CSCI-1680
Network Layer:
More

Chen Avin

Based partly on lecture notes by David Mazières, Phil Levis, John Jannotti, Peterson & Davie, Rodrigo Fonseca

and “Computer Networking: A Top Down Approach” - 6th edition

Administrivia

- **Homework 2 is due Tuesday**
  - So we can post solutions before the midterm!
- **Exam on Thursday**
  - All content up to today (including!)
  - Questions similar to the homework
  - Book has some exercises, samples on the course web page

Today: IP Wrap-up

- **BGP - extra**
- **IP Service models**
  - Unicast, Broadcast, Anycast, Multicast
- **IPv6**
  - Tunnels

BGP – cont.

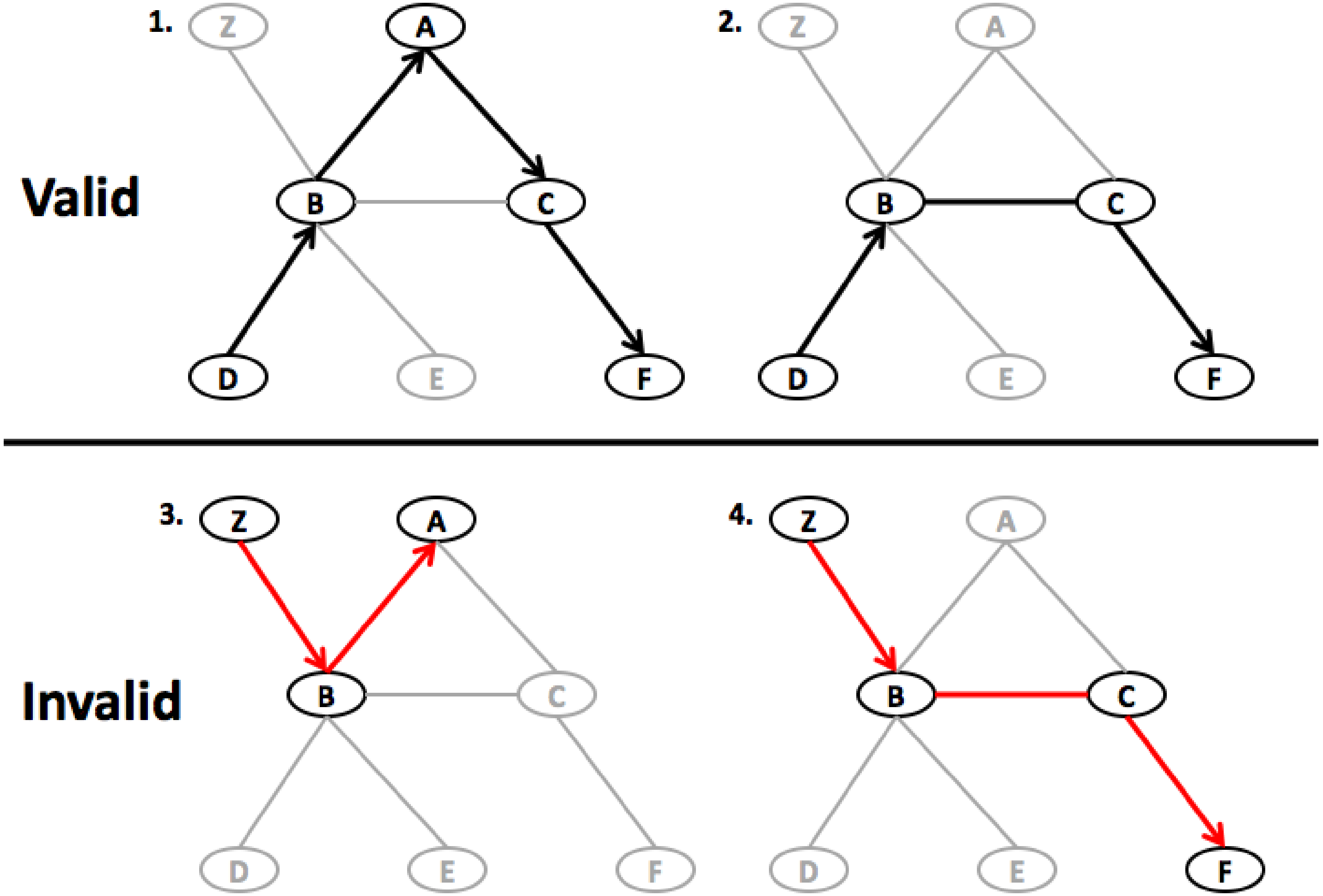
Structure of ASs

- **3 Types of relationships (Customer, Provider, Peer)**
  - Customer-Provider: customer AS pays provider AS for access to rest of Internet: provider provides transit service
    - End customers pay ISPs, and ISPs in lower “tiers” pay ISPs in higher tiers
  - Peers: ASs that allow each other transit service
    - ISPs on same tier, usually involvesno fees
- **Customer-Backup Provider**: Provider if primary provider fails. May be peers otherwise

AS BGP Policies

- AS Policy for its customers - an AS gives its customers transit services toward all of its neighboring ASes.
- AS Policy for its providers - an AS gives its providers transit services only toward its customers.
- AS Policy for its peers - an AS gives its peers transit services only toward its customers.
- “Valley free” paths.

“Valley free”

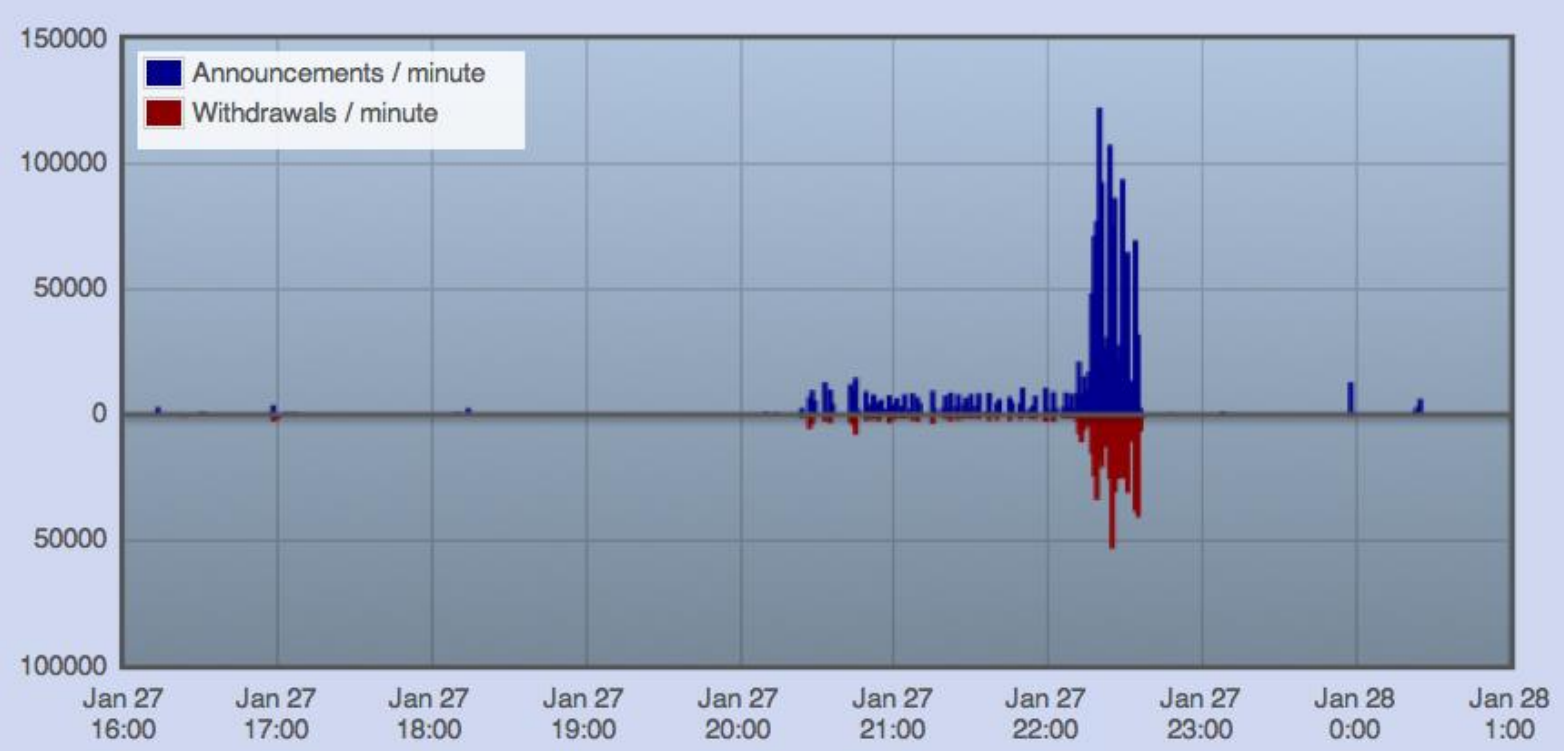


Peering Drama

- **Cogent vs. Level3 were peers**
- **In 2003, Level3 decided to start charging Cogent**
- **Cogent said no**
- **Internet partition: Cogent’s customers couldn’t get to Level3’s customers and vice-versa**
  - Other ISPs were affected as well
- **Took 3 weeks to reach an undisclosed agreement**

“Shutting off” the Internet

- **Starting from Jan 27th, 2011, Egypt was disconnected from the Internet**
  - 2769/2903 networks withdrawn from BGP (95%)!



