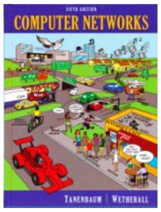
# Solutions – IPv4



*The solutions below are based on our capture and use of tools. Your answers will differ in the details if they are based on your own capture and use of tools in a different network setting. Nonetheless, we expect our solutions to help you understand whether your answers are correct.*

## Step 3: IP Packet Structure

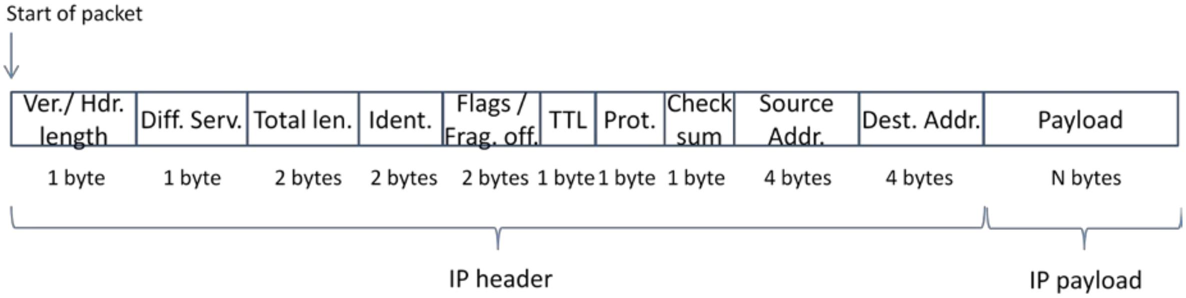


Figure 1: Structure of an IP packet

This drawing differs from the text drawing in Fig. 5-46 in only minor respects:

- There are no IP options in the IP packets we observe for regular traffic.
- The figure gives sizes in bytes, so bit fields are combined (specifically version and header length, and flags and fragment offset).
- The wording is slightly different in ways that are not material, e.g., Wireshark says "Header length" versus "IHL" for IP header length.

The answers to the questions are:

1. For our trace, the IP address of the computer is 128.208.2.151 and the IP address of the server (www.uwa.edu.au) is 130.95.128.40. To find these values, look in the source and destination addresses of a packet sent from your computer to the server, such as the GET. As a check, the destination IP address should also appear in the traceroute output.
2. Yes, the total length covers both IP header and IP payload. You can confirm this by looking at the sizes of the overall packet and various headers.
3. Identification changes on each new packet. It is set independently by each sending computer, so it is not related in different directions. Often, it is implemented as a simple counter, in which case this should be apparent by looking at successive packets. On some operating systems, it is changed in pseudo-random ways.
4. In our case the initial value is 128. It is the maximum of 255 on some computers, but more commonly it is set to a smaller initial value such as 128 or 64. It only needs to be sufficient for the packet to cross the Internet.
5. A packet cannot be a fragment if the Don't Fragment bit is set. If not set, a packet is not a fragment if the fragment offset is zero and the More Fragments bit is zero. These settings imply that

the packet is complete. Conversely if either the fragment offset is not zero or the More Fragments bit is set then another part is needed, so the packet must be a fragment. Note that the total length cannot be used to tell if a packet is a fragment as it is changed to give the length of fragments.

6. The header length of our packets is 20 bytes; your header will be 20 bytes or possibly longer. But with 4 bits we can only encode values up to 16. Therefore the header length is not given in bytes. In fact, it is given in multiples of 4 bytes or 32 bits, so the value 5 means 5x4 or 20 bytes.

