# Communication Networks (0368-3030) / Spring 2011

The Blavatnik School of Computer Science, Tel-Aviv University

Allon Wagner

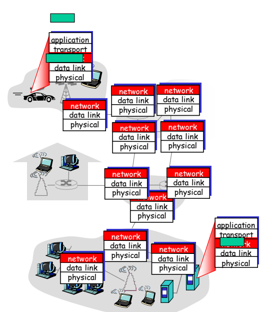
# Network Layer – Forwarding

Kurose & Ross, Chapter 4 (5th ed.)

Many slides adapted from:
J. Kurose & K. Ross \
Computer Networking: A Top Down Approach (5th ed.)
Addison-Wesley, April 2009.
Copyright 1996-2010, J.F Kurose and K.W. Ross, All Rights Reserved.

## Network layer

- transport segment from sending to receiving host
- on sending side encapsulates segments into datagrams
- on rcving side, delivers segments to transport layer
- network layer protocols in *every* host, router
- router examines header fields in all IP datagrams passing through it






## Two Key Network-Layer Functions

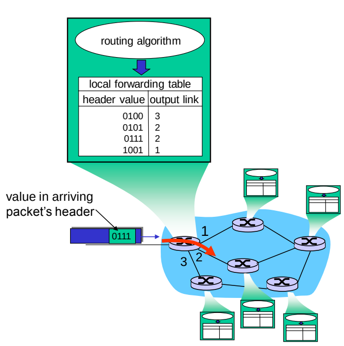
- *forwarding:* move packets from router's input to appropriate router output
- *routing:* determine route taken by packets from source to dest.
  - *routing algorithms*

analogy:

- routing: process of planning trip from source to dest
- forwarding: process of getting through single interchange



## Interplay between routing and forwarding






## Network service model

Q: What *service model* for "channel" transporting datagrams from sender to receiver?

example services for individual datagrams:

- guaranteed delivery
- guaranteed delivery with less than 40 msec delay

example services for a flow of datagrams:

- in-order datagram delivery
- guaranteed minimum bandwidth to flow
- restrictions on changes in inter-packet spacing

## Network layer service models:

| Network Architecture | Service Model | Guarantees ? Bandwidth | Loss | Order | Timing | Congestion feedback |
|---|---|---|---|---|---|---|
| Internet | best effort | none | no | no | no | no (inferred via loss) |
| ATM | CBR | constant rate | yes | yes | yes | no congestion |
| ATM | VBR | guaranteed rate | yes | yes | yes | no congestion |
| ATM | ABR | guaranteed minimum | no | yes | no | yes |
| ATM | UBR | none | no | yes | no | no |

**IP provides best-effort service**



## Virtual circuits

"source-to-dest path behaves much like telephone circuit"
- performance-wise
- network actions along source-to-dest path

- call setup, teardown for each call *before* data can flow
- each packet carries VC identifier (not destination host address)
- *every* router on source-dest path maintains "state" for each passing connection
- link, router resources (bandwidth, buffers) may be *allocated* to VC (dedicated resources = predictable service)



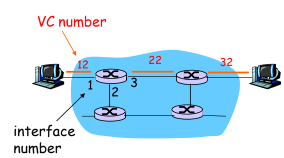
## VC implementation

a VC consists of:
1. path from source to destination
2. VC numbers, one number for each link along path
3. entries in forwarding tables in routers along path

- packet belonging to VC carries VC number (rather than dest address)
- VC number can be changed on each link.
  - New VC number comes from forwarding table



## VC Forwarding table




Forwarding table in northwest router:

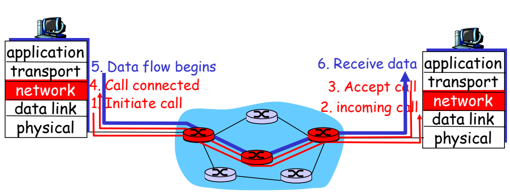
| Incoming interface | Incoming VC # | Outgoing interface | Outgoing VC # |
|---|---|---|---|
| 1 | 12 | 3 | 22 |
| 2 | 63 | 1 | 18 |
| 3 | 7 | 2 | 17 |
| 1 | 97 | 3 | 87 |
| … | … | … | … |

Routers maintain connection state information!



