

# IP Forwarding

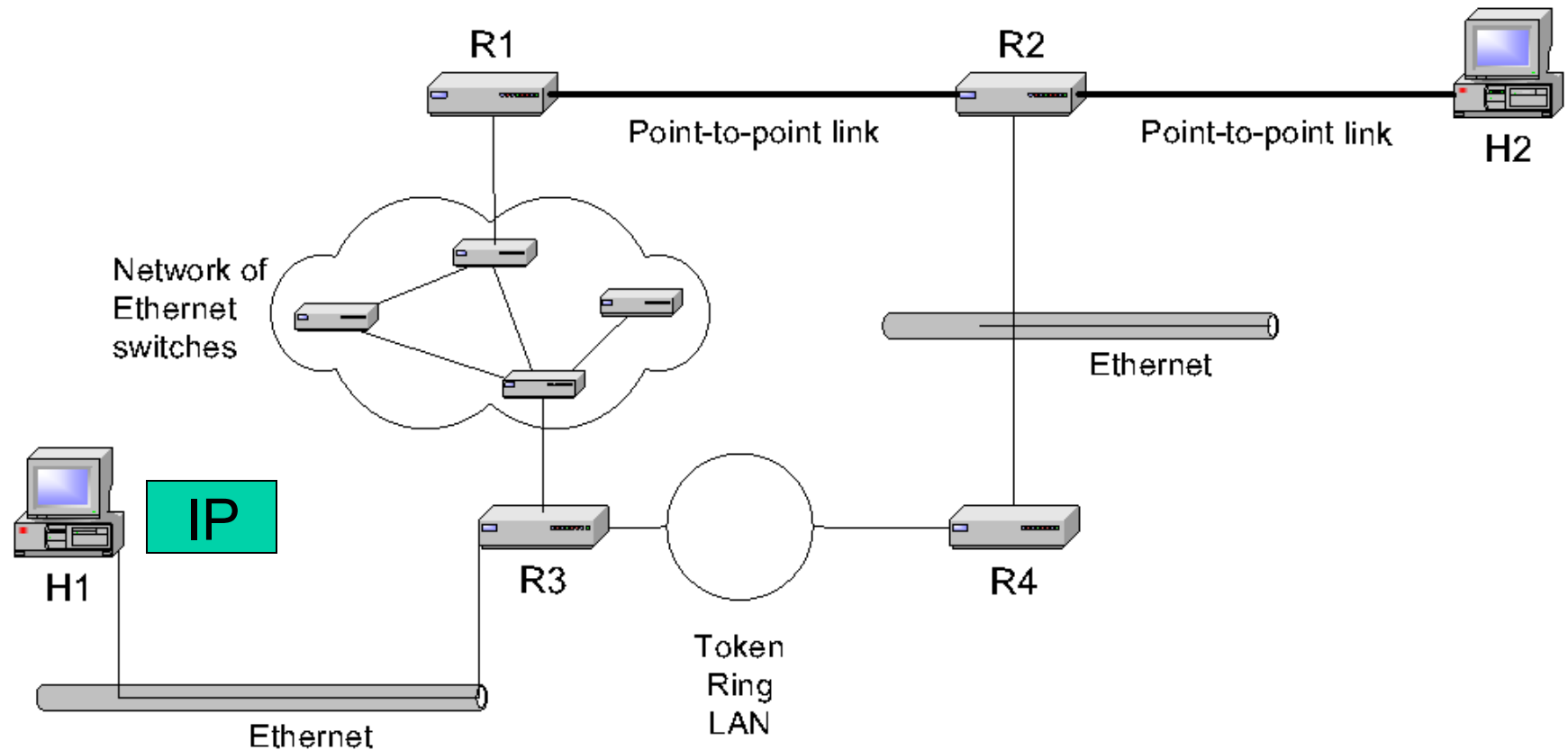
---

**Relates to Lab 3.**

Covers the principles of end-to-end datagram delivery in IP networks.

