E6998-02: Internet Routing

Lecture 18
Overlay Networks

John Ioannidis
AT&T Labs – Research

ji+ir@cs.columbia.edu

Announcements

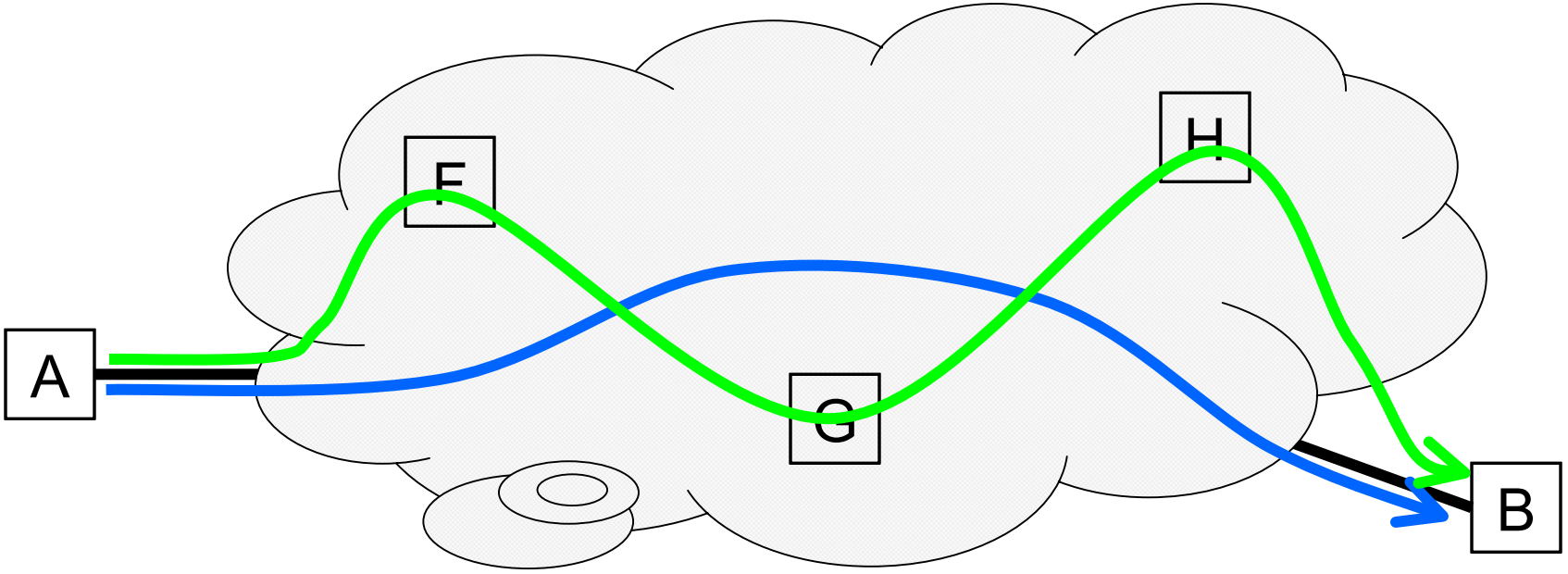
Lectures 1-18 are available.

There are no more announcements.

Source Routing

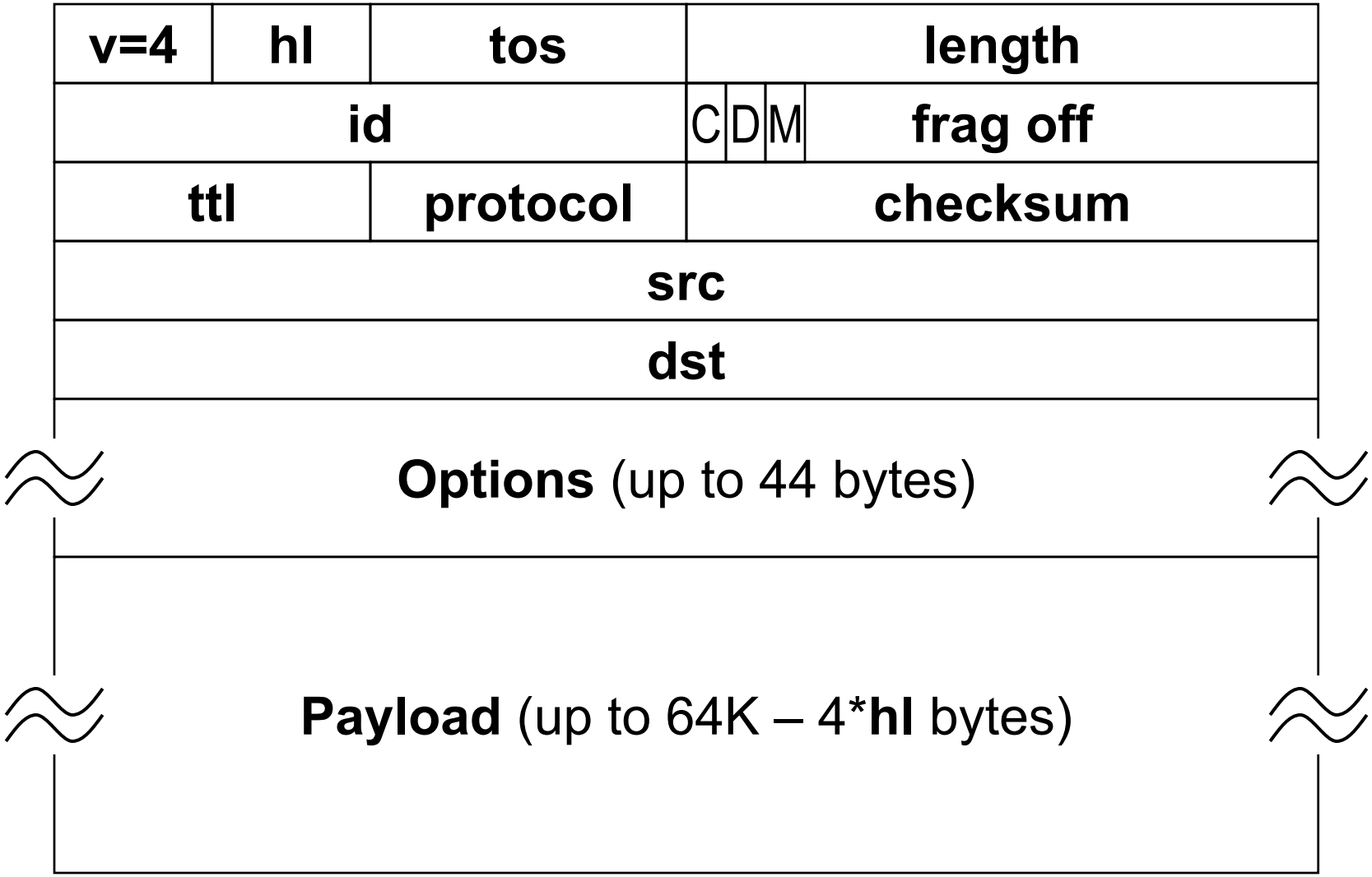
- Normally, the path a packet takes from its source to its destination is determined by each intermediate router.
- A packet can be *source-routed*: the source determines
  - all (strict source routing)
  - some (loose source routing)

  of the routers that the packet will pass through.



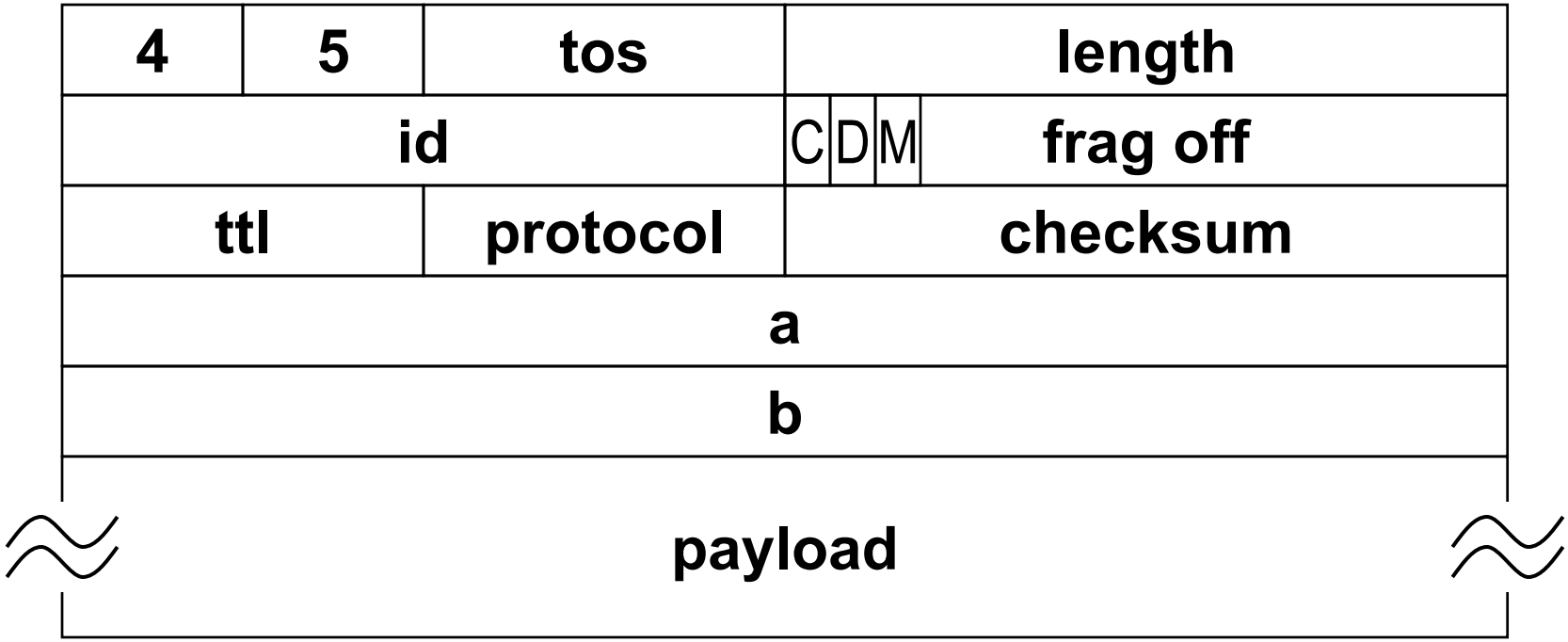
Source Routing in IPv4

- Recall the IPv4 packet format:



