

CSCE 515: Computer Network Programming TCP/IP

Wenyuan Xu

Department of Computer Science and
Engineering
University of South Carolina

IP – Network Layer

IP Datagrams

- IP is the network layer
 - packet delivery service (host-to-host).
 - translation between different data-link protocols.

CSCE515 – Computer Network Programming

IP Datagram

1 byte	1 byte	1 byte	1 byte
VER	HL	Service	Total Length
Datagram ID		FLAG	Fragment Offset
TTL	Protocol	Header Checksum	
Source Address			
Destination Address			
Options (if any)			
Data			

CSCE515 – Computer Network Programming

IP Addresses

- IP addresses are not the same as the underlying data-link (MAC) addresses. **WHY?**
- IP is a network layer - it must be capable of providing communication between hosts on different kinds of networks (different data-link implementations).
- The address must include information about what *network* the receiving host is on. This is what makes routing feasible.

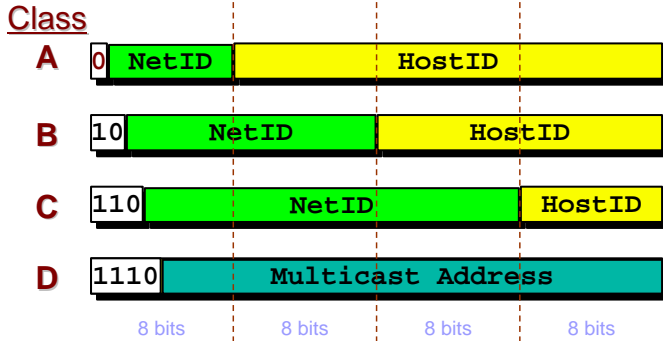
CSCE515 – Computer Network Programming

IP Addresses

- IP addresses are *logical* addresses (not physical)
- 32 bits.
 - IP Addresses are usually shown in *dotted decimal* notation: 1.2.3.4
- Includes a network ID and a host ID.
- Every host must have a unique IP address.
- IP addresses are assigned by a central authority (*American Registry for Internet Numbers* for North America).
 - <http://www.iana.org/ipaddress/ip-addresses.htm>

CSCE515 – Computer Network Programming

The four formats of IP Addresses



CSCE515 – Computer Network Programming

Class A

- 128 possible network IDs
- over 16M host IDs per network ID

Class B

- 16K possible network IDs
- 64K host IDs per network ID

Class C

- over 2 million possible network IDs
- about 256 host IDs per network ID

CSCE515 – Computer Network Programming

Network and Host IDs

- A Network ID is assigned to an organization by a global authority.
- Host IDs are assigned locally by a system administrator.
- Both the Network ID and the Host ID are used for routing.

CSCE515 – Computer Network Programming

IP Addresses

- IP Addresses are usually shown in *dotted decimal* notation:

1.2.3.4 00000001 00000010 00000011 00000100

- cse.sc.edu is 129.252.138.8

10000001 11111100 10001010 00001000

↖ CSE has a class B network

CSCE515 – Computer Network Programming

Host and Network Addresses

- A single network interface is assigned a single IP address called the *host* address.
- A host may have multiple interfaces, and therefore multiple *host* addresses.
- Hosts that share a network all have the same IP *network* address (the network ID).

CSCE515 – Computer Network Programming

Special IP addresses

- An IP broadcast addresses has a host ID of all 1s.
- An IP address that has a host ID of all 0s is called a *network address* and refers to an entire network.
- localhost: 127.0.0.1

CSCE515 – Computer Network Programming

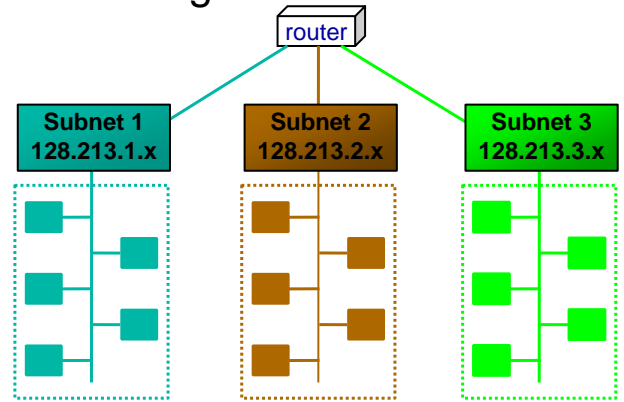
Subnet Addresses

- An organization can subdivide its host address space into groups called subnets.
- The subnet ID is generally used to group hosts based on the physical network topology.