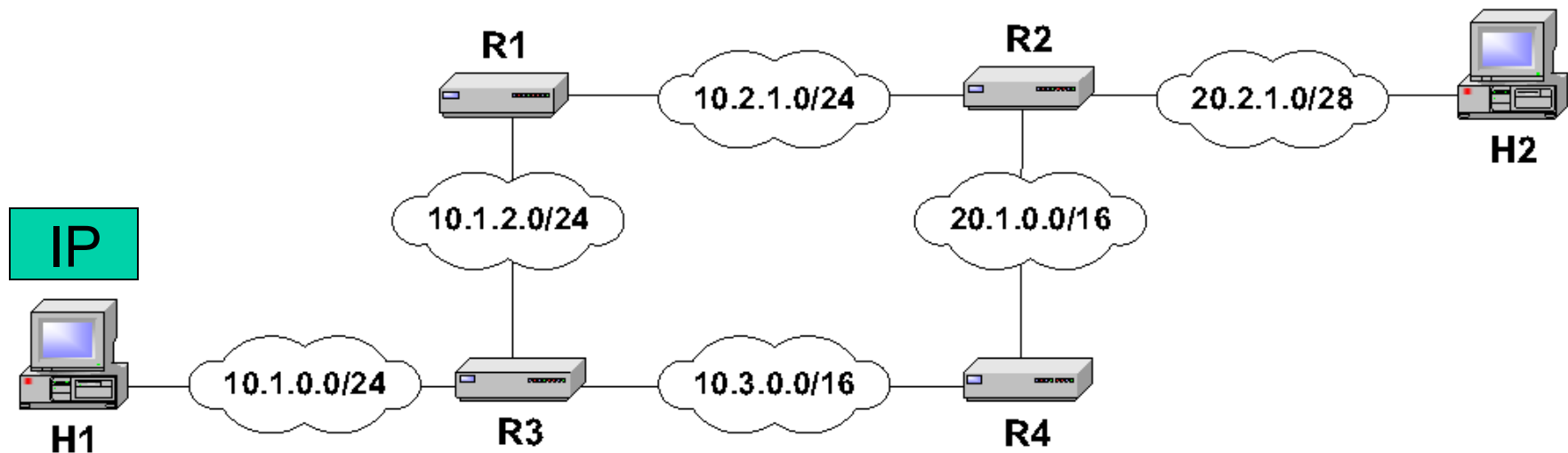
# Delivery of an IP datagram

- View at the data link layer layer:
  - Internetwork is a collection of LANs or point-to-point links or switched networks that are connected by routers



# Delivery of an IP datagram

- View at the IP layer:
  - An IP network is a logical entity with a network number
  - The IP delivery service thinks of IP networks as “clouds”, ignoring the data link layer view



# Tenets of end-to-end delivery of datagrams

---

The following conditions must hold so that an IP datagram can be successfully delivered


1. The network prefix of an IP destination address must correspond to a unique data link layer network (=LAN or point-to-point link or switched network).  
(The reverse need not be true!)
2. Routers and hosts that have a common network prefix must be able to exchange IP datagrams using a data link protocol (e.g., Ethernet, PPP)
3. A sequence of alternating data link layers and routers must exist from the source to the destination.

# Routing tables

- Each router and each host keeps a **routing table** which tells the router how to process an outgoing packet, i.e., take it closer to destination
- Main columns:
  1. **Destination address:** where is the IP datagram going to?
  2. **Next hop:** how to send the IP datagram?
  3. **Interface:** what is the output port?

Routing table of a host or router

IP datagrams can be directly delivered (“direct”) or is sent to a router (“R4”)