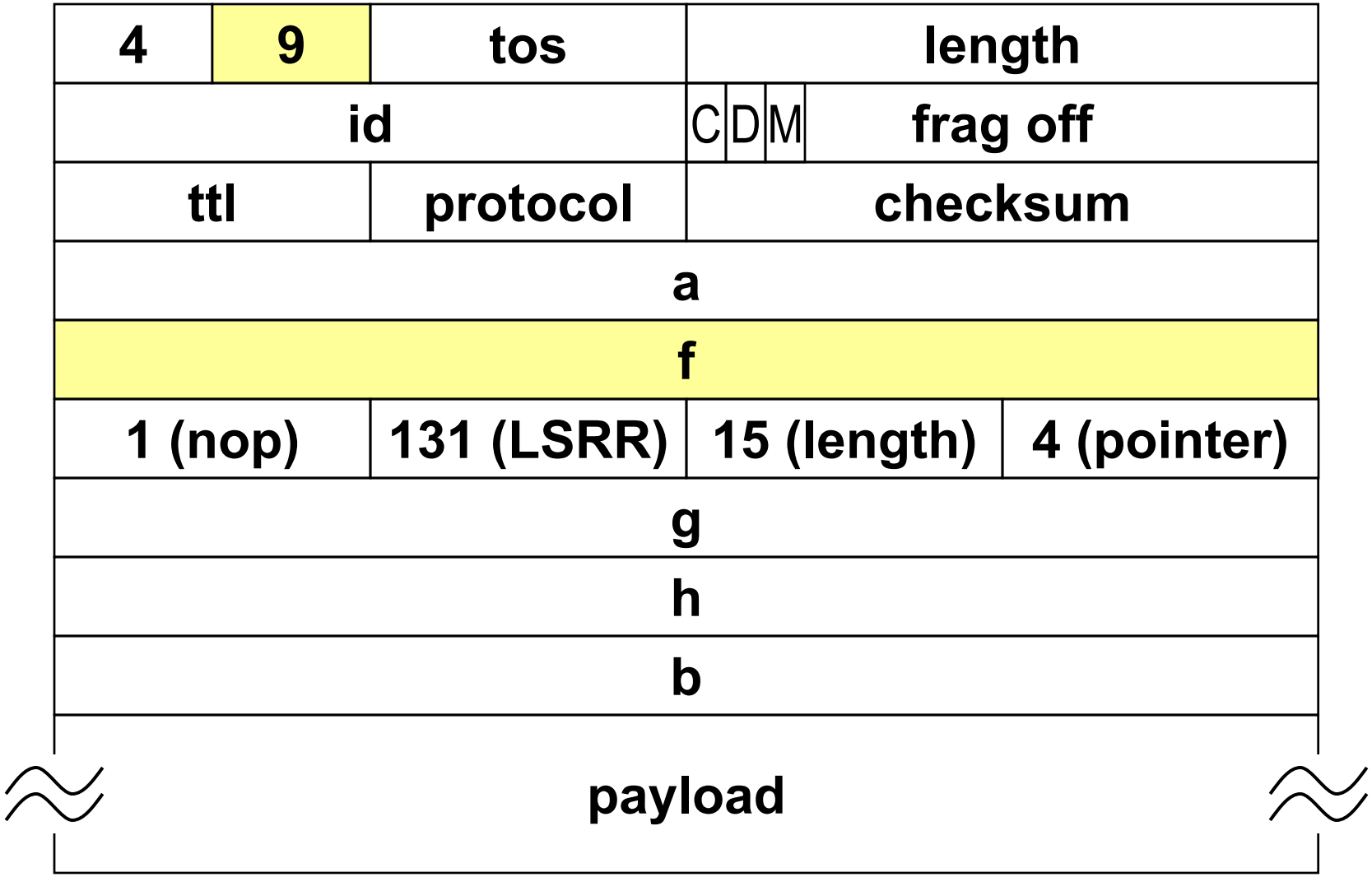
Source Routing in IPv4

- When sending from **a** to **b** (no source routing):



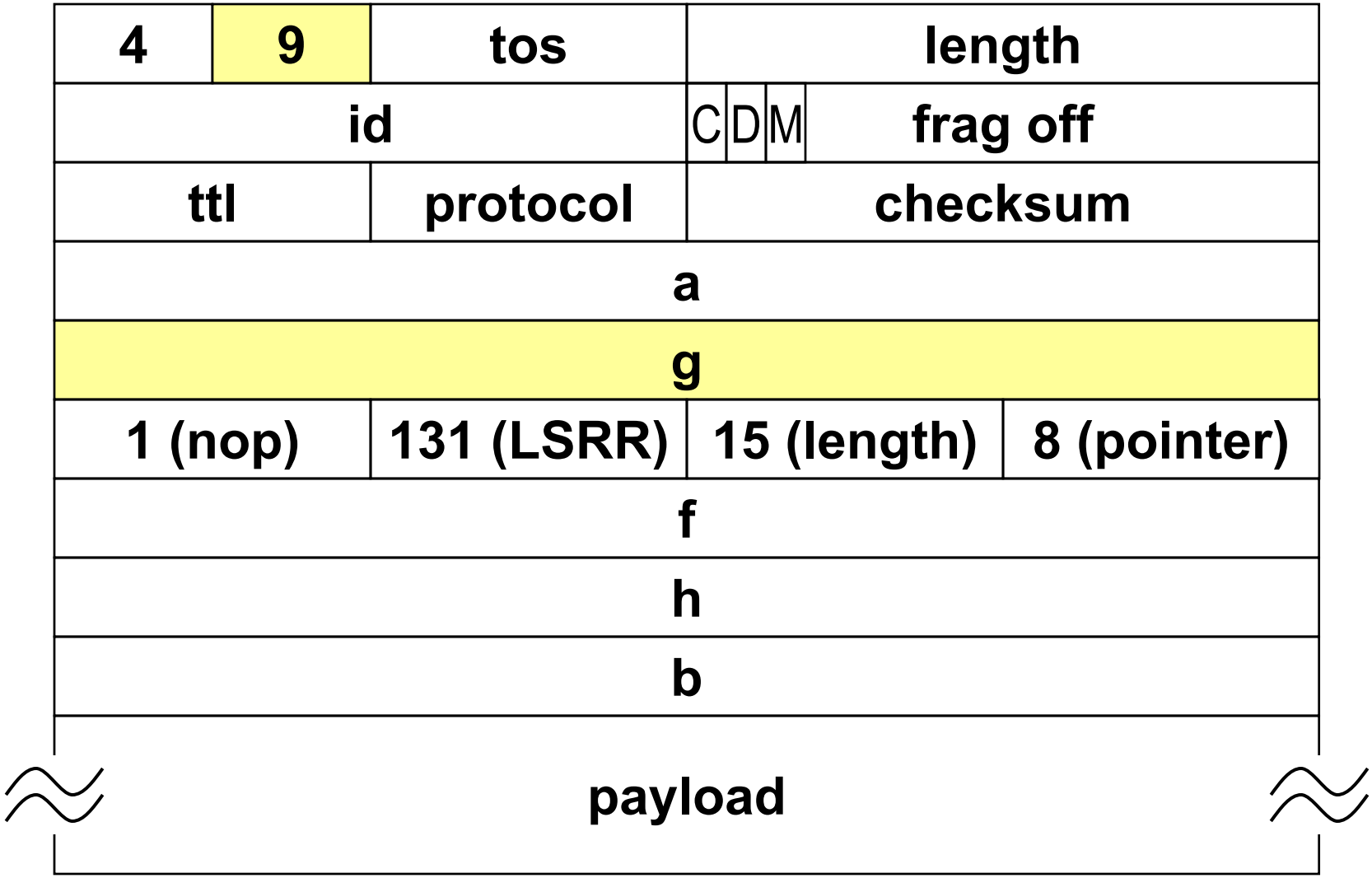
Source Routing in IPv4

- When sending from **A** to **B** (loose source routing via **F**, **G**, **H**):