Static view on BGP activity for prefixes originating from Egyptian organisations between 27 Jan 16:00 UTC and 28 Jan 01:00 UTC

Source: RIPEStat - http://stat.ripe.net/egypt/

Some BGP Challenges

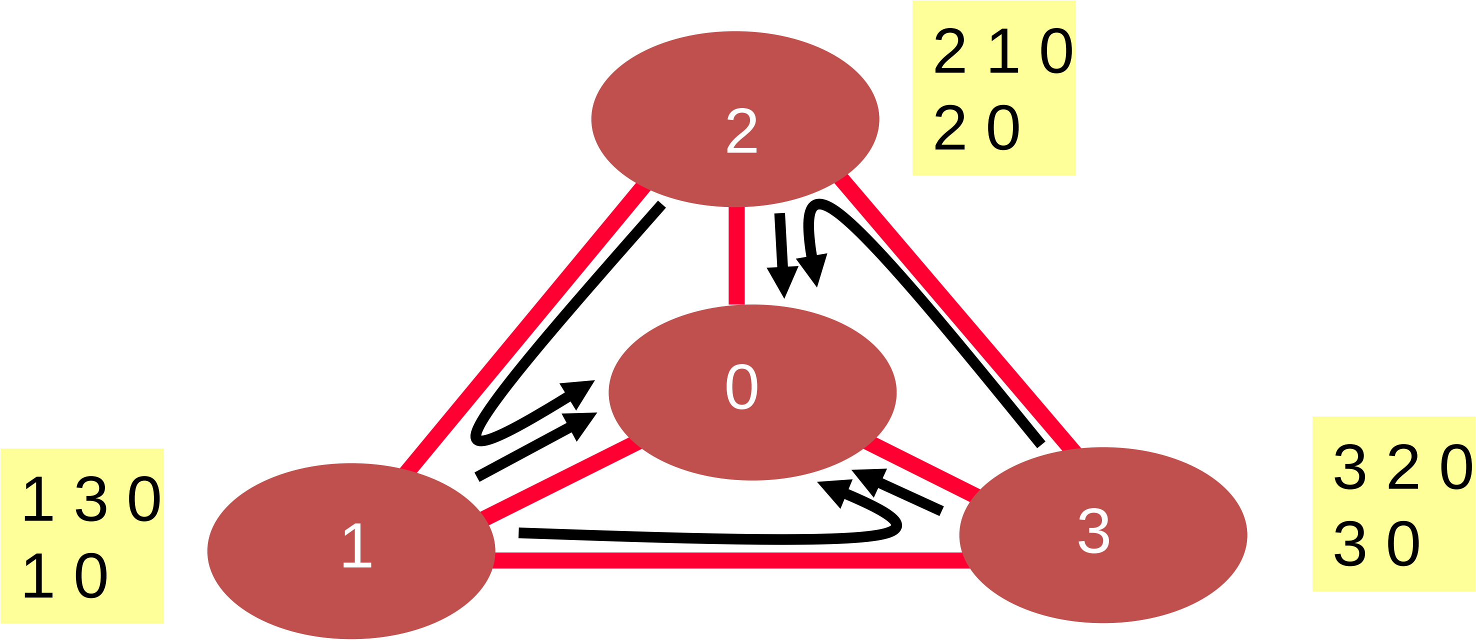
- Convergence
- Scaling (route reflectors)
- Security
- Traffic engineering

Convergence

- **Given a change, how long until the network re-stabilizes?**
  - Depends on change: sometimes never
  - Open research problem: “tweak and pray”
  - Distributed setting is challenging
- **Some reasons for change**
  - Topology changes
  - BGP session failures
  - Changes in policy
  - Conflicts between policies can cause oscillation

Unstable Configurations

- **Due to policy conflicts (Dispute Wheel)**



2 1 0
2 0

1 3 0
1 0

3 2 0
3 0

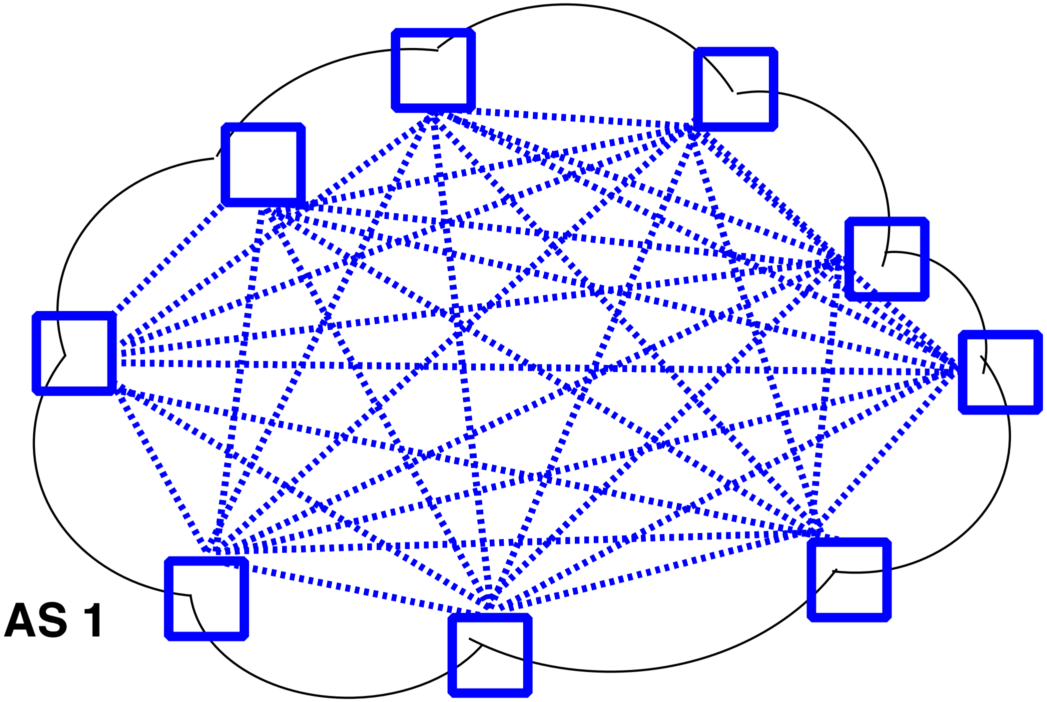
Avoiding BGP Instabilities

- **Detecting conflicting policies**
  - – Centralized: NP-Complete problem!
  - – Distributed: open research problem
  - – Requires too much cooperation
- **Detecting oscillations**
  - – Monitoring for repetitive BGP messages
- **Restricted routing policies and topologies**
  - – Some topologies / policies proven to be safe*