CSCE515 – Computer Network Programming

Subnetting



CSCE515 – Computer Network Programming

Subnetting

- Subnets can simplify routing.
- IP subnet broadcasts have a hostID of all 1s.
- It is possible to have a single wire network with multiple subnets?*



CSCE515 – Computer Network Programming

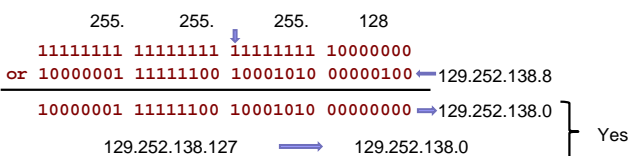
Subnet Mask

- How do we know how many bits are allocated for subnet ID, how many bits are allocated for host ID?
- Example: subnet mask: 255.255.255.0
- Question: what is the subnet id for 129.252.138.8 if subnet is:
 - 255.255.255.0
 - 255.255.255.192
 - 255.255.127.0

CSCE515 – Computer Network Programming

Subnet mask

- Question: if the subnet mask is 255.255.255.128, are 129.252.138.8 and 129.252.138.127 on the same subnet?
 - Calculate the subnet ID
 - If the subnet IDs are the same => Yes
 - Otherwise => No

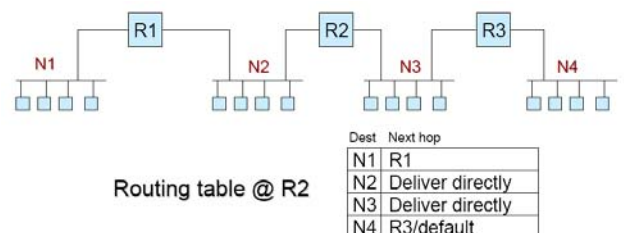


2007

CSCE515 – Computer Network Programming

IP Routing

- Q: How do you get a packet from one network to another?



CSCE515 – Computer Network Programming

Routing table

- Destination IP address. Either host address or a network address
- IP address of the next hop Router
- Flags
- Network interface

/sbin/route

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
129.252.130.0	*	255.255.255.0	U	0	0	0	eth1
loopback	*	255.0.0.0	U	0	0	0	lo
default	SWG130.cse.sc.edu	0.0.0.0	UG	1	0	0	eth1

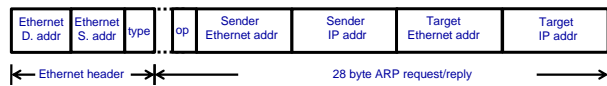
Mapping IP Addresses to Hardware Addresses

- IP Addresses are not recognized by hardware.
- If we know the IP address of a host, how do we find out the hardware address ?
- The process of finding the hardware address of a host given the IP address is called

