

Networking and the Internet Protocol

Amin Vahdat
CSE 123b
April 13, 2006

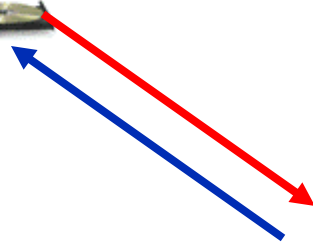
Annoucements

- First assignment available
 - HTTP Server: due Apr 21, 5 pm

Example: What Happens When You Click on a Web Link?

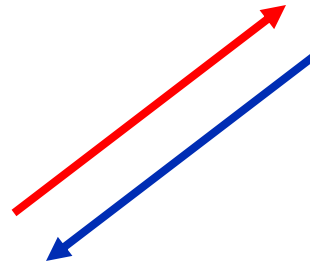
What Happens When You Click on a Web Link?

Your computer



?

www.google.com

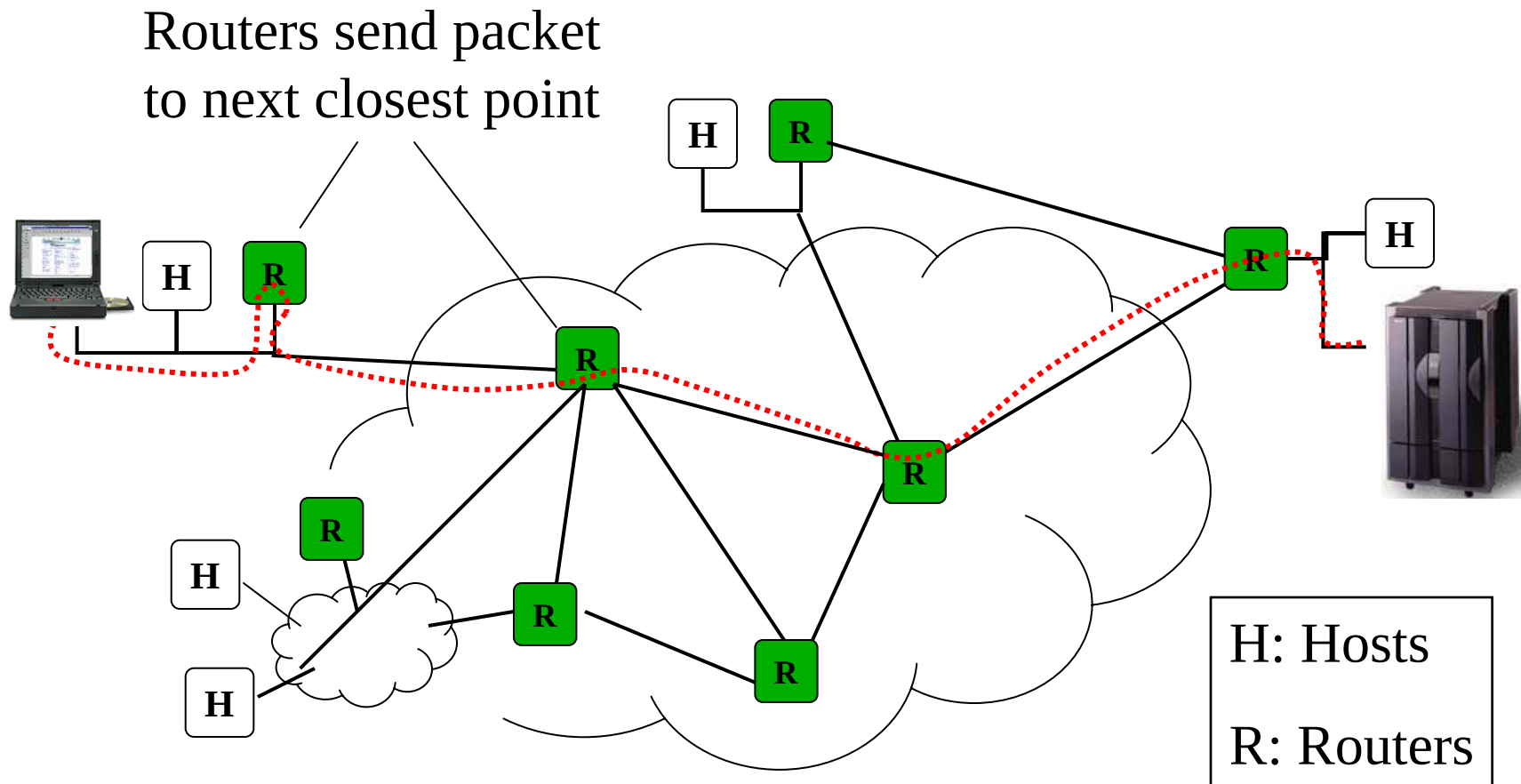


Internet

Steps in Retrieving a Web Page

- Extract hostname from URL
 - <http://www.google.com/foo> → www.google.com
- Use DNS to translate www.google.com to IP address
 - Used for Internet routing
- Establish a TCP (socket) connection to 66.102.7.104
 - Protocol agreement for browser and server to speak HTTP
 - TCP handle network problems (drops, corruption, etc.)
 - TCP layered on top of IP/Ethernet
- Routers determine efficient path to 66.102.7.104

How Does a Packet Get Through the Internet?