## Virtual circuits: signaling protocols

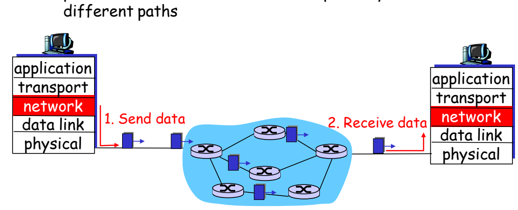
- used to setup, maintain teardown VC
- used in ATM, frame-relay, X.25
- not used in today's Internet






## Datagram networks

- no call setup at network layer
- routers: no state about end-to-end connections
  - no network-level concept of "connection"
- packets forwarded using destination host address
  - packets between same source-dest pair may take different paths



## Datagram or VC network: why?

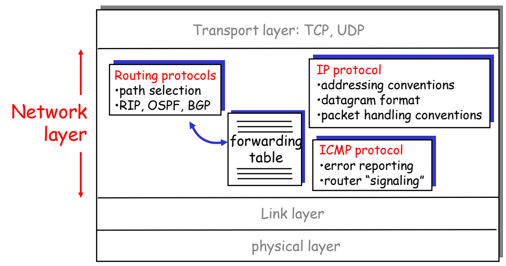
**Internet (datagram)**
- data exchange among computers
  - "elastic" service, no strict timing req.
- "smart" end systems (computers)
  - can adapt, perform control, error recovery
  - simple inside network, complexity at "edge"
- many link types
  - different characteristics
  - uniform service difficult

**ATM (VC)**
- evolved from telephony
- human conversation:
  - strict timing, reliability requirements
  - need for guaranteed service
- "dumb" end systems
  - telephones
  - complexity inside network



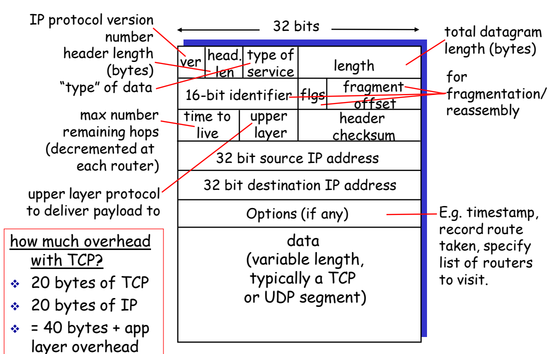
## The Internet Network layer

Host, router network layer functions:






## IP datagram format



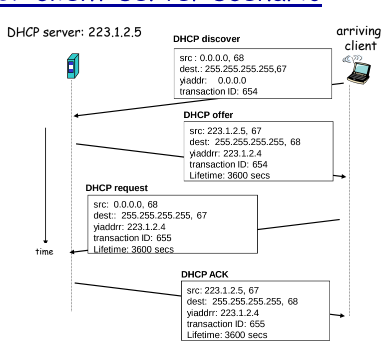





## Dynamic Host Configuration Protocol

- How can a newly connected host get an IP address?
  - (And other useful information: its network mask, an IP address of DNS server etc.)
  - Manual configuration by a system administrator
  - DHCP
    - Plug and play – host obtains this information automatically
    - Defined in RFCs 2131 & 2132.
    - Sent over UDP. Server's port: 67, client's port: 68.
    - As usual, we'll give an overview and not go into all the technical details.