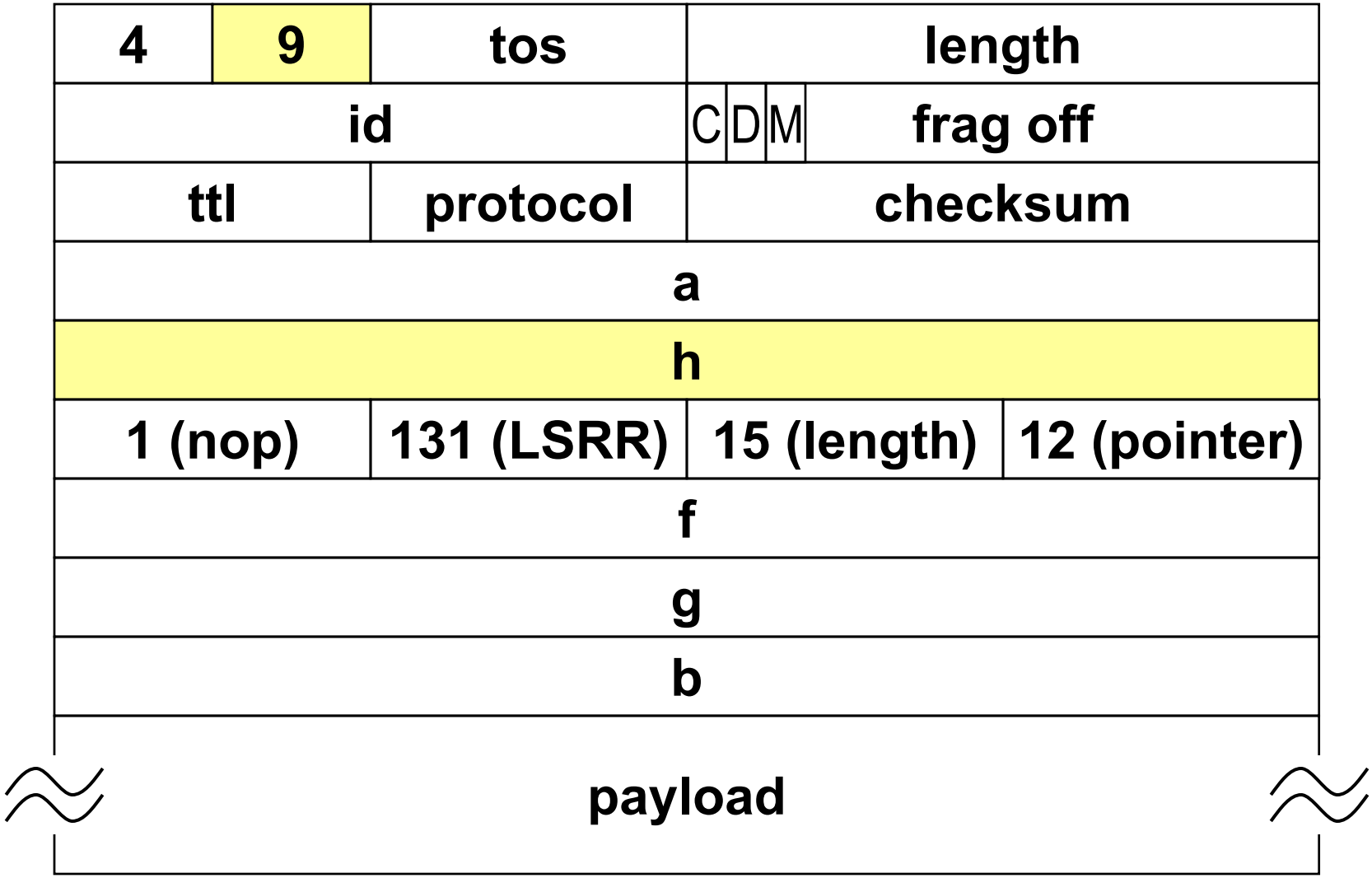
Source Routing in IPv4

- We're at F…



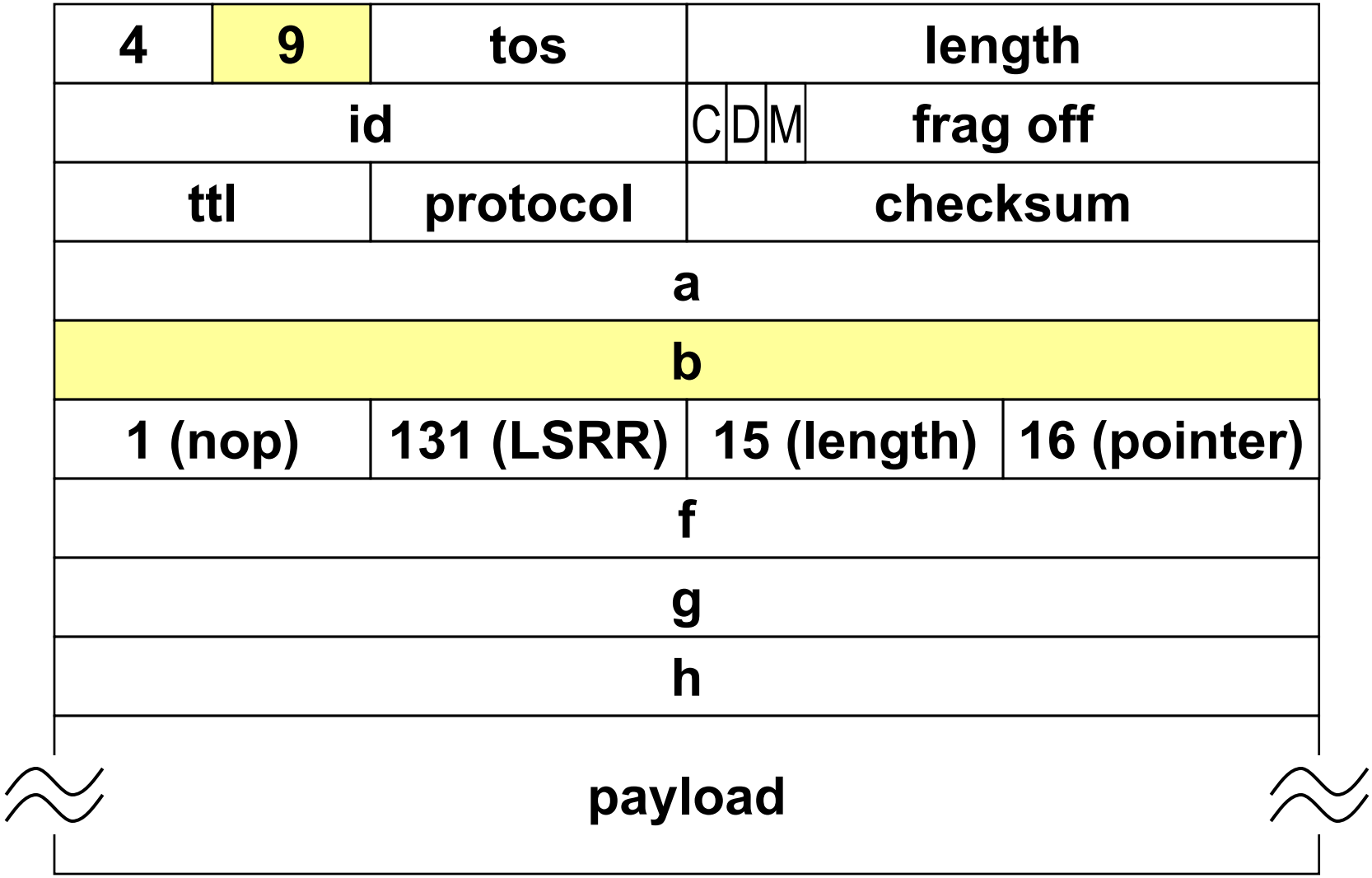
Source Routing in IPv4

- We're at G…



Source Routing in IPv4

- We're at H…



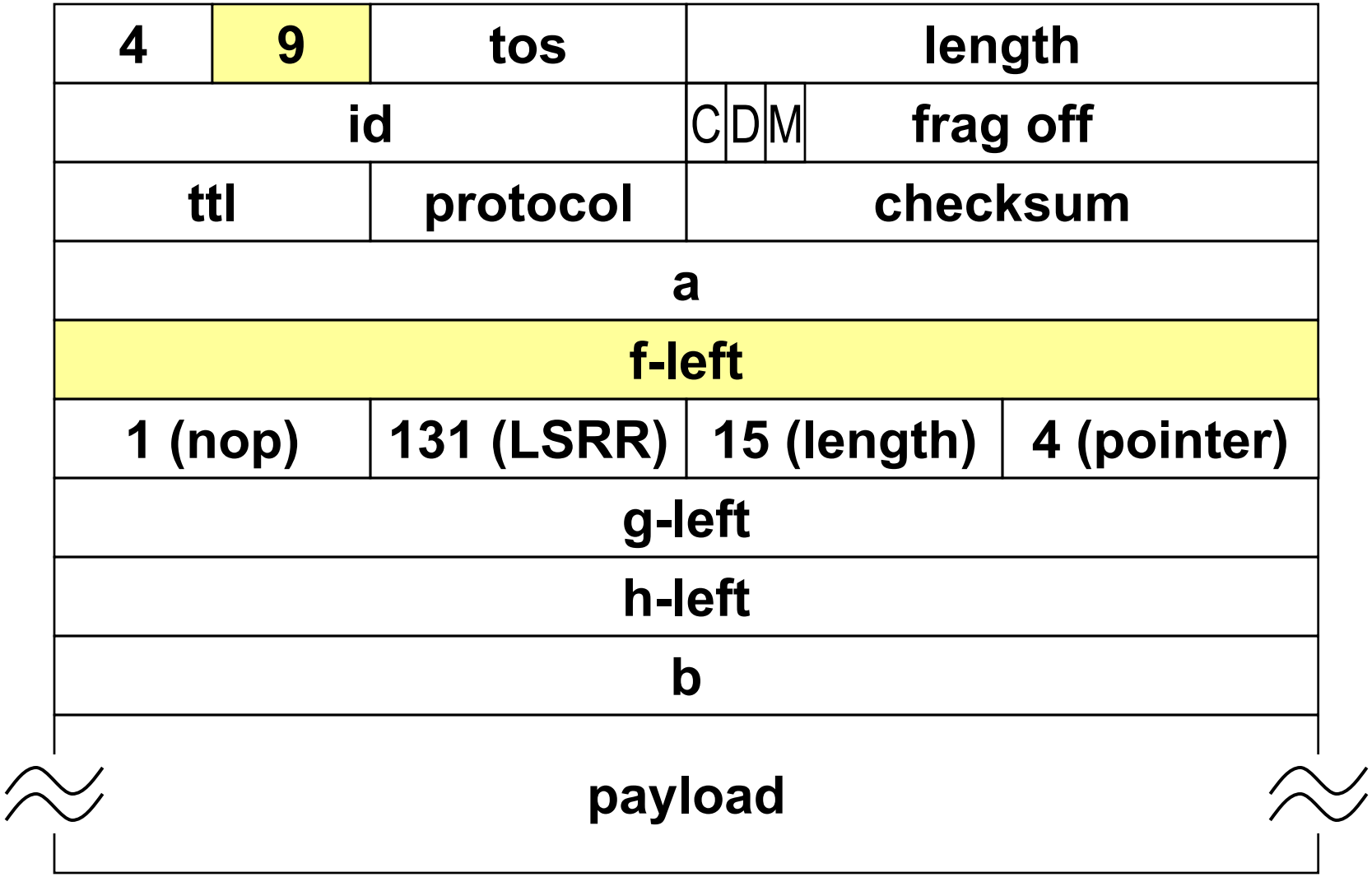
Source Routing in IPv4

- B sees that the LSRR pointer is beyond the end of the option, so it accepts the packet as its own.

- Actually, the addresses replaced in the option are not exactly as shown.
  - The next address should be the next-hop interface (“left”).
  - The replaced address is the egress interface (“right”).
  - This way, the LSRR is ready to be reversed and use by B.

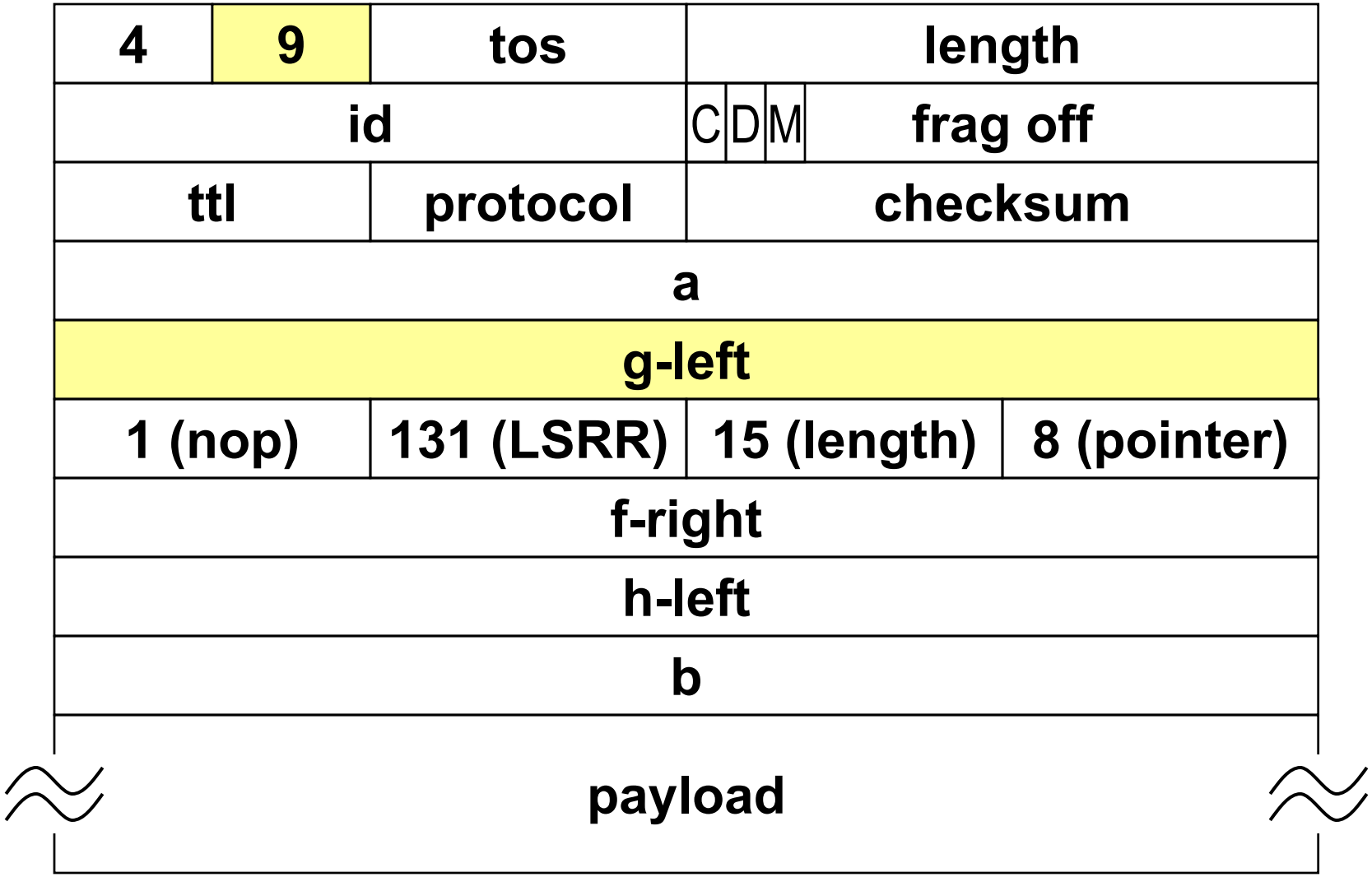
Back to A

- A fills out the “left” addresses of F, G, H:



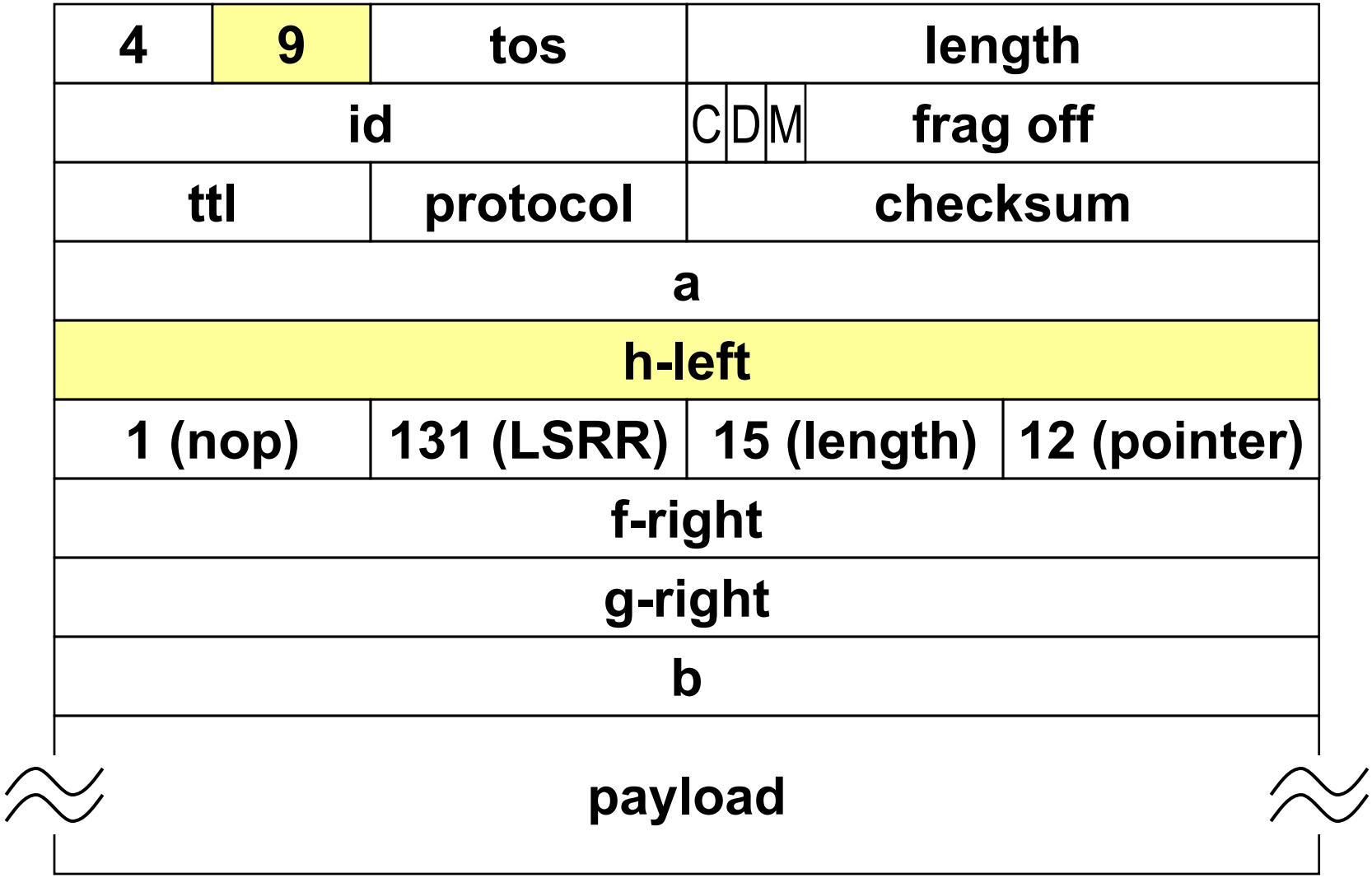
At F

- F puts its egress address in the option.