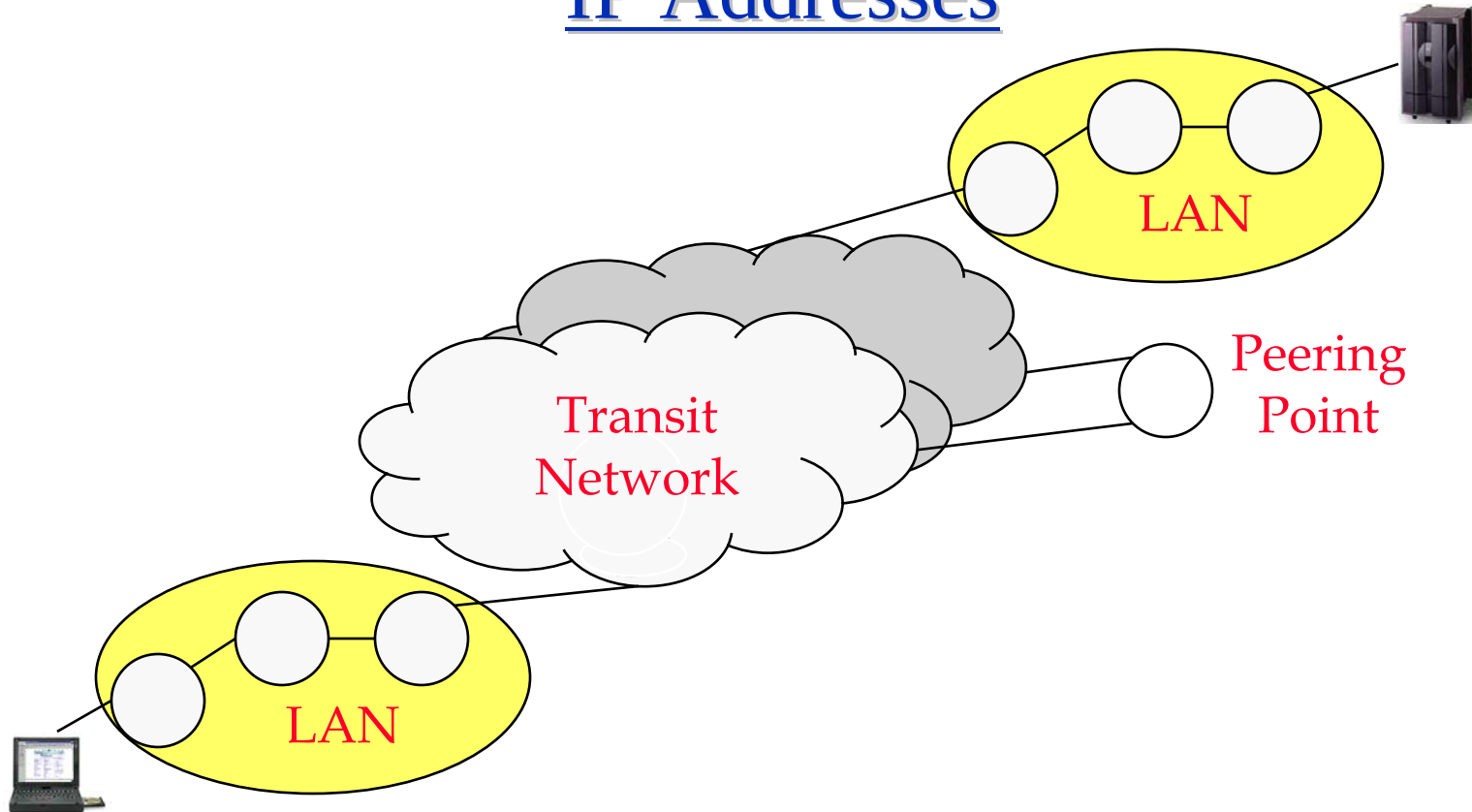


Router

- Juniper Router
- 32 line cards
- 16 OC-192 line cards
 - 10 Gbps
- 16 4 port OC-48
 - 2.5 Gbps
- Aggregate bisection bandwidth: 320 Gbps
 - 160,000 cable modems going at full speed



IP Addresses



- At each step, figure out how to get closer to destination

Building Routing Tables

- Graph theory to compute “shortest path”

- Switches = nodes
- Links = edges
- Delay, hops = cost

Cost function changes dynamically and rapidly

- Static route computation

- Does not address failure
- Does not address addition of new links
- Assumes that link “cost” is constant

⇒ Need dynamic updates to adapt to change

Two Approaches

