Computer Networking

COMP 177 | Fall 2020 | University of the Pacific | Jeff Shafer

Internet Control Message Protocol (ICMP)

Recap

Past Topics

- Overview of networking and layered architecture
- Wireshark packet sniffer
- Ethernet, WiFi, VLANs
- IPv4
- ARP

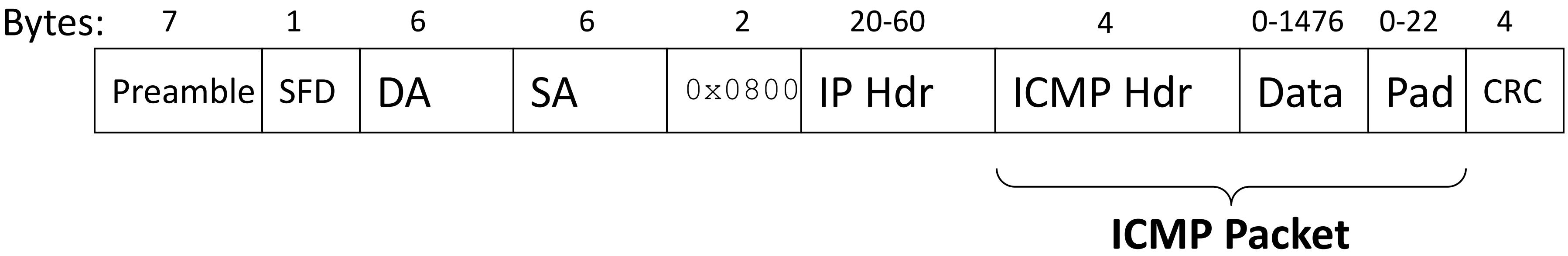
Today's Topics

- Internet Control Message Protocol (ICMP)

Internet Control Message Protocol

- One of the core protocols in the Internet
- Primarily used to communicate errors among routers and hosts
  - IP datagram errors
  - Communicate routing information/errors
  - Communicate diagnostics
- Not (typically) used by applications
  - Applications communicate application-level errors using higher level protocols
  - Ping and traceroute are the exceptions

ICMP Packets



Bytes:	7	1	6	6	2	20-60	4	0-1476	0-22	4
	Preamble	SFD	DA	SA	`0x0800`	IP Hdr	ICMP Hdr	Data	Pad	CRC

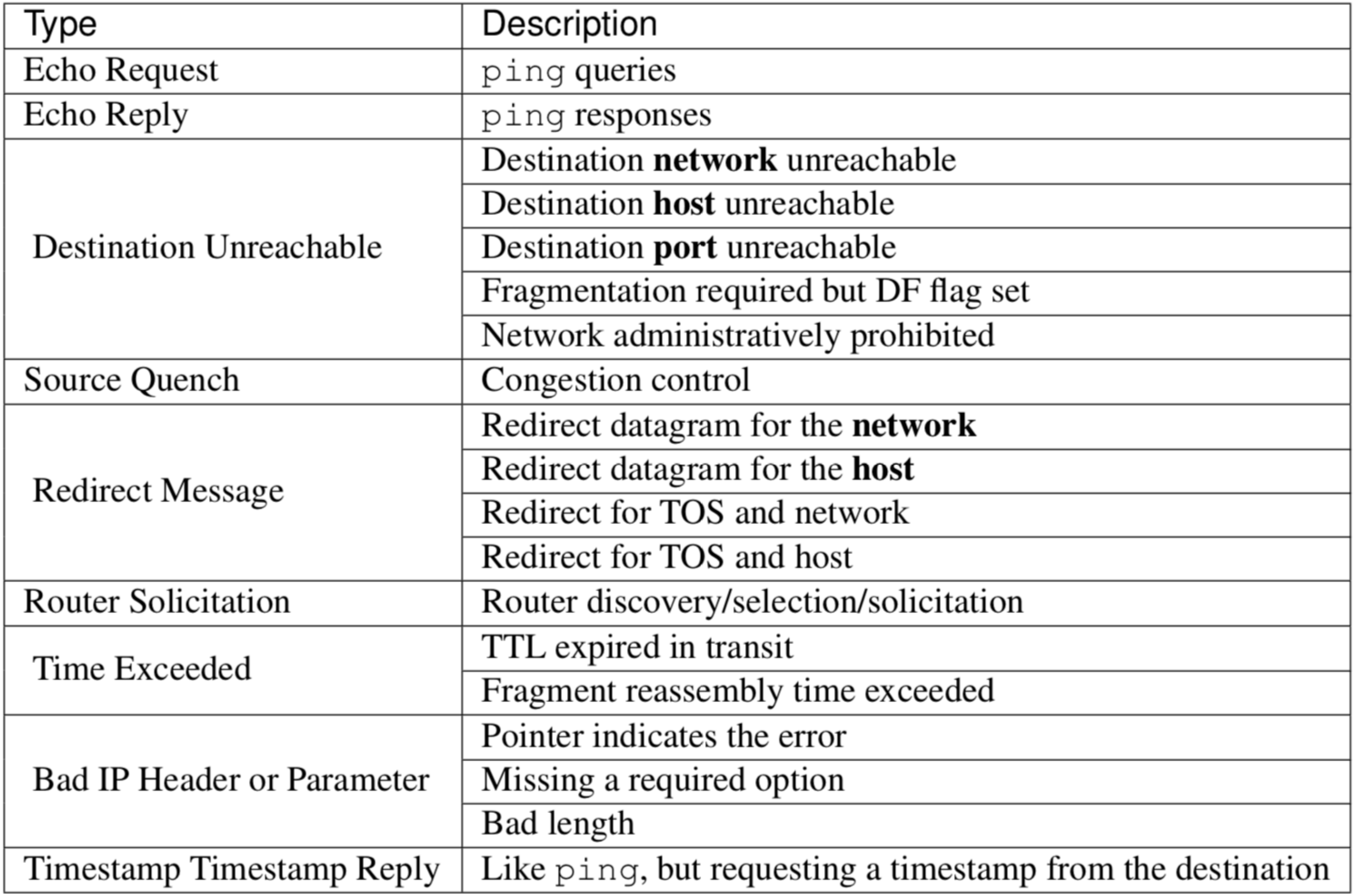
ICMP Packet

- ICMP packets are encapsulated in IP datagrams
  - IP protocol field: ICMP (`0x01`)
- Header fields
  - Type (1 byte)
  - Code (1 byte)
  - Checksum (2 bytes)

ICMP in IP in Ethernet

<table>
<tr><td colspan="8">Destination MAC Address</td></tr>
<tr><td colspan="4">Destination MAC Address</td><td colspan="4">Source MAC Address</td></tr>
<tr><td colspan="8">Source MAC Address</td></tr>
<tr><td colspan="4">Type (0x0800)</td><td>Version</td><td>HdrLen</td><td colspan="2">Type of Service</td></tr>
<tr><td colspan="4">Total Length</td><td colspan="4">Identification</td></tr>
<tr><td>Flags</td><td colspan="3">Fragment Offset</td><td colspan="2">Time-To-Live</td><td colspan="2">Protocol (0x01)</td></tr>
<tr><td colspan="4">Header Checksum</td><td colspan="4">Source IP Address</td></tr>
<tr><td colspan="4">Source IP Address</td><td colspan="4">Destination IP Address</td></tr>
<tr><td colspan="4">Destination IP Address</td><td colspan="4">Options and Padding</td></tr>
<tr><td colspan="4">Options and Padding</td><td colspan="2">Type</td><td colspan="2">Code</td></tr>
<tr><td colspan="4">Checksum</td><td colspan="4">Payload</td></tr>
<tr><td colspan="8">Ethernet CRC</td></tr>
</table>

ICMP Message Types



Type	Description
Echo Request	`ping` queries
Echo Reply	`ping` responses
Destination Unreachable	Destination **network** unreachable
	Destination **host** unreachable
	Destination **port** unreachable
	Fragmentation required but DF flag set
	Network administratively prohibited
Source Quench	Congestion control
Redirect Message	Redirect datagram for the **network**
	Redirect datagram for the **host**
	Redirect for TOS and network
	Redirect for TOS and host
Router Solicitation	Router discovery/selection/solicitation
Time Exceeded	TTL expired in transit
	Fragment reassembly time exceeded
Bad IP Header or Parameter	Pointer indicates the error
	Missing a required option
	Bad length
Timestamp Timestamp Reply	Like `ping`, but requesting a timestamp from the destination

ICMP Status Reporting

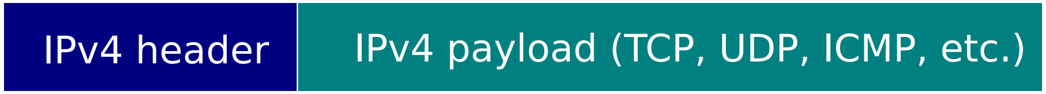
- ICMP status reporting messages have either
  - ICMP Echo type, or
  - ICMP Timestamp type
- Using ICMP status reporting types, sender may query the receiver about its status. The receiver, upon receiving the request sends a reply
- Each request includes a 16-bit query ID
- The reply uses the same query ID, in response

ICMP Error Reporting

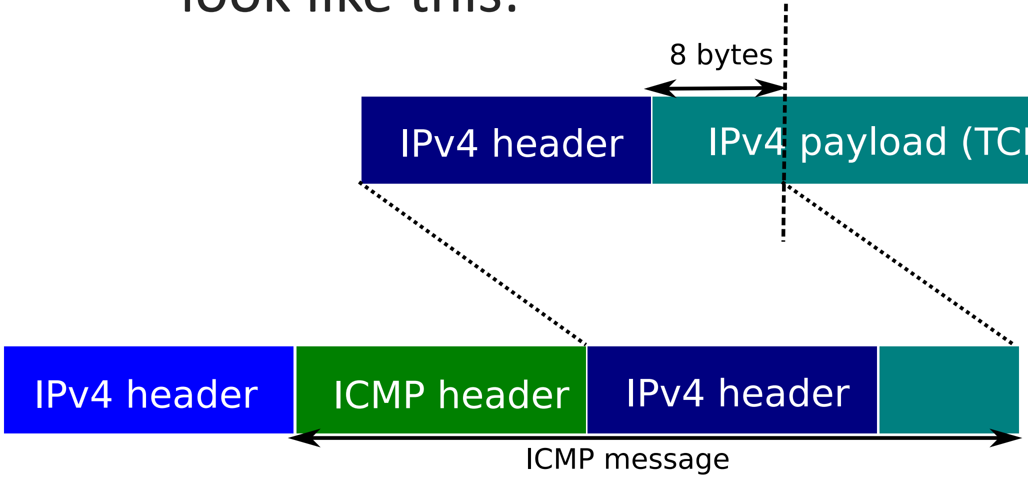
- ICMP error reporting messages are caused by the transmission of IPv4 datagrams
- Each ICMP error reporting message includes
  - The IPv4 header of the packet that has caused the error (e.g. 20 bytes)
  - First 8 bytes of the IPv4 packet payload that has caused the error
    - If TCP or UDP is the payload of IPv4, then port numbers are within this 8 bytes!