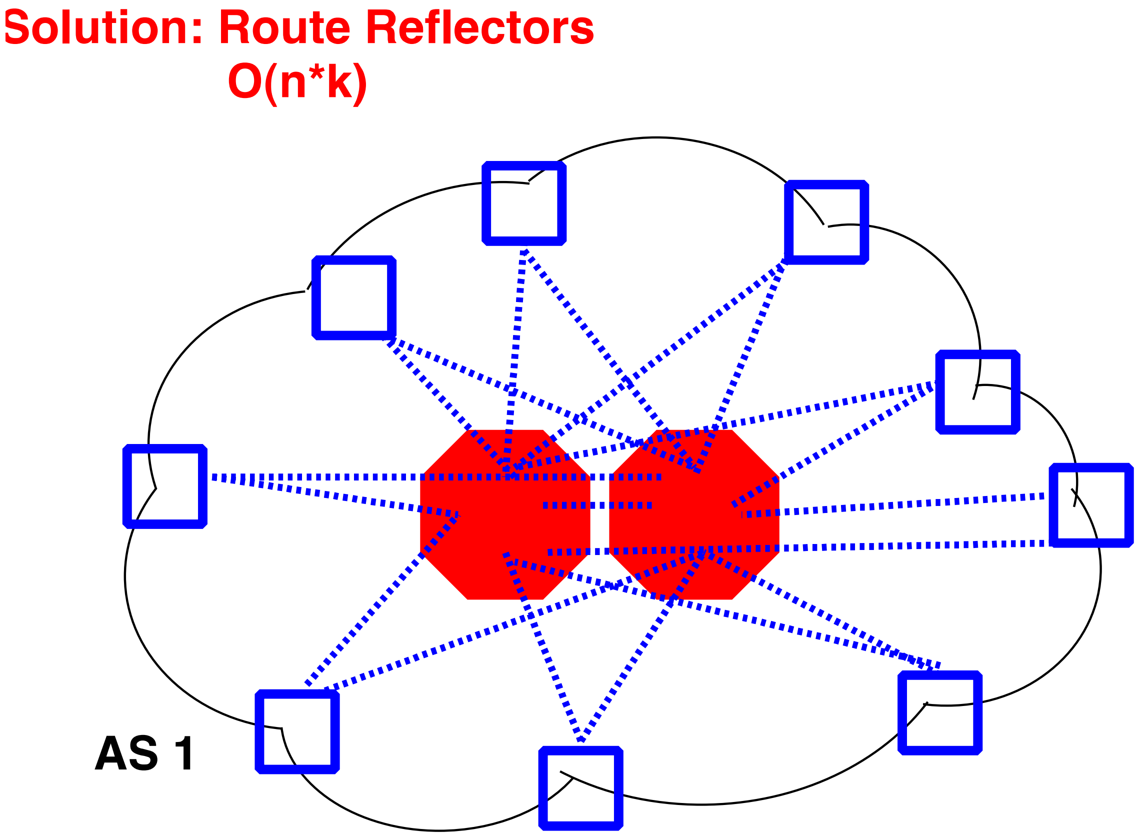
* Gao & Rexford, “Stable Internet Routing without Global Coordination”, IEEE/ACM ToN, 2001

Scaling iBGP: route reflectors

iBGP Mesh == O(n^2) mess



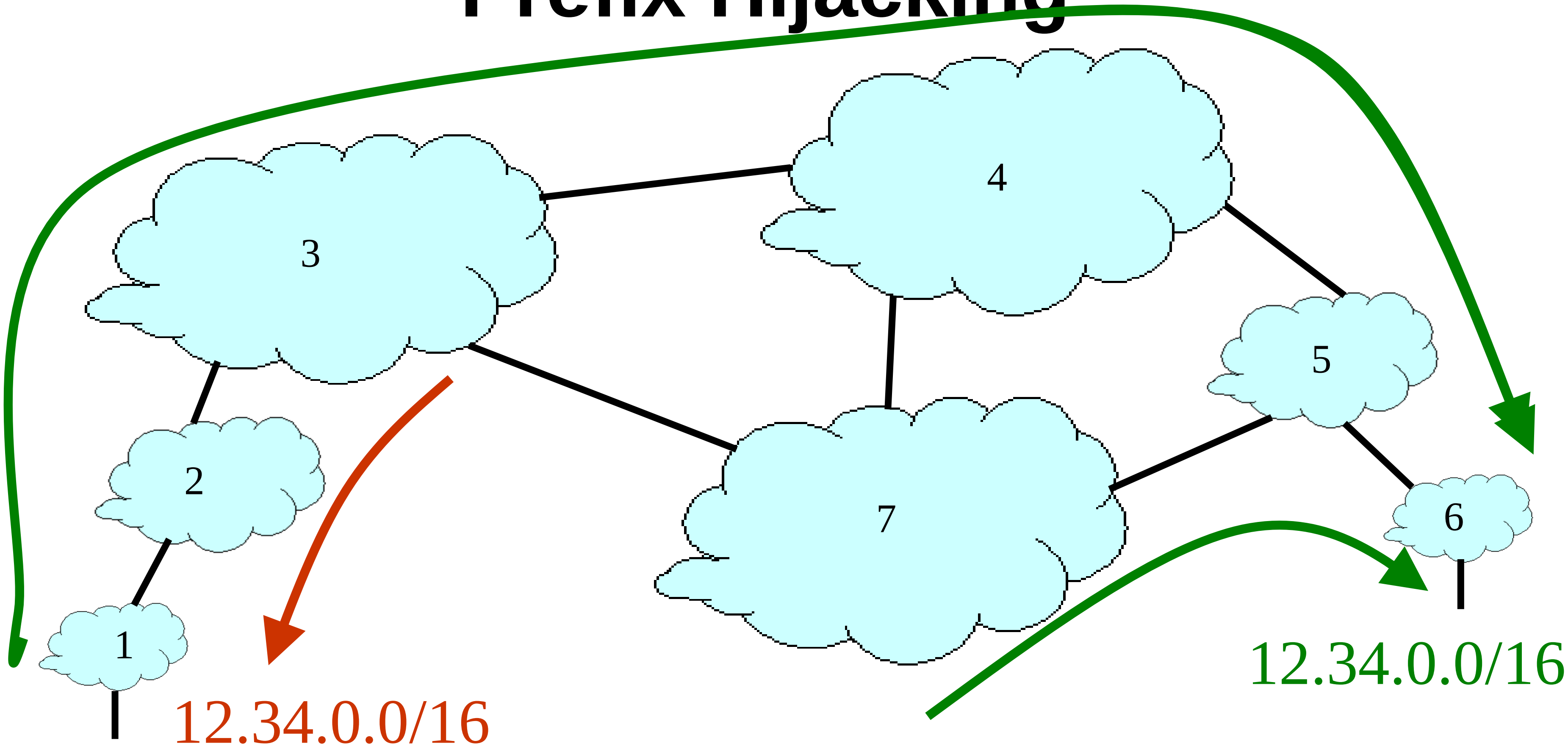
Scaling iBGP: route reflectors



BGP Security Goals

- **Confidential message exchange between neighbors**
- **Validity of routing information**
  - Origin, Path, Policy
- **Correspondence to the data path**

Prefix Hijacking



- **Consequences for the affected ASes**
  - Blackhole: data traffic is discarded
  - Snooping: data traffic is inspected, and then redirected
  - Impersonation: data traffic is sent to bogus destinations