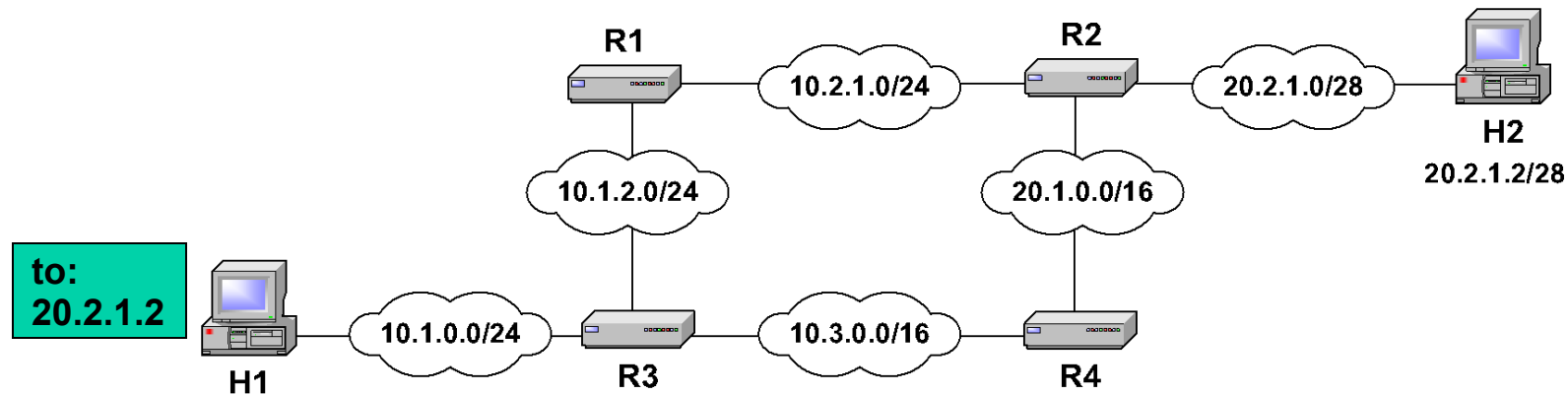
Destination	Next Hop	interface
10.1.0.0/24	direct	eth0
10.1.2.0/24	direct	eth0
10.2.1.0/24	R4	serial0
10.3.1.0/24	direct	eth1
20.1.0.0/16	R4	eth0
20.2.1.0/28	R4	eth0

# Delivery with routing tables

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
20.2.0.0/16	R2
30.1.1.0/28	R2

Destination	Next Hop
10.1.0.0/24	R1
10.1.2.0/24	R1
10.2.1.0/24	direct
10.3.1.0/24	R4
20.1.0.0/16	direct
20.2.1.0/28	direct

Destination	Next Hop
10.1.0.0/24	R2
10.1.2.0/24	R2
10.2.1.0/24	R2
10.3.1.0/24	R2
20.1.0.0/16	R2
20.2.1.0/28	direct



Destination	Next Hop
10.1.0.0/24	direct
10.1.2.0/24	R3
10.2.1.0/24	R3
10.3.1.0/24	R3
20.1.0.0/16	R3
20.2.1.0/28	R3

Destination	Next Hop
10.1.0.0/24	direct
10.1.2.0/24	direct
10.2.1.0/24	R4
10.3.1.0/24	direct
20.1.0.0/16	R4
20.2.1.0/28	R4

