Lecture Notes: The Internet

The Internet

The Internet is a collection of interconnected networks. Each network is independent from its neighbours, and so ultimately, no one country or organization "owns" the Internet.

For instance, the computer network in the lab is connected to the overall network at the College, which connects to the YTG network, which connects to the NorthwesTel network for the North, which connects to networks in the rest of Canada, North America, and the World.

As do most technological advances, the Internet began as a military project. The story goes that its initial design requirement was to be nuclear blast-proof. It probably wasn't designed that way, but the Internet is very reliable even in the face of widespread outage. This is because it has no single point of failure: whole countries can go offline while outside the borders the Internet hums along (perhaps with degraded performance, but messages can still find a way to get through).

The Internet's design also mandates that all hosts (computers) connected through the network are equal, or peers. This facet of the design permits the "no single point of failure" goal since there are no easily-targeted "master" servers required to keep traffic moving.

An illustration of the growth of the Internet: in 1969, only 4 computers made up the then ARPANet; by 1971, the Internet had grown to 15 computers (the first e-mail message was sent in 1972); in 1973 the number was 37; skipping ahead to January 2002, the estimate ballooned to 147 million computers; by January 2004, 233 million; and by July of 2006, 439 million. Of course, like all statistics, these figures should be taken with a grain of salt: some estimates are twice as high.

Internet Layers

In order to understand a little better how the Internet actually works, we can divide it into four layers of responsibility. Starting with the hardware and moving upwards, the layers are: Physical, Address, Transport, and Application. (Technical references often divide networks into 7 layers, and use different names, but these 4 layers should be sufficient for our level of inquiry.)

Each of the layers on a single computer (or host) connected to the Internet communicates with the corresponding layer on other hosts. This communications is carried out through a number of protocols. A protocol is just the set of rules for such a conversation (a quick example that we're familiar with is that people are expected to say "Hello" or something similar, when answering the telephone).

Physical Layer

The physical layer is just that: physical. It contains all of the stuff we can see and touch, together with the protocols that allow this hardware to communicate with other hardware.

Examples of physical layer hardware: modems, network cables, network interface cards (NIC), wireless cards and antennae (for Bluetooth or Wi-Fi), Ethernet hubs (Ethernet is the most popular physical layer protocol, and is the one used in the College labs), and now even G3 cell phones can connect to data networks.

The protocols at this layer just make sure data can pass between these pieces of hardware; they have no concept of how this hardware is strung together to form the larger Internet.

When speaking of the physical layer, the most often bandied-about term is bandwidth, meaning the amount of raw data that can travel through the wires in a given measure of time, usually bits per second, or "bps". A slow telephone modem runs at about 28Kbps (thousand bits per second), faster modems at up to 56Kbps. A cable or DSL modem can run between 500Kbps and 5Mbps (million bits per second). Wireless (Wi-Fi) networks usually runs at 54Mbps. And an internal network like in the College labs can run between 10Mbps and 100Mbps.

Address Layer

The address layer is responsible for figuring out where a specific host is on the Internet, just given its address. A reasonable analogy would be the post office, who somehow figure out how to deliver mail around the world using only the address printed on an envelope.

The address layer also breaks single messages into bite-sized chunks. These chunks, or packets as they're often called (also streams, segments, and frames), are sent across the network(s) independently. This means that the
many packets that make up a single message may take completely different routes through the Internet from source to destination. This design better ensures reliable delivery of information in the face of Internet outages, but requires the transport layer (see next) to perform some extra work to guarantee that the multi-packet message arrives intact.

The address layer doesn’t pre-calculate the entire route for a packet to take on the Internet. Instead it simply figures out what the next part of the route should be as the packet bounces from one server to the next as it moves from its source to its intended destination. In this way, whole sections of the Internet can go down (perhaps following a nuclear attack, but more likely due to blackouts) but messages can still get through as the address layer figures out alternative routes through Internet nodes that are still functioning.