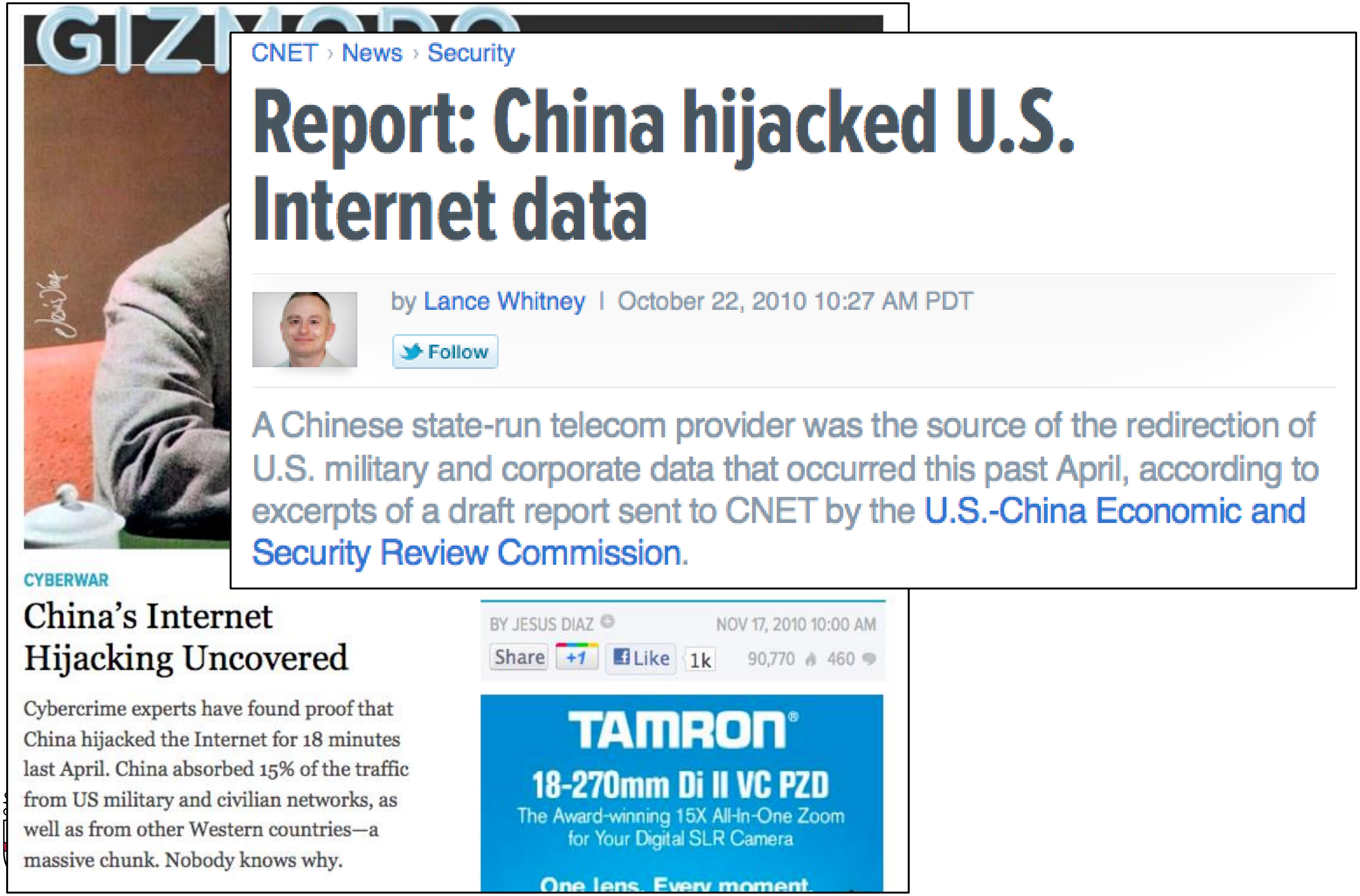
Hijacking is Hard to Debug

- **Real origin AS doesn’t see the problem**
  - Picks its own route
  - Might not even learn the bogus route
- **May not cause loss of connectivity**
  - E.g., if the bogus AS snoops and redirects
  - … may only cause performance degradation
- **Or, loss of connectivity is isolated**
  - E.g., only for sources in parts of the Internet
- **Diagnosing prefix hijacking**
  - Analyzing updates from many vantage points
  - Launching traceroute from many vantage points

Pakistan Youtube incident

- Youtube’s has prefix 208.65.152.0/22
- Pakistan’s government order Youtube blocked
- Pakistan Telecom (AS 17557) announces 208.65.153.0/24 in the wrong direction (outwards!)
- Longest prefix match caused worldwide outage
- http://www.youtube.com/watch?v=IzLPKuAOe50

News



CNET › News › Security

Report: China hijacked U.S. Internet data

by Lance Whitney | October 22, 2010 10:27 AM PDT

Follow

A Chinese state-run telecom provider was the source of the redirection of U.S. military and corporate data that occurred this past April, according to excerpts of a draft report sent to CNET by the U.S.-China Economic and Security Review Commission.

GIZMODO

CYBERWAR

China's Internet Hijacking Uncovered

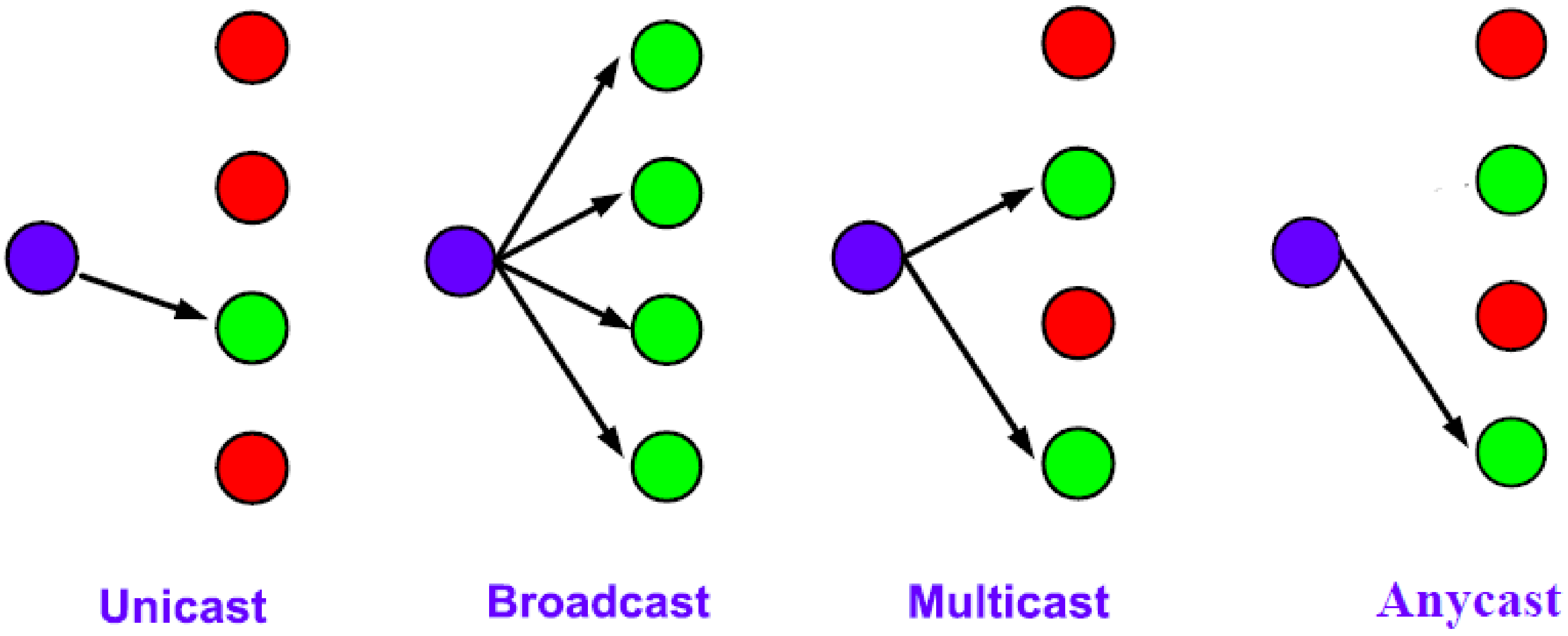
BY JESUS DIAZ NOV 17, 2010 10:00 AM

Share +1 Like 1k 90,770 460

Cybercrime experts have found proof that China hijacked the Internet for 18 minutes last April. China absorbed 15% of the traffic from US military and civilian networks, as well as from other Western countries—a massive chunk. Nobody knows why.

IP Service models

IP Routing



Unicast

Broadcast

Multicast

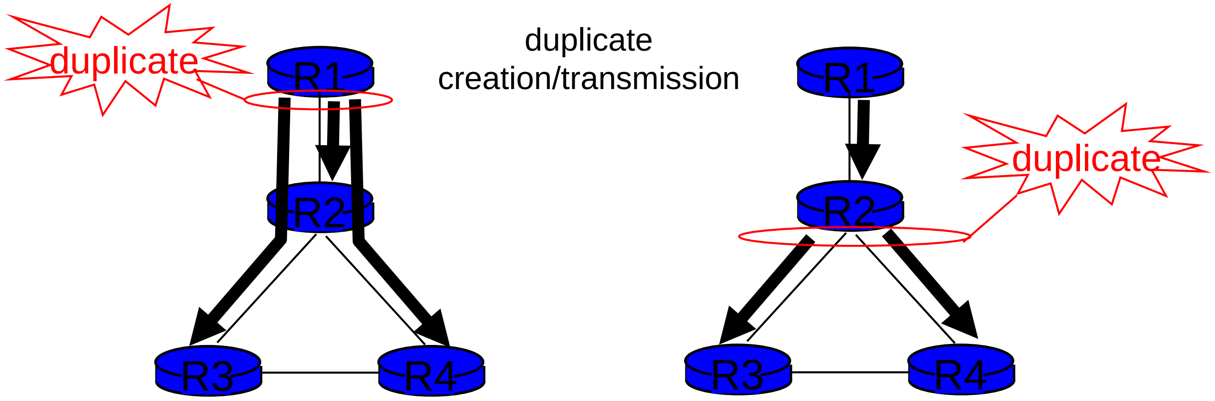
Anycast

Multicast

- **Send messages to many nodes: "one to many"**
- **Why do that?**
  - Snowcast, Internet Radio, IPTV
  - Stock quote information
  - Multi-way chat / video conferencing
  - Multi-player games
- **What's wrong with sending data to each recipient?**
  - Link stress
  - Have to know address of all destinations

Broadcast routing

- deliver packets from source to all other nodes
- source duplication is inefficient:



source duplication

in-network duplication

- source duplication: how does source determine recipient addresses?