ICMP Error Reporting

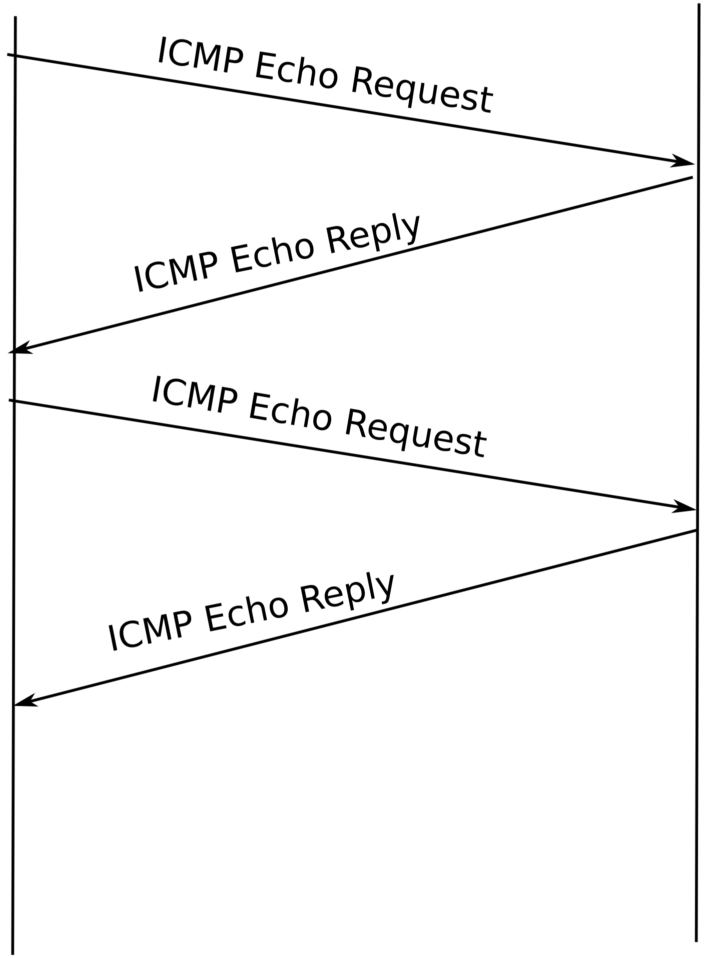
- If the following IPv4 datagram causes an error…



- …then the ICMP error reporting message would look like this:



ICMP Echo: Ping



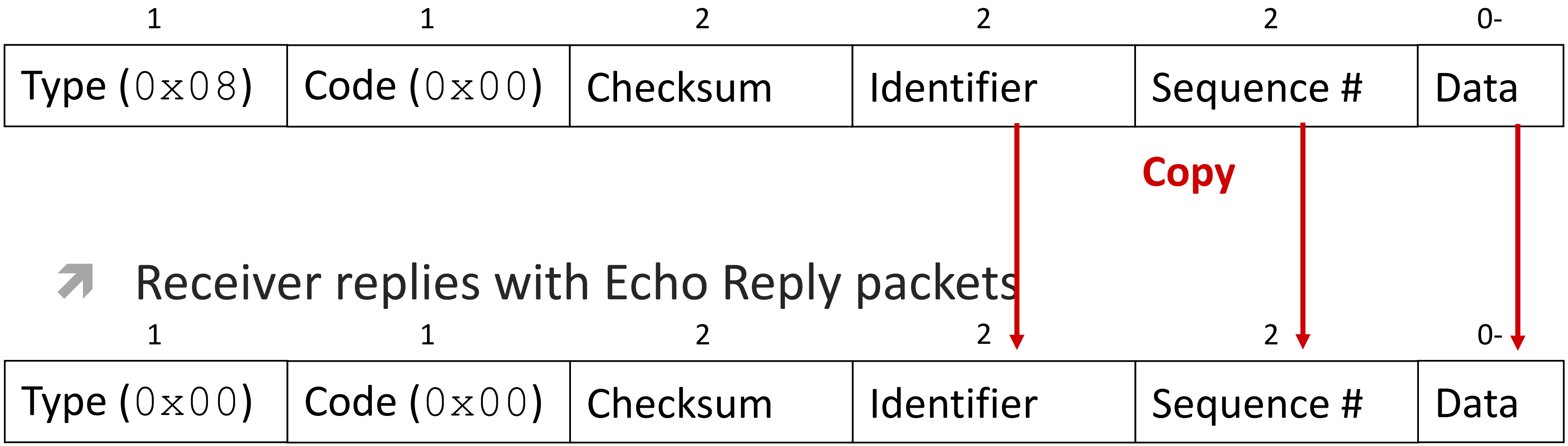
- Common tool used to test basic network connectivity
  - Is target host alive?
  - Is there a route to the target host? *And back?*
  - How long does it take to reach the target host?

Ping

```
dhcp-10-10-207-20:~ shafer$ ping -c 3 www.pacific.edu
PING www.pacific.edu (192.168.200.100): 56 data bytes
64 bytes from 192.168.200.100: icmp_seq=0 ttl=252 time=0.738 ms
64 bytes from 192.168.200.100: icmp_seq=1 ttl=252 time=1.025 ms
64 bytes from 192.168.200.100: icmp_seq=2 ttl=252 time=0.776 ms

--- www.pacific.edu ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.738/0.846/1.025/0.127 ms

dhcp-10-10-207-20:~ shafer$ ping -c 3 www.google.com
PING www.l.google.com (74.125.19.103): 56 data bytes
64 bytes from 74.125.19.103: icmp_seq=0 ttl=56 time=7.534 ms
64 bytes from 74.125.19.103: icmp_seq=1 ttl=56 time=7.295 ms
64 bytes from 74.125.19.103: icmp_seq=2 ttl=56 time=7.661 ms

--- www.l.google.com ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 7.295/7.497/7.661/0.152 ms
```

ICMP Echo

- Ping uses ICMP
  - ICMP Echo Request (type 8)
  - ICMP Echo Reply (type 0)
- Sender creates Echo Request packets



1	1	2	2	2	0-
Type (0x08)	Code (0x00)	Checksum	Identifier	Sequence #	Data

- Receiver replies with Echo Reply packets

1	1	2	2	2	0-
Type (0x00)	Code (0x00)	Checksum	Identifier	Sequence #	Data

ICMP Destination Unreachable

- ICMP Destination Unreachable message type has different codes, representing different error conditions
  - Network unreachable
  - Host unreachable
  - Port unreachable
  - Fragmentation required but DF (“don’t fragment”) set
  - Administratively prohibited

Network Unreachable

- If a router in the path cannot find a matching entry in its forwarding table for the destination IP address
  - It drops the packet, and
  - It sends an ICMP Network Unreachable message back to the sender

Network Unreachable

- **Example** - Consider an IPv4 packet
  - Source IP address: 137.82.251.11
  - Destination IP address: 88.211.92.56
  - Assume that a router in the path with IP address 55.37.127.69 has the forwarding table

Destination	Interface
132.55.0.0/16	int0
178.69.22.0/24	int1
211.8.0.0/16	int2
55.0.0.0/8	int3

Network Unreachable

- **Example** (continued) – The router cannot forward the packet (no match in forwarding table)
- Generates ICMP error message
  - Type: Destination Unreachable
  - Code: Network Unreachable
  - Includes original IPv4 header (w/destination IP and source IP)
- ICMP message is encapsulated in IPv4 packet
  - Source: 55.37.127.69 *(the router)*
  - Destination: 137.82.251.11 *(original source)*

Host Unreachable

- If the last router in the path between the source and destination cannot reach the destination (which is in the same LAN as router interface)
  - The router drops the packet, and
  - The router sends back an ICMP Host Unreachable message to the sender

Host Unreachable

- **How does the router realize that the host is not reachable?**
- The router receives the IPv4 packet and checks its destination IP address
- The router does longest prefix match and determines that the destination is in a directly-connected subnet (on egress interface X)
- In order to send the IPv4 packet to the destination, that packet should be encapsulated within a link layer frame. So, destination MAC address is required
- The router queries its ARP cache to pick up the MAC address matching the destination IP address. There is not any valid entry in the ARP cache...
- The router broadcasts an ARP request through interface X, asking for the MAC address that belongs to the destination IP address
- No reply comes back? **The host is unreachable**

Port Unreachable

- Used for UDP packets encapsulated within IPv4 packets
- If the packet is deliverable to its destination, but on that host the UDP port is closed, then the host sends back ICMP Port Unreachable message to the sender of the packet
- Question: Why not TCP?
  - TCP is connection-oriented. If TCP port is closed, TCP handles it itself! (With a “Reset” message)
  - UDP however is silent about closed ports

Other Destination Unreachable Messages

- Fragmentation required, but DF set
  - If the IPv4 packet size exceeds MTU and DF flag on IPv4 header is set
    - The packets is dropped
    - ICMP error message is sent back to the sender
- Administratively prohibited
  - If the last router in the path is able to forward the packet to the ultimate destination, but for some reason it has been configured to drop the packets to that destination
    - The router drops the packet
    - The router sends back an ICMP network administratively prohibited message to the sender

ICMP Time Exceeded

- Each network layer service, in the path between the two endpoints, decrements the TTL value in the IPv4 header

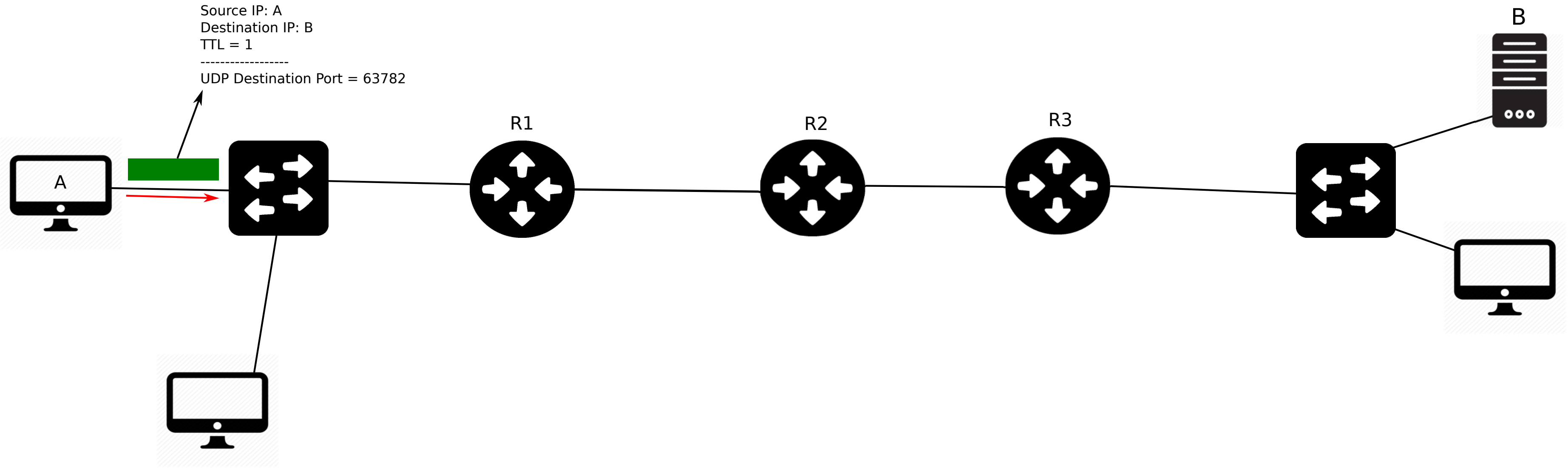
- If after decrementing the TTL becomes 0:
  - The packet is dropped
  - An ICMP Time Exceeded message is sent back to the sender

ICMP Time Exceeded

- **How can a host identify the path between itself and a remote host?**
- By causing the nodes in the path to send back ICMP Time Exceeded messages
  - The source IP address on the IPv4 packet encapsulating the ICMP message reveals those nodes
- How can we cause nodes in the path to generate this type of ICMP message?
  - By setting the TTL value to a certain amount!
  - **Traceroute tool**