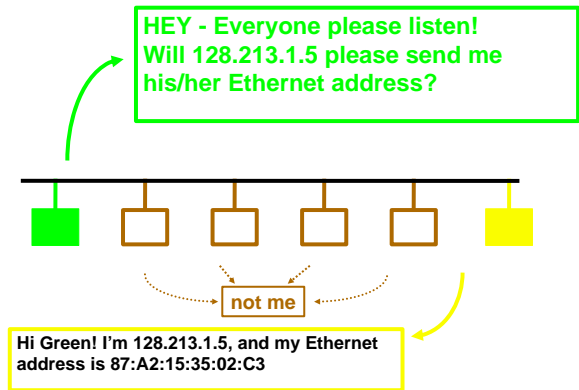
Address Resolution

ARP

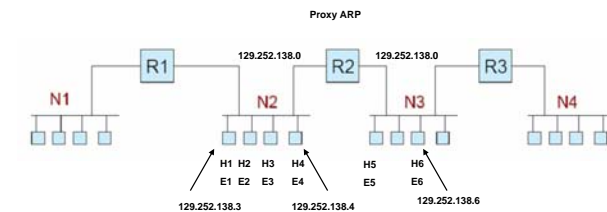
- The *Address Resolution Protocol* is used by a sending host when it knows the IP address of the destination but needs the Ethernet (or whatever) address.
- ARP is a broadcast protocol - every host on the network receives the request.
- Each host checks the request against it's IP address - the right one responds.



ARP conversation

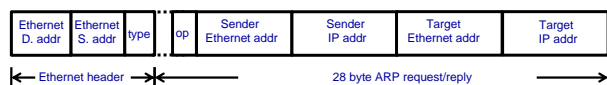


Example– Proxy ARP

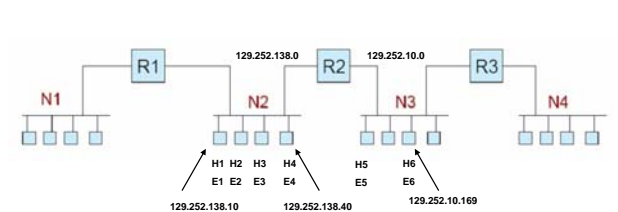


Routing table @ R2

Dest	Next hop
N1	R1
N2	Deliver directly
N3	Deliver directly
N4	R3/default

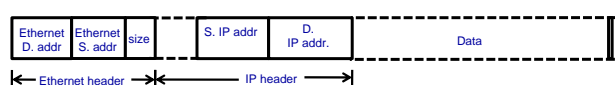


Example– routing



Routing table @ R2

Dest	Next hop
N1	R1
N2	Deliver directly
N3	Deliver directly
N4	R3/default



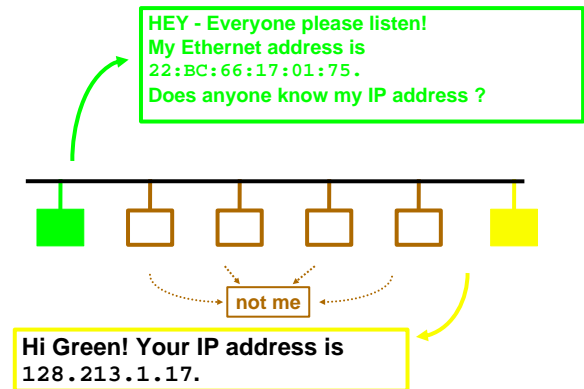
Reverse Address Resolution

- The process of finding out the IP address of a host given a hardware address is called

Reverse Address Resolution

- Reverse address resolution is needed by diskless workstations when booting (which used to be quite common).

RARP conversation



Services provided by IP

- **Connectionless** Delivery (each datagram is treated individually).
- **Unreliable** (delivery is not guaranteed).
- Fragmentation / Reassembly (based on hardware MTU).
- Routing.
- Error detection.

IP Datagram

1 byte		1 byte		1 byte		1 byte	
VERS	HL	Service		Fragment Length			
Datagram ID				FLAG	Fragment Offset		
TTL		Protocol		Header Checksum			
Source Address							
Destination Address							
Options (if any)							
Data							

IP Datagram Fragmentation

- Each fragment (packet) has the same structure as the IP datagram.
- IP specifies that datagram reassembly is done only at the destination (not on a hop-by-hop basis).
- If any of the fragments are lost - the entire datagram is discarded (and an ICMP message is sent to the sender).

IP Flow Control & Error Detection

- If packets arrive too fast - the receiver discards excessive packets and sends an ICMP message to the sender (SOURCE QUENCH).
- If an error is found (header checksum problem) the packet is discarded and an ICMP message is sent to the sender.