Multicast Service Model

- Receivers join a multicast group G
- Senders send packets to address G
- Network routes and delivers packets to all members of G
- Multicast addresses: class D (start 1110)

  224.x.x.x to 229.x.x.x

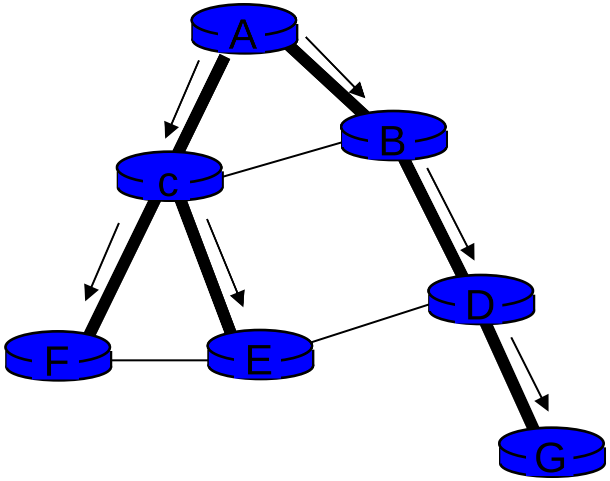
  - 28 bits left for group address

In-network duplication

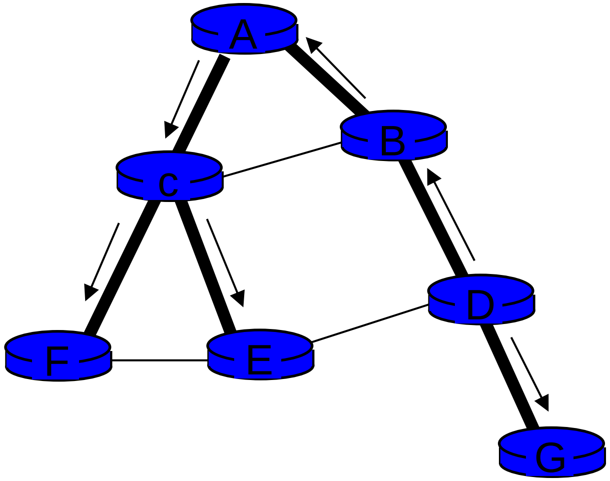
- ***flooding:*** **when node receives broadcast packet, sends copy to all neighbors**
  - problems: cycles & broadcast storm
- ***controlled flooding:*** **node only broadcasts pkt if it hasn't broadcast same packet before**
  - node keeps track of packet ids already broadacsted
  - or reverse path forwarding (RPF): only forward packet if it arrived on shortest path between node and source
- ***spanning tree:***
  - no redundant packets received by any node

Spanning tree

- first construct a spanning tree
- nodes then forward/make copies only along spanning tree



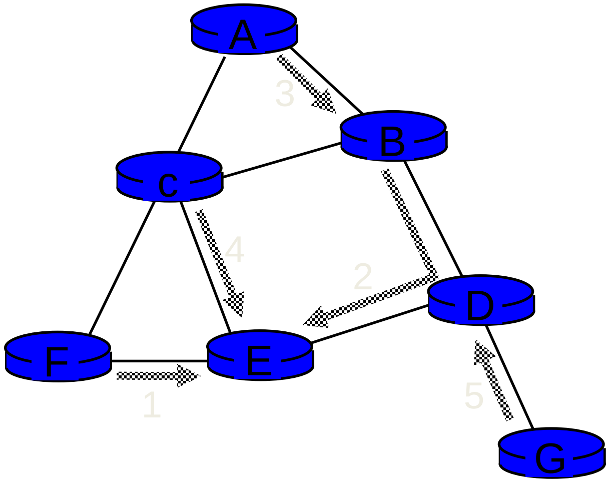
(a) broadcast initiated at A



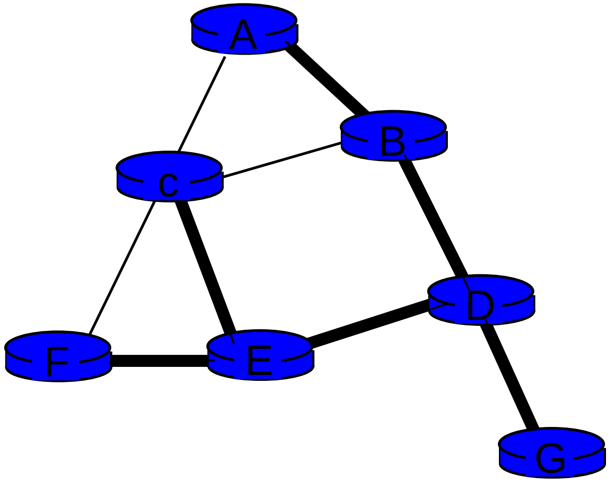
(b) broadcast initiated at D

Spanning tree: creation

- **center node**
- **each node sends unicast join message to center node**
  - message forwarded until it arrives at a node already belonging to spanning tree



(a) stepwise construction of spanning tree (center: E)

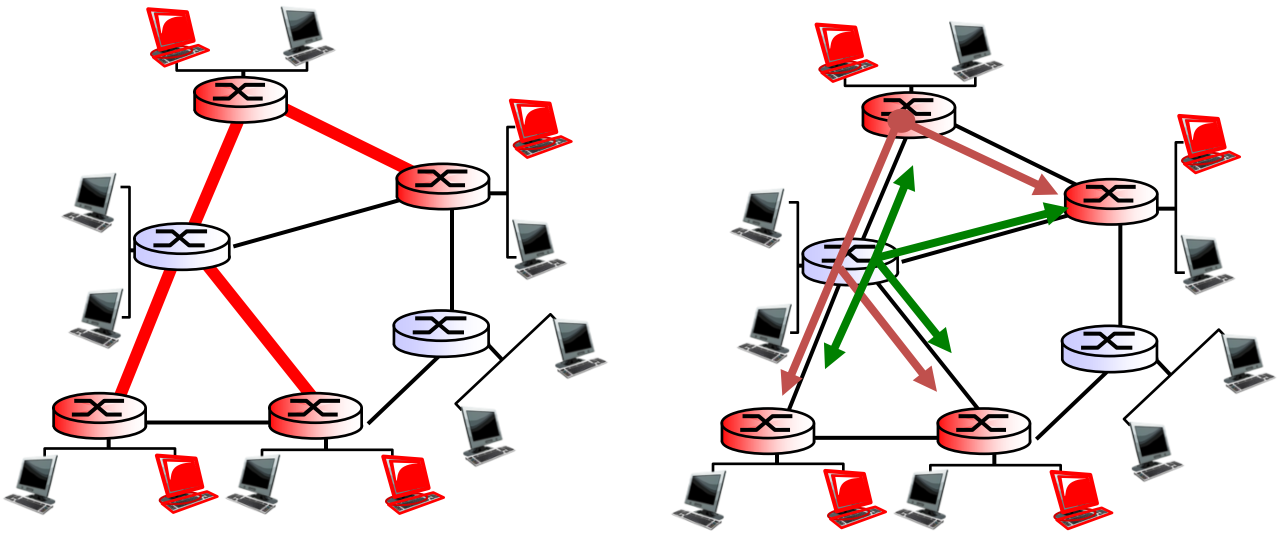


(b) constructed spanning tree

Multicast routing: problem statement

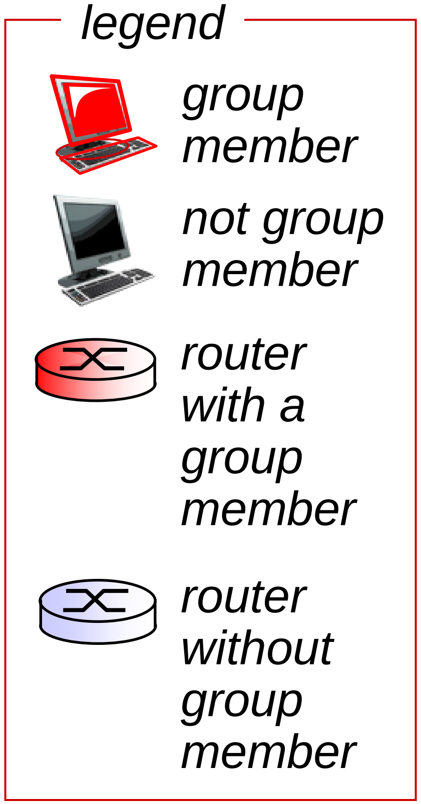
goal: find a tree (or trees) connecting routers having local mcast group members

