Internetworking

Problem

• Aim: Build networks connecting millions of users around the globe
  – also spanning networks based on any technology

• Problems: heterogeneity and scalability
  – bridges can be used to connect different LANs (extended LANs)
  – heterogeneity: need to support different LANs, point-to-point technologies, switched networks, different addressing formats
  – scalability: addressing (management and configuration) and routing must be able to handle millions of hosts
  – in this lecture, we examine the (original) IP protocol, IP addressing, packet forwarding
Outline

- Internet architecture
- IP service model
- IP forwarding
- Address translation (ARP)
- Automatic host configuration (DHCP) and error reporting (ICMP)
- Virtual Private Networks (VPNs)

• Terminology
  - network = network based on LAN or extended LAN technology
  - internet = “network of networks”
  - Internet = internet using IP
  - routers = nodes connecting networks
  - IP = Internet Protocol, current version IPv4 (IP Version 4)
IP design principles

• Cerf and Kahn’s internet design principles (1974)
  – minimalism, autonomy
    • no internal changes required to interconnect networks
    • network is self-configuring as much as possible
    • network can survive node and link failures
  – best effort service model
    • packets are not offered any guarantees
    • simplifies packet processing
  – stateless routers
    • network does not store information of any “connections” or user state
    • routers forward autonomous packets
  – decentralized control
    • enables high survivability (in presence of, e.g., link or node failures)

Internet architecture

• Internet architecture has only 4 layers
  – L4 (Application layer): FTP, HTTP, ...
  – L3 (Transport layer): TCP (reliable byte transfer) and UDP (unreliable datagram delivery) provide logical channels to applications
  – L2 (IP layer): IP protocol interconnects multiple networks into a single logical network
  – L1 (“Link” layer): wide variety of LAN and point-to-point protocols

• Internet architecture features
  – Does not imply strict layering
  – IP defines a common way for exchanging packets among widely differing networks
  – “Hour glass”-shape

• Aim: heterogeneity and scalability
IP protocol stack

IETF (Internet Engineering Task Force)

- Majority of Internet development (standardization) done by IETF
  - offers a mutual forum for the development of the Internet to vendors, users, researchers, service providers and network managers
  - develops architectures and protocols for solving technical issues
  - gives recommendations on the use of protocols
  - performs dissemination of the recommendations of IRTF (Internet Research Task Force) which is responsible for long term development of Internet
  - IETF requires always working implementations before any protocol specification is accepted as a standard (“we believe in running code”)

- Working methods
  - has meetings 3 times a year
  - work conducted within study groups (> 100 study groups)
    - joining a group done via e-mail to the mailing list
  - study groups belong to 8 different fields

- work reported in Internet drafts and RFCs (Request for Comments)
  - Internet drafts have no official status, serve as basis for RFCs
  - not all RFCs are standards (Informational, Best Current Practice, …)
  - http://www.ietf.org
Outline

- Internet architecture
- IP service model
  - IP forwarding
  - Address translation (ARP)
  - Automatic host configuration (DHCP) and error reporting (ICMP)
  - Virtual Private Networks (VPNs)

Service model

- Idea in the Internet service model:
  - Make it undemanding enough that IP can be run over anything
  - One of the major reasons for the success of IP technology

- Service model consists of 2 parts:
  - Model for data delivery
  - Addressing scheme
Data delivery model

- Data delivery in Internet
  - IP network connectionless (datagram-based)
  - IP network offers best-effort delivery (unreliable service)
    - packets are lost
    - packets are delivered out of order
    - duplicate copies of a packet are delivered
    - packets can be delayed for a long time
    - ⇒ “intelligence” implemented at the end hosts
  - datagram format (next slide)