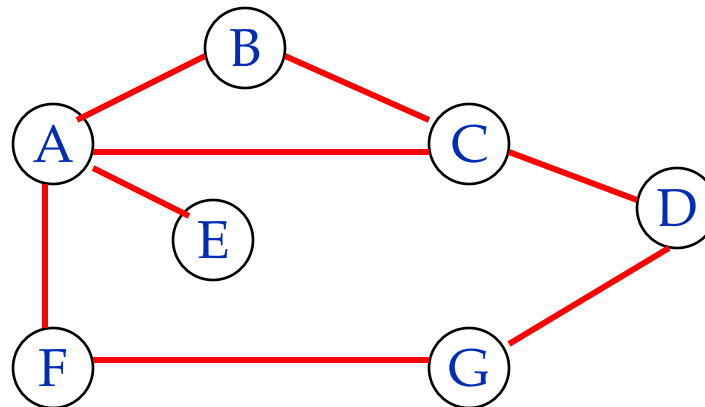
- Distance vector (RIP, BGP)
 - Exchange routing tables with neighbors
 - No one knows complete topology
 - Now used between administrative domains
- Link state (OSPF)
 - Send (flood to) everyone costs to reach directly connected neighbors
 - Everyone computes shortest path
 - Now used within administrative domains

Distance Vector Algorithm

- Initially, can get to self with cost 0
- Iterate
 - Exchange tables with neighbors
 - If neighbor has lower cost, update table

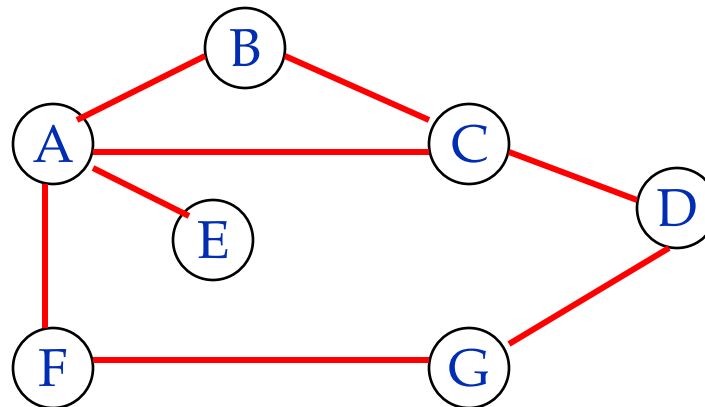
	A	B	C	D	E	F	G
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	1	∞	∞	∞
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

