TCP/IP

General description of the TCP/IP protocols

- TCP/IP is a layered set of protocols.
    - protocol to communicate reliably between the two computers.
    - TCP/IP is based on the "catenet model". (This is described in more detail in IEN 48.) This model assumes that there are a large number of independent networks connected together by gateways. The user should be able to access computers or other resources on any of these networks.
    - Datagrams will often pass through a dozen different networks before getting to their final destination. The routing needed to accomplish this should be completely invisible to the user. As far as the user is concerned, all he needs to know in order to access another system is an "Internet address". This is an address that looks like 128.6.4.194. It is actually a 32-bit number. However it is normally written as 4 decimal numbers, each representing 8 bits of the address. Of course we normally refer to systems by name, rather than by Internet address. When we specify a name, the network software looks it up in a database, and comes up with the corresponding Internet address. Most of the network software deals strictly in terms of the address. (RFC 882 describes the name server technology used to handle this lookup.)
        - Information is transfered as a sequence of "datagrams".
            - A datagram is a collection of data that is sent as a single message.
            - Each of these datagrams is sent through the network individually
    - TCP/IP is built on "connectionless" technology.

.

The Protocols

- TCP (the "transmission control protocol")
    - responsible for breaking up the message into datagrams
        - TCP breaks it up into manageable chunks. (In order to do this, TCP has to know how large a datagram your network can handle. Actually, the TCP's at each end say how big a datagram they can handle, and then they pick the smallest size.)
            - TCP puts a header at the front of each datagram.
                - contains at least 20 octets,
                    - most important ones

- source and destination "port number"
- "sequence number".
    - Each datagram has a sequence number.
    - the other end can make sure that it gets the datagrams in the right order, and that it hasn't missed any. (See the TCP specification for details.) TCP doesn't number the datagrams, but the octets.
    - Checksum. This is a number that is computed by adding up all the octets in the datagram (more or less - see the TCP spec). The result is put in the header. TCP at the other end computes the checksum again. If they disagree, then something bad happened to the datagram in transmission, and it is thrown away. So here's what the datagram looks like now.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        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        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   your data ... next 500 octets                             |
|   ......                                                    |
```

-
  - reassembling them at the other end, resending anything that gets lost
  - putting things back in the right order.

- IP (the "internet protocol")
  - TCP sends each of these datagrams to IP.
  - IP's job is simply to find a route for the datagram and get it to the other end. In order to allow gateways or other intermediate systems to forward the datagram, it adds its own header.
  - responsible for routing individual datagrams.
  - A connection may require the datagram to go through several networks
  - Keeping track of the routes to all of the destinations and handling incompatibilities among different transport media turns out to be a complex job.
- Interface between TCP and IP
  - fairly simple.
  - TCP simply hands IP a datagram with a destination.
  - IP doesn't know how this datagram relates to any datagram before it or after it.

Internet Addresses

- Internet addresses are 32-bit numbers, normally written as 4 octets (in decimal), e.g. 128.6.4.7.
  - There are actually 3 different types of address.
    - address has to indicate both the network and the host within the network
    - Addresses beginning with 1 to 126 use only the first octet for the network number.
      - But few normal organizations get one of these "class A" addresses.
      - class B" addresses are used. Class B addresses use the first two octets for the network number. Thus network numbers are 128.1 through 191.254. (We avoid 0 and 255, for reasons that we see below. We also avoid addresses beginning with 127, because that is used by some systems for special purposes.) The last two octets are available for host addesses, giving 16 bits of host address. This allows for 64516 computers, which should be enough for most

organizations. (It is possible to get more than one class B address, if you run out.)

    - Finally, class C addresses use three octets, in the range 192.1.1 to 223.254.254. These allow only 254 hosts on each network, but there can be lots of these networks. Addresses above 223 are reserved for future use, as class D and E (which are currently not defined).

- Because 0 and 255 are used for unknown and broadcast addresses,
    - normal hosts should never be given addresses containing 0 or 255.
    - Addresses should never begin with 0, 127, or any number above 223.