- *tree:* not all paths between routers used
- *shared-tree:* same tree used by all group members
- *source-based:* different tree from each sender to rcvrs



shared tree

source-based trees



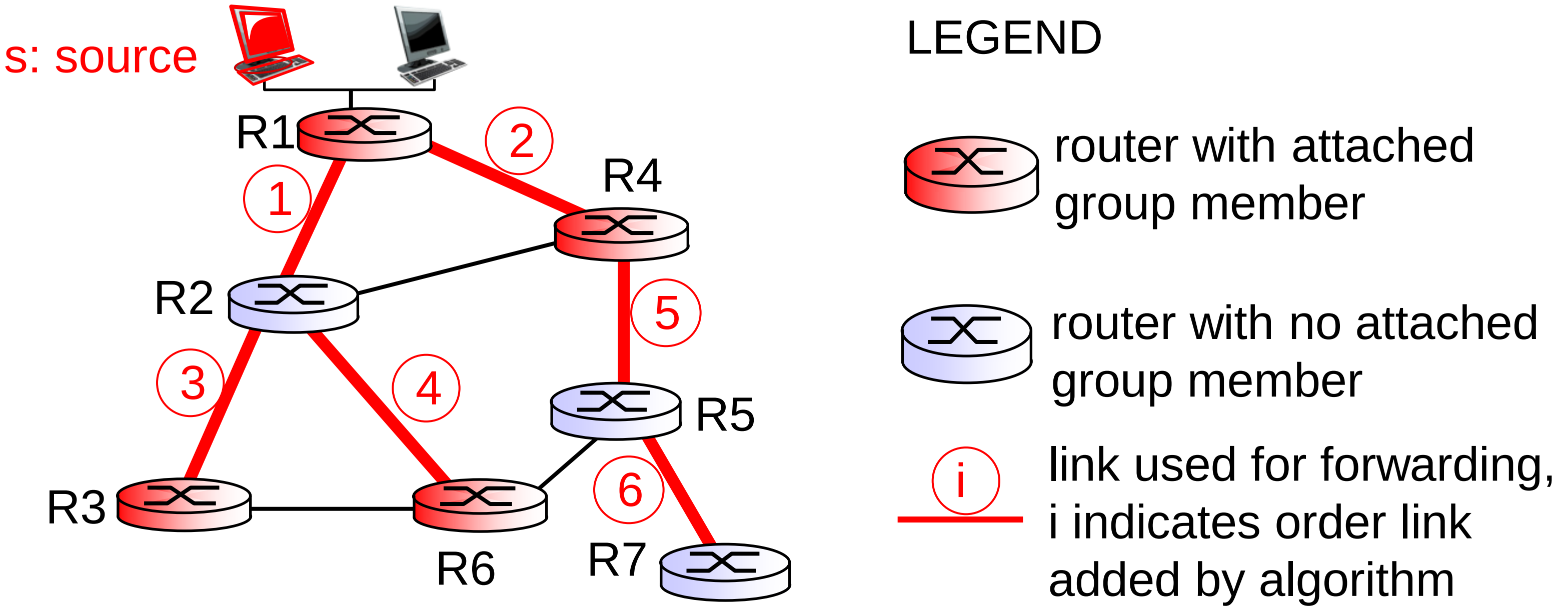
Approaches for building mcast trees

- **approaches:**
- ***source-based tree:*** **one tree per source**
  - shortest path trees
  - reverse path forwarding
- ***group-shared tree:*** **group uses one tree**
  - minimal spanning (Steiner)
  - center-based trees

…we first look at basic approaches, then specific protocols adopting these approaches

Shortest path tree

- **mcast forwarding tree: tree of shortest path routes from source to all receivers**
  - Dijkstra’s algorithm



LEGEND

router with attached group member

router with no attached group member

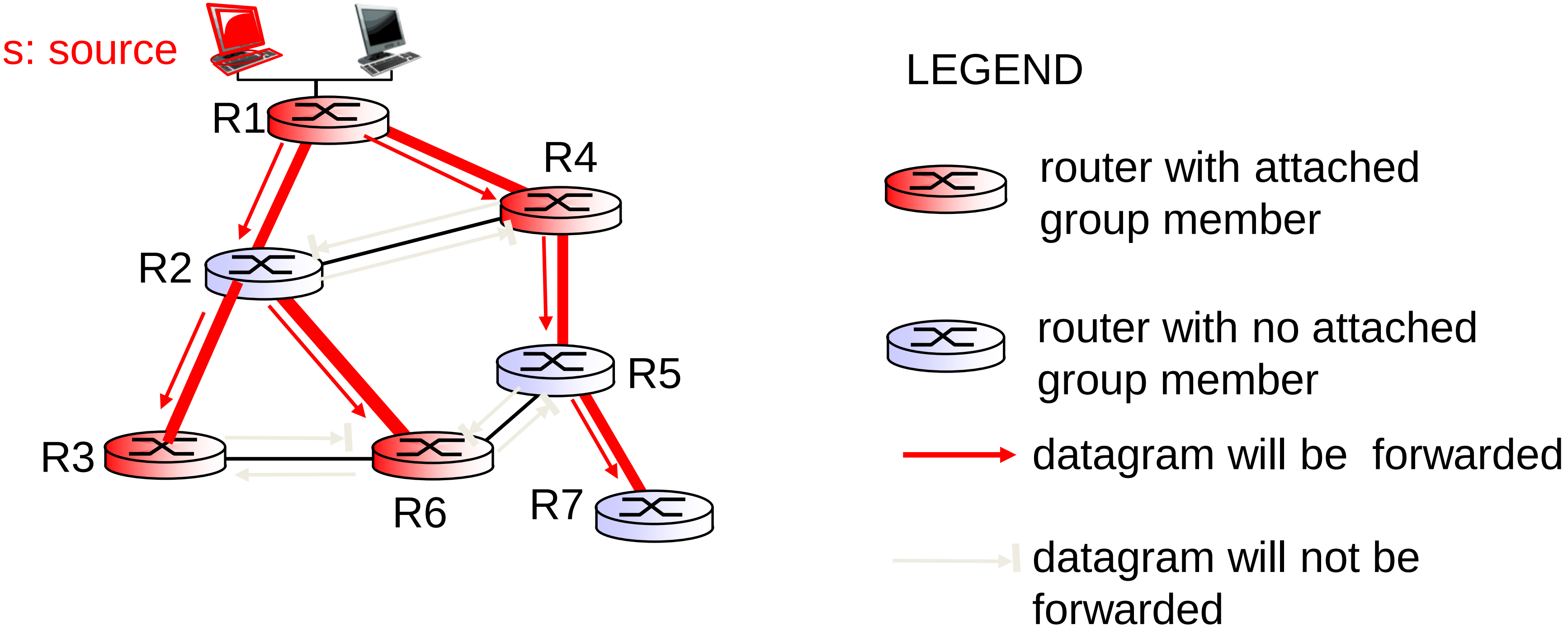
link used for forwarding, i indicates order link added by algorithm

Reverse path forwarding

- rely on router’ s knowledge of unicast shortest path from it to sender
- each router has simple forwarding behavior:

```
if (mcast datagram received on incoming link on
   shortest path back to center)
  then flood datagram onto all outgoing links
  else ignore datagram
```