## DHCP client-server scenario






## DHCP client-server scenario 1

- Arriving client sends a DHCP discover message
  - Src IP: 0.0.0.0 ("this host")
  - Dst IP: 255.255.255.255 ("broadcast")
  - Transaction ID: some number x.
  - Message is broadcast to all nodes on the subnet

## DHCP client-server scenario 2

- Server replies with a DHCP offer message
  - Src IP: server's IP
  - Dst IP: 255.255. 255.255 ("broadcast")
    - Why? The client still can't receive direct messages – it has no IP address.
  - Transaction ID: x
  - Message contains the proposed IP address .

## DHCP client-server scenario 3

- Why aren't the previous messages enough?
  - The network might contain more than one DHCP server; a client may receive multiple DHCP offers.
- Client sends a DHCP request message
  - Src IP: 0.0.0.0 ("this host")
  - Dst IP: 255.255.255.255 ("broadcast")
    - Why? To allow other servers know their offer was declined
  - DHCP Server ID: the IP of the server whose offer the client wish to accept
  - Transaction ID: some number y

## DHCP client-server scenario 4

- Server responds with a DHCP ACK message
  - Src IP: Server's IP
  - Dst IP: 255.255.255.255 ("broadcast")
    - The client still doesn't have an IP address
  - Transaction ID: y

## Lease times

- DHCP servers assigns a lease-time for each IP address allocation
  - A client may renew its allocation when it is about to expire
  - A client may relinquish its allocation

## Internet Control Message Protocol

- ICMP – defined in RFC 792.
- Carried directly over IP
  - No transport protocol used
- Used by hosts and routers to communicate network layer information – usually report errors.

## ICMP: Internet Control Message Protocol

- used by hosts & routers to communicate network-level information
  - error reporting: unreachable host, network, port, protocol
  - echo request/reply (used by ping)
- network-layer "above" IP:
  - ICMP msgs carried in IP datagrams
- ICMP message: type, code plus first 8 bytes of IP datagram causing error

| Type | Code | description |
|---|---|---|
| 0 | 0 | echo reply (ping) |
| 3 | 0 | dest. network unreachable |
| 3 | 1 | dest host unreachable |
| 3 | 2 | dest protocol unreachable |
| 3 | 3 | dest port unreachable |
| 3 | 6 | dest network unknown |
| 3 | 7 | dest host unknown |
| 4 | 0 | source quench (congestion control - not used) |
| 8 | 0 | echo request (ping) |
| 9 | 0 | route advertisement |
| 10 | 0 | router discovery |
| 11 | 0 | TTL expired |
| 12 | 0 | bad IP header |

Network Layer 4-24

## ICMP (cont.)

- For instance: "Destination unreachable":
  - A router was unable to find a path to host B specified in host A's request
  - The router sends an ICMP message (type 3) to A to indicate the error
- Another common use: ping
  - Host A sends an ICMP message (type 1) to host B
  - Host B sees this, and replies to A with another type 1 ICMP message.

## Traceroute and ICMP

- Source sends series of UDP segments to dest
  - first has TTL =1
  - second has TTL=2, etc.
  - unlikely port number
- When nth datagram arrives to nth router:
  - router discards datagram
  - and sends to source an ICMP message (type 11, code 0)
  - ICMP message includes name of router & IP address
- when ICMP message arrives, source calculates RTT
- traceroute does this 3 times

Stopping criterion

- UDP segment eventually arrives at destination host
- destination returns ICMP "port unreachable" packet (type 3, code 3)
- when source gets this ICMP, stops.

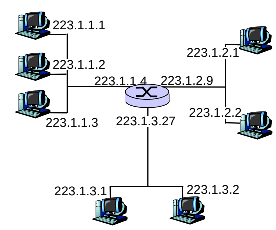


## Extra slides

Review of lecture, if time permits

## IP Addressing: introduction

- IP address: 32-bit identifier for host or router *interface*
- *interface:* connection between host/router and physical link
  - router's typically have multiple interfaces
  - a host has typically a single interface
  - IP addresses associated with interface, not host, or router