IP datagram format details

- Format aligned at 32 bit words
  - simplifies packet processing in sw
- Fields
  - Version: currently version 4 (6 is coming)
  - HLen: header length, 32 bit words (min 5)
  - TOS: type of service, used to give priorities to packets (QoS lecture)
  - Length: datagram+header length, in bytes
  - 2nd word for fragmentation/reassembly
  - TTL: time to live, nof times packet allowed to be forwarded (nof hops), default 64, detects packets caught in routing loop
  - Protocol: identifies upper layer protocols, TCP (6), UDP (17)
  - Checksum: erroneous packets discarded
  - Addresses: global Internet addresses
  - Options: rarely used
Fragmentation and reassembly

- Each network has an MTU (Maximum Transfer Unit)
  - Ethernet 1500 bytes, FDDI 4500 bytes, PPP 512 bytes
- Strategy
  - fragment when necessary (MTU < datagram length)
  - try to avoid fragmentation at source host
    - host sets datagram size equal to MTU of home network
    - for ATM MTU based on CS-PDU size (not cell size)
  - fragments are self-contained datagrams
    - each fragment contains a common identifier in Ident field
    - Flags (M-bit) and Offset used to guide fragmentation process
      - Offset measured in 8B units
    - fragmented packet can be again re-fragmented
  - reassembly performed only at destination host
  - reassembly does not try to recover from lost fragments

Fragmentation/reassembly example

- Original message 1400B + 20B header
IP addressing

• Properties
  – globally unique, 32 bits
  – hierarchical: network + host
  – address identifies interface
    • end host has 1 interface
    • router has many interfaces
  – IP address ≠ domain name

• Original classful addressing
  – class A, B and C networks
  – defines different sized networks
  – idea: small nof WANs, modest nof campus networks, large nof LANs

• Dot Notation
  – 32 bit addresses represented as group of 8 bit integers
  – e.g., 10.3.2.4, 128.96.33.81

Outline

• Internet architecture
• IP service model
• IP forwarding
  • Address translation (ARP)
  • Automatic host configuration (DHCP) and error reporting (ICMP)
  • Virtual Private Networks (VPNs)
IP forwarding (1)

- Some terminology:
  - **forwarding**:
    - process of taking a packet from input interface, and …
    - based on the contents of the **forwarding table**, determining the correct output interface for the packet
  - **routing**:
    - process of constructing forwarding tables that enable efficient routing of traffic in the network (lecture 5)

IP forwarding (2)

- Preliminaries
  - Every datagram contains destination’s address
  - Every node has a forwarding table
    - normal hosts with one interface have only **default router** configured
    - routers maintain forwarding tables with multiple entries (constructed via routing process)
    - forwarding table maps network number into next hop router number or local interface number

- Strategy
  - Any node receiving a packet (router/host) checks destination **network address** of datagram and ...
    - if directly connected to destination network, then forward to host
      - need to map IP address to physical LAN address ⇒ **ARP**
    - if not directly connected to destination network, then forward to next hop router
IP forwarding example

- $H_1 \rightarrow H_3$: forwarding on the same network
- $H_1 \rightarrow H_8$: via $R_1$ and $R_2$

Forwarding table of $H_1$

<table>
<thead>
<tr>
<th>NetworkNum</th>
<th>NextHop</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Default ($R_1$)</td>
</tr>
<tr>
<td>2</td>
<td>Interface 0</td>
</tr>
<tr>
<td>3</td>
<td>Default ($R_1$)</td>
</tr>
<tr>
<td>4</td>
<td>Default ($R_1$)</td>
</tr>
</tbody>
</table>

Forwarding table of $R_2$

<table>
<thead>
<tr>
<th>NetworkNum</th>
<th>NextHop</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$R_3$</td>
</tr>
<tr>
<td>2</td>
<td>$R_1$</td>
</tr>
<tr>
<td>3</td>
<td>Interface 1</td>
</tr>
<tr>
<td>4</td>
<td>Interface 0</td>
</tr>
</tbody>
</table>