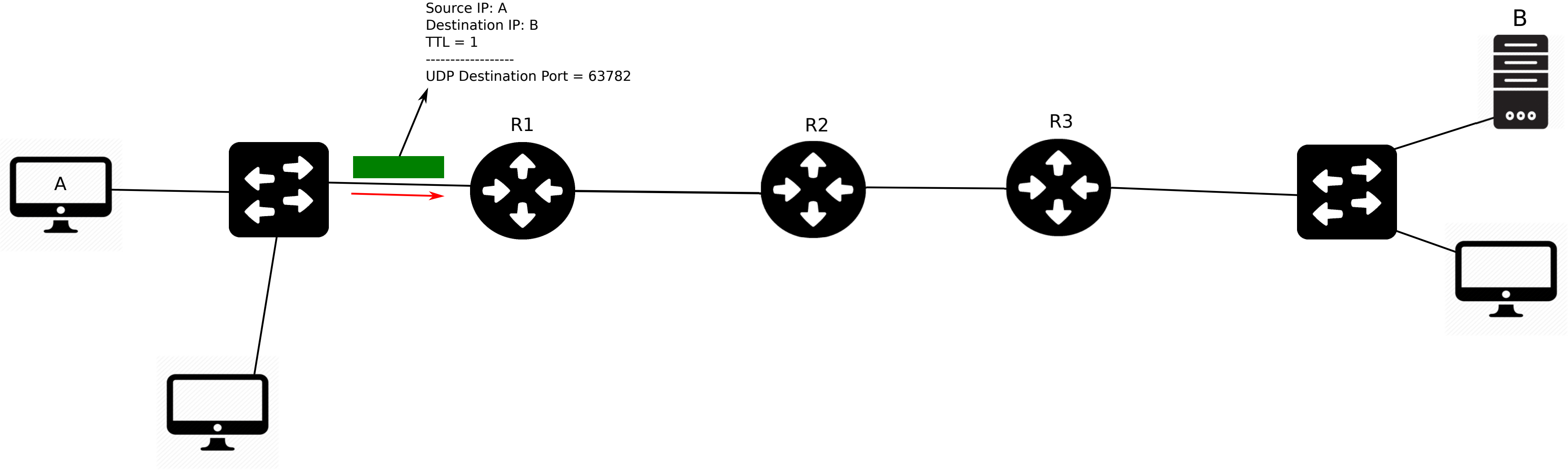
Traceroute (1)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.



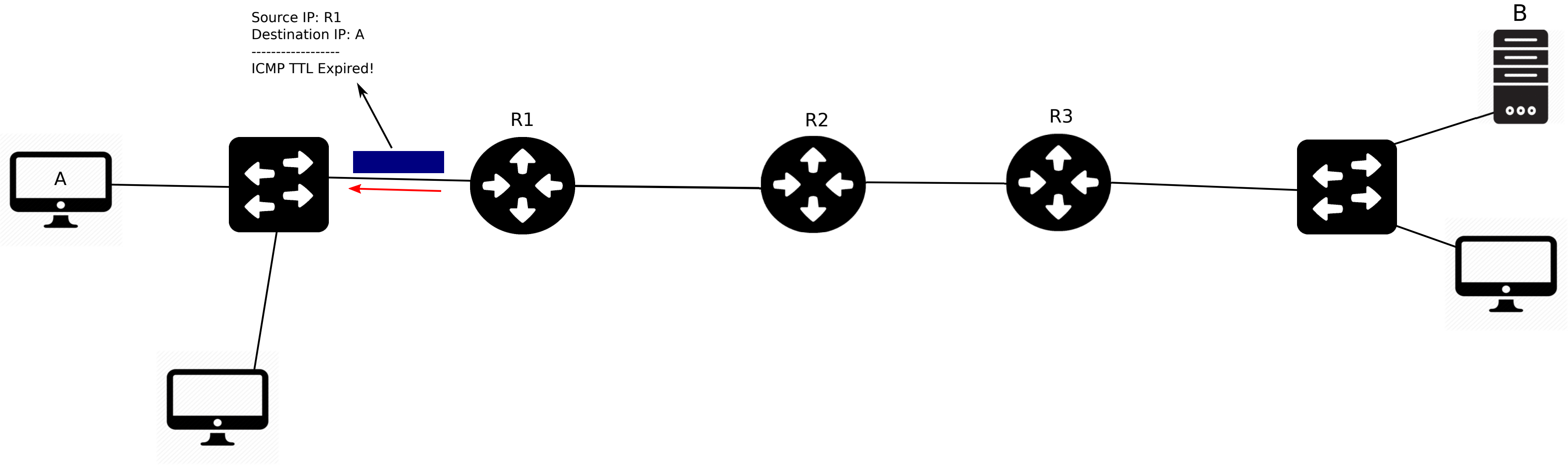
Traceroute (2)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.



Traceroute (3)

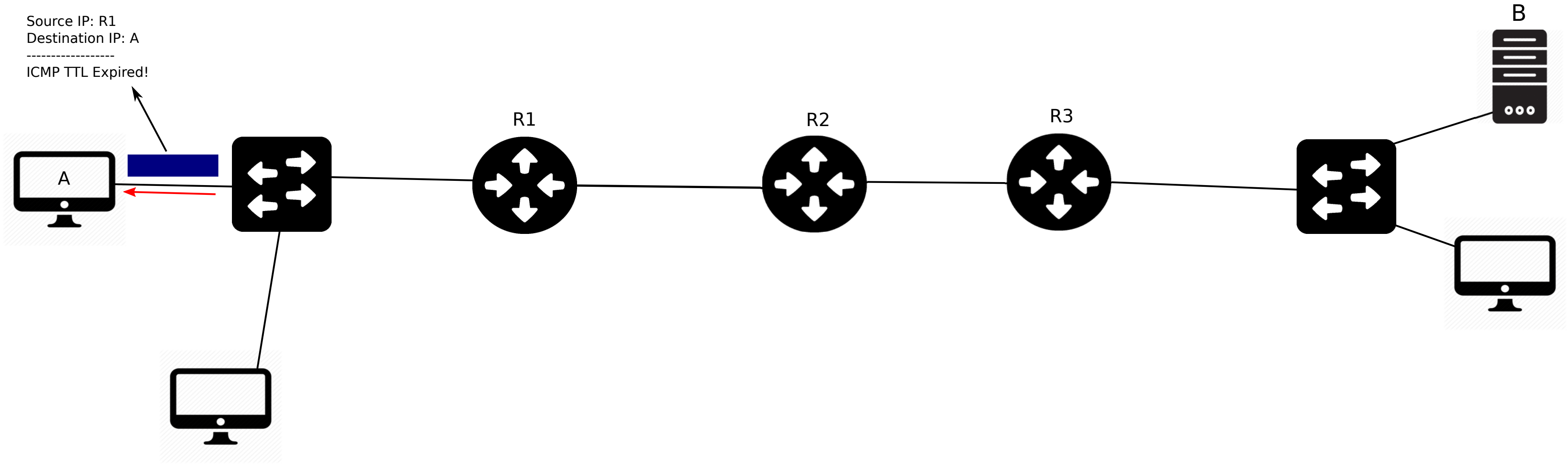
- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.



Traceroute (4)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

Hop	Router
1	R1



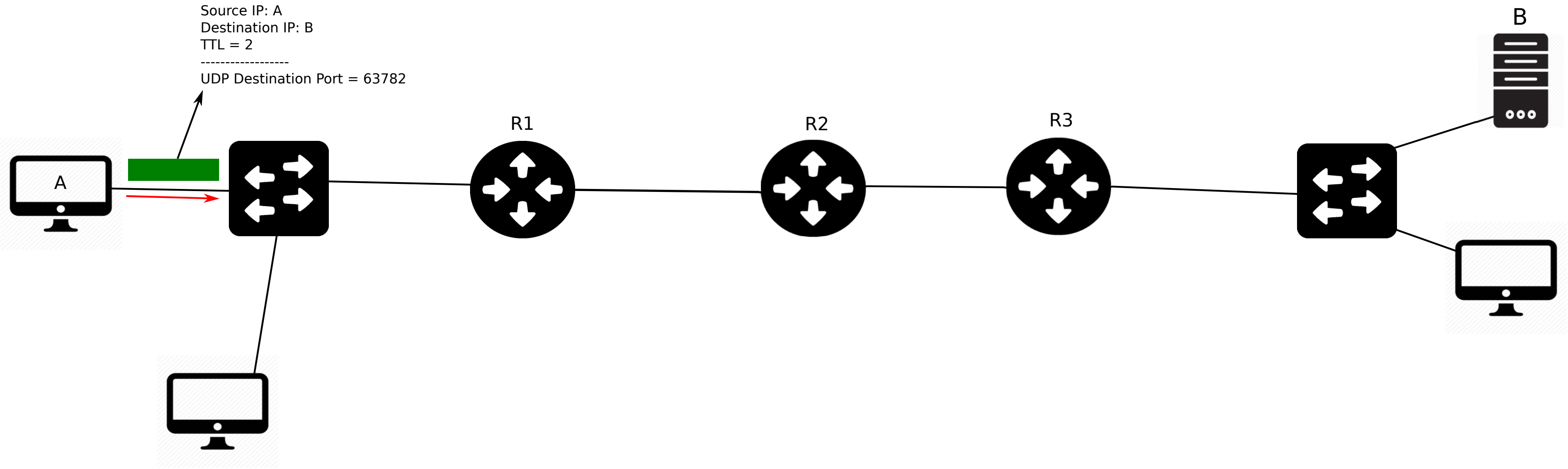
Source IP: R1
Destination IP: A

ICMP TTL Expired!