ICMP *Internet Control Message Protocol*

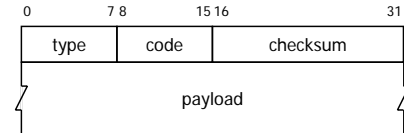
- ICMP is a protocol used for exchanging control messages.
- Two main categories
 - *Query message*
 - *Error message*
- Usage of an ICMP message is determined by *type* and *code* fields
- ICMP uses IP to deliver messages.
- ICMP messages are usually generated and processed by the IP software, not the user process.



20 bytes

CSCE515 – Computer Network Programming

ICMP Message Format



CSCE515 – Computer Network Programming

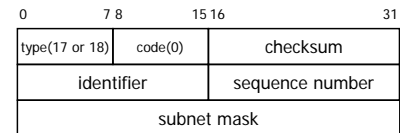
ICMP Message Types

- Echo Request
- Echo Response
- Destination Unreachable
- Redirect
- Time Exceeded
- there are more ...

CSCE515 – Computer Network Programming

ICMP Address Mask Request and Reply

- intended for a diskless system to obtain its subnet mask.
- Id and seq can be any values, and these values are returned in the reply.
 - Match replies with request



CSCE515 – Computer Network Programming

Transportation Layer

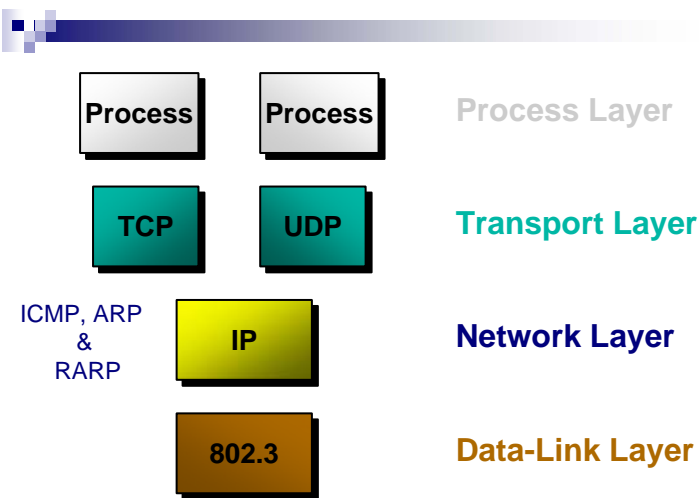
Transport Layer & TCP/IP

Q: We know that IP is the network layer - so TCP must be the transport layer, right ?

A: No... well, almost.

TCP is only part of the TCP/IP transport layer - the other part is UDP (User Datagram Protocol).

CSCE515 – Computer Network Programming



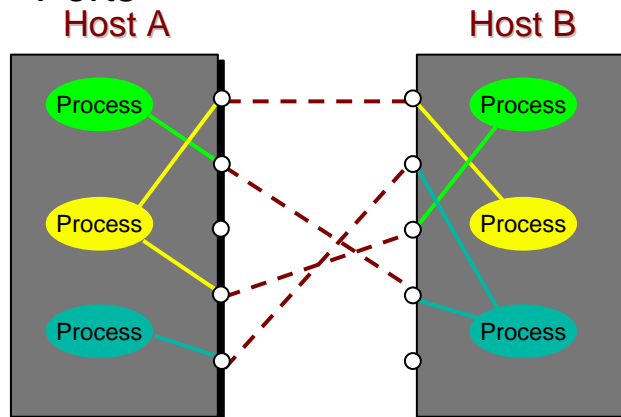
UDP User Datagram Protocol

- UDP is a transport protocol
 - communication between processes
- UDP uses IP to deliver datagrams to the right host.
- UDP uses *ports* to provide communication services to individual processes.

Ports

- TCP/IP uses an abstract destination point called a protocol port.
- Ports are identified by a positive integer.
- Operating systems provide some mechanism that processes use to specify a port.

Ports



UDP

The term *datagram* is also used to describe the unit of transfer of UDP!

- Datagram Delivery
- Connectionless
- Unreliable
- Minimal

UDP Datagram Format

Source Port	Destination Port
Length	Checksum
Data	

TCP

Transmission Control Protocol

- TCP is an alternative transport layer protocol supported by TCP/IP.
- TCP provides:
 - Connection-oriented
 - Reliable
 - Full-duplex
 - Byte-Stream

Wow!