Routers vs. bridges

- **Bridge (+/-)**
  + bridge operation simple, requires less processing
  + transparent (no configuration needed when new nodes added to LAN)
    - restricted topology (forwarding determined by a spanning tree)
    - LANs use a flat addressing space (no hierarchical network structure)
- **Router (+/-)**
  + arbitrary topologies, enables use of efficient routing algorithms for distributing traffic (helps traffic management)
  + hierarchical addressing enables scalability:
    - scalability requires minimization of address info stored in routers
    - routing based on network numbers ⇒ forwarding tables contain info on all networks, **not** all nodes
      - requires IP address configuration
      - packet processing more demanding
- **Summary:** bridges do well in small (~100 hosts) networks while routers are used in large networks (1000s of hosts)
Outline

• Internet architecture
• IP service model
• IP forwarding
• Address translation (ARP)
  • Automatic host configuration (DHCP) and error reporting (ICMP)
  • Virtual Private Networks (VPNs)

Address translation

• Earlier, we skipped the part what to do when router/host notes that it is connected directly to the network where an arriving packet is destined.

• Need to map IP addresses into physical LAN addresses
  – destination host
  – next hop router

• Techniques
  – encode physical LAN address in host part of IP address
    • not scalable
  – table-based (maintain IP address, PHY address pairs)
    • $\Rightarrow$ ARP
ARP details

- ARP (Address Resolution Protocol)
  - utilizes LAN's broadcast capabilities
  - each node maintains table of IP to physical LAN address bindings
  - broadcast request if IP address not in table
  - target machine responds with its physical LAN address

- ARP request contains also source addresses (physical and IP)
  - all "interested" parties can learn the source address

- Node (host/router) actions:
  - table entries timeout in about 10 minutes
  - if node already has an entry for source, refresh timer
  - if node is the target, reply and update table with source info
  - if node not target and does not have entry for the source, ignore source info

- ARP info can be incorporated in the contents of forwarding table

ARP Packet Format

- Request Format
  - HardwareType: type of physical network (e.g., Ethernet)
  - ProtocolType: type of higher layer protocol (e.g., IP)
  - HLen & PLen: length of physical and upper layer addresses
  - Operation: request or response
  - Physical/IP addresses of Source and Target

<table>
<thead>
<tr>
<th>0</th>
<th>8</th>
<th>16</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware type = 1</td>
<td>ProtocolType = 0x0800</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HLen = 48</td>
<td>PLen = 32</td>
<td>Operation</td>
<td></td>
</tr>
<tr>
<td>SourceHardwareAddr (bytes 0 – 3)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SourceHardwareAddr (bytes 4 – 5)</td>
<td>SourceProtocolAddr (bytes 0 – 1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SourceProtocolAddr (bytes 2 – 3)</td>
<td>TargetHardwareAddr (bytes 0 – 1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TargetHardwareAddr (bytes 2 – 5)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TargetProtocolAddr (bytes 0 – 3)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Classical IP over ATM

- Problem: ARP uses broadcast, but
  - ATM is connection oriented (no broadcasting)
- Solution:
  - LANE not useful if nodes spread over large area
  - Classical IP over ATM and ATMARP server
- Classical IP over ATM
  - group nodes of ATM network into several LIS (Logical IP Subnet)
  - nodes in same LIS have same IP network number
  - nodes in same LIS communicate with each other directly using ATM (AAL5)
  - nodes in different LIS communicate via IP router
  - can connect large nof hosts and routers to a big ATM network without assigning addresses from same IP network
  - scalability: ATMARP handles smaller nof hosts

ATMARP

- ATMARP server
  - resolves ATM addresses to IP addresses (like ARP translates ETH to IP)
  - does not rely on broadcast
- Functionality
  - each node in a LIS sets up VC to ATMARP and registers (sends own ⟨ATM, IP⟩ address pair)
  - ARP server builds table of ⟨ATM, IP⟩ address pairs for all registered nodes
  - nodes make queries to ARP server
  - nodes can keep cache of ⟨ATM, IP⟩ address mappings
    - like in traditional ARP
  - VC to a destination can be kept alive as long as needed
- Note! In Classical IP over ATM two nodes in same ATM network cannot communicate directly if they are in different subnets.
Outline