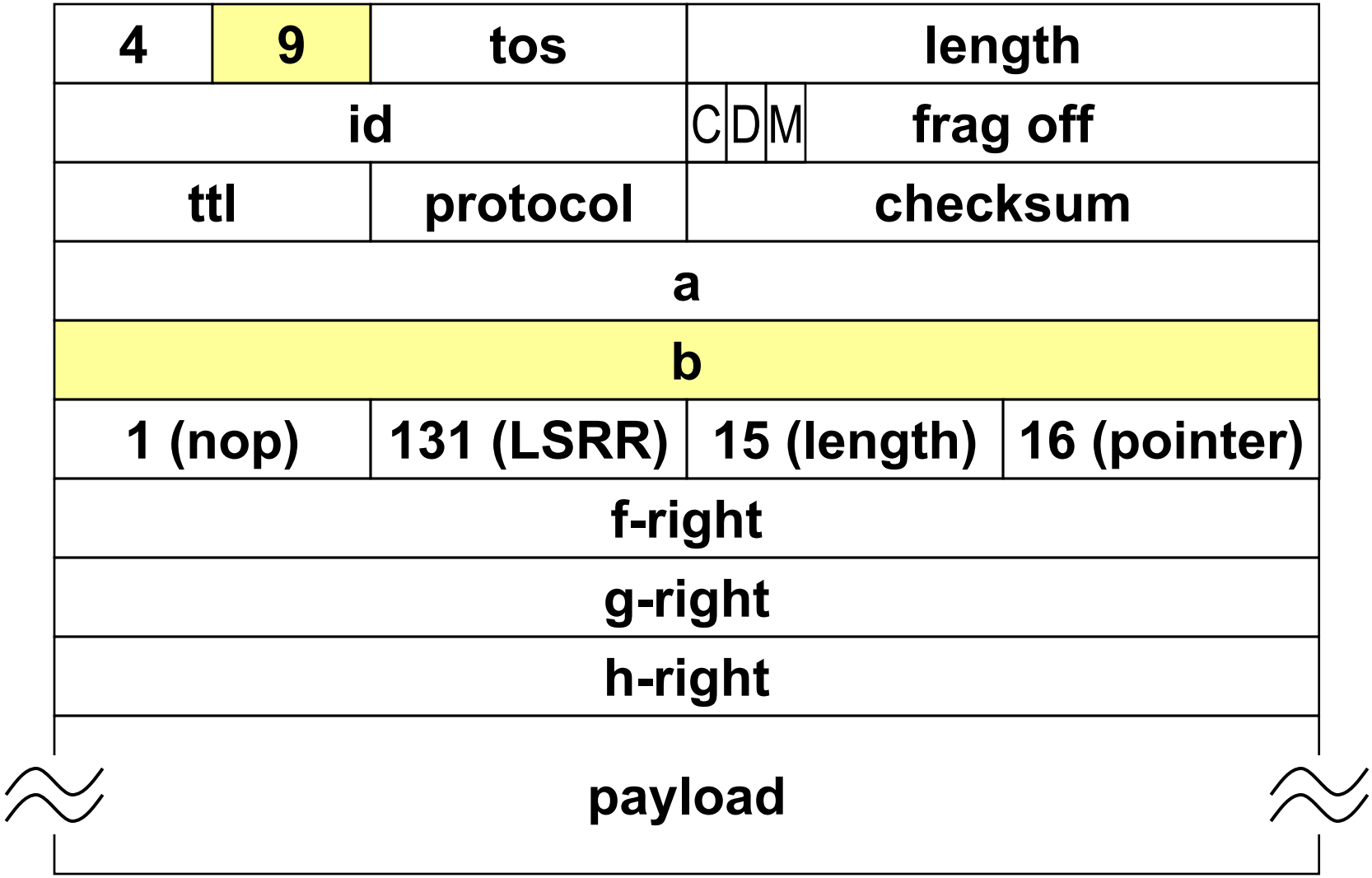
At G

- G puts its egress address in the option.



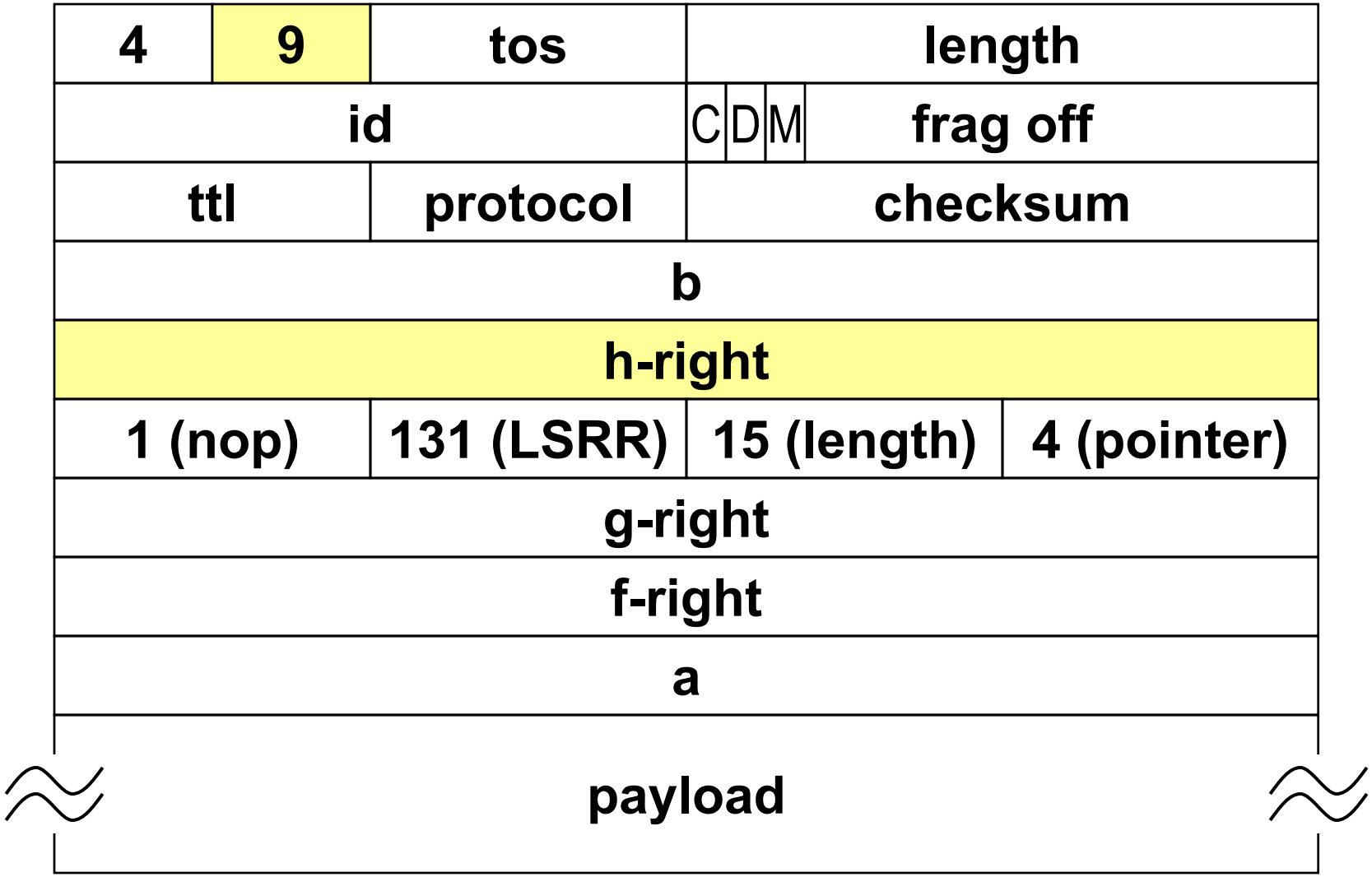
At H

- About to take the last hop



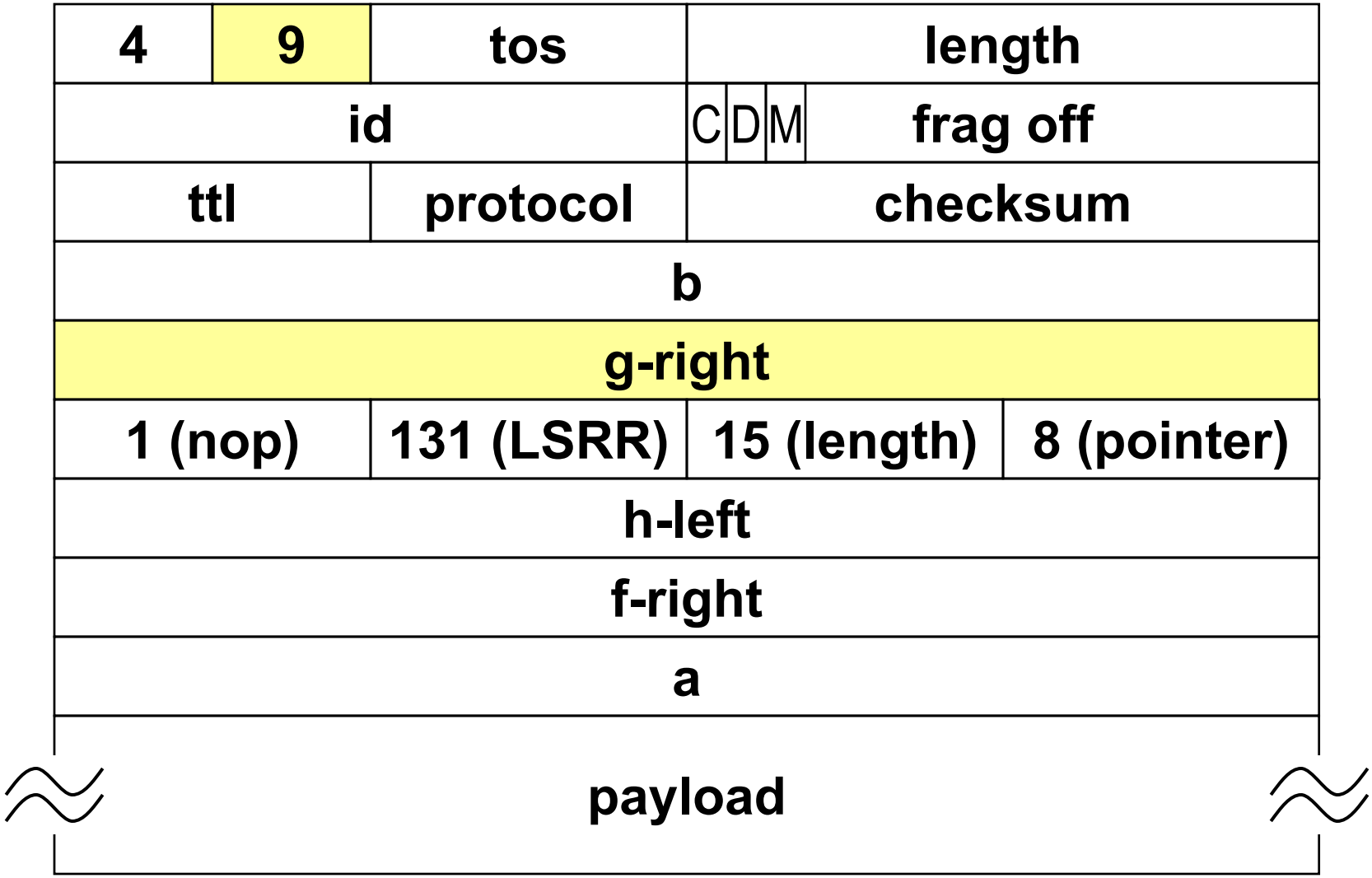
Reversing the Source Routing

- To source-route the packet back to A, B reverses the Recorded Route:



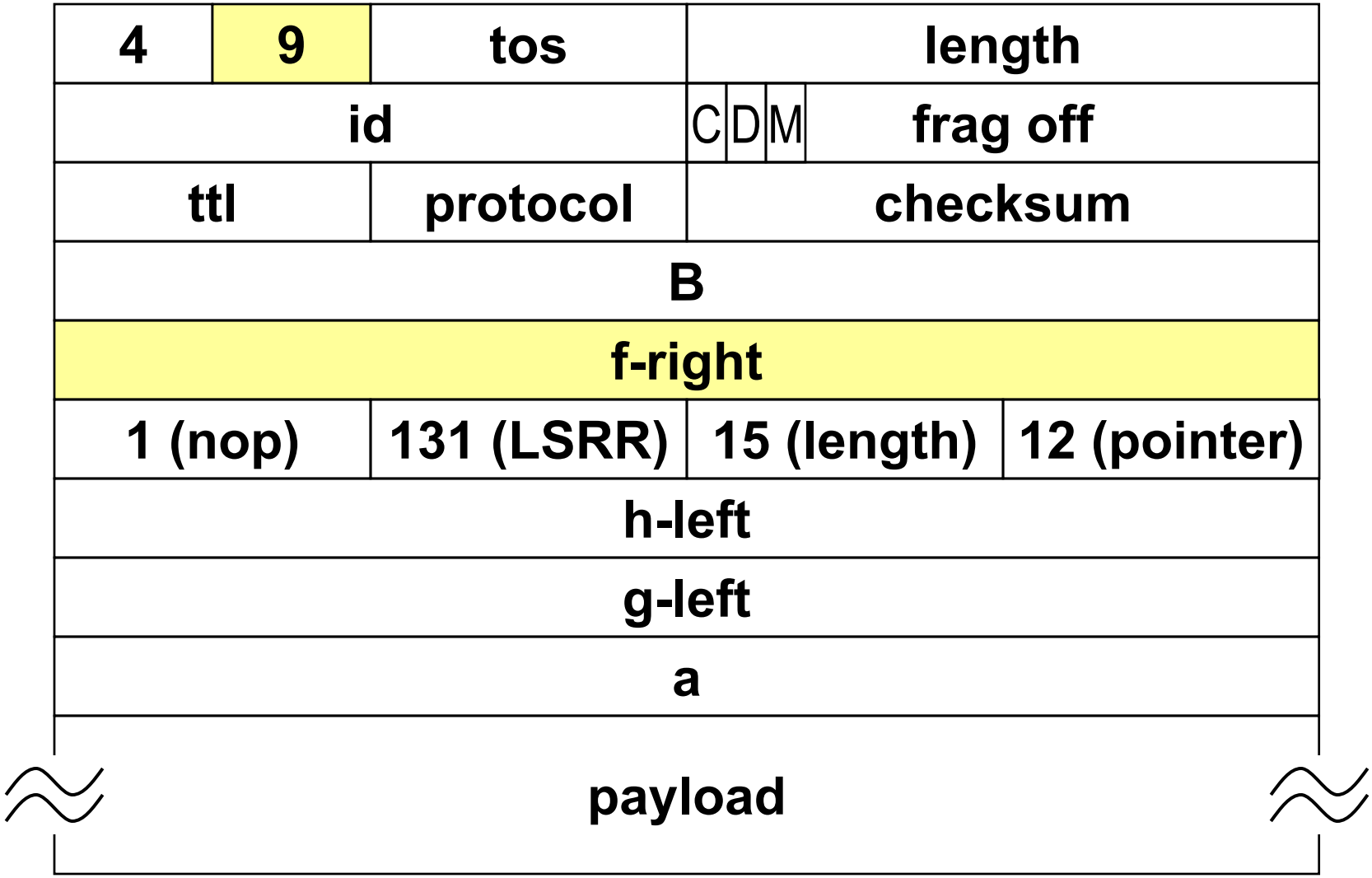
Back at H

- Otherwise, LSRR processing proceeds the same way.



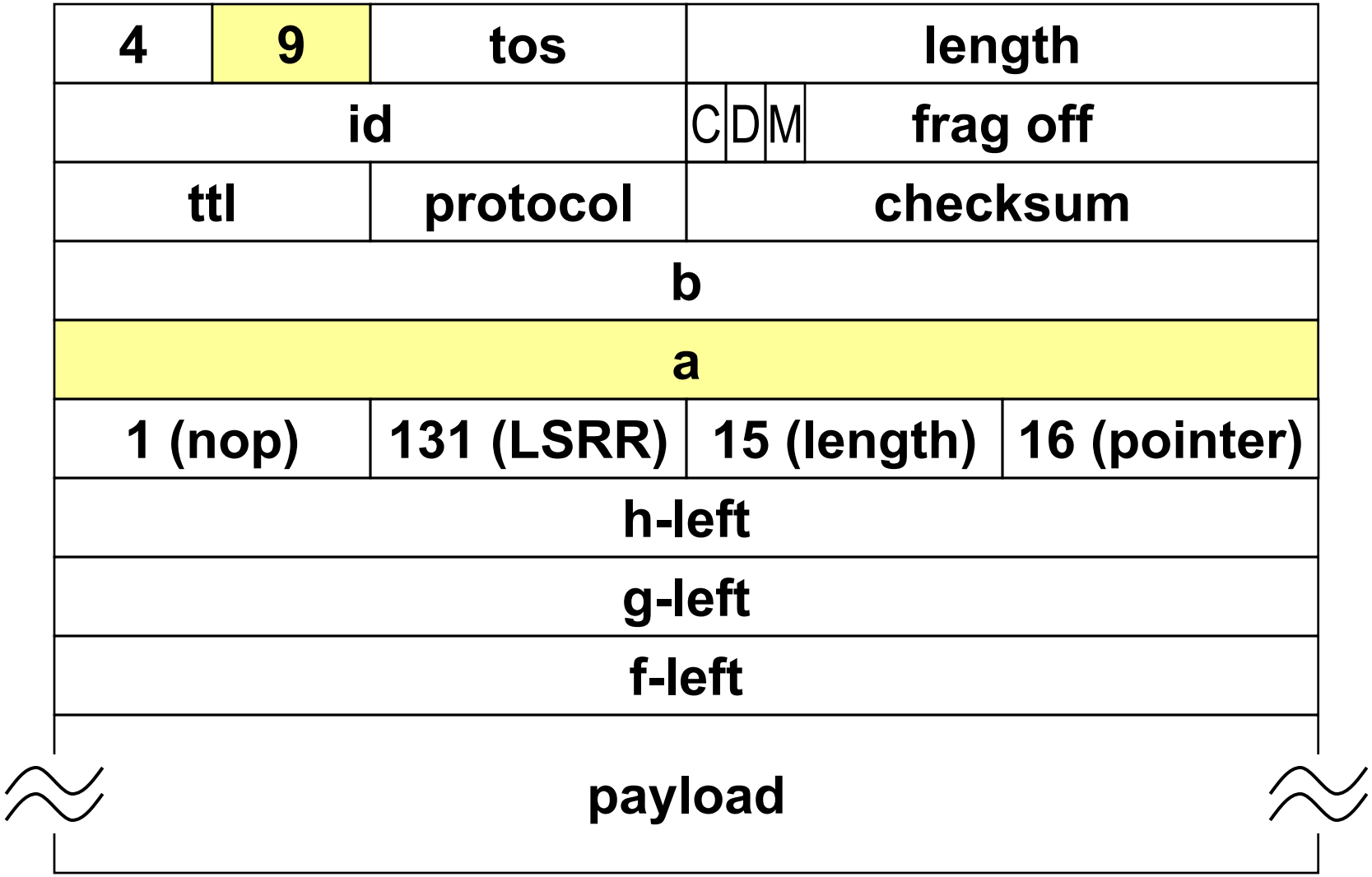
Back at G

- On to F…



Back at F

- Final hop…



Back at A again

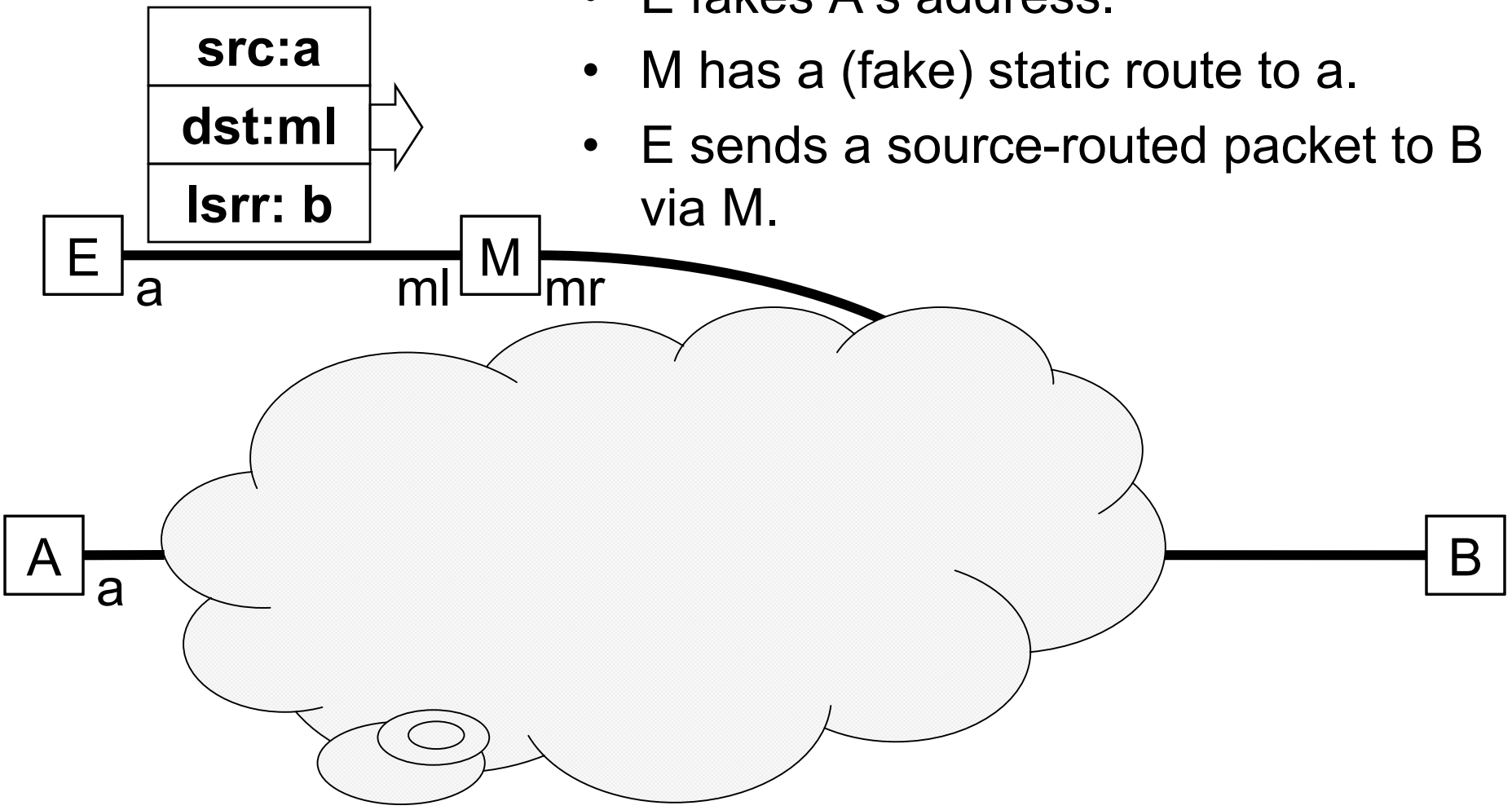
- A now gets the packet with the same LSRR it sent it out.
- We could have used SSRR, but we would have had to list (and know) **all** intermediate routers.
- There is also a RR option that just records the route.
  - Not very useful, only good for 10 hops.

Source Routing Considered Harmful

- All routers in the path (not just the ones listed in the LSRR option) are affected.
  - Routers are optimized for packets with no options.
  - If options are present, CPU has to get involved.
  - Slows things down.
- Routing systems should know best path.
  - But LSRR allows packets to go through when routing is broken.
- Source should not dictate return path to destination.
  - Return path may not even be routable.
  - But destination can always ignore it.
- Security issues.
  - But hosts shouldn't be relying on IP address for auth.