Reverse path forwarding: example



LEGEND

router with attached group member

router with no attached group member

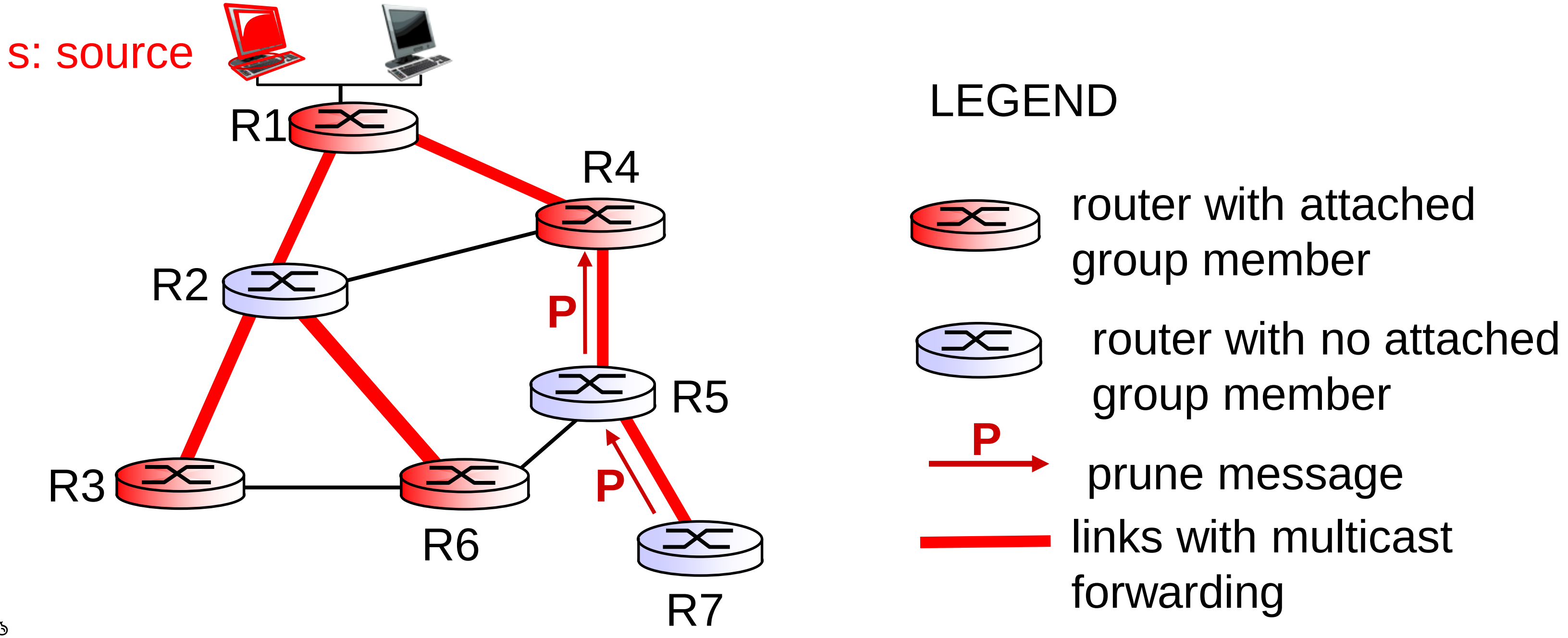
datagram will be forwarded

datagram will not be forwarded

- result is a source-specific *reverse* SPT
  - may be a bad choice with asymmetric links

Reverse path forwarding: pruning

- **forwarding tree contains subtrees with no mcast group members**
  - no need to forward datagrams down subtree
  - "prune" msgs sent upstream by router with no downstream group members



LEGEND

router with attached group member

router with no attached group member

prune message

links with multicast forwarding

Anycast

- **Multiple hosts may share the same IP address**
- **“One to one of many” routing**
- **Example uses: load balancing, nearby servers**
  - DNS Root Servers (e.g. f.root-servers.net)
  - Google Public DNS (8.8.8.8)
  - IPv6 6-to-4 Gateway (192.88.99.1)

Anycast Implementation

- **Anycast addresses are /32s**
- **At the BGP level**
  - Multiple ASs can advertise the same prefixes
  - Normal BGP rules choose one route
- **At the Router level**
  - Router can have multiple entries for the same prefix
  - Can choose among many
- **Each packet can go to a different server**
  - Best for services that are fine with that (connectionless, stateless)

IPv6 – in a nutshell

IPv6: motivation

- ***initial motivation:*** **32-bit address space soon to be completely allocated.**
- **additional motivation:**
  - header format helps speed processing/forwarding
  - header changes to facilitate QoS

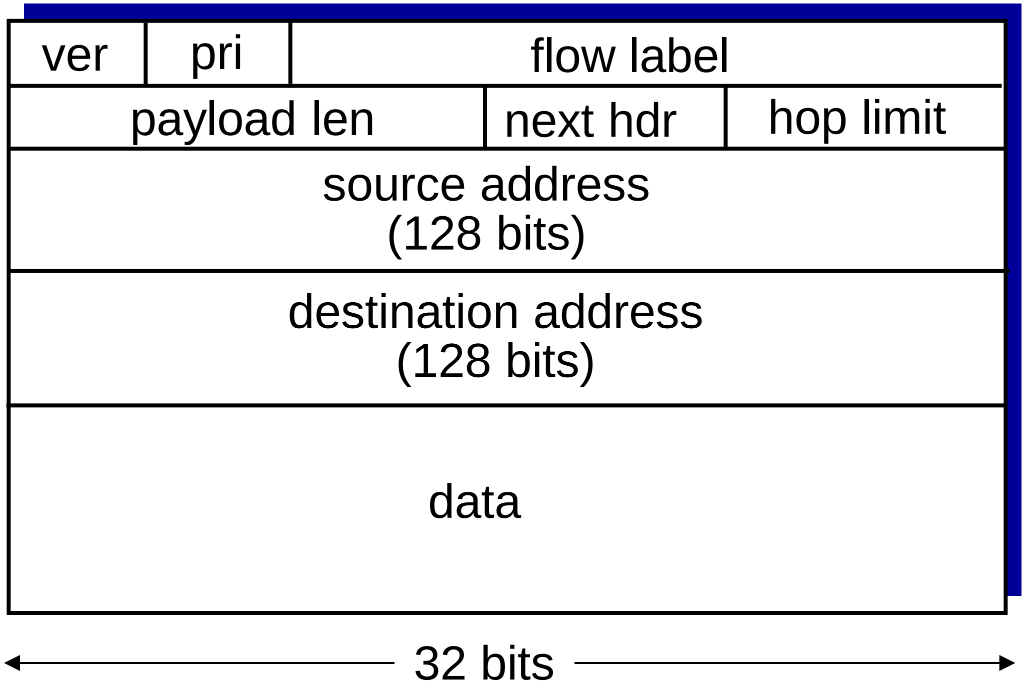
- ***IPv6 datagram format:***
  - fixed-length 40 byte header
  - no fragmentation allowed

IPv6 datagram format

priority: identify priority among datagrams in flow

flow Label: identify datagrams in same "flow."
(concept of"flow" not well defined).

next header: identify upper layer protocol for data



ver

pri

flow label

payload len

next hdr

hop limit

source address
(128 bits)

destination address
(128 bits)

data

32 bits

IPv6 Address Representation

- **Groups of 16 bits in hex notation**

  47cd:1244:3422:0000:0000:fef4:43ea:0001

- **Two rules:**
  - Leading 0’s in each 16-bit group can be omitted

    47cd:1244:3422:0:0:fef4:43ea:1

  - One contiguous group of 0’s can be compacted

    47cd:1244:3422::fef4:43ea:1

IPv6 Addresses

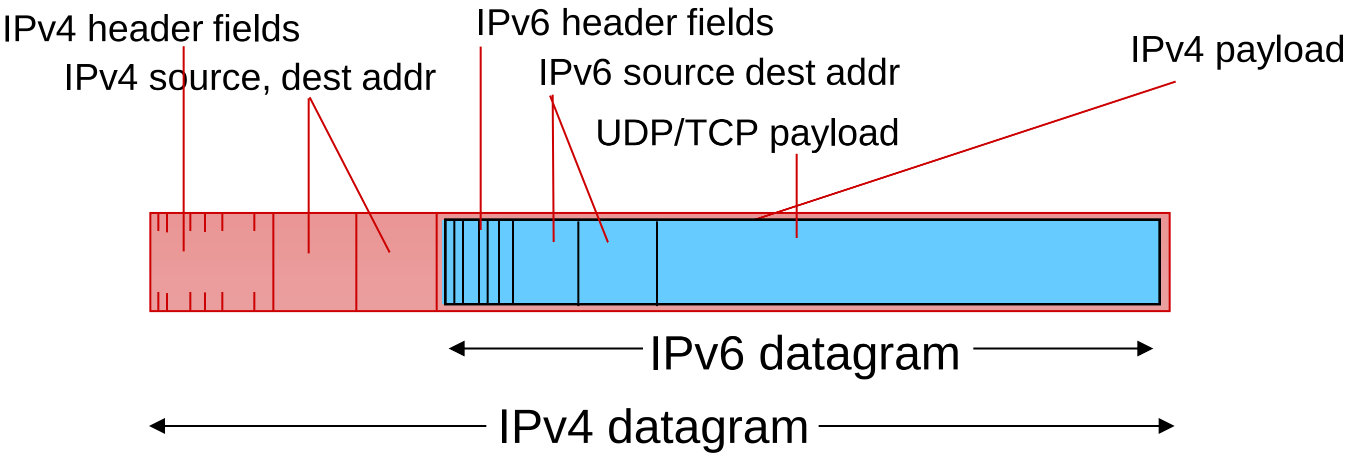
- **Break 128 bits into 64-bit network and 64-bit interface**
  - Makes autoconfiguration easy: interface part can be derived from Ethernet address, for example
- **Types of addresses**
  - All 0’s: unspecified
  - 000…1: loopback
  - ff/8: multicast
  - fe8/10: link local unicast
  - fec/10: site local unicast
  - All else: global unicast