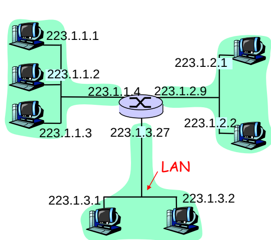

223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1



## IP Addressing

- IP address is divided into two parts:
  - network prefix
    - K high order bits
  - host number
    - remaining low order bits
- This partitioning of the address depends on the context network in which we see this NIC
  - networks are nested inside each other



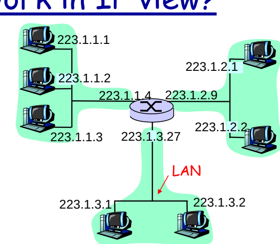

**Qn:** What is the router's IP address in the drawing we see?



## What is a network in IP view?

IP network terminology:

- *a Subnet is:*
  - a set of devices that can physically reach each other without intervening router(s)
  - e.g. a LAN
- a Network is:
  - a subnet , or:
  - the union of several subnets that are interconnected by links



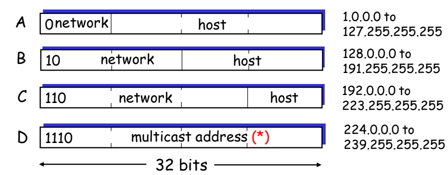

three subnets (LANs) 223.1.1.*, 223.1.2.*, 223.1.3.*, together they form a larger network with prefix 223.1 (16 bits) **(OR MORE bits?)**

# IP Addresses

given notion of "network", let's re-examine IP addresses:

"classful" addressing:

(does not need mask or /K indicator)




(*) this range used as multicast also in CIDR method



# IP addressing: CIDR

- classful addressing:
  - inefficient use of address space, address space exhaustion
  - e.g., class B net allocated enough addresses for 65K hosts, even if only 2K hosts in that network
- CIDR: Classless InterDomain Routing
  - network portion of address of arbitrary length
  - address format: a.b.c.d/x, where x is # bits in network portion of address
  - Requires inclusion of mask or "/K" in routing table

network part | host part

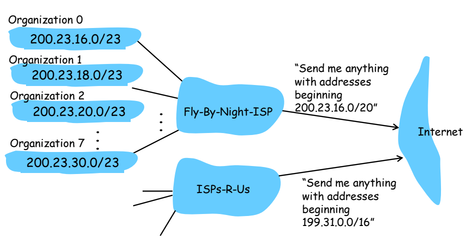
11001000 00010111 00010000 00000000

200.23.16.0/23



# Hierarchical addressing: route aggregation

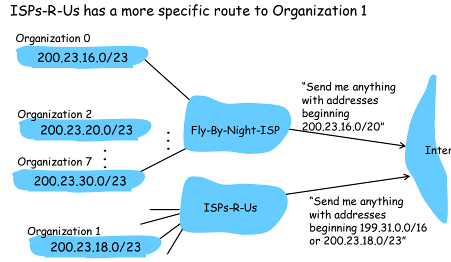
Hierarchical addressing allows efficient advertisement of routing information:






# Hierarchical addressing: more specific routes

ISPs-R-Us has a more specific route to Organization 1






# IP addresses: how to get one?

Q: How does *network* get subnet part of IP addr?

A: gets allocated portion of its provider ISP's address space

| | | |
|---|---|---|
| ISP's block | 11001000 00010111 00010000 00000000 | 200.23.16.0/20 |
| Organization 0 | 11001000 00010111 00010000 00000000 | 200.23.16.0/23 |
| Organization 1 | 11001000 00010111 00010010 00000000 | 200.23.18.0/23 |
| Organization 2 | 11001000 00010111 00010100 00000000 | 200.23.20.0/23 |
| ... | ….. …. | …. |
| Organization 7 | 11001000 00010111 00011110 00000000 | 200.23.30.0/23 |



# IP addressing: the last word...

Q: How does an ISP get block of addresses?

A: ICANN: Internet Corporation for Assigned Names and Numbers
- allocates addresses
- manages DNS
- assigns domain names, resolves disputes