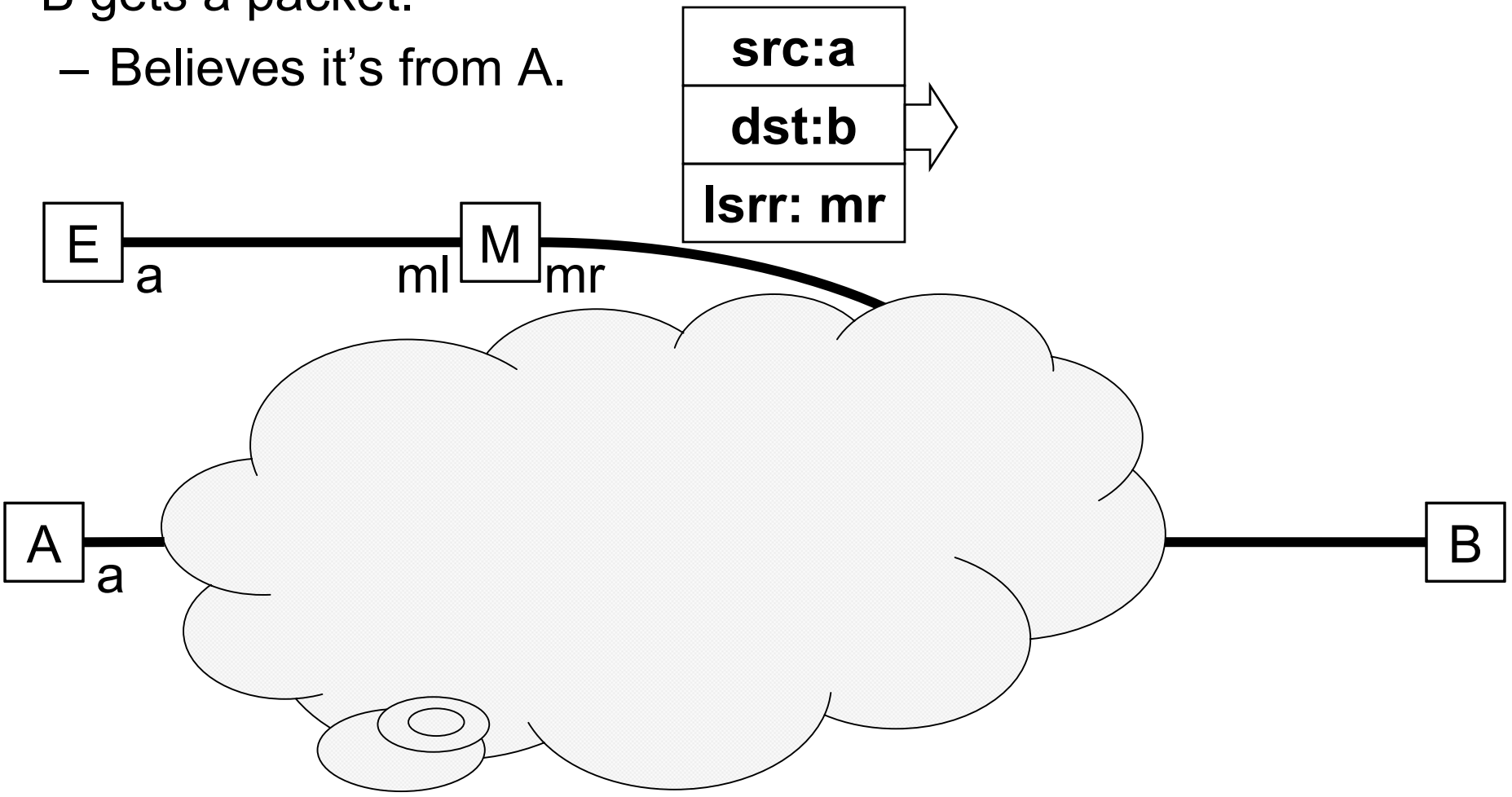
E fakes A's address

- E fakes A's address.
- M has a (fake) static route to a.
- E sends a source-routed packet to B via M.



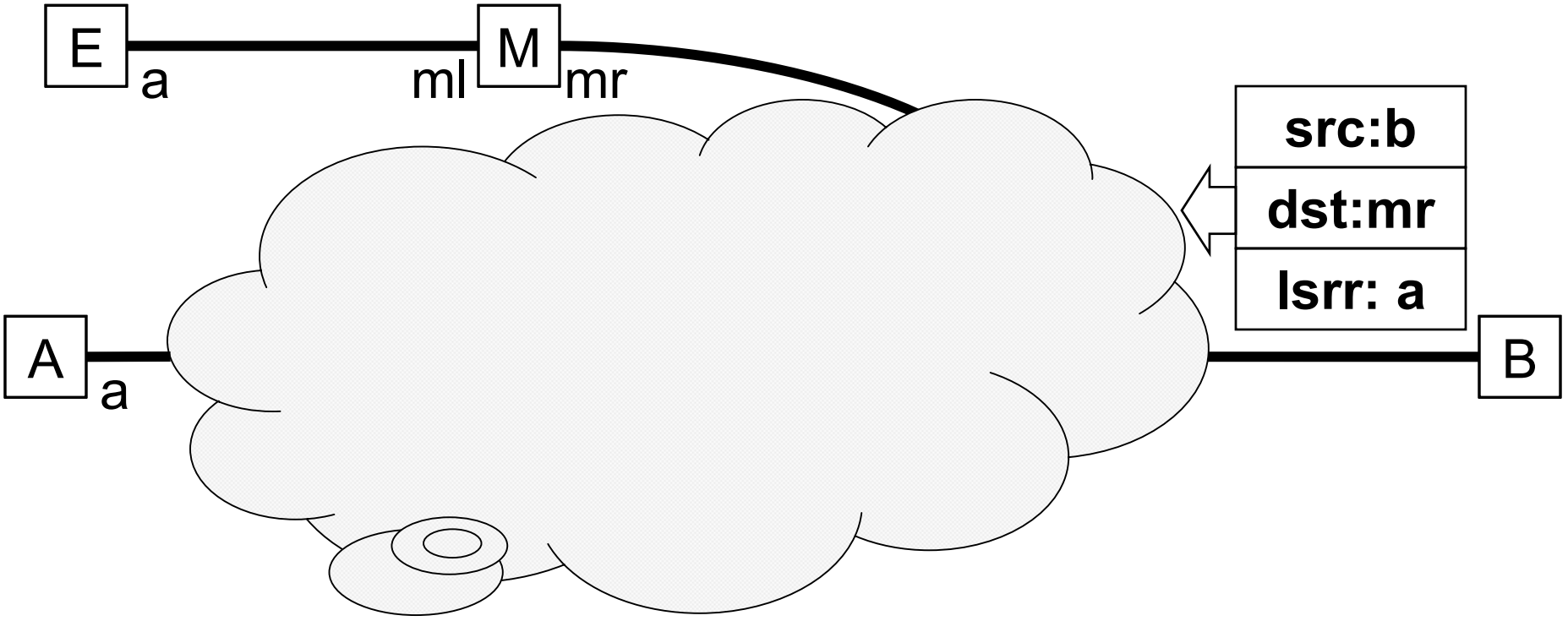
M Colludes

- M forwards the packet.
- B gets a packet.
  - Believes it's from A.



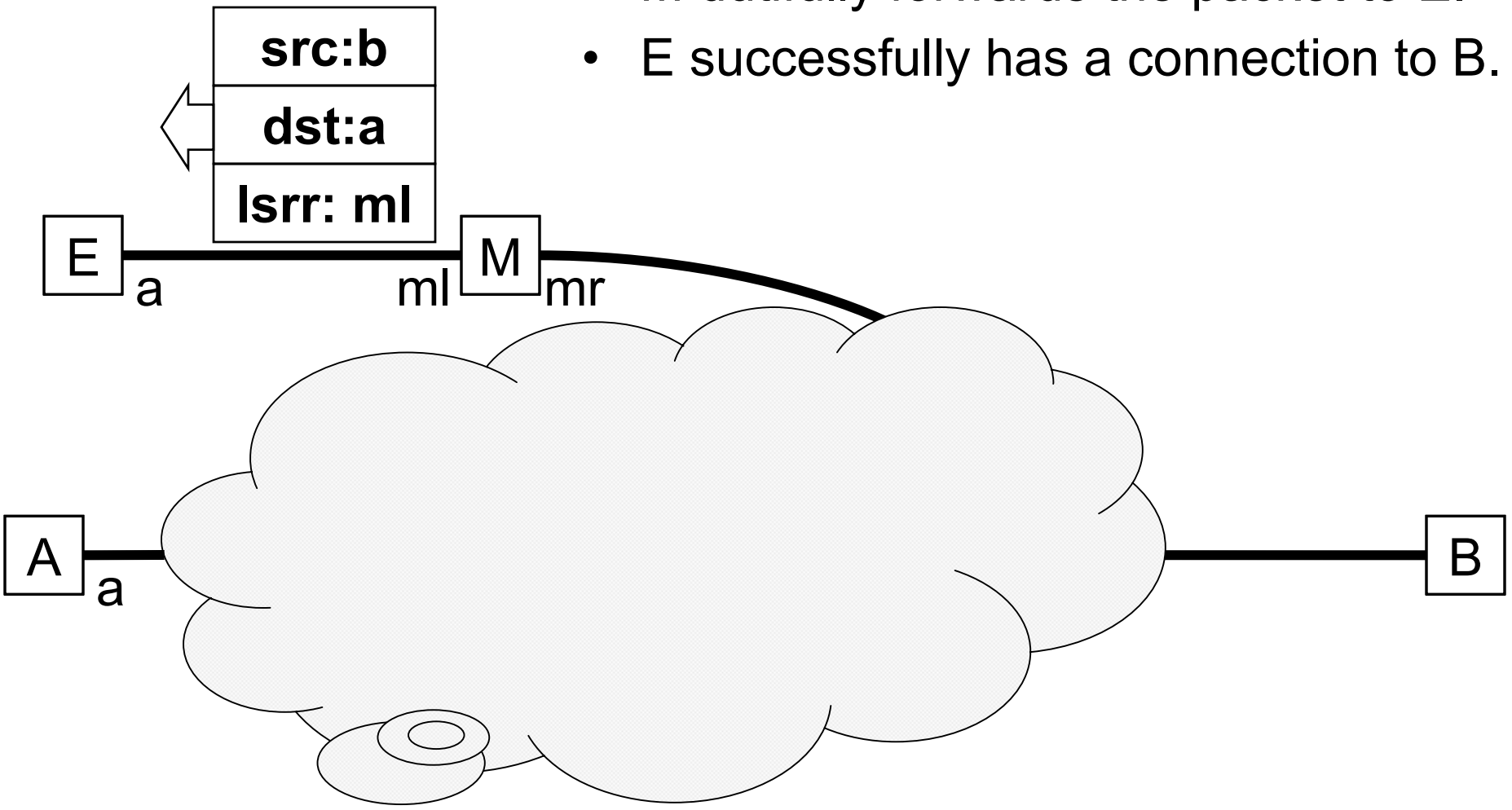
B Responds

- B reverses the LSRR
  - Thinking it's sending to A.
- Packet is source-routed to M.



