# Introduction to Networking and the Internet

## 1 *The Internet* vs. *internet*

This course is titled *Internet Programming*. What is the Internet? All of us use the Internet every day. Basically the Internet is a network.

People may use the term 'internet' in two different ways. 'The Internet', with the capital 'I', specially refers to the international computer network linking together thousands of individual networks around the world. People may also use 'internet', referring to any network comprised of two or more smaller networks that connect with one other in some way.

The term in 'Internet Programming' is of course the first kind of usage. So we, by learning *Internet Programming*, aim to know how to program, how to develop applications that make the communication and interaction between any two computers in the Internet possible.

## 2 Basic Networking

Before we discuss high-level Internet Programming techniques, let us first review how a computer network works at the low level so that even if you happened to have missed your network classes, you may still have a chance to know some points crucial to Internet Programming, or in another word, basics.

What is a network? Easily, we define it as a system connecting computers together. Each computer in the network has a unique identity system-wide so that data may be transmitted from a sender to a specified receiver. The send/receive operation is actually copying bytes from the sender to the receiver. This is how knowledge is widely spread nowadays in such a cheap way. That's also what the Hollywood stars and movie production companies worry about.

There are roughly two different networking technologies: *circuit switched* and *packet switched*.

## 2.1 Circuit Switched

Circuit switched is a technology traditionally used in the phone network, which is probably the only interconnected system comparable to the Internet. In the phone network, which is formally called PSTN[1], people communicate with each other by making calls. The "call" operation sets up an end-to-end connection, meaning there is a wire connection all the way between the phones.

The advantages of circuit switched include:

- Latency and bandwidth available don't vary during the call, which is important for some applications, e.g. live 2-way audio or video stream.
- If the call is long, the set up cost is relatively low since it's a one-time cost amortized over all the traffic for the call.
- "Routing" is not a problem during the call. It's all set up one time when the call is placed.

Of course it also has apparent disadvantages:

- If the call is short, the setup is relatively costly.
- The connection, once set up, is used exclusively. The resources is occupied even during periods of silence, which is potentially inefficient.

## 2.2 Packet Switched

Differently, packet switched is used for binary data transmission based on the fact that the sender may divide the data to be sent into a series of relatively small packets, which may be re-assembled at the destination into its original format.

To be directed to the destination and re-assembled in order, each packet is made of header and body. The header contains the addresses of both the sender and the receiver, as well as the sequential number. The body otherwise carries the data for the receiver, which may be text, video, audio, or anything else.

Each packet starts at the source and travels through several routers to the destination. At each router, the header of the packet will be examined to determined where it should be sent to, or

[1] PSTN (public switched telephone network) is the world's collection of interconnected voice-oriented public telephone networks.

what its next stop should be. Due to this nature, packet switched will tend to be bursty and a little unreliable: silence, then 5 packets show up in one group, more silence, 1 packet, silence, 8 packets, and so on. This is fine with text and web pages, but bad for real time audio/video streams.

Compared to circuit switched,

- Packet connection routing is harder, since at each stop, there is a decision making process for each packet.
- Packet connection is bursty.
- Packet connection is cheaper to set up.
- Packet connection is potentially efficient. The packets from many concurrent connections can share resources along the way. Over time, the silence in one connection is utilized by traffic from another.

  Suppose you are sending stuff from NY to SF. In the case of circuit switched, we reserve you an entire container. You put your stuff in it. There tends to be unused space in the container. With packet switched, you give your stuff to the shipper. The shipper packs your stuff along with the stuff of others into the containers. All the containers on the ship are full.

# 3 Ethernet

## 3.1 A Viewpoint of Layered Networking

The two networking technologies we discuss above, circuit switched and packet switched, roughly describe how a data network works differently from traditional networks. Actually to make a huge computer network to work is difficult and complicated, since there are so many different networking technologies and they all need to be accommodated in the larger network environment, so that any two computers can communicate with each other. Based on how computers are connected with one another, a big network, say the Internet, is considered made up of many different parts, each called a LAN (Local Area Network) and constructed with a different networking technology.

