CS 168
Spring 2025

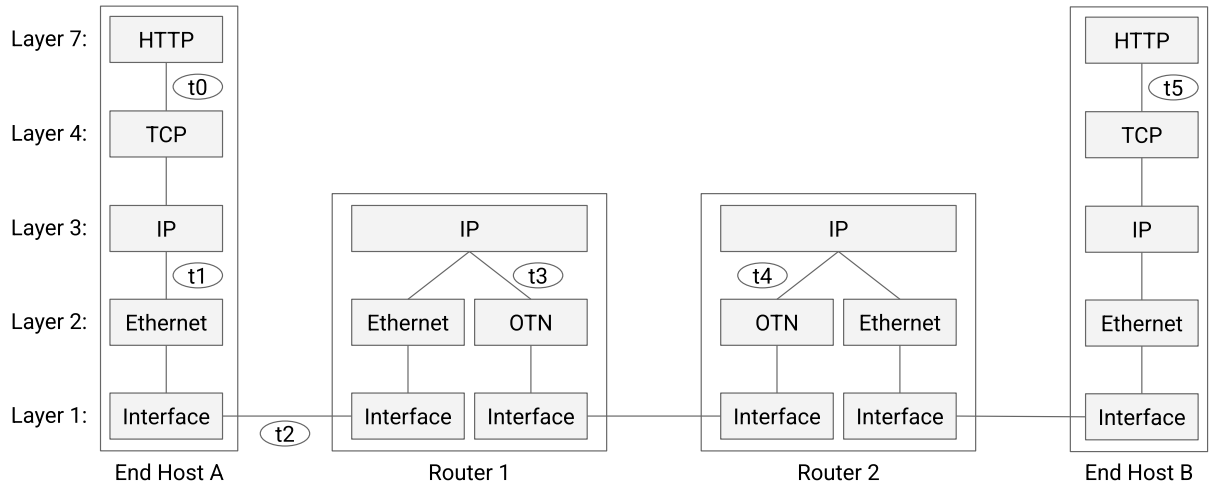
Layers, Design, Traceroute

Discussion 1

1 Internet Layers

In this question, we'll explore what headers are on the packet as it travels through different layers and protocols of the network. Consider the diagram below, which shows Host A sending a packet to Host B. Along the way, the packet is forwarded by intermediate routers R1 and R2.

At each of the labeled time steps, fill in the boxes to describe which headers are attached to the payload. Not all boxes may be used. The headers at $t = 2$ are filled in for you as an example.



t0				HTTP	Payload
t1		IP	TCP	HTTP	Payload
t2	Ethernet	IP	TCP	HTTP	Payload
t3		IP	TCP	HTTP	Payload
t4		IP	TCP	HTTP	Payload
t5		IP	TCP	HTTP	Payload

2 Design Philosophy (Clark Paper)

Today's Internet architecture derives its core design principles from the ARPANET. The ARPANET was the first packet-switched network, developed by DARPA[1] in the 1970s and 1980s.

The core design objectives of the ARPANET are described in the paper *The design philosophy of the DARPA internet protocols* (1988), by David Clark. Clark was a chief protocol architect for the Internet in the 1980s.

Link to the paper: https://dl.acm.org/doi/10.1145/52325.52336

According to the paper, the DARPA architecture has one fundamental goal, and several secondary goals:

1. **Fundamental Goal:** The Internet must offer an "effective" technique for multiplexed utilization of existing interconnected networks (e.g. autonomous systems).

   Specifically, the Internet should be a packet-switched communications facility that interconnects individual networks through packet switches called gateways, where such interconnection still preserves the individual networks as separately administrated, relatively autonomous entities.
2. **Secondary Goal:** Internet communication must continue despite loss of networks or gateways.
3. **Secondary Goal:** The Internet must support multiple types of communications service.
4. **Secondary Goal:** The Internet architecture must accommodate a variety of networks (e.g. autonomous systems).
5. **Secondary Goal:** The Internet architecture must permit distributed management of its resources.
6. **Secondary Goal:** The Internet architecture must be cost effective.
7. **Secondary Goal:** The Internet architecture must permit host attachment with a low level of effort.
8. **Secondary Goal:** The resources used in the Internet architecture must be accountable.

[1] Defense Advanced Research Projects Agency

For each of the following proposals, determine whether it is in violation of the goals listed above. If a proposal is in violation, also indicate which of the above goals the proposal violates.

Also, for subparts 2.2 and 2.4, determine whether the proposals violate the end-to-end principle.

Note: This question is open-ended, and our answers are subjective and open to debate.

2.1 The routers of all Internet carriers must accommodate circuit switching.

Recall circuit switching: End hosts can reserve capacity in the network in advance.

It violates 1 5 6

2.2 When forwarding packets, all routers must also perform additional communication to ensure that the packet is received by the next router.