Traceroute (5)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

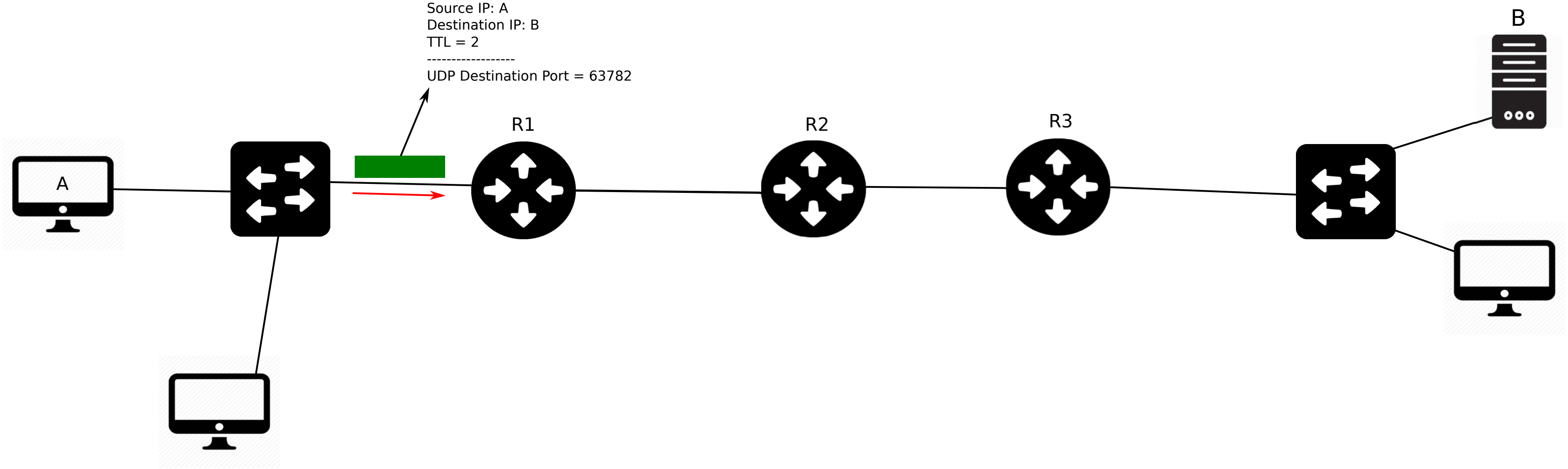
Hop	Router
1	R1



Traceroute (6)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

Hop	Router
1	R1

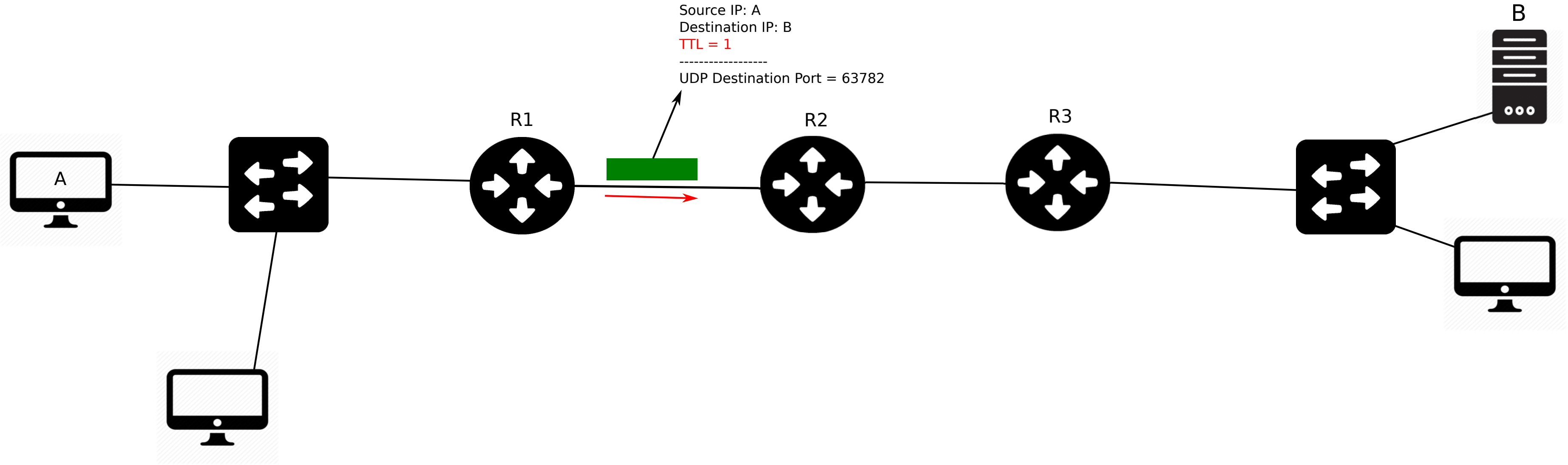

Source IP: A
Destination IP: B
TTL = 2

UDP Destination Port = 63782

Traceroute (7)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

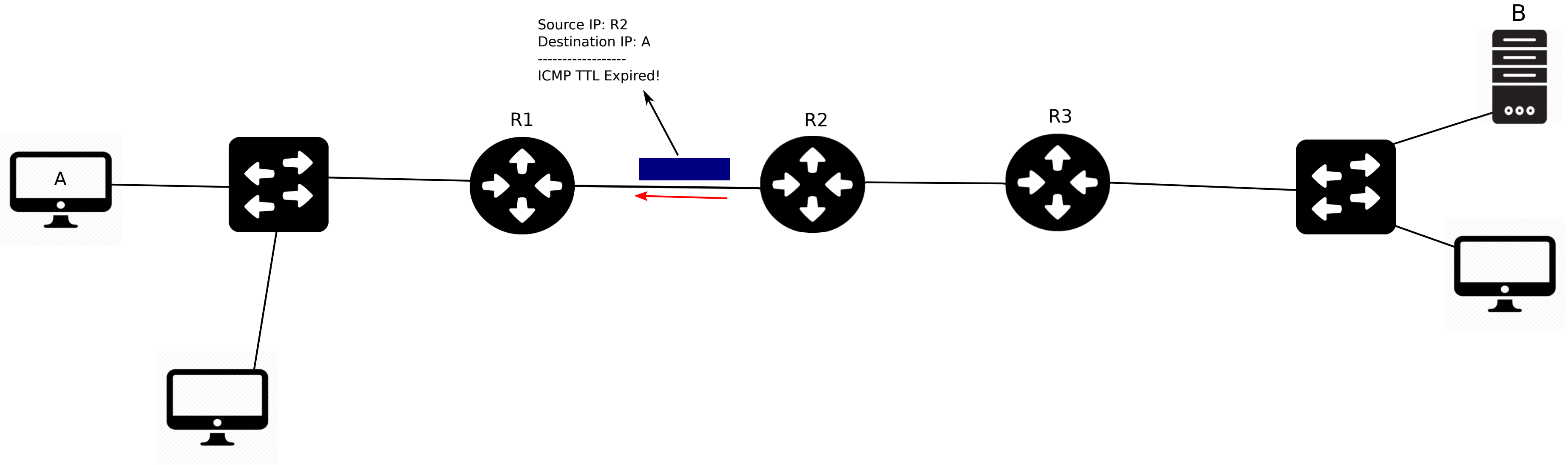
Hop	Router
1	R1



Traceroute (8)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

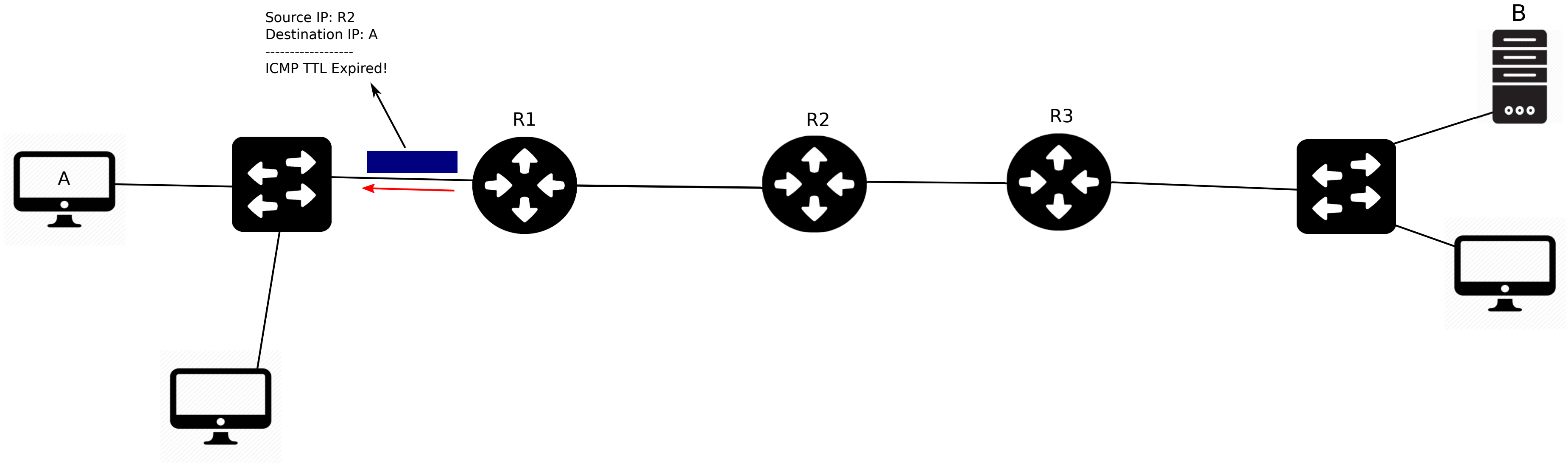
Hop	Router
1	R1



Traceroute (9)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

Hop	Router
1	R1

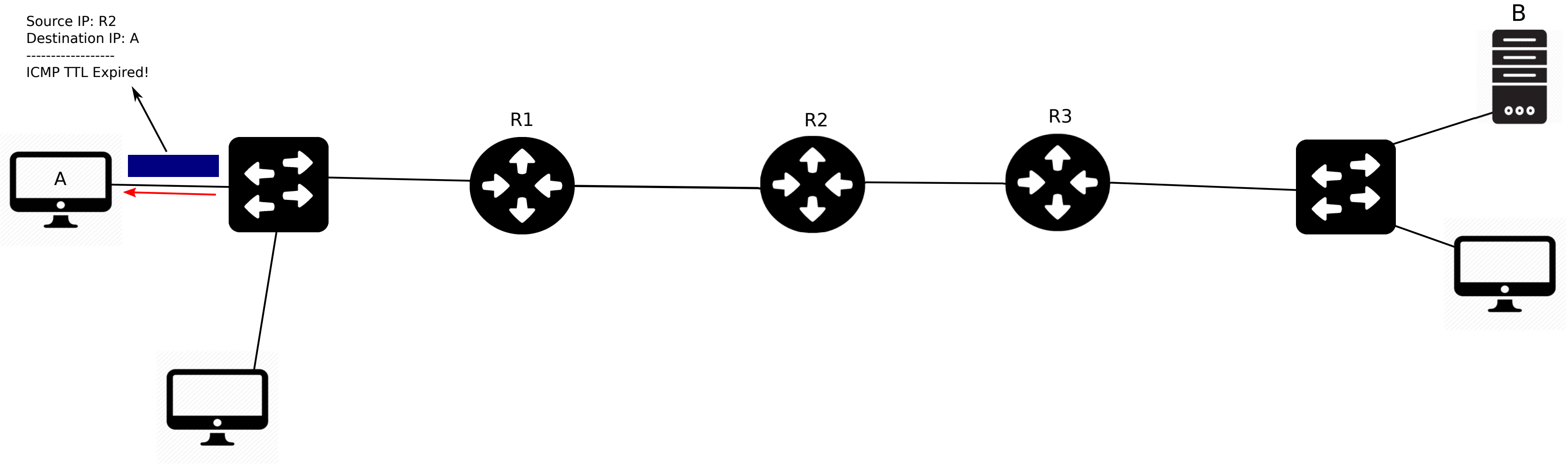

Source IP: R2
Destination IP: A

ICMP TTL Expired!

Traceroute (10)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

Hop	Router
1	R1
2	R2



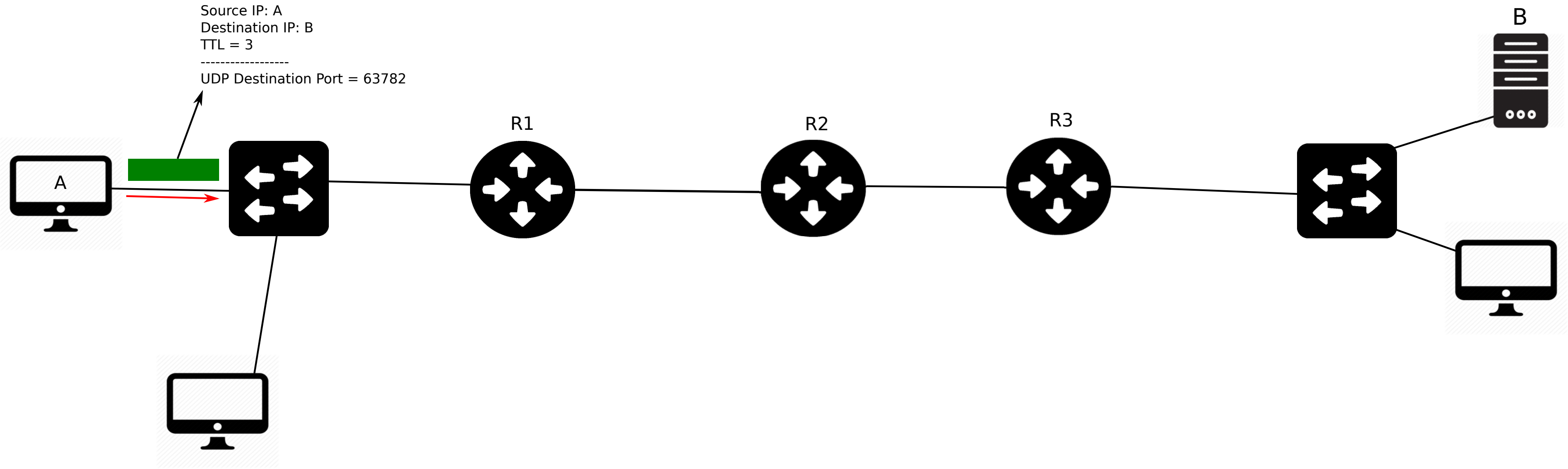
Source IP: R2
Destination IP: A

ICMP TTL Expired!