Round 1



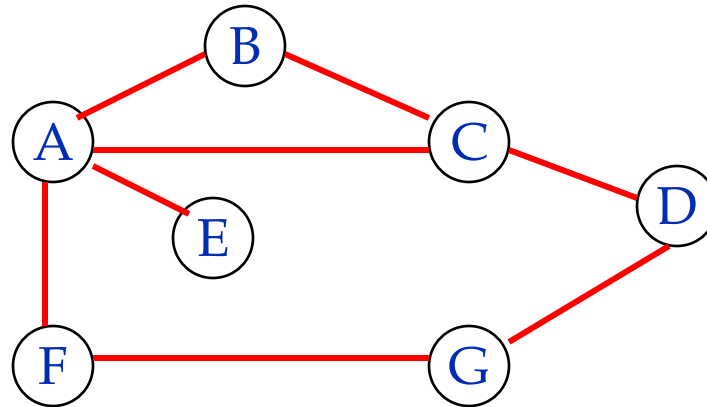
	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	∞
C	1	1	0	1	2	2	2
D	2	2	1	0	∞	2	1
E	1	2	2	∞	0	2	∞
F	1	2	2	2	2	0	1
G	2	∞	2	1	∞	1	0

Round 2



	A	B	C	D	E	F	G
A	0	1	1	2	1	1	2
B	1	0	1	2	2	2	3
C	1	1	0	1	2	2	2
D	2	2	1	0	3	2	1
E	1	2	2	3	0	2	3
F	1	2	2	2	2	0	1
G	2	3	2	1	3	1	0

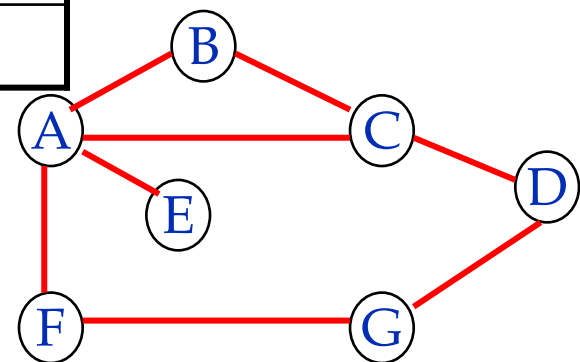
Round 3



Sample Routing Table

- Routing table at node B:

Destination	Cost	Next Hop
A	1	A
C	1	C
D	2	C
E	2	A
F	2	A
G	3	A

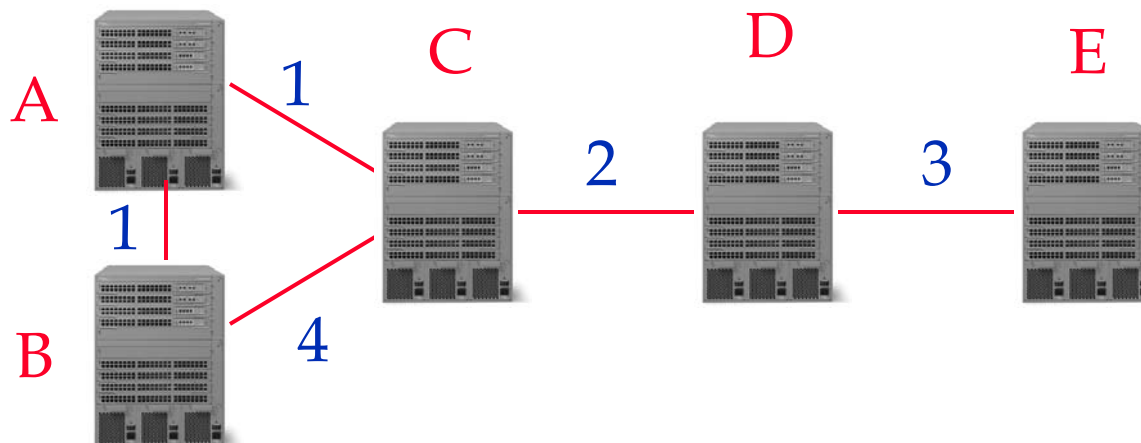


Link State

- Each node gets complete topology via reliable flooding
 - Each node identifies direct neighbors, puts in numbered link state packet (LSP)
 - Includes Node ID, cost to directly connected neighbors, sequence number, TTL
 - If get link state packet from neighbor Q
 - If seen already → drop
 - Else process and forward everywhere but Q
- Given complete topology, compute shortest path using graph algorithm
 - Dijkstra's shortest-path algorithm

Link State Example

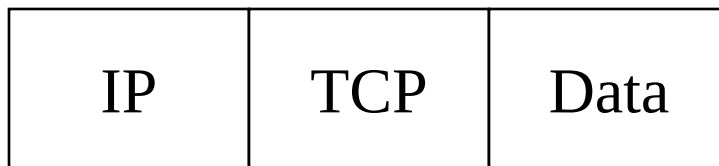
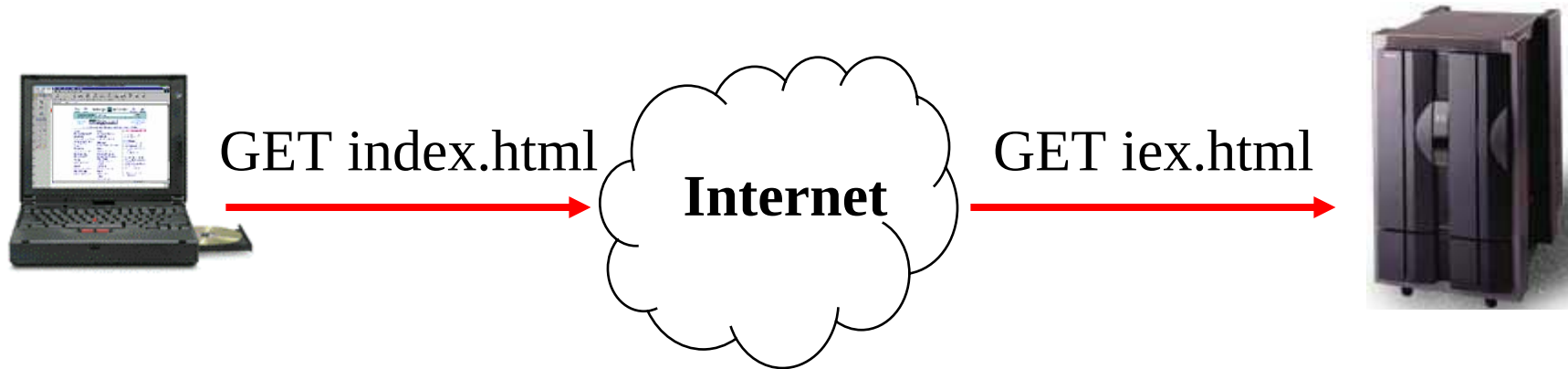
- Flood following information through the network
 - A: (A→C,1) (A→B,1)
 - B: (B→A,1) (B→C,4)
 - C: (C→A,1) (C→B,4) (C→D,2)
 - D: (D→C,2) (D→E,3)
 - E: (E→D,3)



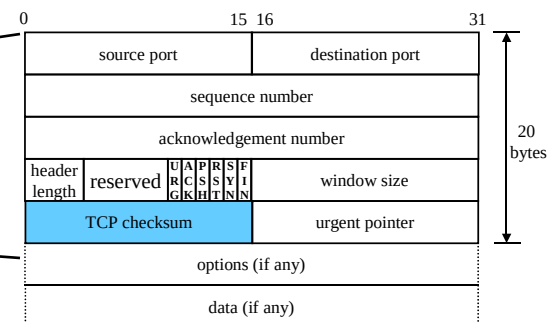
Have Address, Now Send Data

- Real networks must deal with congestion, bit errors, etc.
 - Data can be corrupted
 - Data can get lost
 - Data might not fit in a single packet
 - Data can be delivered in the wrong order
 - ...