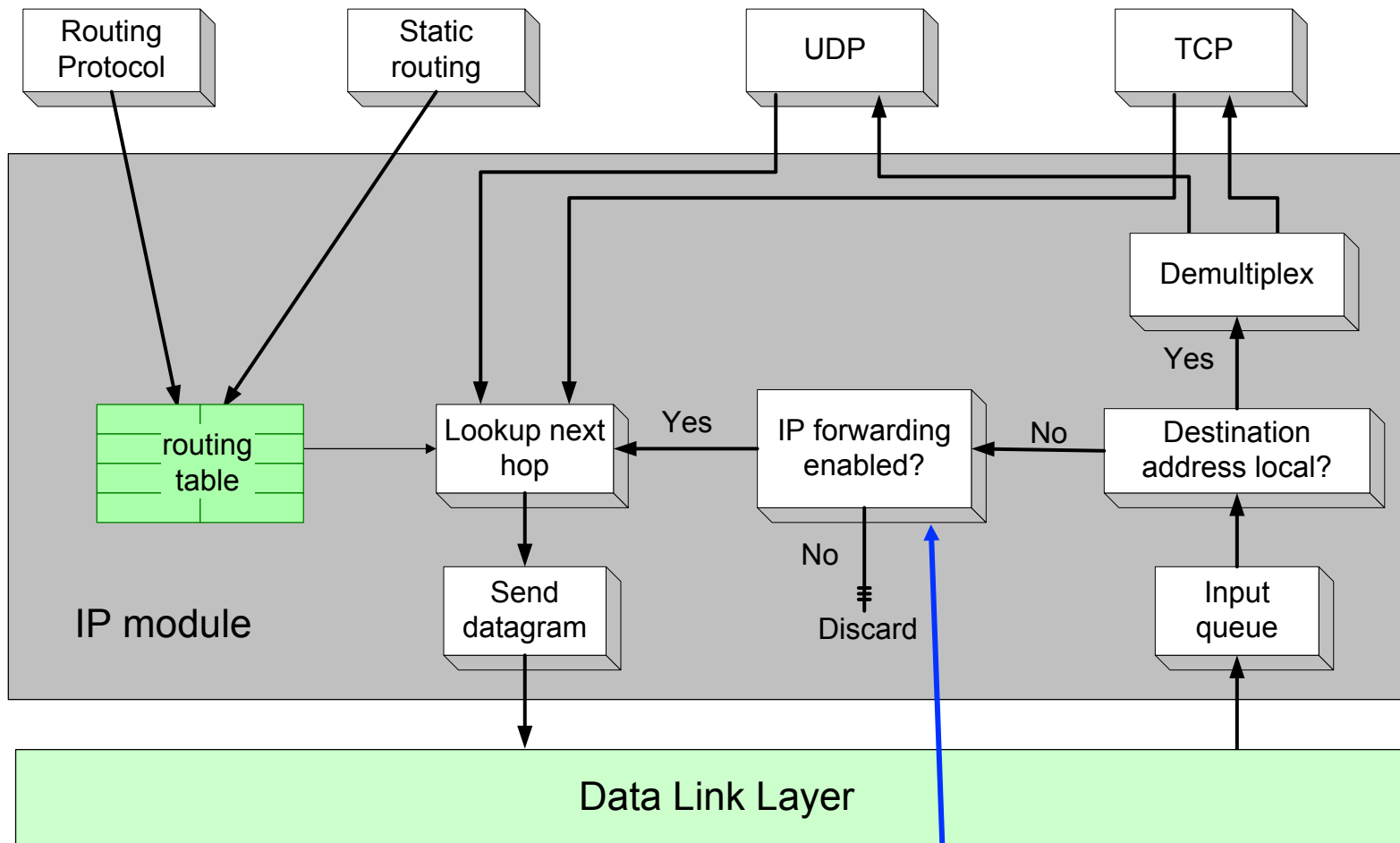
Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	R3
10.2.1.0/24	R2
10.3.1.0/24	direct
20.1.0.0/16	direct
20.2.1.0/28	R2

# Delivery of IP datagrams

---

- There are two distinct processes to delivering IP datagrams:
  1. **Forwarding:** How to pass a packet from an input interface to the output interface?
  2. **Routing:** How to find and setup the routing tables?
- Forwarding must be done as fast as possible:
  - on routers, is often done with support of hardware
  - on PCs, is done in kernel of the operating system
- Routing is less time-critical
  - On a PC, routing is done as a background process

# Processing of an IP datagram in IP



IP router: IP forwarding enabled  
Host: IP forwarding disabled 8



# Processing of an IP datagram in IP

---

- Processing IP datagrams very similar on IP router and host
  - **Main difference: “IP forwarding” is enabled on router and disabled on host by default**
- **IP forwarding enabled**
  - if a datagram is received, but it is not for the local system, the datagram will be sent to a different system
- **IP forwarding disabled**
  - if a datagram is received, but it is not for the local system, the datagram will be dropped

# Processing of an IP datagram at a router

---

Receive an  
IP datagram



1. IP header validation
2. Process options in IP header
3. Parsing the destination IP address
4. Routing table lookup
5. Decrement TTL
6. Perform fragmentation (if necessary)
7. Calculate checksum
8. Transmit to next hop
9. Send ICMP packet (if necessary)

# Routing table lookup

---

**Routing table lookup:** Use the IP destination address as a key to search the routing table.

<b>Destination address</b>	<b>Next hop/ interface</b>
network prefix <i>or</i> host IP address <i>or</i> loopback address <i>or</i> default route	IP address of next hop router  <i>or</i>  Name of a network interface

