# IPv4 in Practice: The Life of a Packet

In this lesson, we'll consolidate everything we have learned about the network layer so far by tracing the journey of a packet.

**We'll cover the following**



At this point in the description of IPv4, it is useful to have a detailed look at how an IPv4 implementation sends, receives and forwards IPv4 packets.

The simplest case is when a host needs to send a transport layer segment in an IPv4 packet. In order to do so, it performs two operations.

1. First, it must **decide on which interface the packet will be sent**.
2. Second, it must **create the corresponding IP packet(s)**.

To simplify the discussion in this section, we ignore the utilization of IPv4 options. This is not a bad idea as most of the traffic today consists of IP packets that don’t make any use of IP options. Furthermore, we also assume that only point-to-point links are used.

An IPv4 host with $n$ data link layer interfaces manage $n + 1$ IPv4 addresses:

- The $127.0.0.1/32$ IPv4 address assigned by convention to its loopback address.
- One $A.B.C.D/p$ IPv4 address assigned to *each* of its $n$ data link layer interfaces.

- The host maintains a forwarding table that contains one entry for its loopback address and one entry for each subnet identifier assigned to its interfaces.
- Furthermore, the host usually uses one of its interfaces as the default interface when sending packets that are not addressed to a directly connected destination. This is represented by the default route: 0.0.0.0/0 that is associated with one interface.

## Sending a Packet #

- When a transport protocol running on the host requests the transmission of a segment, it usually provides the IPv4 destination address to the IPv4 layer in addition to the segment.
- The IPv4 implementation first performs a longest prefix match with the destination address in its forwarding table. The lookup returns the identification of the interface that must be used to send the packet.
- The host can then create the IPv4 packet that contains the segment! The source IPv4 address of the packet is the IPv4 address of the host on the interface returned by the longest prefix match.
- The **Protocol** field of the packet is set to the identification of the local transport protocol which created the segment.
- The **TTL** field of the packet is set to the default TTL used by the host.
- The host must now choose the packet's **Identification**. This Identification is important if the packet becomes fragmented in the network, as it ensures that the destination is able to reassemble the received fragments.
    - Ideally, a sending host should never send a packet twice with the same identification to the same destination host, in order to ensure that all fragments are correctly reassembled by the destination.
    - Unfortunately, a $16$-bit Identification field and an expected MSL of $2$ minutes, and maximum packet size of $65535$ implies that the maximum bandwidth to a given destination is limited to roughly $286$ Mbps. Here's the derivation:
        - The identification field is 16 bits so there can be $65536$ ($2^{16}$)

different unique packets between a specific client-server pair.

    - Hence, at most $65536$ unique packets may be sent in one maximum segment lifetime. Then, another $65536$ packets in the next MSL and so on.
    - Suppose you are sending $65535$ byte packets (the maximum size), then you are transmitting
      $65535 \times 8 \times 65536 = 34,359,214,080$ bits per MSL.
    - Assuming that the MSL is $120$ seconds long, that implies a rate of $34,359,214,080/120$ bits per second which is $\approx 286$ Mbps.

However, with a more realistic $1500$ bytes MTU, that bandwidth drops to $6.4$ Mbps to make fragmentation possible. This is very low and is another reason why hosts are highly encouraged to avoid fragmentation. If despite all of this, the MTU of the outgoing interface is smaller than the packet's length, the packet is fragmented.

- Finally, the packet's checksum is computed before transmission.