• Internet architecture
• IP service model
• IP forwarding
• Address translation (ARP)
• Automatic host configuration (DHCP) and error reporting (ICMP)
• Virtual Private Networks (VPNs)

Network management and scalability

• Mechanisms in IP that enable heterogeneity and scalability
  – heterogeneity:
    • best effort service model that makes minimal assumptions on underlying network capabilities
    • common packet format, fragmentation used for networks with different MTUs
    • global address space (ARP maps physical addresses to IP)
  – scaling:
    • hierarchical aggregation of routing information (network/host number)
  – above focuses on minimizing network state info in devices

• Important also to consider management complexity as network grows
  – example: configuration of IP addresses via DHCP
Need for automatic configuration

- IP addresses need to be reconfigurable
  - Ethernet addresses hardwired onto the network adapter
  - IP address consists of network and host part
  - hosts can move between networks ⇒ host gets new address in each network
- Need for automated host configuration
  - hosts need other configuration info, e.g., the default router
  - configuration manually impossible (too much work and errors)
  - ⇒ Dynamic Host Configuration Protocol (DHCP)
- DHCP server
  - at least one DHCP server for each administrative domain
  - centralized repository for configuration info
  - two operation modes:
    - administrator chooses host addresses and configures them to DHCP
    - DHCP manages the addresses by allocating addresses dynamically from a pool of available addresses (more sophisticated)

DHCP operation

- Server discovery: host sends DHCPDISCOVER msg to IP broadcast address (255.255.255.255)
- Msg broadcasted only on same network
- If server on same network, host receives its IP address
- If not, msg picked up by DHCP relay agent
- Relay agent knows address of DHCP server, forwards the msg to DHCP server and host receives its IP address
- Use of DHCP relay agent makes it possible to have fewer DHCP servers (relay agent configuration simpler than DHCP server configuration)
DHCP packet format, etc.

- **Packet format**
  - carried on top of UDP
  - based on older protocol BOOTP (unused fields)
  - client puts its hardware address in chaddr
  - DHCP server puts client's IP address in yiaddr
  - other info placed in options (default router, subnet mask, DNS server)

- **Handling dynamic addresses**
  - problem: hosts may not return addresses (host crashes, is turned off, ...)
  - DHCP addresses only "leased" for a period of time
  - if lease is not refreshed, address placed back in pool

- **DHCP improves manageability of network**

Internet Control Message Protocol (ICMP)

- **ICMP used for reporting errors in Internet**

- **Messages**
  - Echo (ping)
  - Redirect (from router to source host if router knows of a better route to packet's destination)
  - Destination unreachable (protocol, port, or host)
  - TTL exceeded (so datagrams don’t cycle forever)
  - Checksum failed
  - Reassembly failed
  - Cannot fragment
Virtual private networks (VPN)

- Problem:
  - group of isolated networks
  - geographically distant from each other
  - need to connect different networks together into a “private” network
  - e.g., company with many branch offices

- Solution:
  - VPN
  - connect individual networks together through a public network

- Technologies
  - leased virtual circuits from an ATM network operator or Frame Relay operator
  - possible with IP, but requires IP tunneling
VPN and IP tunneling

- Problem with IP
  - not possible to connect to Internet via router without the whole Internet also knowing about your network

- Tunneling
  - virtual point-to-point link btw. two nodes separated by arbitrary nof networks
  - created in R1 by providing it with address of R2
  - R1 encapsulates original packet in a new packet addressed to R2
  - packet forwarded normally inside IP network
  - R2 receives packet and strips off packet header and notices payload contains an encapsulated packet addressed to some host inside network 2

- IP tunneling used in
  - VPNs, Mobile IP
  - building logical networks of multicast or QoS enabled routers