Distributed Software Development

TCP/IP overview

Chris Brooks

Department of Computer Science

University of San Francisco
Goal: Understand how a network transmits messages at different layers.

How is a network composed?

What really happens when Firefox opens a connection to a web server?

Note: this will be an overview: For more details, you should take Prof. Buckwalter’s class.
Modern network design takes advantage of the idea of *layering*

A particular service or module is constructed as a black box.

Users of that service do not need to know its internals, just its interface.

This makes it easy to later build new modules (or layers) that use the lower layers.

For example, HTTP is built on top of TCP.

△ A web browser does not typically need to worry about the implementation of TCP, just that it works.

Unlike modules in a typical OO system, the layers in a networked system comprise protocols that span multiple machines.
ISO (a standards body) developed a reference model called OSI that defines the different layers needed for communication, and specifies which should do each job.

The goal is to produce an open protocol that allows for heterogeneous, extensible systems.

A protocol is a specification describing the order and format of messages.

An open protocol is one in which all of this information is publicly available.
2b-3: The OSI seven-layer model

- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical
An application (such as a web browser) wants to send a message to another computer.

That application constructs a message and passes it to the application layer.

The application layer attaches a header to the message and passes it to the presentation layer.

The presentation layer attaches a header and passes it to the session layer, and so on.

On the other end, the message is received by the physical layer, who strips off the appropriate header and passes the message up to the data link layer.

This continues until the message reaches the application layer of the receiving machine.

High-level layers don’t need to worry about lower-level layers.
This is the lowest-level layer, responsible for transmitting 0s and 1s.

Govern transmission rates, full or half-duplex, etc.

A modem works at the physical layer.

Lots of interesting problems at this level that we won’t get into ...
The data link layer provides error handling for the physical layer.

Individual bits are grouped together into frames.

A checksum is then computed to detect transmission errors.

The data link layer can then request a retransmission of an error is detected.

Messages are numbered; receiver can request re-transmission of any message in a sequence.

Each frame is a separate, distinct message.

The Data link layer provides error-free transmission to upper-level layers.
The network layer is responsible for routing and flow control.

The network layer removes the data link header and examines the resulting packet for a destination, and then forwards it as appropriate.

The Internet Protocol (IP) is one of the best-known network-layer protocols.

Primary role: move packets from a sending host to a receiving host. This involves:

- **Routing**: determine the path that a packet should take to get to its destination.
- **Forwarding**: When an incoming packet is received, place it on the output link that takes it to the next hop in its route.
A router contains a *forwarding table* - when an incoming packet is received, the router compares it to this table to determine where to send it next.

- This is forwarding.

These forwarding tables are configured by means of a routing algorithm.

For example, the link-state algorithm is a version of Djikstra’s algorithm - this computes a global routing table.

Internet routing algorithms (such as BGP) are more complex, and use a decentralized routing table.

In a nutshell, BGP lets subnets figure out how to reach other subnets via a gateway. That gateway is then responsible for routing within the subnet.
The network layer still operates at the level of individual packets, or datagrams.

Packets may get lost, or arrive out of order.

TCP is a transport-level protocol that provides *connection-oriented* service.
- Guaranteed, in-order delivery.
- State is maintained.

This layer will also manage quality-of-service and some congestion control.

UDP is also a transport level protocol, albeit one that does not provide connection-oriented delivery.
The session layer was designed to provide support for access rights and synchronization.

In practice, it is not widely used, and is not present in the TCP/IP suite.
2b-11: Presentation Layer

- The presentation layer controls display of packet information.
- This may include encryption/decryption, compression, translation between character formats.
This is the layer that most of us are most familiar.

It consists of user-level protocols built on top of the existing layers.

- HTTP
- FTP
- SMTP
- P2P protocols
- Instant messaging
- RTSP/streaming video
- etc.
HTTP is the protocol that drives the Web.
△ A side note/axe to grind: WWW != Internet!!

It is a stateless protocol that uses TCP as its underlying protocol.
△ The client sends a request, which is processed by the server.
△ The server sends a reply, and the exchange is ended.
HTTP has a very simple message format.

GET /~brooks/index.html HTTP/1.1
Host: www.cs.usfca.edu
Connection: close
User-agent: Mozilla/4.0
Accept-language: en

You can try this out for yourself with telnet ...
There are lots of wrinkles and extensions to HTTP
  △ Cookies to help save state
  △ CGI, SOAP to execute code as the result of an HTTP request.
  △ Web caching to store data closer to clients.

These are all possible because HTTP is an open protocol.

This is also what makes it possible for different companies to write web browsers and web servers that seamlessly work together.
the modern networking stack can be conceptually broken into a set of layers.

Each layer has a specific, well-defined function.
- Acts as a black box

Higher-level layers build on the functionality of lower-level layers.

We’ll be primarily concerned with the Transport and Application layers.