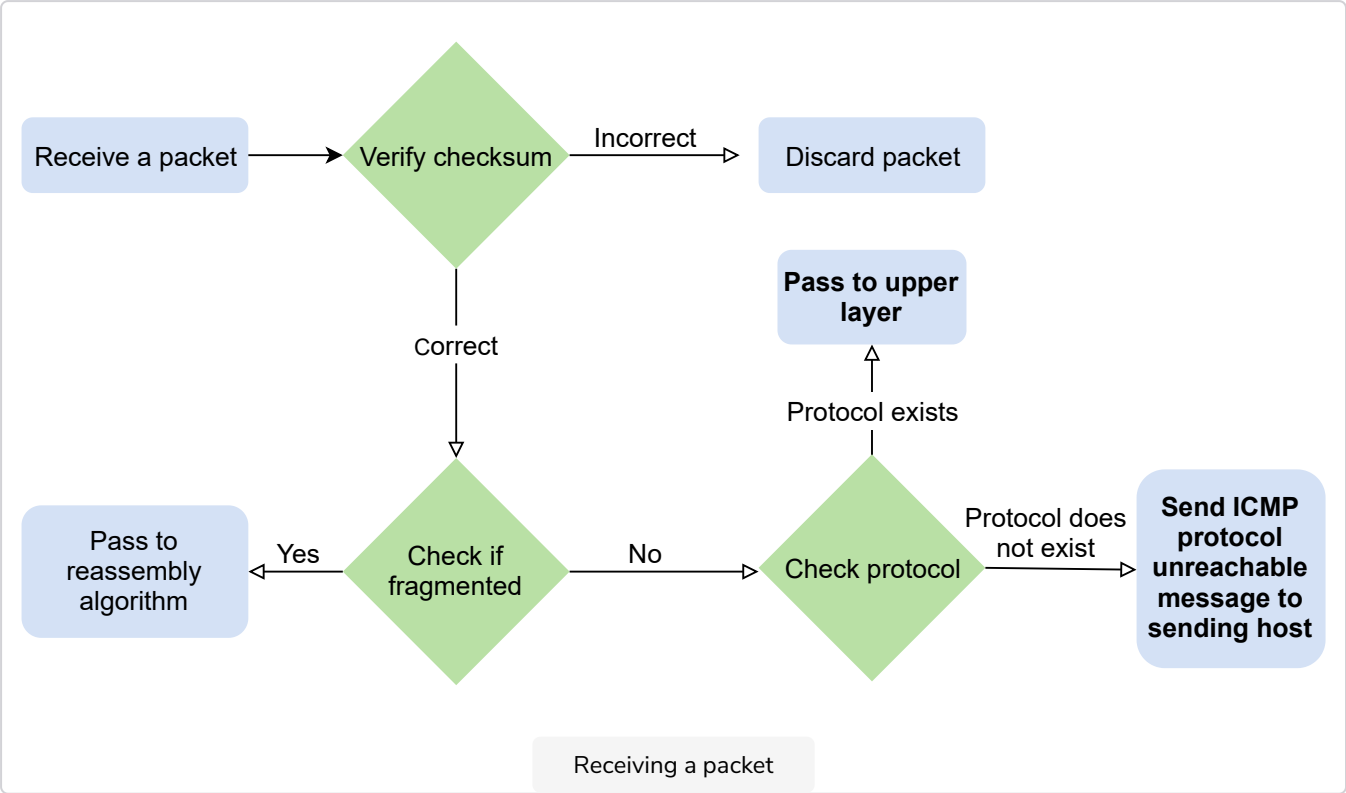
## Receiving A Packet #

When a host receives an IPv4 packet destined to itself, there are several operations that it must perform.

- First, it must check the packet's **checksum**. If the checksum is incorrect, the packet is discarded.

- Then, it must check whether the packet has been fragmented. If yes, the packet is passed to the reassembly algorithm described earlier. Otherwise, the packet must be passed to the upper layer. This is done by looking at the Protocol field (6 for TCP and 17 for UDP).

- If the host doesn't implement the transport layer protocol corresponding to the received Protocol field, it sends a **Protocol unreachable ICMP message** to the sending host.




Receiving a packet

## If ICMP is Received #

If the received packet contains an ICMP message (with the protocol field set to 1), the processing is more complex.

- An **Echo-request ICMP message** triggers the transmission of an ICMP Echo-reply message.
- The other types of ICMP messages, except for ICMP Echo Response, indicate an error that was caused by a previously transmitted packet. These ICMP messages are usually forwarded to the transport protocol that sent the erroneous packet. This can be done by inspecting the contents of the ICMP message that includes the header and the first 64 bits of the erroneous packet.
- If the IP packet did not contain options, which is the case for most IPv4 packets, the transport protocol can find in the first 32 bits of the transport header the source and destination ports to determine the affected transport flow. This is important for Path MTU discovery for example.

## How Routers Handle Packets #

When a router receives an IPv4 packet, it must:

- First check the packet's checksum. If the checksum is invalid, it's discarded.
- Otherwise, the router must check whether the destination address is one of the IPv4 addresses assigned to the router. If so, the router must behave as a host and process the packet as described above. Although routers mainly forward IPv4 packets, they sometimes need to be accessed as hosts by network operators or network management software.
- If the packet is not addressed to the router, it must be forwarded on an outgoing interface according to the router's forwarding table.
- The router first decrements the packet's TTL.
    - If the TTL reaches 0, a TTL Exceeded ICMP message is sent back to the source.
    - As the packet header has been modified, the checksum must be recomputed. Fortunately, as IPv4 uses an arithmetic checksum, a router can incrementally update the packet's checksum.
- Then, the router performs a longest prefix match for the packet's destination address in its forwarding table.
    - If no match is found, the router must return a **Destination unreachable ICMP** message to the source.
    - Otherwise, the lookup returns the interface over which the packet must be forwarded.
- Before forwarding the packet over this interface, the router must first compare the length of the packet with the MTU of the outgoing interface.
    - If the packet is smaller than the MTU, it is forwarded.
    - Otherwise, a **Fragmentation needed ICMP message** is sent if the DF flag was sent or the packet is fragmented if the DF was not set.

## Quick Quiz! #

1 When is an ICMP protocol unreachable message sent?

A) When the selected protocol in the packet is not available

B) When the selected protocol is TCP

C) When the host is unavailable

COMPLETED 0%

1 of 3

In the next lesson, we'll study IPv6!