# Type of routing table entries

---

- **Network route**
  - Destination addresses is a network address (e.g., 10.0.2.0/24)
  - Most entries are network routes
- **Host route**
  - Destination address is an interface address (e.g., 10.0.1.2/32)
  - Used to specify a separate route for certain hosts
- **Default route**
  - Used when no network or host route matches
  - The router that is listed as the next hop of the default route is the **default gateway (for Cisco: “gateway of last resort)**
- **Loopback address**
  - Routing table for the loopback address (127.0.0.1)
  - The next hop lists the loopback (lo0) interface as outgoing interface

# Routing table lookup: Longest Prefix Match

- **Longest Prefix Match:** Search for the routing table entry that has the longest match with the prefix of the destination IP address

1. Search for a match on all 32 bits
2. Search for a match for 31 bits
- .....
32. Search for a match on 0 bits

Host route, loopback entry  
→ 32-bit prefix match

Default route is represented as 0.0.0.0/0  
→ 0-bit prefix match

128.143.71.21



Destination address	Next hop
10.0.0.0/8	R1
128.143.0.0/16	R2
128.143.64.0/20	R3
128.143.192.0/20	R3
128.143.71.0/24	R4
128.143.71.55/32	R3
default	R5



**The longest prefix match for 128.143.71.21 is for 24 bits with entry 128.143.71.0/24**

**Datagram will be sent to R4**<sup>13</sup>

# Route Aggregation

- Longest prefix match algorithm permits to aggregate prefixes with identical next hop address to a single entry
- This contributes significantly to reducing the size of routing tables of Internet routers

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
20.2.0.0/16	R2
30.1.1.0/28	R2



Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
20.0.0.0/8	R2

# How do routing tables get updated?

---

- Adding an interface:
  - Configuring an interface eth2 with 10.0.2.3/24 adds a routing table entry:

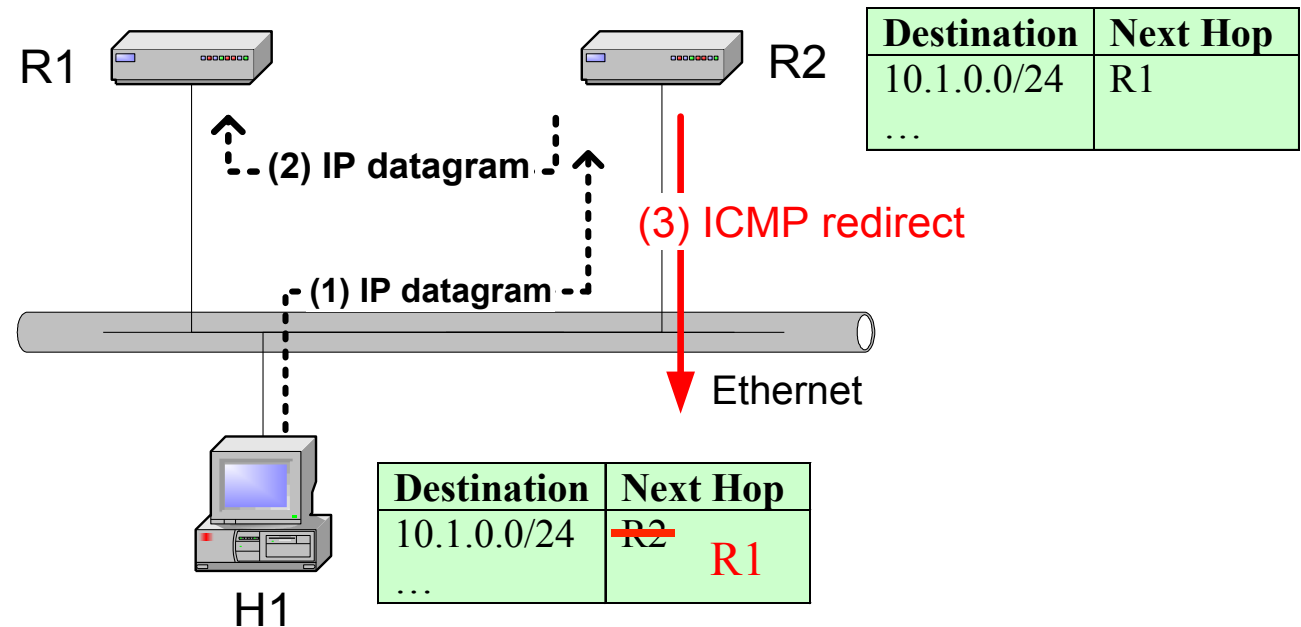
<b>Destination</b>	<b>Next Hop/ interface</b>
10.0.2.0/24	eth2