Connection-Oriented

- *Connection oriented* means that a virtual connection is established before any user data is transferred.
- If the connection cannot be established - the user program is notified (finds out).
- If the connection is ever interrupted - the user program(s) is finds out there is a problem.

CSCE515 – Computer Network Programming

Reliable

Reliable does not mean that things don't go wrong, it means that we find out when things go wrong.

- *Reliable* means that every transmission of data is acknowledged by the receiver.
- If the sender does not receive acknowledgement within a specified amount of time, the sender retransmits the data.

CSCE515 – Computer Network Programming

Byte Stream

- *Stream* means that the connection is treated as a stream of bytes.
- The user application does not need to package data in individual datagrams (as with UDP).

Somebody needs to do this since IP is delivering all the data, it's just that the application layer doesn't need to do this!

CSCE515 – Computer Network Programming

Buffering

- TCP is responsible for buffering data and determining when it is time to send a datagram.
- It is possible for an application to tell TCP to send the data it has buffered without waiting for a buffer to fill up.

CSCE515 – Computer Network Programming

Full Duplex

- TCP provides transfer in both directions (over a single virtual connection).
- To the application program these appear as 2 unrelated data streams, although TCP can piggyback control and data communication by providing control information (such as an ACK) along with user data.

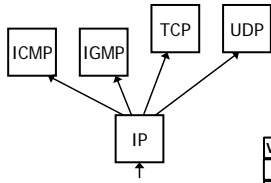
CSCE515 – Computer Network Programming

TCP Ports

- Interprocess communication via TCP is achieved with the use of ports (just like UDP).
- UDP ports have no relation to TCP ports (different name spaces).

CSCE515 – Computer Network Programming

IP Demultiplexing



VERS	HL	Service	Fragment Length
Datagram ID		FLAG	Fragment Offset
TTL	Protocol	Header Checksum	
Source Address			
Destination Address			
Options (if any)			
Data			

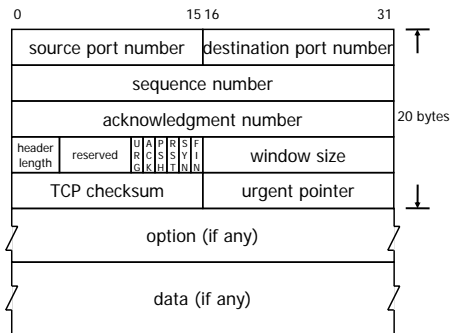
CSCE515 – Computer Network Programming

TCP Segments

- The chunk of data that TCP asks IP to deliver is called a *TCP segment*.
- Each segment contains:
 - data bytes from the byte stream
 - control information that identifies the data bytes

CSCE515 – Computer Network Programming

TCP Segment Format



CSCE515 – Computer Network Programming

Addressing in TCP/IP

- Each TCP/IP address includes:
 - Internet Address
 - Protocol (UDP or TCP)
 - Port Number

NOTE: TCP/IP is a *protocol suite* that includes IP, TCP and UDP.

CSCE515 – Computer Network Programming

TCP vs. UDP

Q: Which protocol is better ?

A: It depends on the application.

TCP provides a connection-oriented, reliable, byte stream service (lots of overhead).

UDP offers minimal datagram delivery service (as little overhead as possible).

CSCE515 – Computer Network Programming

TCP/IP Summary

- IP: network layer protocol
 - unreliable datagram delivery between hosts.
- UDP: transport layer protocol
 - unreliable datagram delivery between processes.
- TCP: transport layer protocol
 - reliable, byte-stream delivery between processes.

CSCE515 – Computer Network Programming



Hmmmmm. TCP or UDP ?

- Electronic commerce?
- Video server?
- File transfer?
- Email ?
- Chat groups?
- Robotic surgery controlled remotely over a network?



Assignment & Next time

- Reading:
 - TI 3,4,5,6,17 **
 - [TCP/IP FAQ](#)
- Next Lecture:
 - Sockets Programming API