To understand how the Internet works, the point of view of layering is extremely important. It is layering that makes the task simpler to integrate all kinds of heterogenous LANs together and form an interoperable Internet. The networking facilities over the Internet may be divided into 5 layers:

| Name of Layer | Purpose of Layer |
|---|---|
| Application Layer | Specifies how a particular application uses a network. |
| Transport Layer | Specifies how to ensure reliable transport of data. |
| Network Layer | Specifies packet format and routing. |
| Data Link Layer | Specifies frame organization and transmittal. |
| Physical Layer | Specifies the basic network hardware. |

Within a LAN, at least the two lowest layers are present, which are enough to guarantee the communication between computers locally. For the interoperability in a heterogenous network environment, the network layer is necessary. The two top layers are available to make life easier and better.

### 3.2 Ethernet LAN

Let us first look at the most common LAN, Ethernet. Ethernet is a LAN architecture developed by Xerox Corporation together with DEC and Intel in 1976. Ethernet uses a bus or star topology and supports data transfer rates of 10 Mbps. The Ethernet specification served as the basis for the IEEE 802.3 standard, which specifies the physical and data link layer functioning. It is one of the most widely implemented LAN standards.

With Ethernet, there is one wire, and all the computers are connected to it. That is they share the wire and only one computer is supposed to transmit at once, but all computers listen to the wire at all times.

With Ethernet, each computer has a 6-byte unique Ethernet address that is burned in at the factory. This is called its MAC address (Media Access Control).

When data need to be transmitted, the sender divides the message into small packets. Every packet has the address of the recipient in its header. The sender then listens, waiting for a period of silence. When there is a period of silence, the sender broadcasts the packet. Everybody listens all the time, ignoring packets not for them. Sometimes two transmissions overlap and collide with each other. This can happen because both senders can start sending before each other's signals have propagated down the wire to each other. The network card can usually detect these collisions and so knows to stop transmitting.

If a collision happens, then the senders follow a "wait/retransmit" protocol to resend packets: wait a random amount of time before retransmitting to avoid colliding with the same other transmitter again. The randomness is part of what makes Ethernet a little unpredictable in terms of performance, especially latency, but this method is a clever, low-overhead way of allowing two computers to coordinate their use of the wire.

Overall, Ethernet is a great design.

- Shared: There is just one wire and everybody uses it.
- Decentralized: There is no central control and it is thus easily scalable.
- Insecure: It is not too hard to listen and pick up packets not indended for you.
- Unpredictable: You cannot say what the effective latency or bandwidth will be for a transmission since the collisions are themselves random and thus make the transmissions random. So this could be a problem for an application system that has real-time requirement, but Ethernet works very well overall.

  There was an interesting story around this characteristic. As we know, IBM's competing "Token Ring" technology provides less uncertain traffic, because in a Token Ring LAN, each computer has to get permission before starting to transmit by obtaining a token and there is only one token in the network. A rumor goes that IBM in purpose exaggerated the uncertainty problem as a marketing strategy to promote their own technology.

## 4 TCP/IP - On Top of LAN

So far, several important LANs have been invented and widely used, including Ethernet, Token Ring, Wireless, ATM, etc.. They are not compatible in both hardware and software - different cabling method, different name space, and different frame formats. The computers on each LAN speak their own language and the languages are not compatible.

This 'balkanized' state is natural for vendors. Companies always try to make profit so they intend to produce different things that are made intentionally incompatible with products from their competitors, so that a customer who once buy their products has to continue with them forever.

Fortunately there are still some good buys in this world, who want freedom and flexibility. So to accommodate different networking technologies, TCP/IP came into being. It deals with heterogeneity and provides interoperability. It presents a single standard language that everyone can use. TCP/IP is so successful that its "network effect" has beaten out all the better-funded, proprietary/vendor-specific protocols.

TCP/IP works on top of the basic LAN technologies and as we discussed above, it is a layered architecture. We use TCP/IP to refer to the protocol stack used for the Internet because TCP and IP are two most important protocols in the Internet environment.

Relating different network devices to this layered architecture may help us to understand the whole thing.

- Hub: is a common connection point for devices in a network. Hubs are commonly used to connect segments of a LAN. A hub contains multiple ports. When a packet arrives at one port, it is copied to the other ports so that all segments of the LAN can see all packets.

  Hubs and all the computers in a LAN support the same protocols in the physical layer and the data link layer.

- Router: is a hardware device that connects two or more networks. Routers are the primary backbone device of the Internet, connecting different network technologies into a seamless whole. Each router is assigned two or more IP addresses because each IP address contains a prefix that specifies a physical network.