What If the Data Gets Corrupted?

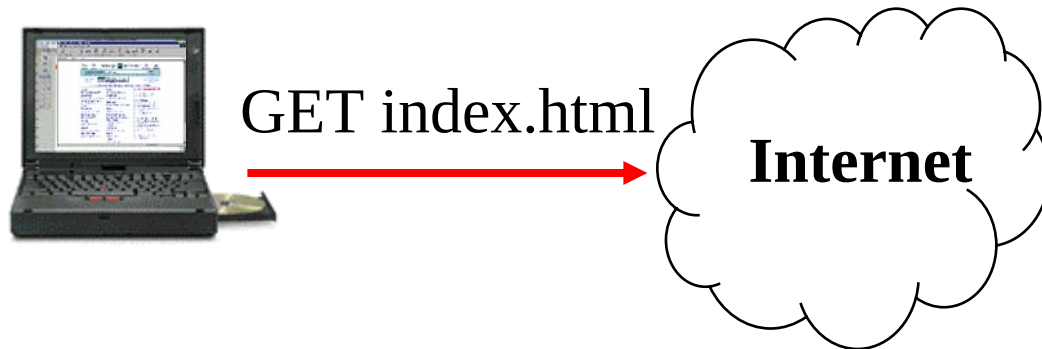


TCP Packet Format

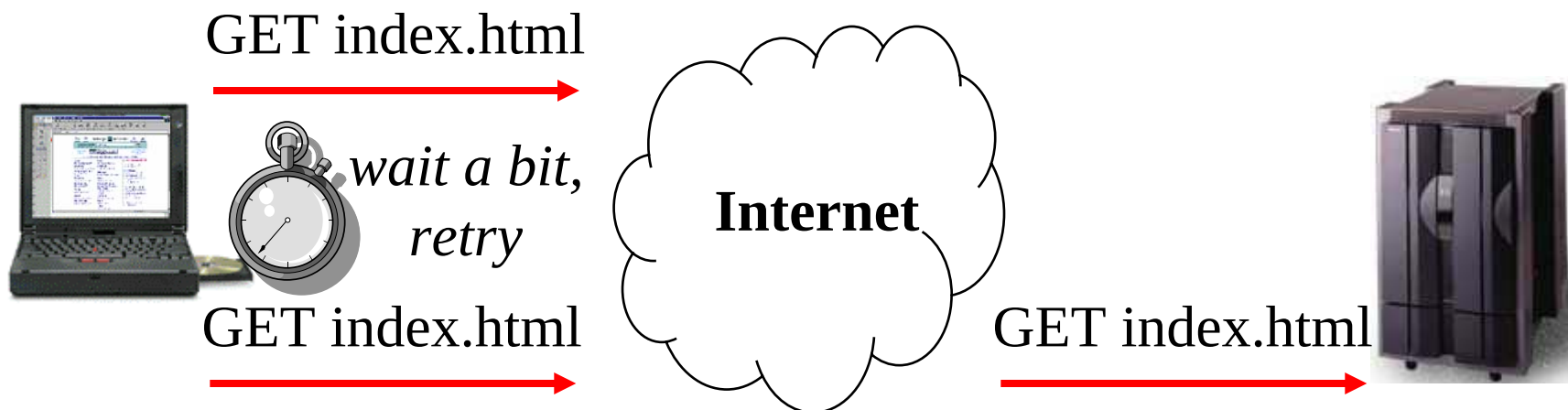


Solution: Add a *checksum*

What If the Data Gets Lost?