Traceroute (11)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

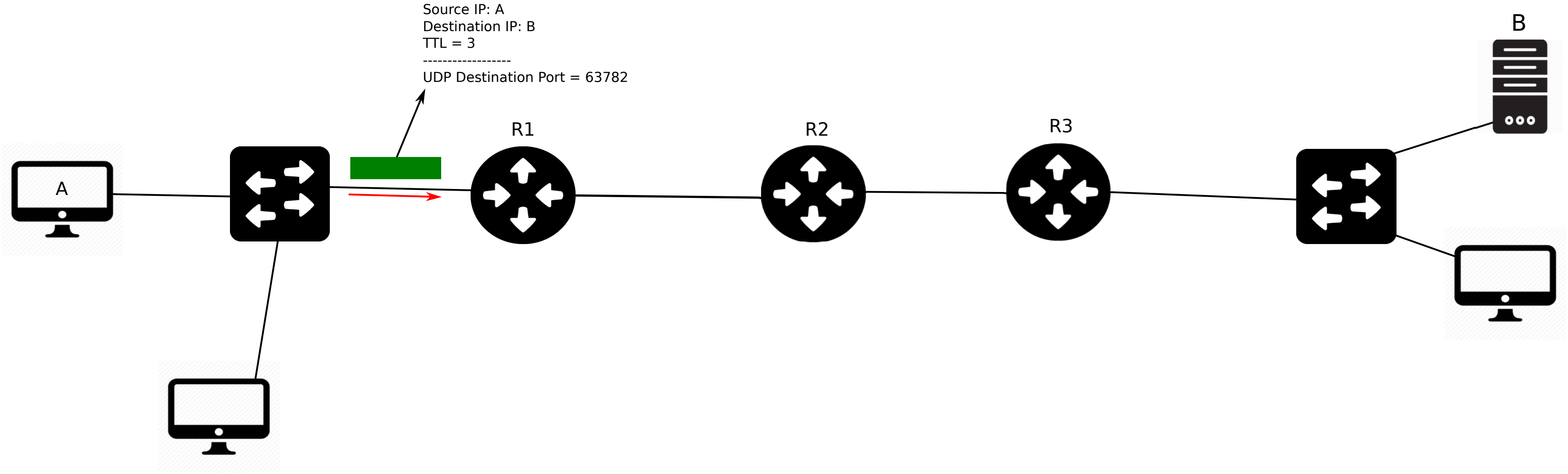
Hop	Router
1	R1
2	R2



Traceroute (12)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

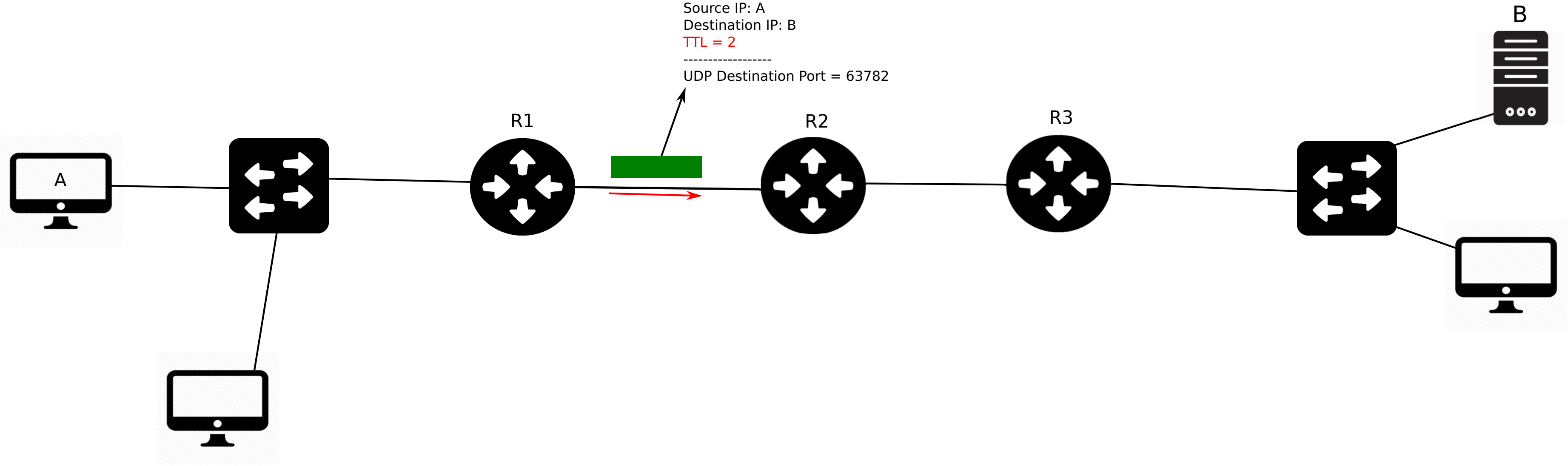
Hop	Router
1	R1
2	R2



Traceroute (13)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

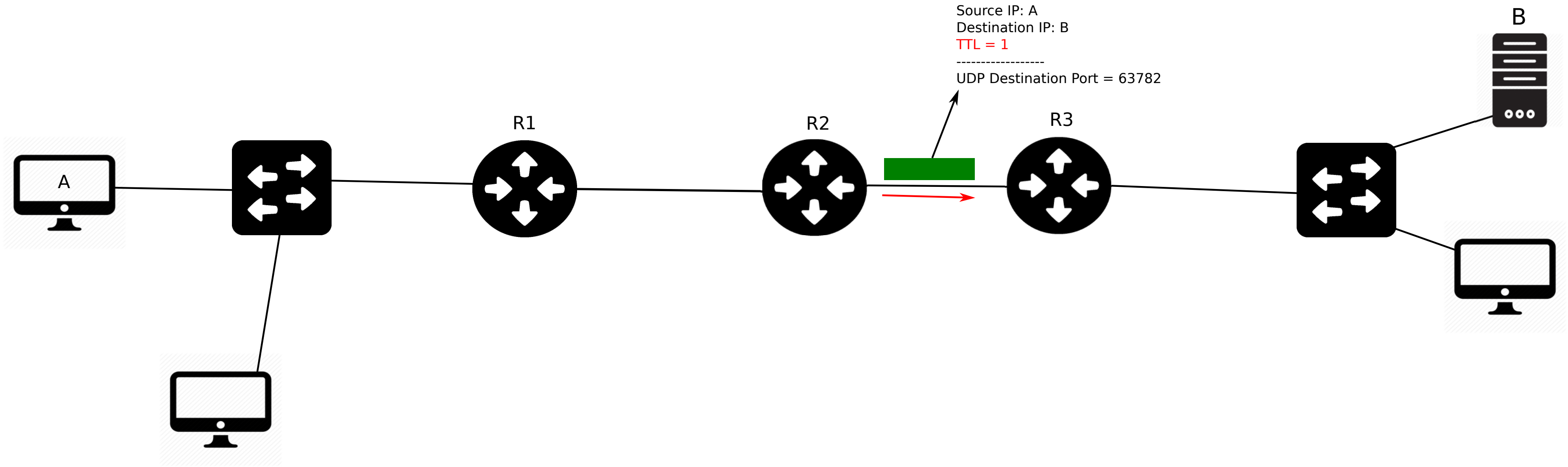
Hop	Router
1	R1
2	R2



Traceroute (14)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

Hop	Router
1	R1
2	R2



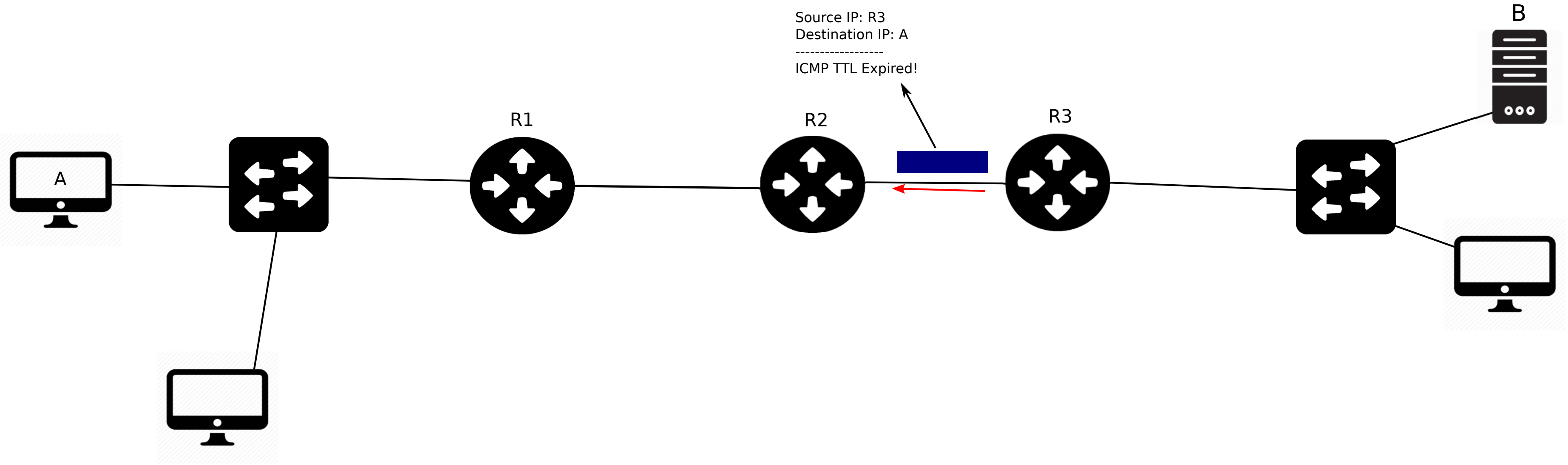
Source IP: A
Destination IP: B
TTL = 1

UDP Destination Port = 63782

Traceroute (15)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

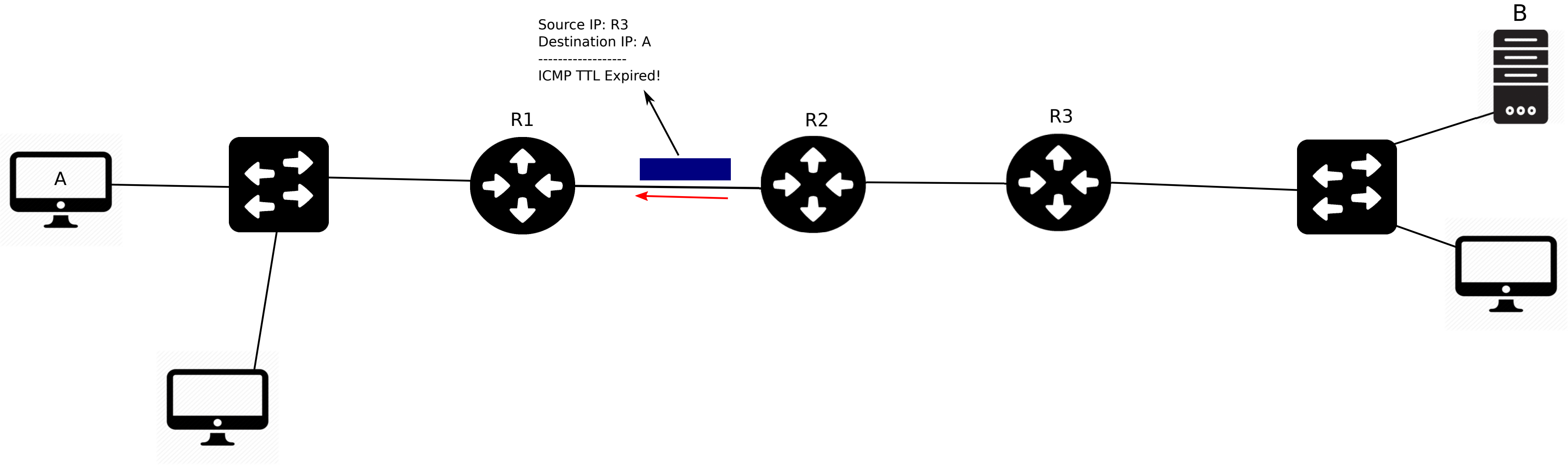
Hop	Router
1	R1
2	R2



