0000000 | P.Q.R.1000 0000 | P345 |

Aggregate

| Subnet Mask | Network/Subnet address | Next Hop/Port |
|---|---|---|
| 255.255.255.1111 0000 | P.Q.R.0000 0000 | P1 |
| 255.255.255.1111 0000 | P.Q.R.0001 0000 | P2 |
| 255.255.255.0000 0000 | P.Q.R.0000 0000 | P345 |





# Supernet Addressing

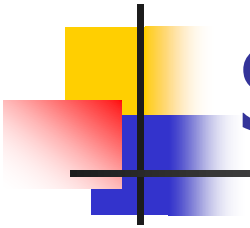
- Use of many IP network addresses for a single organization
- Example:
  - To conserve class B addresses, issue multiple class C address to the same organization
  - Issue: increase in the number of entries in the routing tables for routers outside the network
  - Solutions:
    - Collapse a block of contiguous class C address into the pair: (network address, count) where network address is the smallest number in the block

# Supernet Addressing

- It requires each block to be a power of 2 and uses bit mask to identify the size of the block
- Example

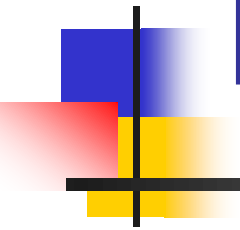
| | Dotted decimal | 32-bit binary equivalent |
|---|---|---|
| Lowest: | 234.170.168.0 | 11101010 10101010 10101000 00000000 |
| Highest: | 234.170.175.255 | 11101010 10101010 10101111 11111111 |

  - A block of 2048 addresses
  - 32-bit mask is 11111111 11111111 11111000 00000000
- Do we really need address classes when we have masks?
  - Answer: NO → CIDR (Classless Inter Domain Routing)

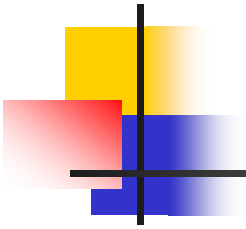


# Supernet Addressing

- In the router, the entry consists of:
  - The lowest address and the 32-bit mask
  - A block of addresses can be subdivided, and separate route can be entered for each subdivision
  - When looking up a route, the routing software uses a longest-match paradigm to select a route