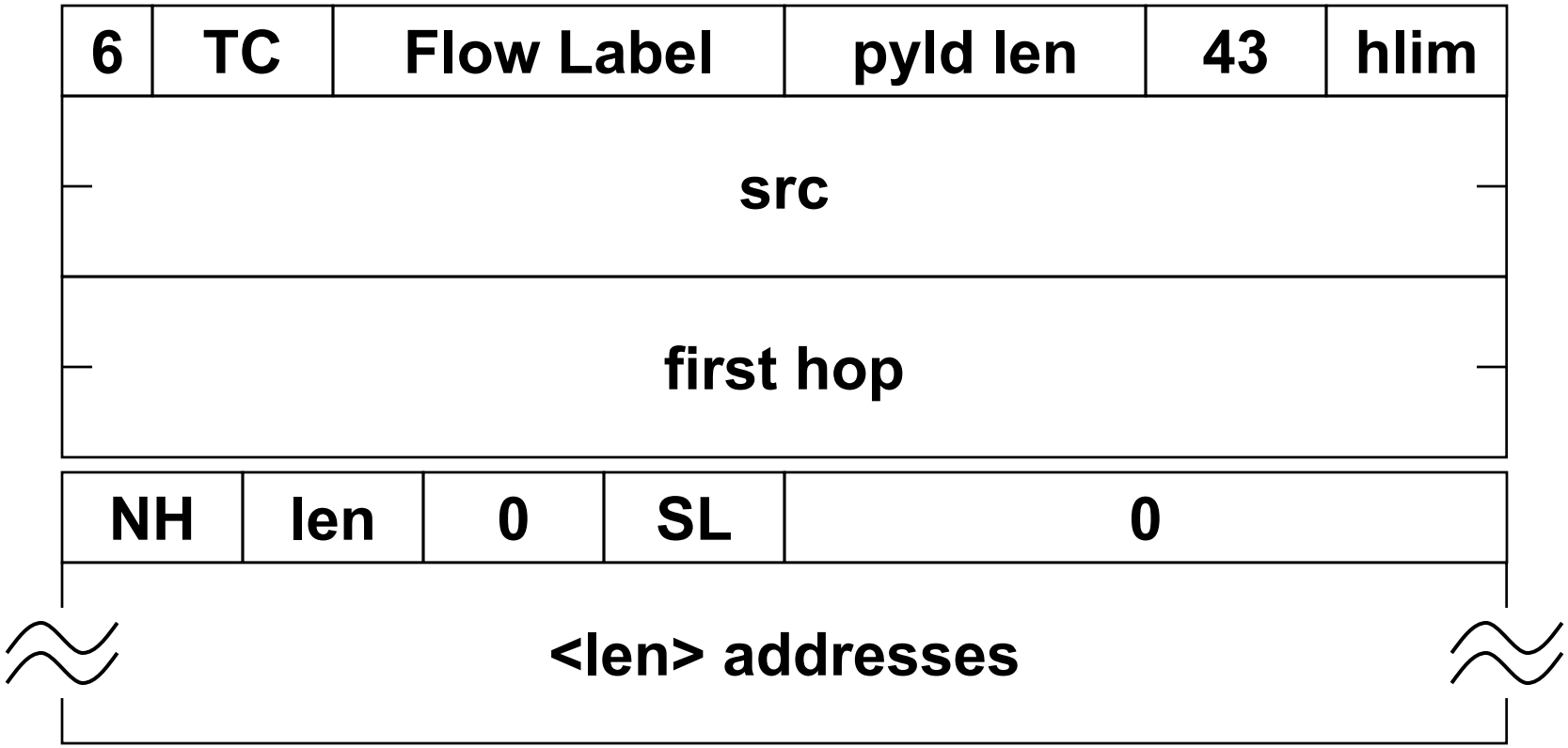
M is Still Colluding

- M dutifully forwards the packet to E.
- E successfully has a connection to B.



Source Routing in IPv6

- Routing Header (43).
- It's a routing header, so it appears directly after the IPv6 header (and any hop-by-hop options).



Tunneling

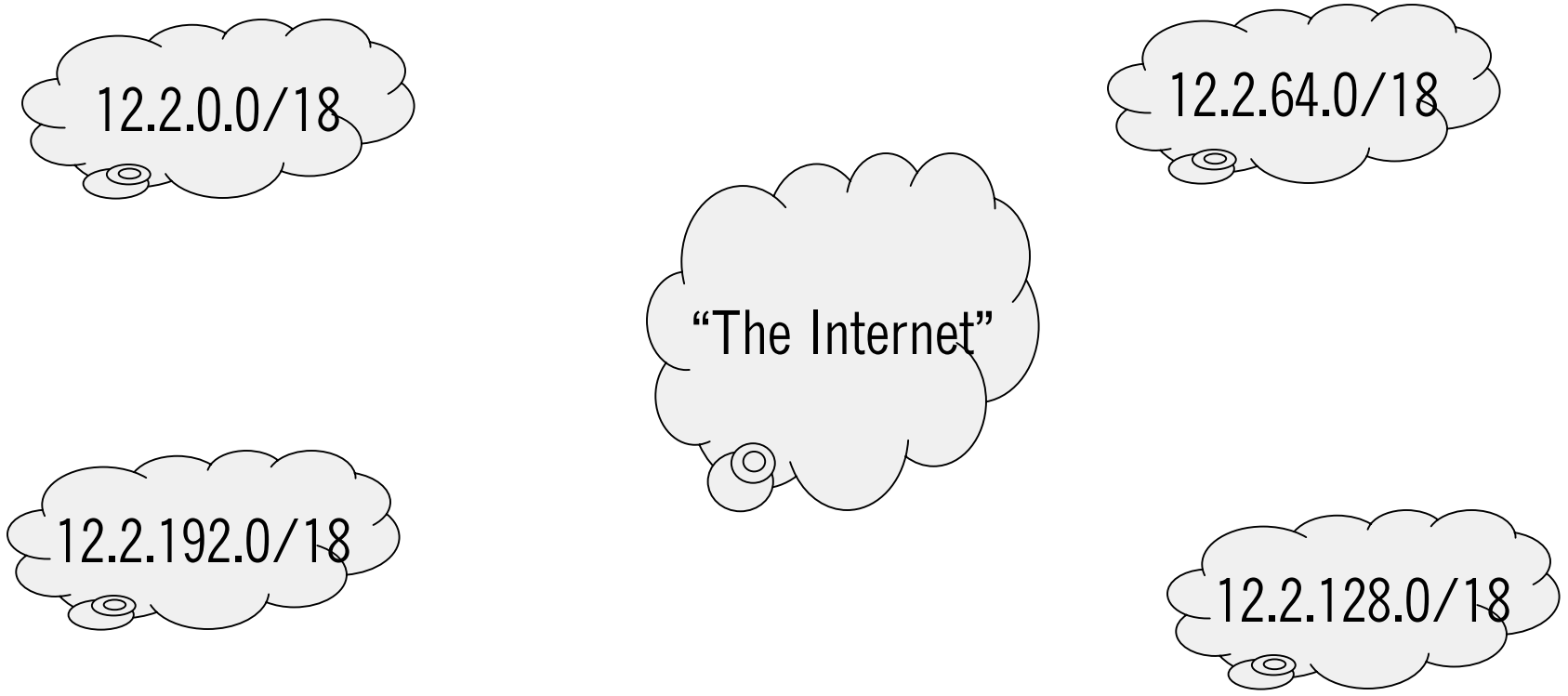
- Source routing is a form of *tunneling*.
- Tunnel: a (virtual) link between two network nodes.
  - At the same layer in the network stack.
  - Uses underlying network as virtual wire.
    - “switching fabric”.
  - Routing is circumvented.
    - In the LSRR examples, we cause traffic to follow a different path from what the underlying routing would have dictated.

Encapsulation

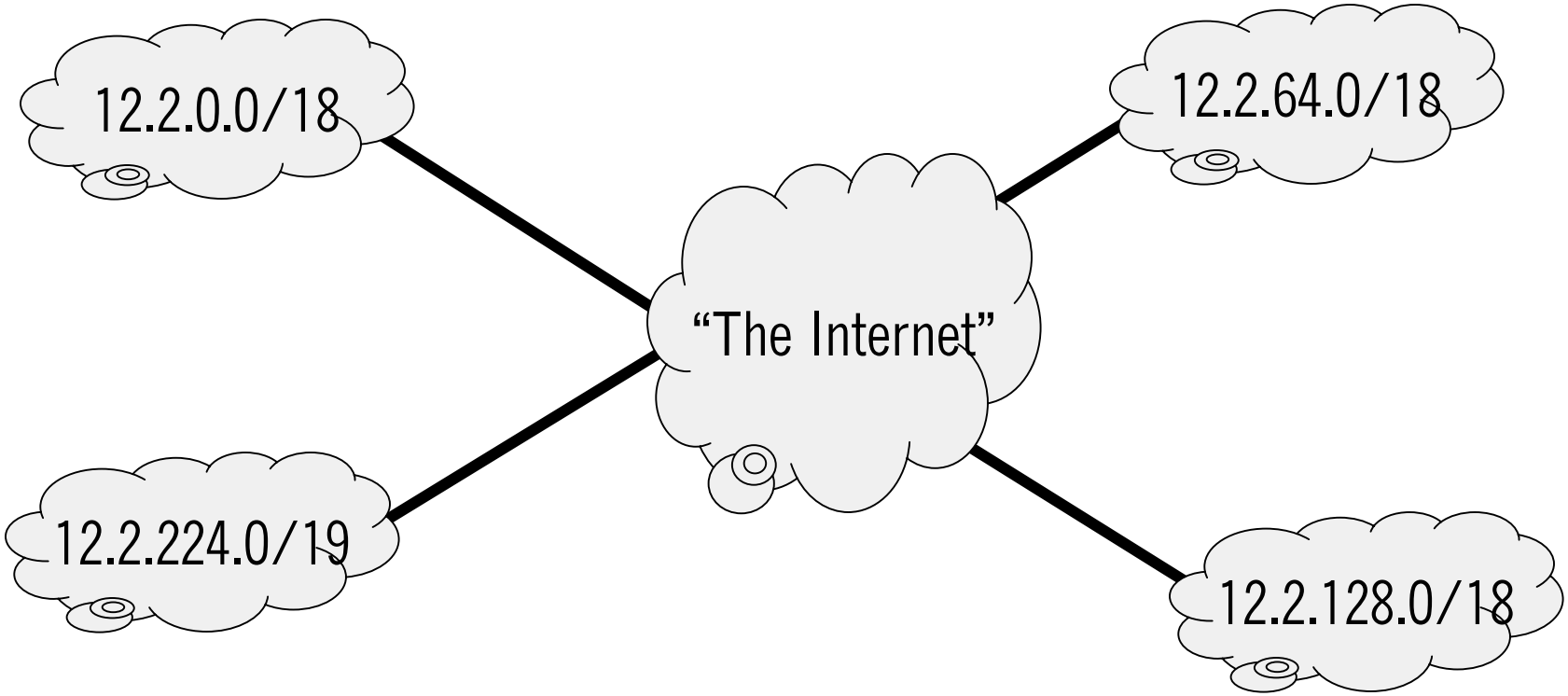
- Putting one kind of network frame inside another.
  - An IP packet is *encapsulated* in ethernet, PPP, etc. frames.
- An IP packet may be encapsulated in another IP packet.
  - IP protocol 4 (“IP-in-IP”).
  - GRE encapsulation, IPIP (proto 94) encapsulation.
- An IP packet may also be encapsulated in a higher-level protocol.
  - IP over UDP (over IP).
  - IP over HTTP (over TCP (over IP)).
- Encapsulation is another Tunneling technique.

Virtual Private Networks (VPNs)

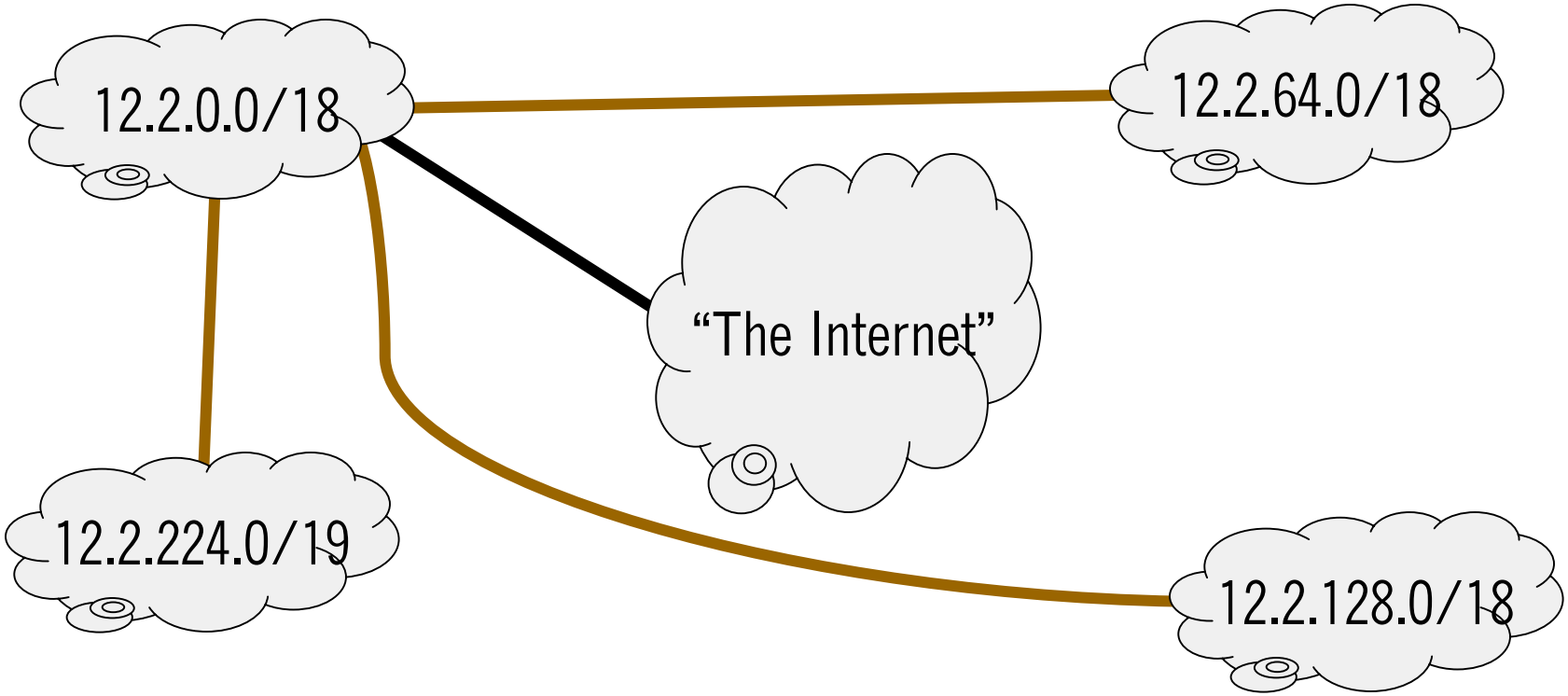
- What's a Private Network?
- "Private" as in "mine", not as in "secret".
- Consider a geographically-distributed organization.
- Various parts want to connect to each other, and potentially to the Internet.