It violates the end to end principle, because the reliability should be checked at ends, implementing at routers raise overheads. violates 1 6

2.3 All Internet carriers must routinely advertise the state of their network to a centralized network controller. Carriers must rely solely on the controller for commands to update their state.

violates 1 2 4 5

2.4 Due to abnormal radio interference at the Berkeley lab (LBNL), routers on LBNL's network must implement reliable delivery when forwarding packets.

violates nothing, doesnt violate end to end, as it implement reliable delivery for performance in the network

2.5 To improve connection stability, the routers of all Internet carriers can only update their state during a two-hour window of each day.

violates 2 5

2.6 The Amazon network devises a more efficient mechanism for load-balancing network traffic. All the routers in Amazon's network subscribe to Amazon's in-house network controller for updating their state.

5

2.7 The Stanford network blocks packets that it suspects are coming from the UC Berkeley network.

violates nothing

3 A Gentle Intro to Traceroute

Traceroute is a utility that attempts to discover the intermediate routers on the network path between you and a given destination host.

This problem provides a basic walkthrough of traceroute, and shows how traceroute exploits the TTL field in IP packets, and the ICMP protocol.

3.1 **IP Time-to-Live**: Layer 3 (IP) packets have a header field called Time-to-Live (TTL). When a packet is first sent out of an end host, its TTL field is initialized to a high number (e.g. 64). Every time a packet is forwarded by an intermediate router to the next-hop router, the TTL field is decremented by 1.

Why might having a TTL field be useful?

Hint: What if packets are mistakenly forwarded in a loop? What should routers do if the packet TTL is decremented to 0?

If the packet cant be delivered to the destination with TTL, it may indicates that the packet is not being delivered correctly, so it may be discarded

3.2 **ICMP**: Network devices can use the Internet Control Message Protocol (ICMP) to communicate various error messages and operational status messages. ICMP is built on top of IP.

For example, intermediate routers can send an ICMP-over-IP packet to an end host to indicate a network error (e.g. "you tried to send a packet to an unreachable destination").

ICMP packets consist of a header and a payload:

- The ICMP header has two fields (*type* and *code*) that indicate the status that the sender wants to communicate (e.g. "unreachable destination", or "this is a response to an `echo` request").
- The ICMP payload includes a copy of the IP header of the original packet that triggered the error, as well as the first 8 bytes of the original packet's IP payload.

In the traceroute project, the original packets you send will be UDP-over-IP packets. You will receive ICMP packets in response, and the ICMP payload will contain the original packet's IPv4 and UDP headers.

Why do the ICMP packets you receive contain this specific payload?

In general, why might ICMP packets include a copy of the original packet's header and part of the original packet's payload?

the end host has to detect which stream the packet is from, therefore the payload data help end host to detect which stream of traffic that error corresponds to.

3.3 **ICMP Status Types**: In the traceroute project, we are interested in two particular status types:

- **Time Exceeded**: Suppose an intermediate router receives a packet, decrements the TTL, and the resulting TTL is 0. The router drops the packet and sends an ICMP Time Exceeded response to the original sender of the packet.
- **Destination Unreachable**: Suppose an intermediate router receives a packet with an unknown destination (i.e. the router doesn't know where the destination is). The router drops the packet and sends an ICMP Destination Unreachable response to the original sender of the packet.
- **Destination Unreachable**: Here's another scenario where this type is used. Suppose a destination host receives a packet with a UDP destination port that the host is not listening on. The host drops the packet and sends an ICMP Destination Unreachable response to the original sender of the packet.

How do intermediate routers know who to send the ICMP packet to?

Hint: What header field should the router look at?

By looking at the the source IP of L3 header

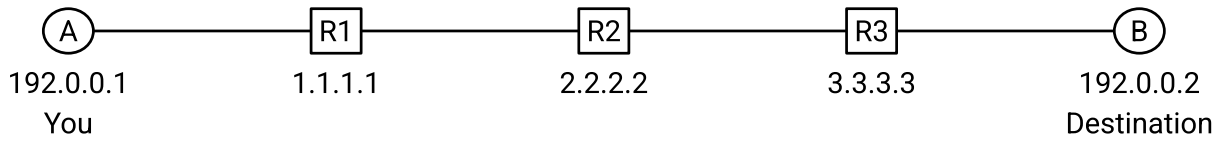
3.4 If an intermediate router generates an ICMP Time Exceeded response to a packet, what value does the router set in the TTL field of the new ICMP-over-IP packet?

What is the value of the TTL set in the copy of the original packet's IPv4 header (inside the ICMP payload)?

1

Motivating Traceroute: Now, let's see how we can send UDP packets and leverage ICMP Time Exceeded responses to reveal information about intermediate routers that are forwarding our packets.