Traceroute (16)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

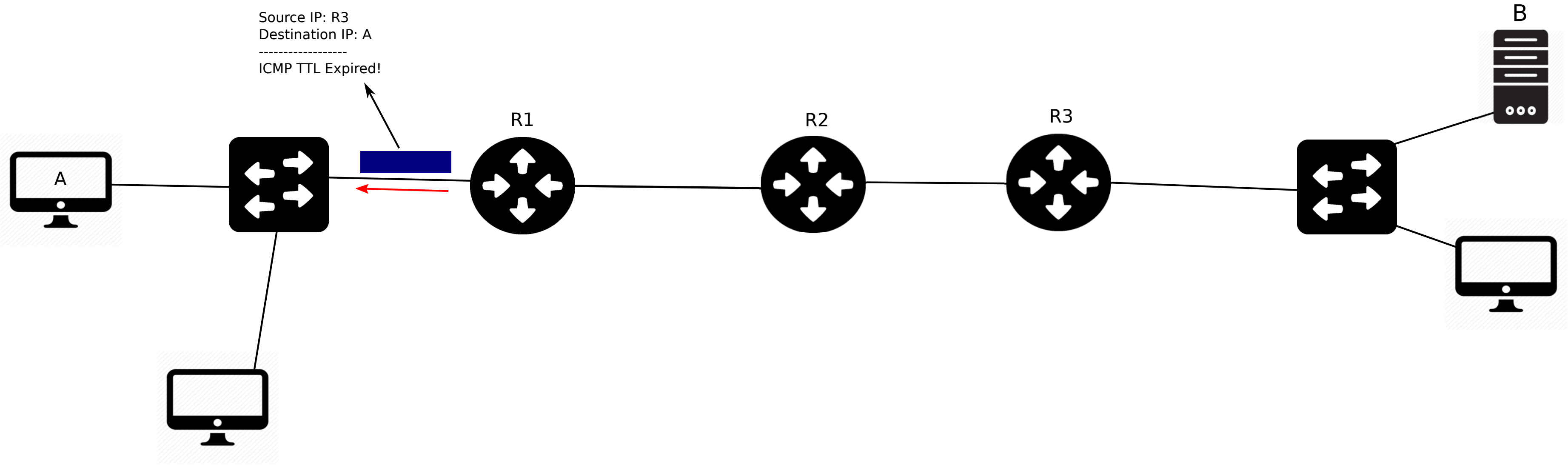
Hop	Router
1	R1
2	R2



Traceroute (17)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

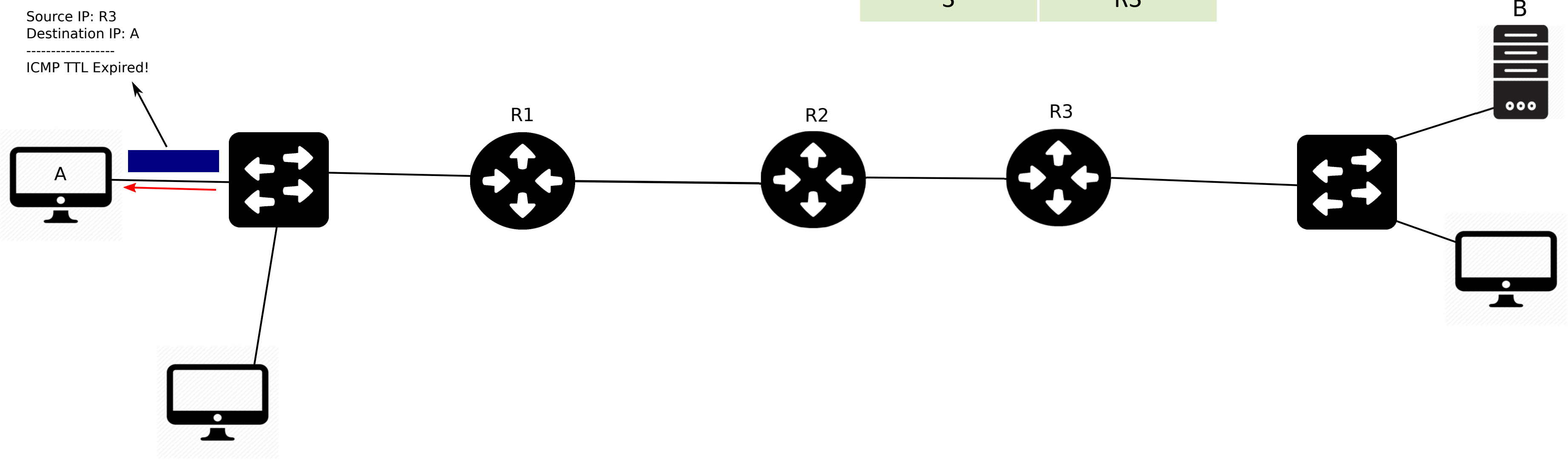
Hop	Router
1	R1
2	R2



Traceroute (18)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

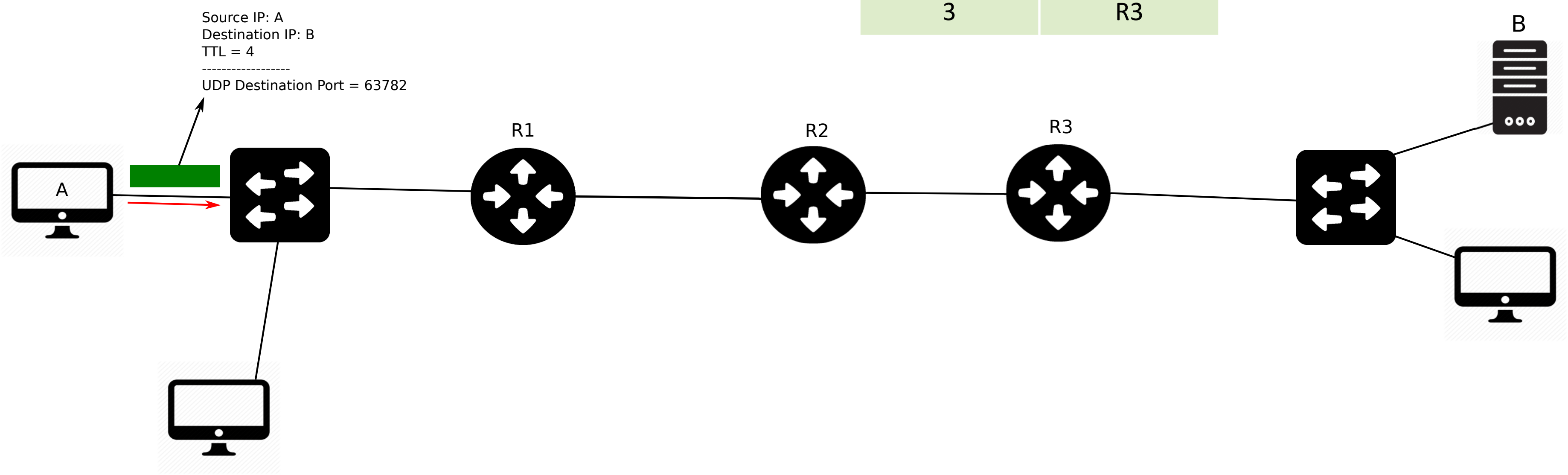
Hop	Router
1	R1
2	R2
3	R3



Traceroute (19)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

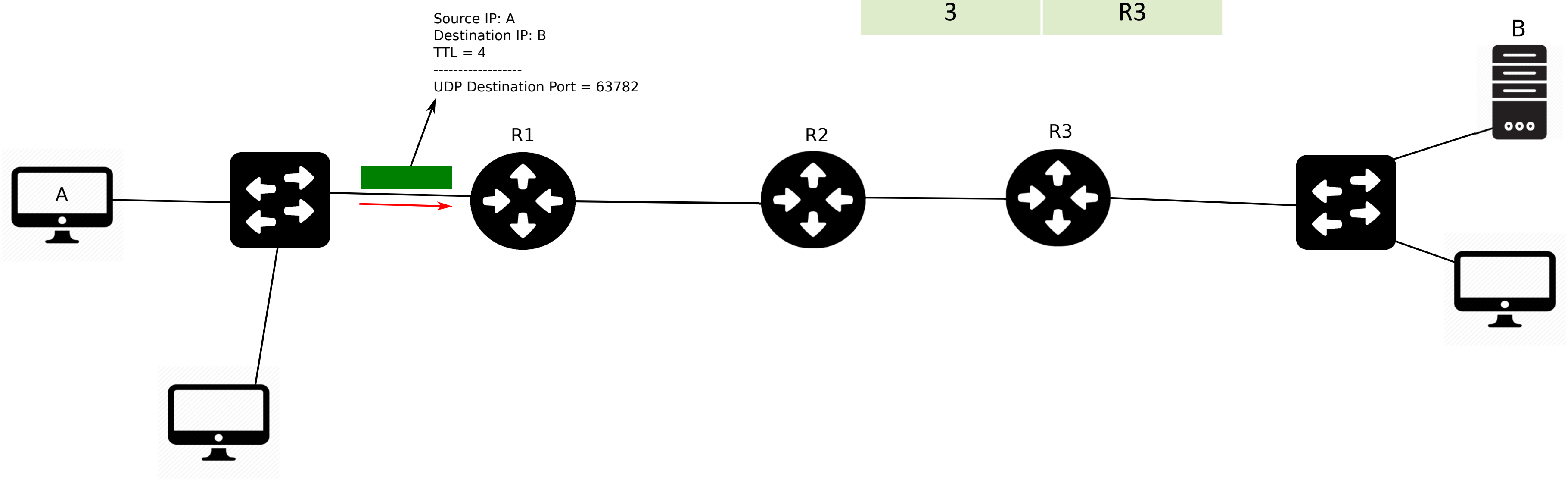
Hop	Router
1	R1
2	R2
3	R3



Traceroute (20)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

Hop	Router
1	R1
2	R2
3	R3



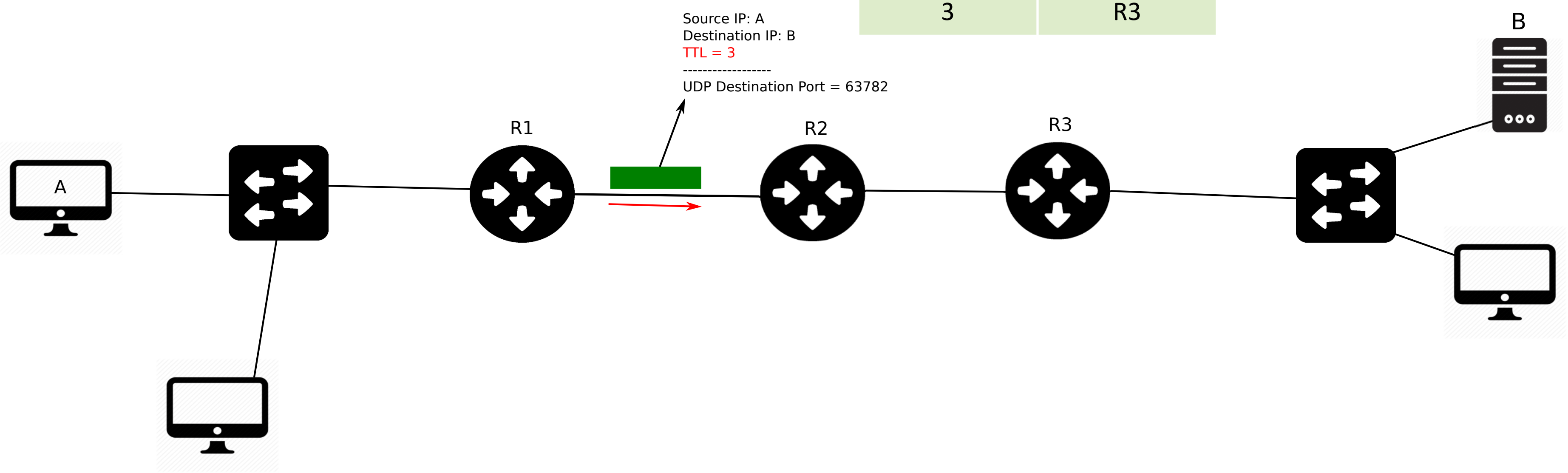
Source IP: A
Destination IP: B
TTL = 4

UDP Destination Port = 63782

Traceroute (21)