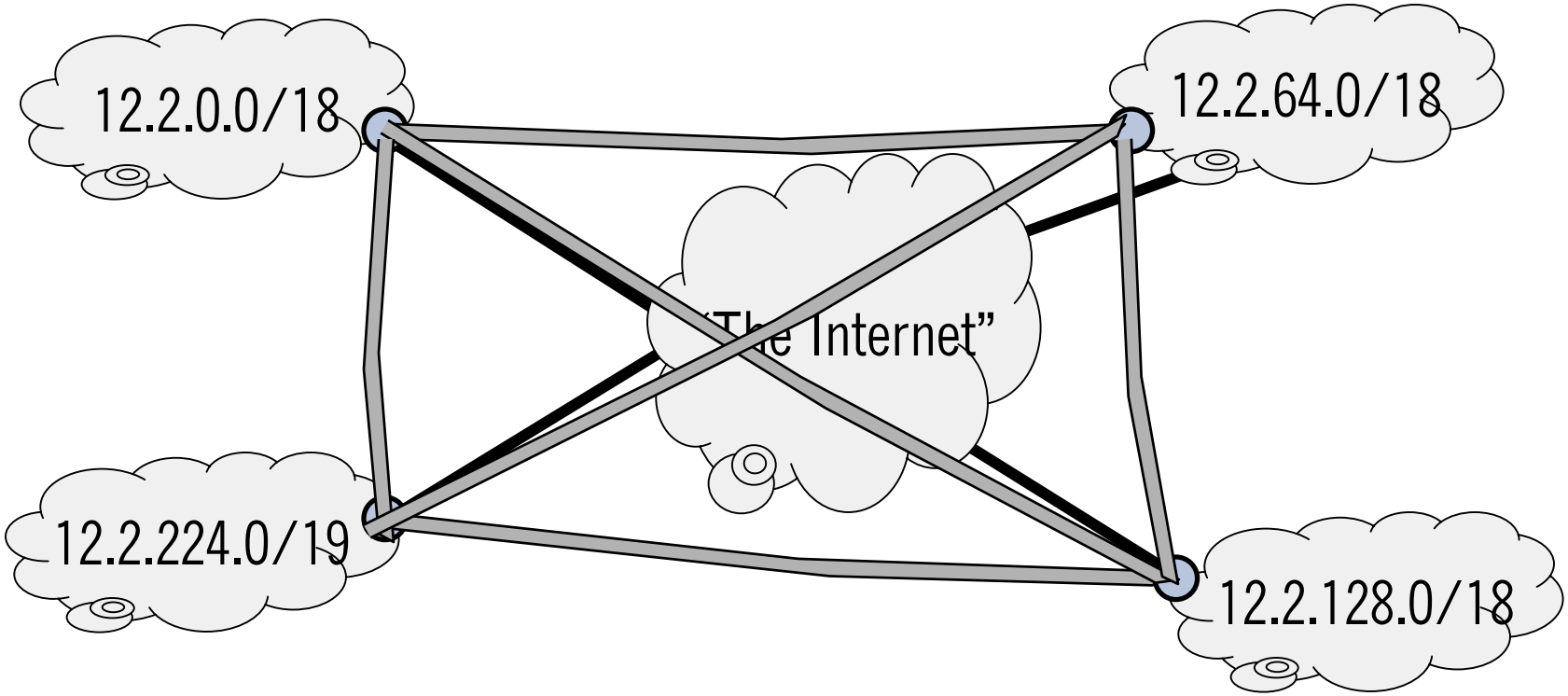
- Each part advertises its more-specific prefix.
- Pollutes the routing tables.
- Was not possible in pre-CIDR days.
- Is still not possible when using private address space.
- Lots of security implications.



- The old way:
  - Get point-to-point circuits from telco.
- The somewhat more modern way:
  - Get fixed ATM or FR virtual circuits from telco.
- Only the aggregate gets advertised to the network.
- But: high cost of point-to-point links.



- The VPN way:
- Looks like a partitioned network.
- Heal the partition with tunnels.
- (There are lots of other ways to heal the partition).



VPN Implementation

- Each *tunnel endpoint* has a "real" IP address (since the "inside" space can be private address space).
- Forwarding rules on each tunnel endpoint.
  - Look like routes.
  - Usually implemented with a virtual interface.
  - "If destination prefix is `foo/n` then tunnel via `bar`.
- Encapsulation can be as simple as IP in IP (protocol 4).
  - "Inner header" *vs*. "outer header".
- GRE encapsulation (RFC1701 and 1702).
- IPsec encapsulation (RFC2401-series RFCs).
- Other forms of IP-in-foo encapsulation.

Problems with IP Encapsulation

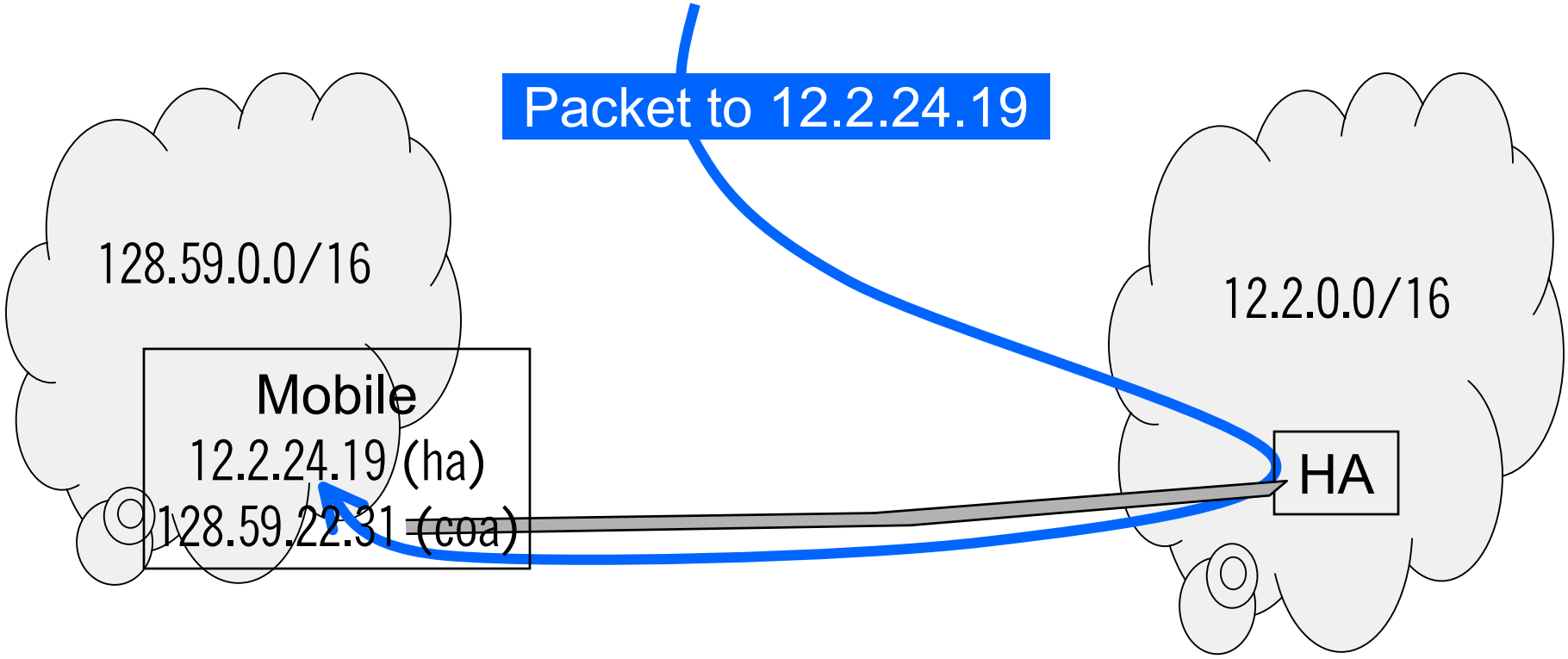
- Max IP packet size and Path MTU:
  - Encapsulation adds an extra header to the packet.
  - Max packet size gets reduced by that amount.
    - Not usually a problem (no one sends 65K datagrams).
  - As does PMTU.
  - Hosts should be running PMTU discovery.
  - Or interfaces should be configured with lower MTUs.
- Who gets ICMP messages?
  - If generated while traversing the tunnel.
  - If generated by the tunnel endpoints.
  - If generated by the end hosts.
- What to do with TOS, DF, TTL.

Overlay Networks

- Any network at layer-n that (recursively) uses layer-n for forwarding.
  - Uses existing network as a *switching fabric* (a virtual link-layer).
  - Bypasses existing routing.
- Examples:
  - VPNs.
  - Multicast.
  - Mobile IP.
  - Peer-to-peer networks (these can also be at higher layers).
  - Route traffic around faults, congestion, etc.

Overlay Networks cont'd

- Overlay networks are used when a different forwarding path is required from the one provided by the underlying routing.
- This is functionally equivalent to source routing.
  - Obviously!
- Example: Mobile IP



Overlay Networks Extend Routing

- Tunneling can also be used to create new routing infrastructures out of existing ones.
- Examples:
  - M-BONE (Multicast backbone).
  - 6BONE (IPv6 backbone).

MBONE

- Multicast was not natively supported by routing protocols in 1992.
- Most providers still don't have it turned on.
- MBONE is a collection of routers that forward multicast traffic.
  - "Links" between these routers are tunnels.
  - They have their own routing protocol.
    - "Next Hop" is really next router in the overlay network.
- "Outside" (tunnel) header src and dst addresses are those of the sending and receiving mbone routers.
- "Inside" source address is the (unicast) address of the originator.
- "Inside" destination address is the multicast destination address.

MBONE cont'd

- Each MBONE router receives a (tunneled) packet.
- Strips the encapsulation.
- Decides which tunnels to forward the packet to.
  - (This is multicast, there will usually be more than one).
- Re-encapsulates the packet for each destination router.

- If the router has (local) hosts in the multicast group, it also sends a copy of the (multicast) packet to the corresponding link-layer multicast address.

- If the router sees a multicast destination address on one of its interfaces, it picks it up and forwards it.

6BONE

- Same idea, for IPv6.
- Couldn’t wait for providers to offer natively-routed IPv6.
- Preconfigured tunnels.
- “Outer” (tunnel) header is IPv4.
- “Inner” header is IPv6.
- Allocated 3ffe::/15, further allocations from there.
- Runs multiprotocol extensions to BGP.
- IPv4 links look like link-layer links.

Six-to-four

- Automatic way to tunnel v6 in v4.
- First sixteen bits are (hex) 2002.
- Next 32 bits show (ipv4) tunnel endpoint.

- Packets from 2002:803b:1012::/48 to 2002:c014:e10f::/48 get tunneled from 128.59.16.18 to 192.20.225.15.