Computers being computers, the addresses for hosts are all numbers. Every host connected to the Internet has a unique address number, which allows any computer anywhere in the world to communicate with precisely one other computer using just this address. This is no different from North American 10-digit telephone numbers: no two are the same. If two people did have the same telephone number, how could you contact just one of them?

The protocol that carries out this addressing is called the Internet Protocol (IP), and so these numeric addresses are called IP addresses. You most likely have run into these addresses before; they are composed of a sequence of four numbers that range between 0 and 255. For example, the IP address for the computer you are connected to as you read this is: 199.247.245.153. These numbers are handed out by ISPs and telephone companies. Most Yukon IP addresses begin with 199.247. The IP address for the Yukon College's webserver is 199.247.245.8.

These numerical IP addresses are terribly inconvenient for humans. Fortunately, we can represent these addresses in a text form called a Uniform Resource Locator (URL). See the section below for more about URLs.

**Transport Layer**

The transport layer’s responsibility is to ensure that messages arrive intact at the destination host.

The most common protocol running in the transport layer is the Transport Control Protocol (TCP). The combination name TCP/IP is often used as the general name for the protocols underlying the Internet.

There is also the User Datagram Protocol (UDP), but it is not used as often and doesn’t make the same reliability guarantees as does TCP.

TCP carries out the transport layer’s tasks by guaranteeing that all packets in a message arrive at the destination, in the proper order, and without any data corruption. The order of the packets is important if you consider an Internet message that transmits video, for example. If the packets' order is shuffled, bits of the video would appear in the wrong order and the overall effect would be unwatchable (unlike, say, Memento or Pulp Fiction, which both reorder the normal sequence of events, but in an eminently watchable fashion).

TCP can define multiple Internet services on a single computer. The Internet Protocol will get messages to a specific computer, but these transport layer protocols are required to direct the message to the proper program running on that computer. They do this by assigning numeric ports (0 through 65535) to certain services. So, if a single computer is running both a webserver program and an e-mail server program, web requests will be directed by TCP to the port used by the webserver program, and e-mail messages will be directed by TCP to the port used by the e-mail server program.

Compare the transport layer’s idea of ports to telephone extension numbers in a business office. There may just be one main telephone number (the IP address) but many extension numbers (ports) for the various departments in the business.

**Application Layer**

At the very top of the Internet layer "cake" is the application layer. The protocols in this layer deliver all of the Internet services we’re familiar with:

- HTTP (HyperText Transfer Protocol, TCP port 80): transmits web pages across the Internet
- FTP (File Transfer Protocol, TCP ports 20 & 21): used for moving files from one machine to another
- SMTP (Simple Mail Transfer Protocol, TCP port 25): for sending e-mail messages
- POP3 (Post Office Protocol, version 3, TCP port 110): for retrieving e-mail messages from a e-mail server
- NNTP (Network News Transfer Protocol, TCP port 119): network newsgroups
- Telnet (TCP port 23): allows users to log on to remote computers just as if they were sitting in front of the
remote computer

- DNS (Domain Name Service, UDP port 53): translates the machine name part of the URL into IP addresses (DNS is the only one of the protocols listed here that used UDP as its transport layer protocol)

There are hundreds, if not thousands, of such protocols, but those are some of the most common ones.

Each protocol is designed specifically for its task: HTTP retrieves web pages immediately; SMTP eventually gets the e-mail to its destination but doesn't require that it get there right away, FTP is tuned for transferring large files, but isn't as fast as HTTP when it comes to smaller files.

**Uniform Resource Locators (URLs)**

Humans being humans, prefer not to have to remember IP addresses like "17.112.152.32". Instead, we'd rather use a little piece of text like "www.apple.com", which is called a **Uniform Resource Locator**, or **URL** (pronounced "You-Are-Elle", or sometimes "Earl"). In fact, both 17.112.152.32 and www.apple.com refer to the same Internet host: Apple Computer's webserver. Don't believe me? Try just entering **17.112.152.32** into a browser and see where you end up.

