Next Previous Contents

8. What Packets Look Like

For the exceptionally curious (and the curiously exceptional), here is a description of what a packet actually looks like. There are several tools which watch what packets are passing in and out of your Linux box: the most common one is `tcpdump' (which understands more than TCP these days), but a nicer one is `ethereal'. Such programs are known as `packet sniffers'.

The start of each packet says where it's going, where it came from, the type of the packet, and other administrative details. This part is called the `packet header'. The rest of the packet, containing the actual data being transmitted, is usually called the `packet body'.

So any IP packet begins with an `IP header': at least 20 bytes long. It looks like (this diagram stolen shamelessly from RFC 791):

```
.-------+-------+---------------+-------------------------------.
|Version|  IHL  |Type of Service|          Total Length         |
|-------+-------+---------------+-------------------------------|
|         Identification         |Flags|      Fragment Offset    |
|---------------+---------------+-------------------------------|
|  Time to Live |    Protocol   |         Header Checksum       |
|---------------+---------------+-------------------------------|
|                       Source Address                          |
|---------------------------------------------------------------|
|                    Destination Address                        |
`---------------------------------------------------------------'
```

The important fields are the Protocol, which indicates whether this is a TCP packet (number 6), a UDP packet (number 17) or something else, the Source IP Address, and the Destination IP Address.

Now, if the protocol fields says this is a TCP packet, then a TCP header will immediately follow this IP header: the TCP header is also at least 20 bytes long:

```
.-------------------------------+-------------------------------.
|          Source Port          |       Destination Port        |
|-------------------------------+-------------------------------|
|                        Sequence Number                        |
|---------------------------------------------------------------|
|                    Acknowledgment Number                      |
|-------------------+-+-+-+-+-+-+-------------------------------|
|  Data |           |U|A|P|R|S|F|                               |
| Offset| Reserved  |R|C|S|S|Y|I|            Window             |
|       |           |G|K|H|T|N|N|                               |
|-------+-----------+-+-+-+-+-+-+-------------------------------|
|           Checksum            |         Urgent Pointer        |
`---------------------------------------------------------------'
```

The most important fields here are the source port, and destination port, which says which service the packet is going to (or coming from, in the case of reply packets). The sequence and acknowledgement numbers are used to keep packets in order, and tell the other end what packets have been received. The ACK, SYN, RST and FIN flags (written downwards) are single bits which are used to negotiate the opening (SYN) and closing (RST or FIN) of connections.

Following this header comes the actual message which the application sent (the packet body). A normal packet is up to 1500 bytes: this means that the most space the data can take up is 1460 bytes (20 bytes for the IP header, and 20 for the TCP header): over 97%.

Next Previous Contents