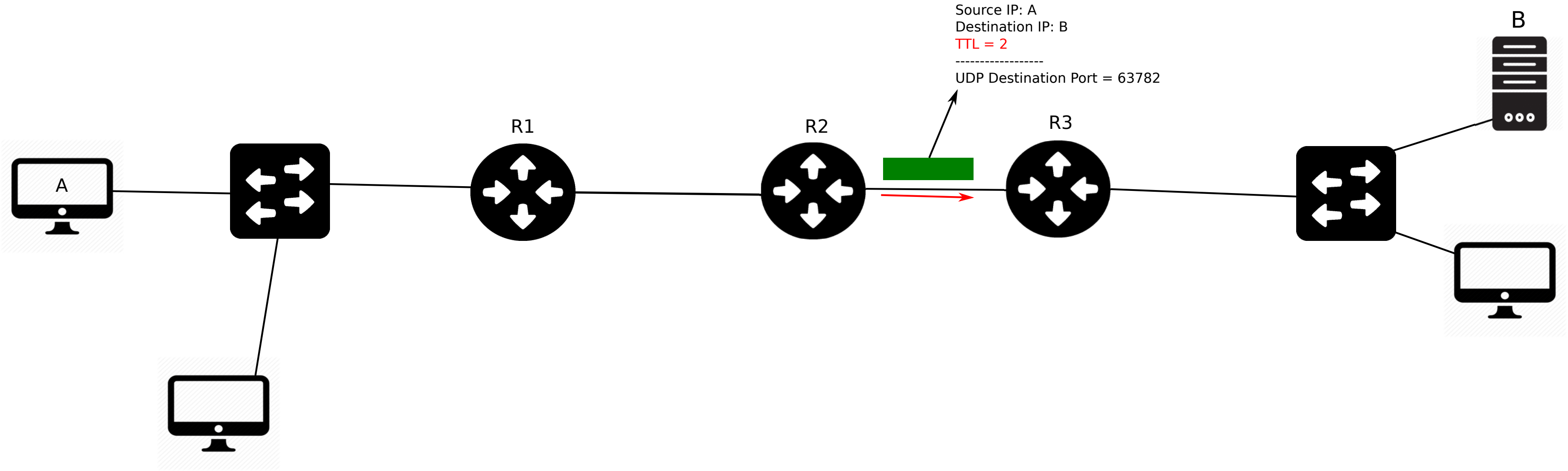
- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

Hop	Router
1	R1
2	R2
3	R3



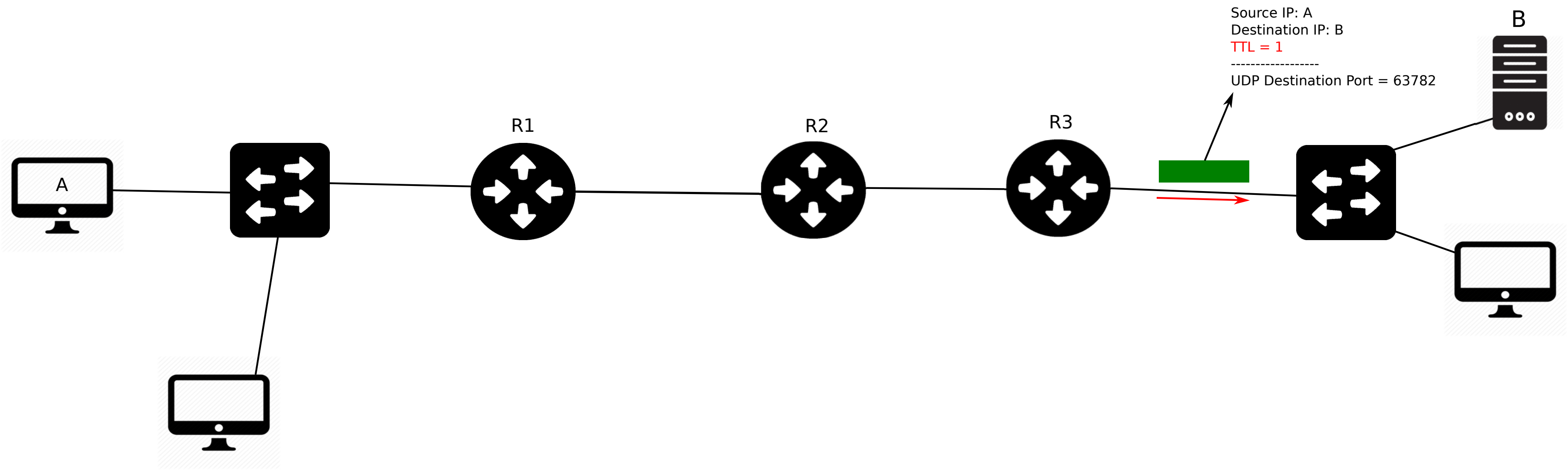
Traceroute (22)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.



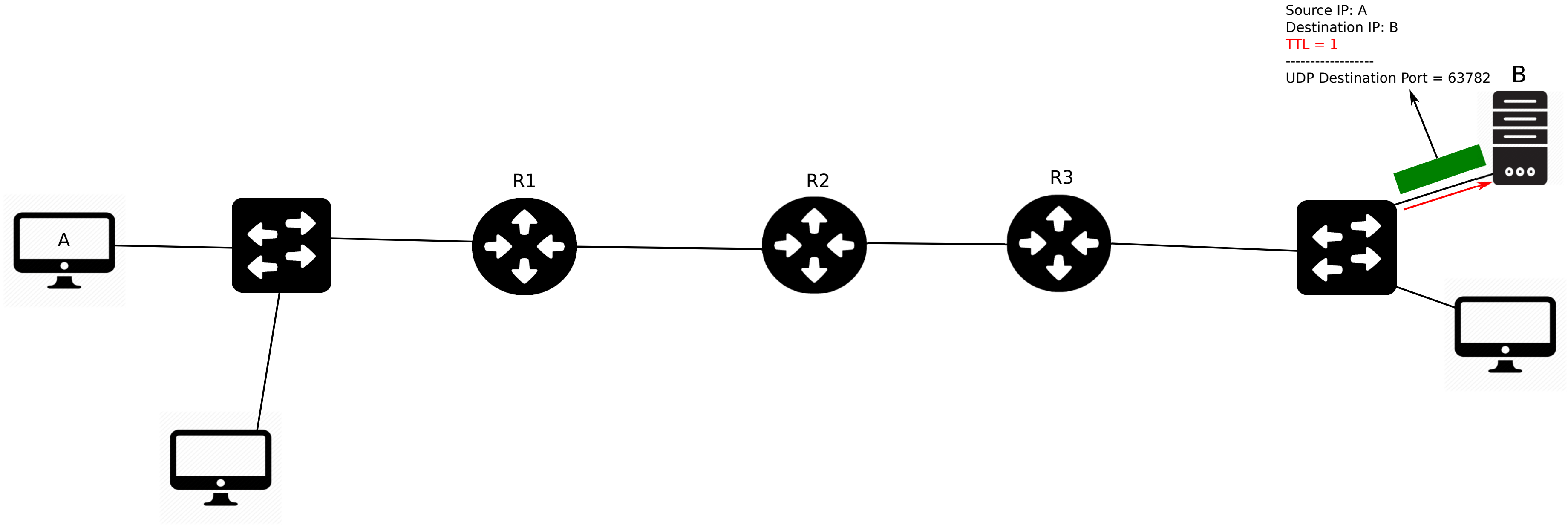
Traceroute (23)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.



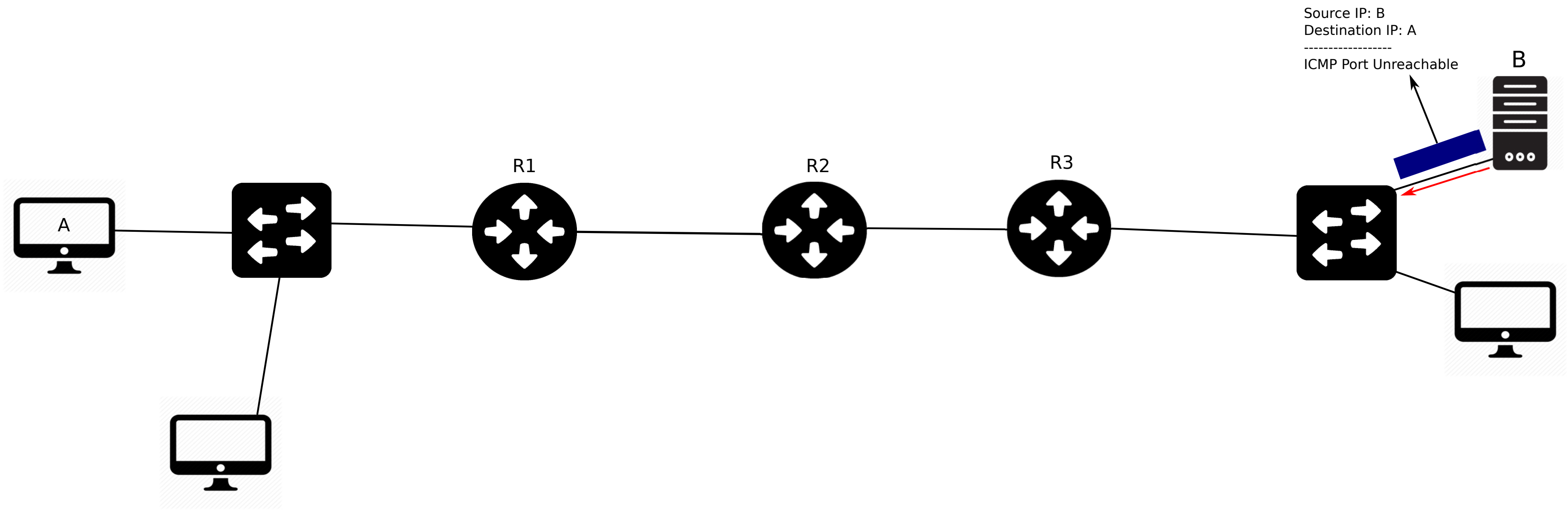
Traceroute (24)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.