URL's are used for more than just websites; they're also used for e-mail addresses, FTP servers, network newsgroups, or any other service running on a computer connected to the Internet.

When we type in a URL, an application-layer service called the **Domain Name Service (DNS)** translates the human-friendly URL into the computer-friendly IP address for us. DNS performs this translation by consulting the databases maintained by the Domain Name Registrars. DNS is responsible for translating "www.apple.com" into "17.112.152.32" whenever we type that into a browser address bar. DNS is the "phone book" of the Internet: look up a name, and find its number.

A URL is made up of many parts. For example, consider the following:

```
```

- **http** is the application-layer protocol. In this case HTTP, the protocol used by the World Wide Web. If you omit the protocol when typing a URL into a browser address box, the browser assumes you meant "http://". (Technically, the whole string "http://" is called the **scheme**.)
- **www.apple.com** is the **Fully Qualified Domain Name (FQDN)**. In turn, the FQDN is made up of pieces as well (from right to left, or most general to most specific):
  - **com** is the **Top Level Domain (TLD)**, in this case meaning "company" or "commercial".
  - **apple** is the specific company name, chosen by Apple Computer.
  - **www** is the name of the webserver hosting Apple Computer's website. Using "www" for the name of the webserver is merely a convention, and you will run into cases where "www" is not used (e.g. the course website at "cpsc100.yukoncollege.yk.ca").
- **/ipod/red/index.html** is the path to the requested HTML page on the webserver. This path and file is used by the webserver to figure out which file to return to a browser. Other application-layer protocols might not need a path or file name.

And now a more complex URL: **http://www.tc.gov.yk.ca/digitization/public/index.php**

- **http** is the HTTP protocol.
- **www.tc.gov.yk.ca** is the Fully Qualified Domain Name:
  - **ca** is the Top Level Domain (TLD), meaning "Canada" in this context. There are only a few designated Top Level Domains. See the section below for a description of these TLDs.
  - **yk** indicates a sub-federal geographical region, in this case the Yukon (why they used "yk" instead of "yt" is anybody's guess).
  - **gov** was chosen by YTG to mean "government".
  - **tc** is a department within YTG (Tourism & Culture in this case).
  - **www** is the host computer.
- **/digitization/public/index.php** is the path to the requested resource (page) on the webserver (PHP pages are just like HTML but with a bit of programming smarts).

A simple URL: **http://slashdot.org**

- **http** is the HTTP protocol.
- **slashdot.org** is the Fully Qualified Domain Name:
- org is the Top Level Domain (TLD), meaning an "organization".
- slashdot is the organization, and it was chosen by the fine folks at Slashdot:
- There is no host name. If no host name is specified, a default server at that domain is used. This just happens to be Slashdot's webserver, so this works fine.
- There is no path to a file. In this case, the default HTML document is selected from the root directory of the webserver.

**Top Level Domains**

There are only a few top level domains, and these are controlled by one of the Internet self-regulating authorities (ICANN [www.icann.org]).

The most common TLDs are as follows:

- .com: companies or commercial ventures
- .net: ISPs or other companies that work with the Internet
- .org: non-profit organizations and institutions
- .mil: US military
- .edu: US education institutions
- .gov: US government

Recently, ICANN specified some new TLDs that haven't really caught on much yet: .info, .biz, .pro, .museum, .aero, etc.

TLDs are also assigned to countries:

- .ca: Canada
- .us: United States
- .uk: United Kingdom
- .tv: the tiny South Pacific island nation of Tuvalu which bases most of its economy on the sale of domains using this TLD

Note that some of these TLDs can be split up into sub-federal regions: .yk.ca is the Yukon, .on.ca is Ontario.

The domain name part of the URL can be chosen by anyone (but you can't pick something that's already owned) and is registered with one of many companies called **Domain Name Registrars**. When you register a domain name, you really register the combination of domain name and TLD, so if someone's already registered yourname.com, you can always try for yourname.org or yourname.net or yourname.ca. Even though you're not really a Internet company nor an organization, you can still use those TLDs.