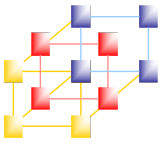
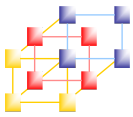
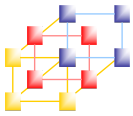


Unit 9
Multicasting



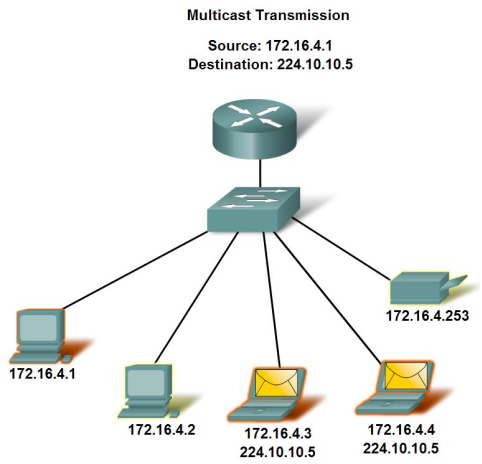
Multicast Basic Concepts (1/4)

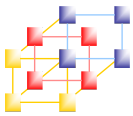
- Unicast
  - Point to point communication
- Broadcast
  - Packets are sent to all
  - Routers limit broadcasts to the local network or subnet, preventing broadcasts form reaching the Internet at large
- Multicast
  - Send packets to many different hosts, but not to everyone.



Multicast Basic Concepts (2/4)

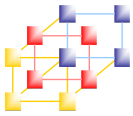
- 一對多(one-to-many)的通訊方式
- 通訊程式送出的訊息可以送往指定的一群接收者
- IP的群播協定 (IP multicast protocol) 支援網際網路上的群播，屬於網路層的協定
- 需要作業系統的支援





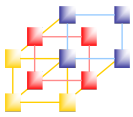
Multicast Basic Concepts (3/4)

- Thinking
  - A real-time video stream goes to 6 million Internet user
  - There is no reason to send a video stream to hosts that are not interested in it
- Multicast: think as a group
  - Like a public meeting
  - People can come and go as they please
  - Send messages to the group and all the people in the group will get the messages
  - People not in the group will not be affected



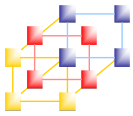
Multicast Basic Concepts (4/4)

- Most of the work is done by routers and should be transparent to application programmers.
- An application simply sends datagram packets multicast IP address. The router makes sure that the packets are delivered to all hosts in the multicast group.
- The problem
  - multicast routers are not yet ubiquitous



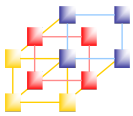
Multicast Address and Groups (1/3)

- A multicast address is the address of a group of hosts called multicast group
  - IPv4 CIDR group: 224.0.0.0/4 (224.0.0.0 to 239.255.255.255)
    - All addresses have the leading four binary digits 1110
  - IPv6 CIDR group: ff00::/8
- Any data sent to the multicast address is relayed to all the members of the group
- Like any IP address, a multicast address can have a hostname
  - ntp.mcast.net = 224.0.1.1 (network time protocol)



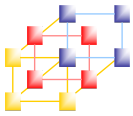
Multicast Address and Groups (2/3)

- Multicast groups can be either permanent or transient
  - Permanent groups have assigned address that remain constant
    - Example: 224.0.0.1 or 224.0.0.2
    - The complete list is available from iana.org
  - Most multicast groups are transient and exist only as long as they have members.
    - Create a new multicast group address from 225.0.0.0 to 238.255.255.255



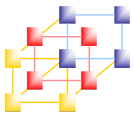
Multicast Address and Groups (3/3)

- Special purposes multicast group
  - all-systems.mcast.net (224.0.0.1) is a multicast group that includes all systems that support multicasting on local subnet
    - This group is commonly used for local testing
  - Also for local testing experiment.mcast.net (224.0.1.20)
  - (224.0.0.0~ 224.0.0.255) are reserved for routing protocols (gateway discovery …)
  - Multicast routers never forward datagrams with destinations in 224.0.0.0~ 224.0.0.255
  - IANA is responsible for handing out permanent multicast addresses



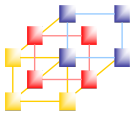
Client and Server

- When a host wants to send data to a multicast group, it puts that data in multicast datagrams (UDP datagrams with an IP address in class D)
- Most multicast data is either audio or video or both. (Small data lost is fine.)
- Multicast data is sent via UDP



Time-To-Live (TTL) (1/2)

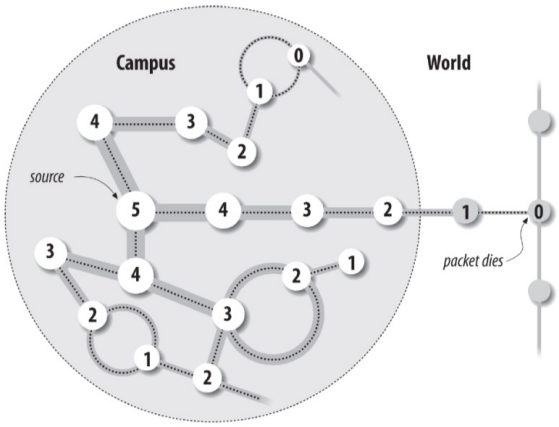
- Routers and hosts must decrement the TIME TO LIVE field by one and remove the datagram from the internet when TTL reaches zero.
  - The TTL acts a "hop limit"
- Two uses
  - It guarantees that datagrams cannot travel around an internet forever.
  - Source might want to intentionally limit the journey of the packet.