Other changes from IPv4

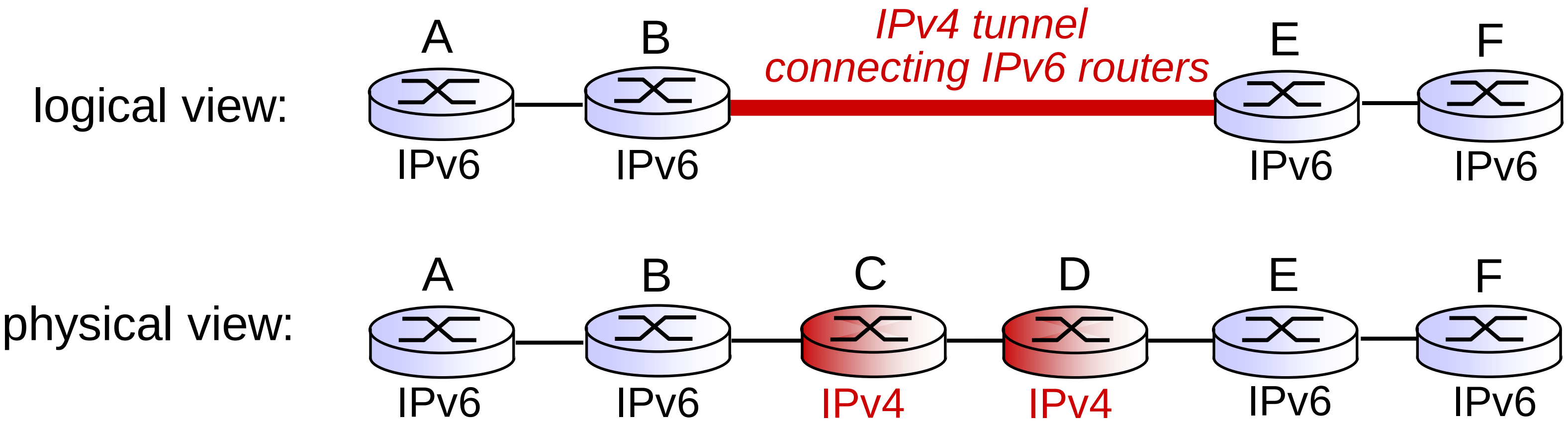
- ***checksum***: **removed entirely to reduce processing time at each hop**
- ***options:*** **allowed, but outside of header, indicated by "Next Header" field**
- ***ICMPv6:*** **new version of ICMP**
  - additional message types, e.g. "Packet Too Big"
  - multicast group management functions

Transition from IPv4 to IPv6

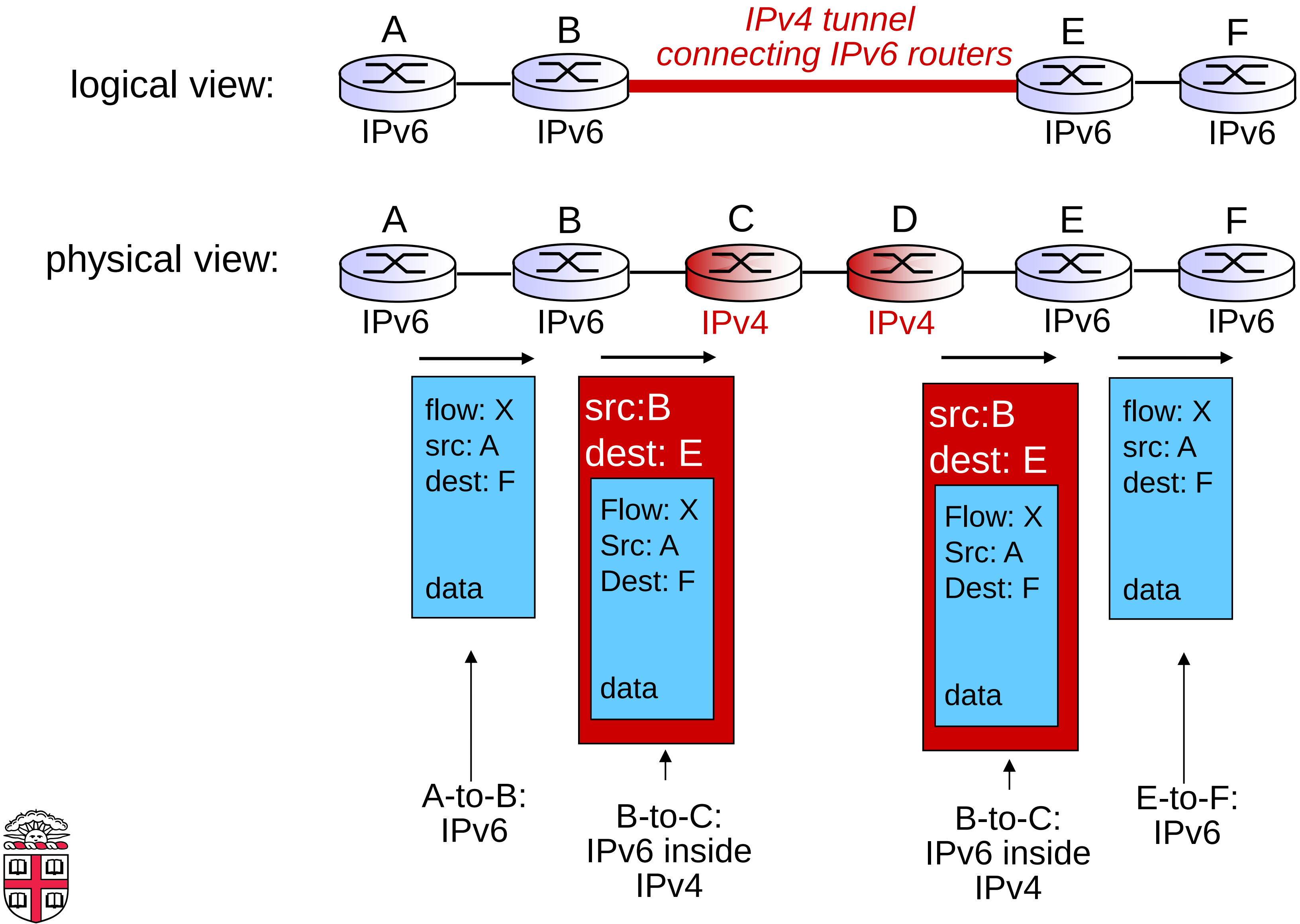
- **not all routers can be upgraded simultaneously**
  - no “flag days”
  - how will network operate with mixed IPv4 and IPv6 routers?
- ***tunneling:* IPv6 datagram carried as *payload* in IPv4 datagram among IPv4 routers**



Tunneling


IPv4 tunnel
connecting IPv6 routers
logical view:
A
B
E
F
IPv6
IPv6
IPv6
IPv6
physical view:
A
B
C
D
E
F
IPv6
IPv6
IPv4
IPv4
IPv6
IPv6

Tunneling



logical view:

A B E F

IPv6 IPv6 IPv6 IPv6

IPv4 tunnel connecting IPv6 routers

physical view:

A B C D E F

IPv6 IPv6 IPv4 IPv4 IPv6 IPv6

flow: X
src: A
dest: F

data

src:B
dest: E

Flow: X
Src: A
Dest: F

data

src:B
dest: E

Flow: X
Src: A
Dest: F

data

flow: X
src: A
dest: F

data

A-to-B:
IPv6

B-to-C:
IPv6 inside
IPv4

B-to-C:
IPv6 inside
IPv4

E-to-F:
IPv6

Good Luck in the exam!

Next wee I’m away, but online…