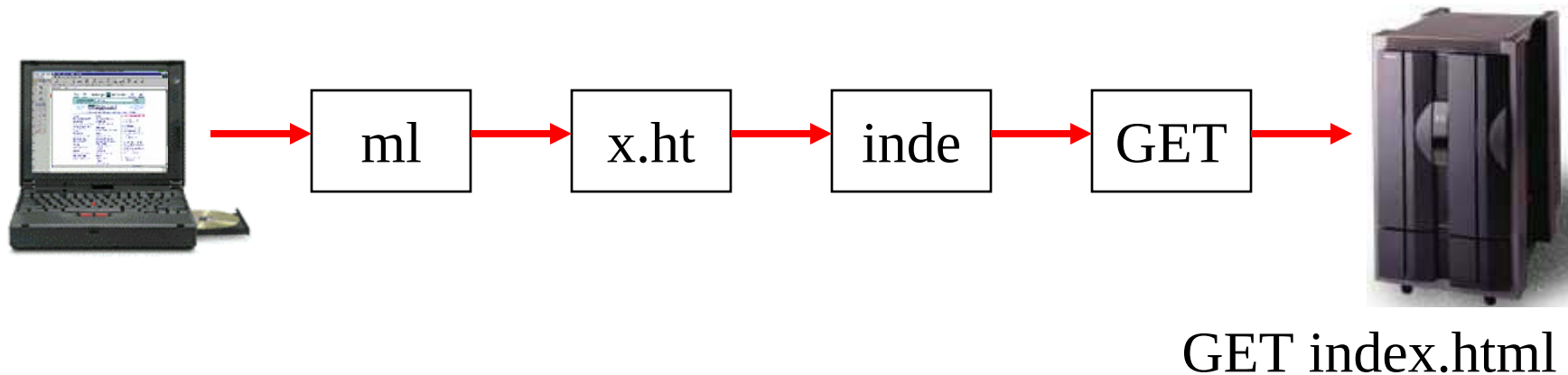
Solution: *Timeout and retransmit*



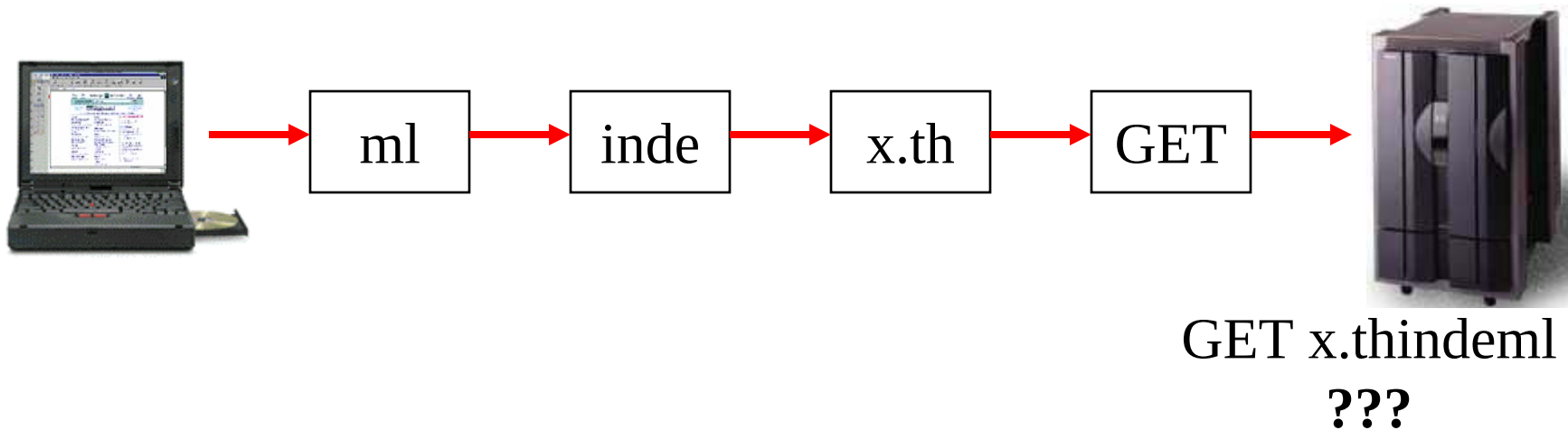
What If the Data Does Not Fit in a Single Packet?