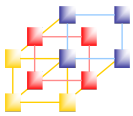


Time-To-Live (TTL) (2/2)

Time-To-Live (TTL) of IP: maximum number of routers that the datagram is allowed

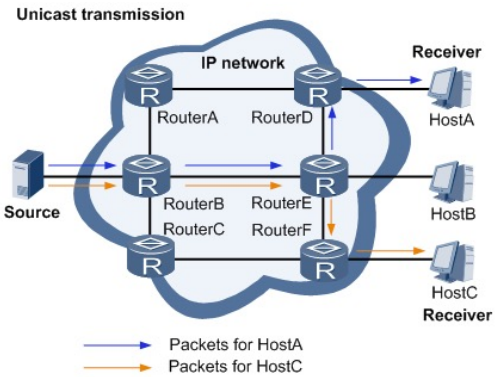
Destinations	TTL value
The local host	0
The local subnet	1
The local campus—that is, the same side of the nearest Internet router—but on possibly different LANs	16
High-bandwidth sites in the same country, generally those fairly close to the backbone	32
All sites in the same country	48
All sites on the same continent	64
High-bandwidth sites worldwide	128
All sites worldwide	255

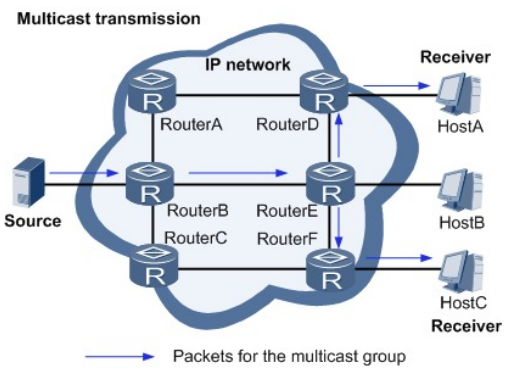


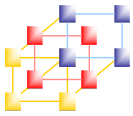


Router and Routing (1/2)

- With multicasting
  - A multicast socket sends one stream of data over the Internet to the clients' router.
  - The router duplicates the stream and sends it to each of the clients
- Without multicasting
  - The server sends separate but identical stream of data to the router
  - The router sends each of the stream to a client.

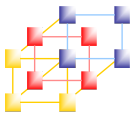






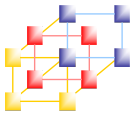
Router and Routing (2/2)

- Note that real-world routes can be much more complex, involving multiple hierarchies of redundant routers
- Goal of multicast sockets
  - No matter how complex the network, the same data should never be sent more than once over any given network
  - Programmers don't need to worry about routing issues.
- To send and receive multicast data beyond the local subnet, you need a multicast router
  - ping all-routers.mcast.net



Multicast Socket (1/4)

- Communication with a multicast group
  - Join a multicast group
  - Send data to the members of the group
  - Receive data from the group
  - Leave the multicast group



Multicast Socket (2/4)

- Receiver
  - Create a UDP socket

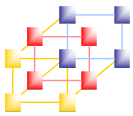
```
recvSocket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
recvSocket.bind(('', PORT))
```

  - Join Multicast group
  - Use setsockopt() to change the IP_ADD_MEMBERSHIP option

```
group = socket.inet_aton(group_addr)
mreq = struct.pack('4sL', group, socket.INADDR_ANY)
recvSocket.setsockopt(socket.IPPROTO_IP, IP_ADD_MEMBERSHIP, mreq)
```

  - Receive message

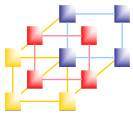
```
data, (rip, rport) = recvSocket.recvfrom(BUFF_SIZE)
```

Multicast Socket (3/4)

- Leave Multicast group
- Use setsockopt() to change the IP_DROP_MEMBERSHIP option

```
group = socket.inet_aton(group_addr)
mreq = struct.pack('4sL', group, socket.INADDR_ANY)
recvSocket.setsockopt(socket.IPPROTO_IP, IP_DROP_MEMBERSHIP, mreq)
```



Multicast Socket (4/4)

- Sender
  - Create UDP socket

```
sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
```

  - Set timeout

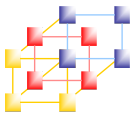
```
sock.settimeout(0.2)
```

  - Config TTL

```
ttl = struct.pack('b', 1)
sock.setsockopt(socket.IPPROTO_IP, socket.IP_MULTICAST_TTL, ttl)
```

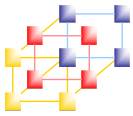
  - Send message

```
sock.sendto(message.encode('utf-8'), (group_addr, port))
```

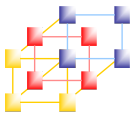
IP群播的原理

- IP群播位址的範圍在224.0.0.0到239.255.255.255
- 群播的範圍
  - time-to-live scoping
  - administrative scoping
- 群播的路由(multicast routing)
  - IGMP (Internet Group Management Protocol)



IP群播路由的組成

- 一種是由所謂的邊緣主機(edge hosts)向鄰接的路由器請求加入或離開群播群組
  - 使用標準化的IGMP(Internet Group Management Protocol)
- 另外一種是處理路由器之間的群播封包
  - 由網路管理者選用非標準化的協定



群播的埠號(port number)

- 由於TCP不適合用在群播中，所以群播一般都使用UDP協定
- 一般不同的群播應用會指定不同的群播位址，所以不需要再使用port number來做所收到的封包的轉送依據，因為從群播位址就知道該送給那個群播應用
- 群播應用還是可以用port number來區隔不同性質的資料傳送