## Step 4: Internet Paths

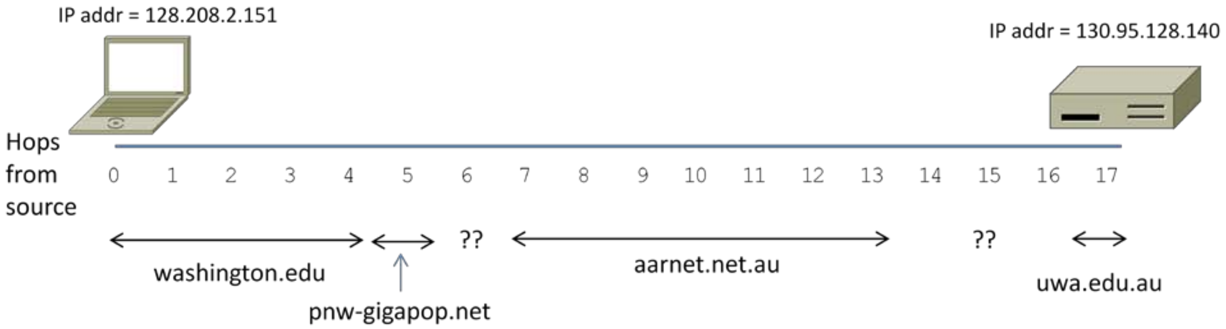
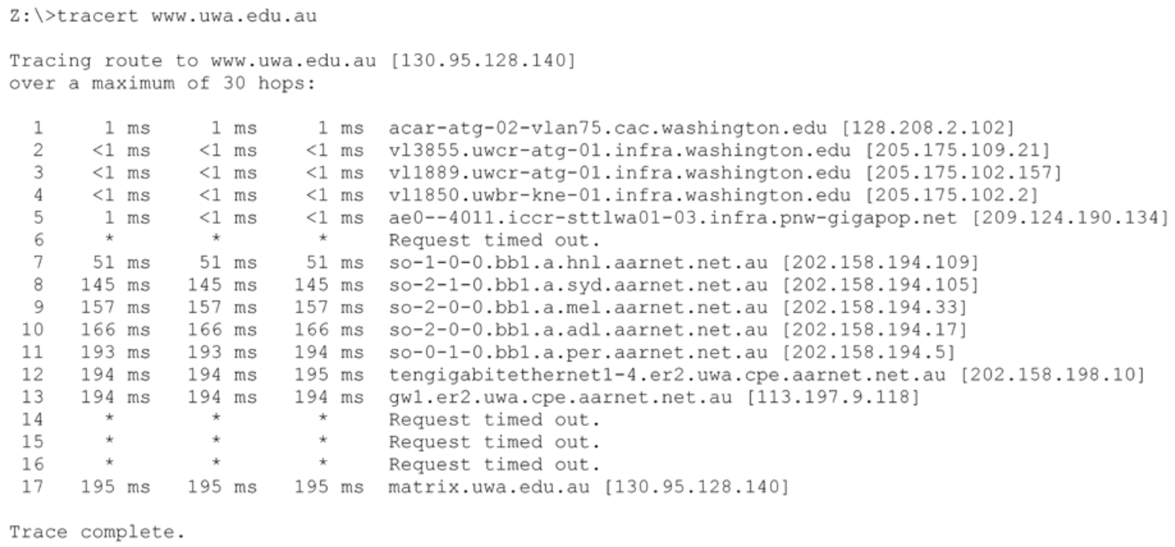


Figure 2: Internet path from our computer to the remote server

Your figure will differ in the details, but is also likely to show a path going through more than a dozen routers and several organizations. There are several features to note:

- The sketch shows 17 router hops because this matches the output of the traceroute.
- The ranges are taken from the DNS names of routers. They are a guess using the last two or three components of the name. The "??" denote ranges for which traceroute could not identify the router. If we wanted to translate names to real-world organizations we could search the web.

For reference, the traceroute output (from a Windows box) that was used is given in below.



```
Z:\>tracert www.uwa.edu.au

Tracing route to www.uwa.edu.au [130.95.128.140]
over a maximum of 30 hops:

  1     1 ms     1 ms     1 ms  acar-atg-02-vlan75.cac.washington.edu [128.208.2.102]
  2    <1 ms    <1 ms    <1 ms  vl3855.uwcr-atg-01.infra.washington.edu [205.175.109.21]
  3    <1 ms    <1 ms    <1 ms  vl1889.uwcr-atg-01.infra.washington.edu [205.175.102.157]
  4    <1 ms    <1 ms    <1 ms  vl1850.uwbr-kne-01.infra.washington.edu [205.175.102.2]
  5     1 ms    <1 ms    <1 ms  ae0--4011.iccr-sttlwa01-03.infra.pnw-gigapop.net [209.124.190.134]
  6     *        *        *     Request timed out.
  7    51 ms    51 ms    51 ms  so-1-0-0.bb1.a.hnl.aarnet.net.au [202.158.194.109]
  8   145 ms   145 ms   145 ms  so-2-1-0.bb1.a.syd.aarnet.net.au [202.158.194.105]
  9   157 ms   157 ms   157 ms  so-2-0-0.bb1.a.mel.aarnet.net.au [202.158.194.33]
 10   166 ms   166 ms   166 ms  so-2-0-0.bb1.a.adl.aarnet.net.au [202.158.194.17]
 11   193 ms   193 ms   194 ms  so-0-1-0.bb1.a.per.aarnet.net.au [202.158.194.5]
 12   194 ms   194 ms   195 ms  tengigabitethernet1-4.er2.uwa.cpe.aarnet.net.au [202.158.198.10]
 13   194 ms   194 ms   194 ms  gw1.er2.uwa.cpe.aarnet.net.au [113.197.9.118]
 14     *        *        *     Request timed out.
 15     *        *        *     Request timed out.
 16     *        *        *     Request timed out.
 17   195 ms   195 ms   195 ms  matrix.uwa.edu.au [130.95.128.140]

Trace complete.
```

Figure 3: Output of `tracert` showing the router-level path

## Step 5: IP Header Checksum

Here is an example checksum calculation for one of our packets (packet #17 of the supplied trace). Your sum will have different numbers but a similar form. Note that you must choose a packet with a non-zero checksum; the zero checksum packets have the checksum stripped out.

```
 4500      version, header len, diff. serv
 058c      total length
 cadd      identification
 4000      flags, fragment offset
 ef06      ttl, protocol
 353b      header checksum
 825f      source addr
 808c      source addr cont.
 80d0      dest addr
+0297      dest addr cont.
-----
3fffc

 fffc
+   3      add back overflow bits
-----
 ffff      desired result – complement of zero
```

[END]