On Ethernet, max IP packet is 1.5kbytes
Typical web page is 10kbytes

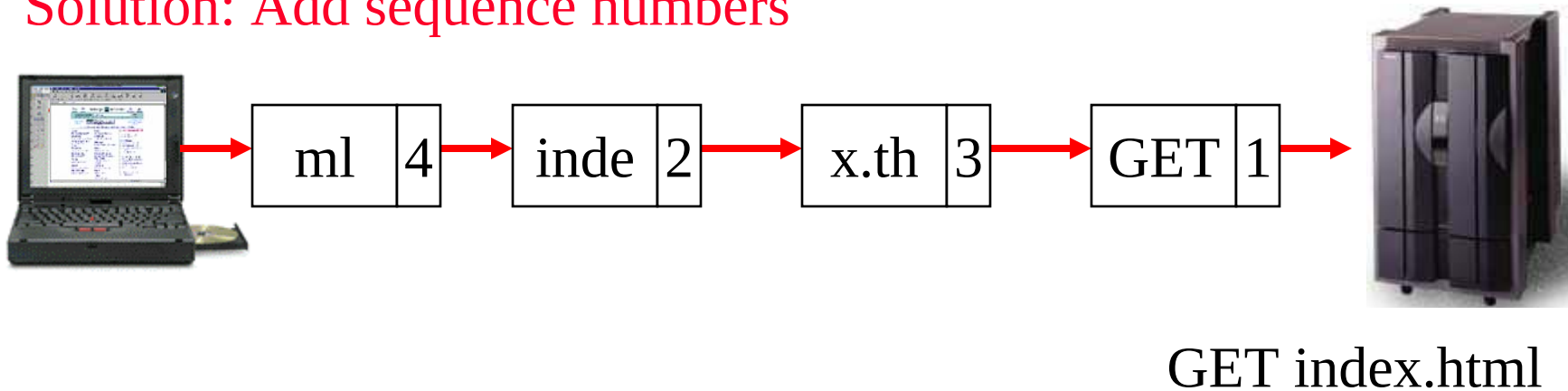
Solution: Fragment data across packets



What If the Data is Out of Order?



Solution: Add sequence numbers



What If Network is Overloaded?

- Data can arrive at router faster than it can be forwarded

What If Network is Overloaded?

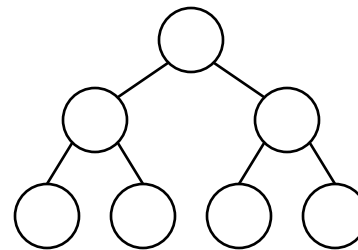
- Data can arrive at router faster than it can be forwarded
- Short bursts: buffer at router
- What if buffer overflows?
 - Packets dropped and retransmitted
 - Sender adjusts rate until load matches available resources
- Called “Congestion control”
 - Broadcast network (Ethernet): bus arbitration
 - Wide-area networks: typically handled by TCP
 - Necessary for global fairness, but in the interests of individuals to ignore

What If Data Has a Deadline?

- Ex: multimedia, teleconferencing
- Original Internet: out of luck!
- To provide guarantees, need
 - Admission control
 - Resource management at routers
- Ex: Telephone network has busy signals + explicit schedules at each switch
- How do we add this to the Internet?
 - Open and active research question
 - Fall back to circuit-switched networks?

What if Multiple Receivers?

- Latest version of software updates, video conferencing
- Send a separate packet to each?
 - What if billions of receivers?
- Multicast
 - Routers form distribution tree
- What if data is dropped?
 - Acks would overwhelm sender
 - Naks? if drop is early in the tree → overwhelm sender



What if Sender is Malicious?

- Every packet has source, destination IP addresses
- But! Host can put *anything* in IP header
 - Packet may have come from anywhere
 - Firewalls to enforce sanity checks
 - ex: source must be from other side of wall
 - ex: only allow reply packets
 - Encryption/digital signatures for authentication/privacy