Consider the following network topology. We are end host A, and we wish to perform traceroute to discover the routers along the path to end host B.



For each outgoing UDP packet we send, describe the packet we will receive from other devices in the network. Do we receive an ICMP-over-IP packet, a UDP-over-IP packet, or something else? What are the header fields (e.g. source/destination IP, ICMP type) in the packet we receive?

For now, assume all routers work as intended, with no router or network errors (e.g. no packets are dropped).

3.5 You send a UDP-over-IP packet with:

- Source IP = A
- Destination IP = B
- Source Port = 53201
- Destination Port = 33434
- TTL = 1

ICMP TTL Exceeded response with source IP 1.1.1.1

3.6 You send a UDP-over-IP packet with:

- Source IP = A
- Destination IP = B
- Source Port = 53201
- Destination Port = 33434
- TTL = 2

ICMP TTL Exceeded response with source IP 2.2.2.2

3.7 You send a UDP-over-IP packet with:

- Source IP = A
- Destination IP = B
- Source Port = 53201
- Destination Port = 33434
- TTL = 3

ICMP TTL Exceeded response with source IP 3.3.3.3

3.8 You send a UDP-over-IP packet with:

- Source IP = A
- Destination IP = B
- Source Port = 53201
- Destination Port = 33434
- TTL = 4

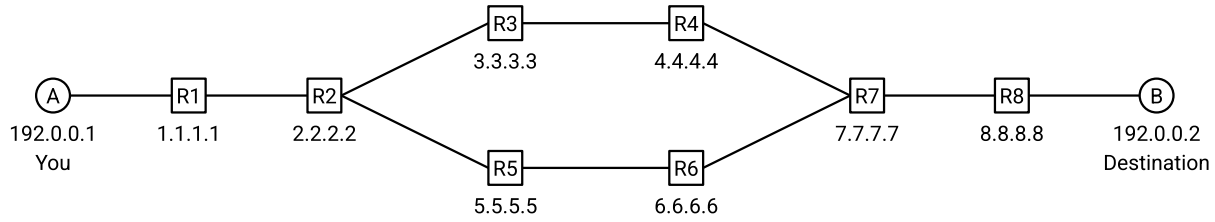
ICMP Destination Unreachable response with source IP 192.0.0.2

You should have observed in the previous part that the packets we received in response to our outgoing "probes" reveal the IP addresses of the intermediate routers along the path to B. This observation forms the basis of traceroute's design!

Here's some high-level pseudocode for traceroute. It iterates through increasing TTL values, and sends 3 probes for each TTL. It then receives responses to the probes, and logs any relevant results.

```
for ttl in range(1, 65):
  ttl_result = []  # empty set to log responses
  for _ in range(3):
    send packet to dst_ip with TTL=ttl
    wait for a corresponding valid/relevant response

    if no valid/relevant response received:
      add an empty address to ttl_result

    if valid/relevant ICMP Time Exceeded response received:
      add source IP of that packet to ttl_result

    if valid/relevant ICMP Destination Unreachable received:
      add dst_ip to ttl_result
      exit outer loop

  print ttl_result  # responses to all probes at this TTL
```

Consider running the pseudocode above on the network below.



You are host A, and you run traceroute with a destination of host B.

For now, assume all routers work as intended, with no router or network errors (e.g. no packets are dropped), with the following exceptions:

- When R2 receives packets from A, it alternates between forwarding the packet to R3, forwarding the packet to R5, and dropping the packet.
- R4 is configured to not send any ICMP responses.

3.9 In the table below, fill in the IP addresses discovered at each TTL. If no response is received for a specific probe, write * in the box.

TTL	Probe 1	Probe 2	Probe 3
1	1.1.1.1	1.1.1.1	.1.1.1.1
2	2.2.2.2	2.2.2.2	2.2.2.2
3	3.3.3.3	5.5.5.5	*
4	*	6.6.6.6	*
5	7.7.7.7	7.7.7.7	*
6	8.8.8.8	8.8.8.8	*

Note on network errors: In the above examples, we assumed that packets would never be delayed, corrupted, or duplicated in-transit. In real life, these errors could happen.

In the traceroute project, your job is to ensure that when reporting results for TTL i, your code ignores ICMP responses that were *not* generated in response to the probes sent at TTL i.

A question for you to consider: Traceroute probes and responses from earlier TTLs (i.e. TTL $\leq i - 1$) can be delayed or duplicated. How can we check that each ICMP response we receive at TTL i is actually generated in response to probes sent at TTL i, and not in response to probes sent at earlier TTLs?

It is not sufficient to check if the entire ICMP payload is a duplicate of an earlier ICMP payload. It's possible that you send two duplicate probes, but receive two distinct ICMP responses. This could happen if, for example, the probes are forwarded along different paths in the network. (Question for you to consider: What header fields end up different in this case?)