- Adding a default gateway:
  - Configuring 10.0.2.1 as the default gateway adds the entry:
- Static configuration of network routes or host routes
- Update of routing tables through routing protocols
- ICMP messages

<b>Destination</b>	<b>Next Hop/ interface</b>
0.0.0.0/0	10.0.2.1

# Routing table manipulations with ICMP

- When a router detects that an IP datagram should have gone to a different router, the router (here R2)
  - forwards the IP datagram to the correct router
  - sends an ICMP redirect message to the host
- Host uses ICMP message to update its routing table

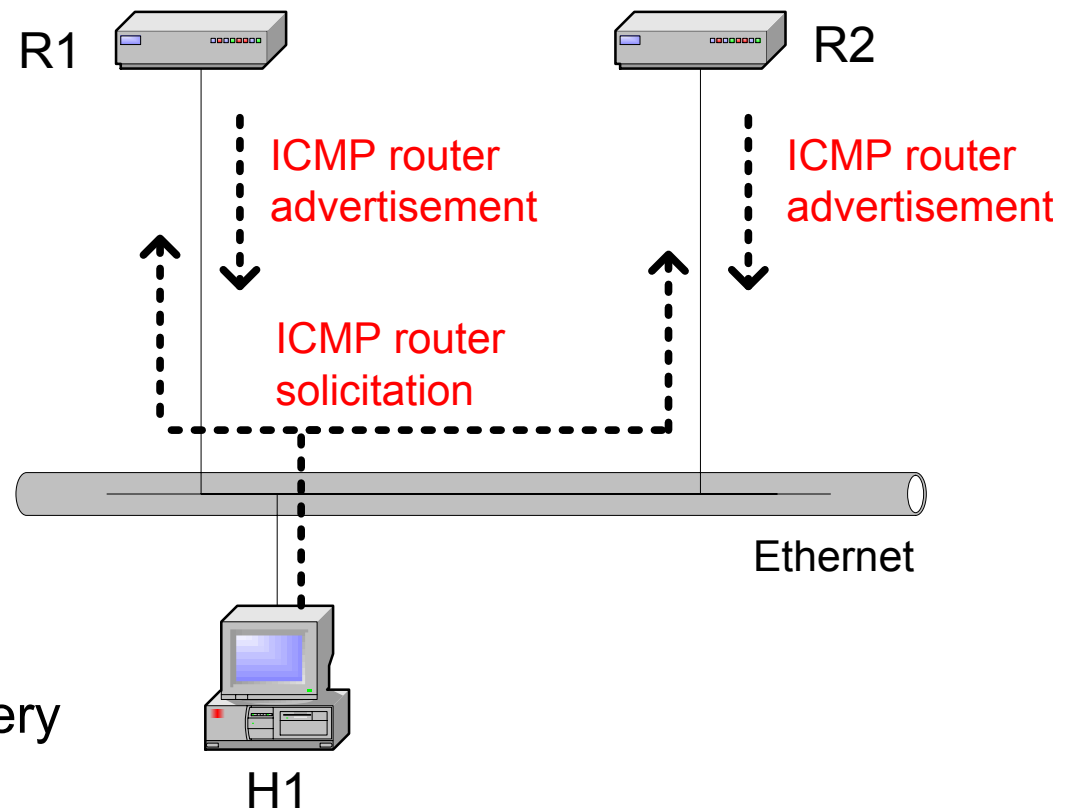




# ICMP Router Solicitation

## ICMP Router Advertisement

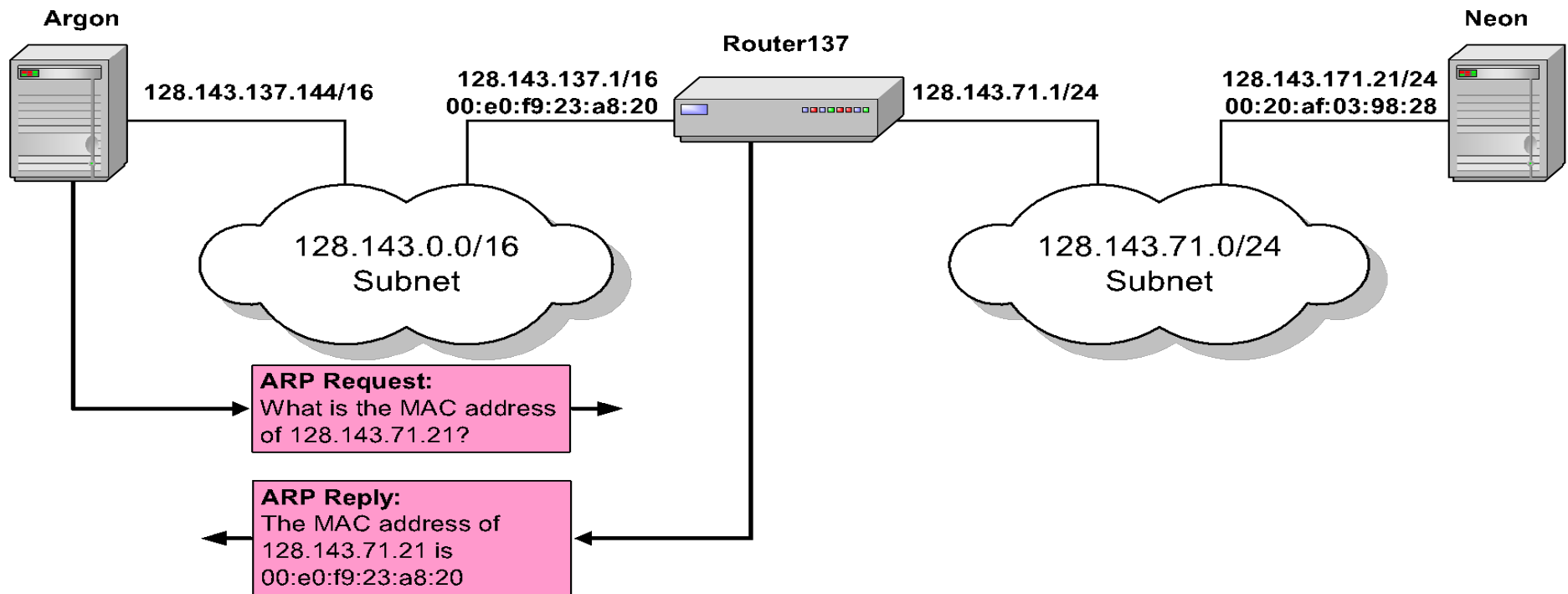
- After bootstrapping, a host broadcasts an **ICMP router solicitation**.
- In response, routers send an **ICMP router advertisement** message
- Routers also periodically broadcast **ICMP router advertisement**



Also called the Router Discovery Protocol

# Proxy ARP

- **Proxy ARP:** Host or router responds to ARP Request that arrives from one of its connected networks for a host that is on another of its connected networks.



# Things to know about ARP

---

- What happens if an ARP Request is made for a non-existing host?

Several ARP requests are made with increasing time intervals between requests. Eventually, ARP gives up.

- On some systems (including Linux) a host periodically sends ARP Requests for all addresses listed in the ARP cache. This refreshes the ARP cache content, but also introduces traffic.
- **Gratuitous ARP Requests:** A host sends an ARP request for its own IP address:
  - Useful for detecting if an IP address has already been assigned.

# Vulnerabilities of ARP

---

1. **No authentication:** Since ARP does not authenticate requests or replies, ARP Requests and Replies can be forged
2. **Stateless:** ARP Replies can be sent without a corresponding ARP Request
  - According to the ARP protocol specification, a node receiving an ARP packet (Request or Reply) must update its local ARP cache with the information in the source fields, if the receiving node already has an entry for the IP address of the source in its ARP cache. (This applies for ARP Request packets and for ARP Reply packets)

Example exploitation of these vulnerabilities:

- A forged ARP Request or Reply can be used to update the ARP cache of a remote system with a forged entry (**ARP Poisoning**)
- This can be used to redirect IP traffic to other hosts