Traceroute (25)

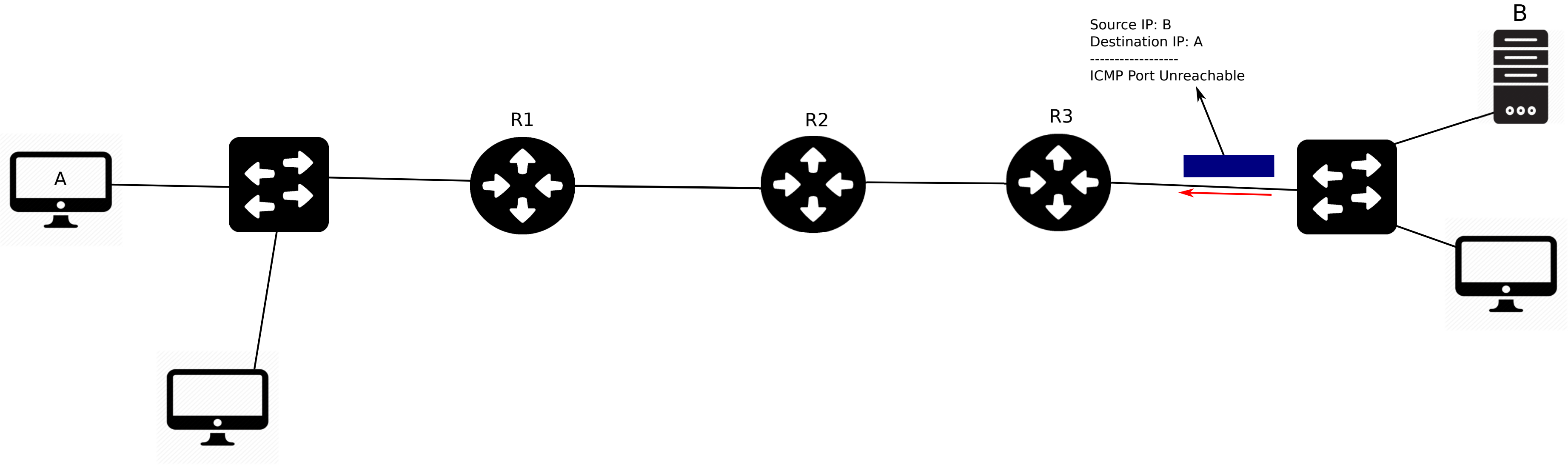
- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.



Traceroute (26)

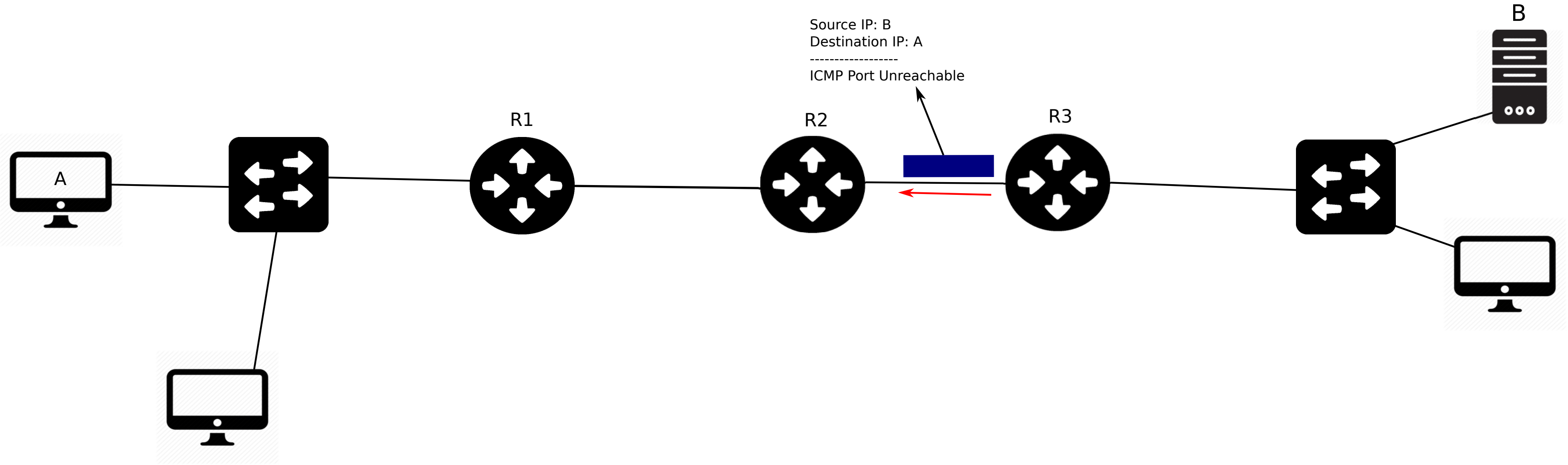
- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number

- Example: A runs traceroute on B.



Traceroute (27)

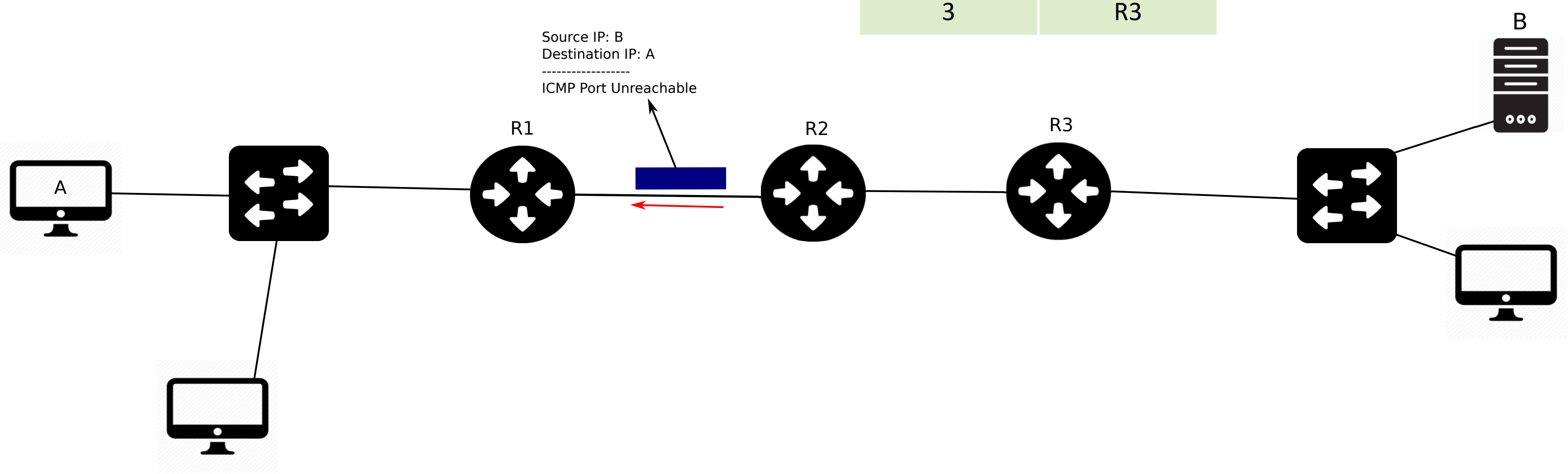
- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.



Traceroute (28)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

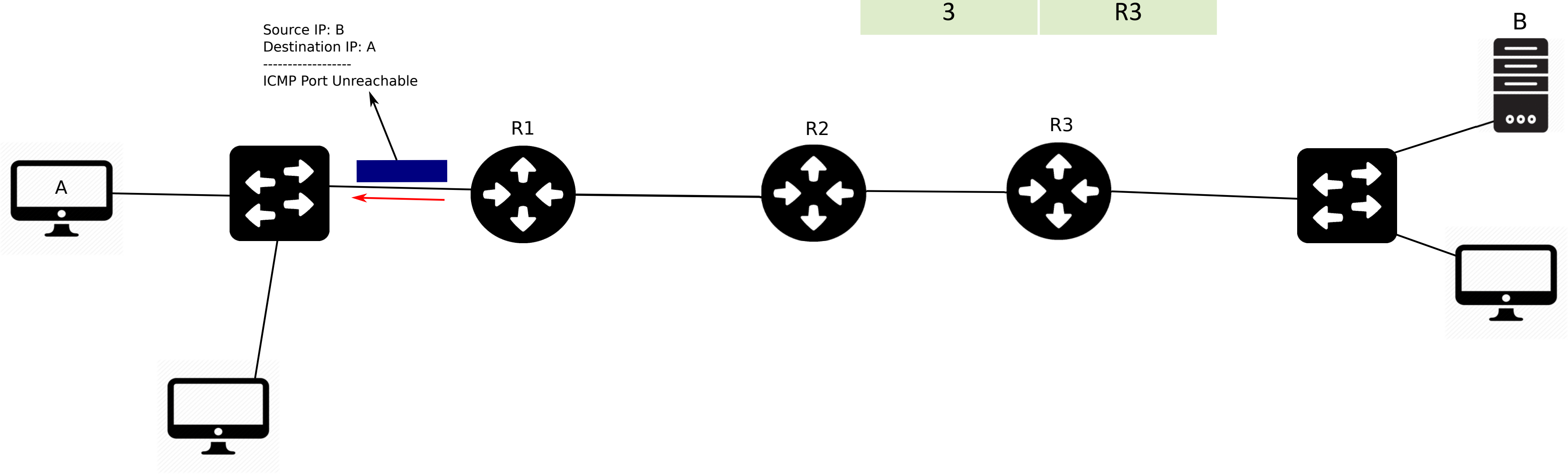
Hop	Router
1	R1
2	R2
3	R3



Traceroute (29)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

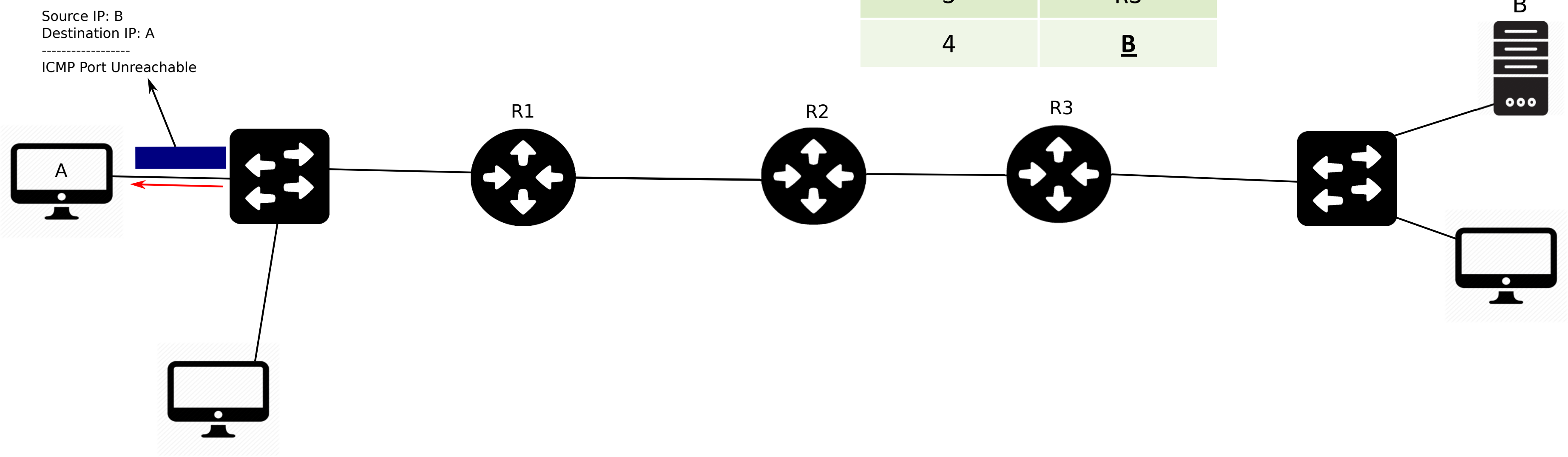
Hop	Router
1	R1
2	R2
3	R3



Traceroute (30)

- **Traceoute** uses ICMP TTL expired messages to identify the path between two hosts
  - Sends IPv4 packet(s) with TTL = 1, then with TTL = 2, then with TTL = 3, ...
  - The IPv4 packets encapsulate a UDP datagram with unconventional destination port number
- Example: A runs traceroute on B.

Hop	Router
1	R1
2	R2
3	R3
4	**B**


Source IP: B
Destination IP: A

ICMP Port Unreachable

Closing Thoughts

Recap

- Today we discussed ICMP
  - Different types
  - Different codes
- Traceroute tool

Next Class

- IPv6

Class Activity

CA.9 – ICMP & Wireshark

Due tonight at 11:59pm

Project 2

Due Oct 7th at 11:59pm