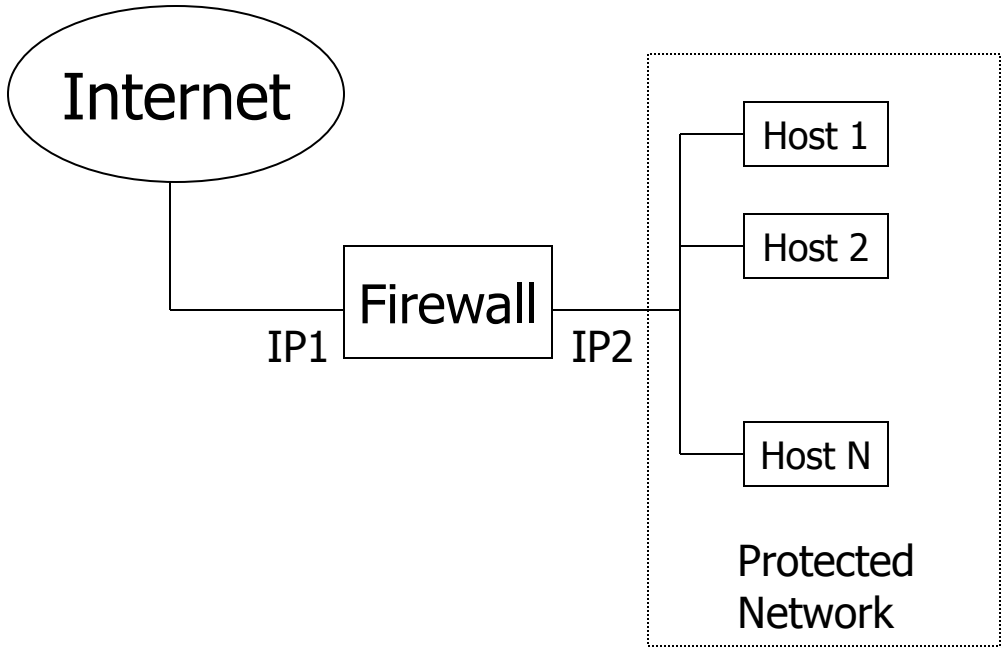
# Network Address Translation



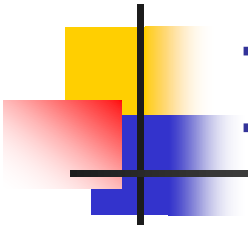
# Private Networks

- Private networks have no “direct” connection to the Internet
- Blocks of addresses have been reserved for the private networks (RFC 1918)
- Blocks in different classes
  - 10.0.0.0 – 10.255.255.255 (1 class A)
  - 172.16.0.0 – 172.31.255.255 (16 class B)
  - 192.168.0.0 – 192.168.255.255 (256 class C)

# Purpose

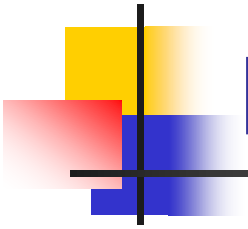



- Machines in the protected network can access the Internet normally
- Packets coming from the protected network all appear to be coming from IP1
- Addresses in the protected network are in the private range



# Implementation

- Hosts inside the private network are configured to use the firewall (IP2) as their gateway
- The firewall rewrites the IP datagram header for the outbound packets, replacing the source IP with IP1
  - All packets "seem" to be coming from IP1
- The destination IP in the packets received from the Internet is IP1; it is rewritten replacing IP1 with the IP address of the internal destination
- Problem: How to figure out what is the right destination in the private network?

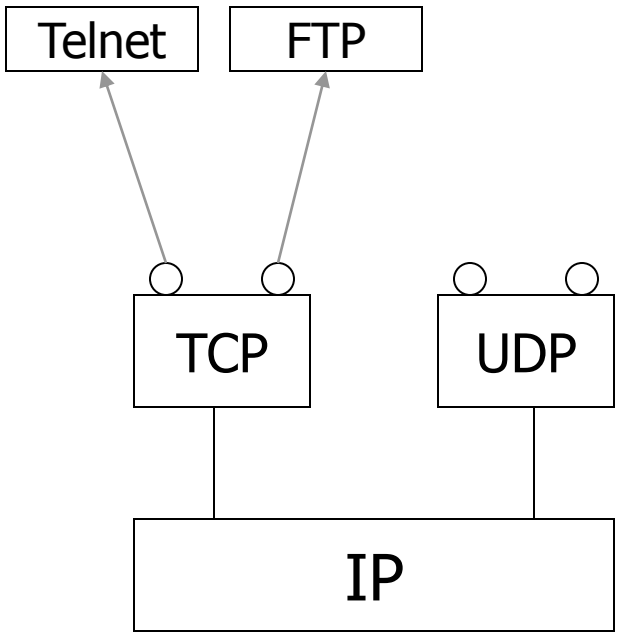


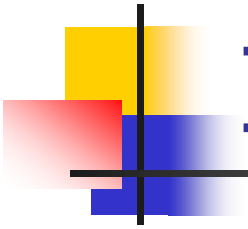
# Demultiplexing Incoming Packets

- There is not enough information in the IP header to demultiplex incoming packets
- It is necessary to use information from the higher layers (transport layer)
- Common transport layers: TCP and UDP
- Transport layer has the concept of port which identifies which process in the host should finally get the packet

# Ports

- 16-bit numbers identifying which process should get the packet
- UDP and TCP ports exist in different spaces
- Each packet carries two port numbers
  - The source port of the process which generated it in the source host
  - The destination port of the process which should get it at the destination





# Implementation (revisited)

- Upon receiving an outbound packet from a host in the private network, the firewall:
  - Rewrites the source IP with its own IP (IP1)
  - Generates a local source port and rewrites the source port in the packet as this port and makes a record of it
- Upon receiving an inbound packet from the Internet, the firewall checks whether the destination port in the packet is in the list of local ports:
  - If not, the packet is dropped
    - Can not initiate connections from outside!
  - If yes, the firewall knows where to send this packet