  If you have broadband Internet connection at home and construct a home network, you probably have experience with a special kind of router, DSL/Cable Router. For example, I have a wireless 4-port DSL/Cable Router at home, which combines an Ethernet LAN and a Wireless LAN seamlessly.

- Web proxy: works on the application layer and relays web access requests and responses between web clients and web servers.

## 4.1 IP

### 4.1.1 Naming - IP Addresses

Each computer in the TCP/IP network has a 4-byte IP address, e.g. 134.74.12.16. Totally there are $2^{32}$ addresses. Each host and router on the Internet has its own IP address which uniquely identifies it for sending/receiving packets.

The left part of the IP address identify the neighborhood (subnet), and the right numbers identify the host in the that subnet. For example, the machines in my lab have IP addresses 134.74.12.16 and 134.74.12.18. The left three numbers are the same because the machines are in the same subnet. That subnet is known as 134.74.16.0. The hosts there have addresses in the range 134.74.16.1 to 134.74.16.255. Actually a subnet may take up more or less addresses.

And note that when we move a host to a different subnet, we also need a different IP address for it, just like phone numbers when we move between towns.

### 4.1.2 Packet Standard

At the level of networking layer, IP datagram is used as a standard packet format for IP protocol. An IP datagram contains the IP addresses of the source and the destination, and

some other routing information.

The IP datagram is the basic building block of TCP/IP communication. Higher services, such as TCP, are built on top of basic IP datagrams.

### 4.1.3 Routing Standard

Now let us consider how to transmit a packet between two computers that belong to different subnets. There is no difficulty based on the above discussion of LANs in transmitting the packet between the end computers and the corresponding routers in their LANs, so the main problem is how the router knows how to forward the packet on.

When we plan our trip, we always have the location of our destination in mind. It is however not the case that the router knows where every other computer in the world is. In fact, each router has a table of next stops, i.e. where to forward a packet based on its subnet. Typically a router knows about a few subnets that are near it, and other traffic will be forwarded to a "superior" router that knows more and is more connected than the local router.

The routing story may be illustrated in this way: suppose you have stuff to be sent from New York to Boston, and you don't know where Boston is, but you do know your friend in Connecticut is nearer to Boston. So you send your stuff to Connecticut. Then similarly, your friend may forward your stuff to his/her neighbor in Rhode Island, and so on and so forth until it arrives its destination in Boston.

Based on the above analogy, we know no router has a global, end-to-end picture of the route a datagram should take. And as we can see, the crucial thing in the strategy is how to construct a table of neighbors and collect knowledge regarding who is nearer to where than you. This issue is beyond the coverage of this course, so to know how this works in details, please refer to related materials or books yourself.

### 4.1.4 IP Port Numbers

Each IP address is logically divided into logical port numbers in the range 1-65535. Traffic is addressed to a specific port number at the IP, which allows a computer with one IP to be in multiple conversations at once. As we'll see, specific port numbers are reserved for specific purposes, e.g. port 80 is used for HTTP traffic. Each end of a connection has a source and destination port number as well as IP address.

#### 4.1.5 Broadcast

It's possible to send a packet on the LAN specifically marked as "broadcast", so everyone reads it. You may be able to see a broadcast packet on your hub if all your "receive" lights blink at once. TCP/IP also has a "broadcast" notion for sending information to an entire subnet.

#### 4.1.6 Special IP Addresses

The IP subnet 192.168.0.0 is set aside as a special "non-routable" network. An organization can use 192.168.0.0 addresses internally, and the routers are not supposed to let those addresses leak outside the local network. That way, another organization can also use 192.168.0.0 addresses, and the two organizations will not conflict. 10.0.0.0 is another non-routable network, but 192.168.0.0 is more commonly used and so is probably more reliable.

#### 4.1.7 Host Configuration and DHCP

For a host to be Internet-enabled, it needs the following information:

- IP address: need to know what IP address to use.
- Subnet mask: this number tells which part of the IP address stands for the subnet and which part is host address
- Router: the IP address of the local router to forward traffic to.
- DNS server address: the IP address of the DNS server(s) to use.

DHCP is short for Dynamic Host Configuration Protocol, allowing a host, at boot time or whatever, to broadcast a query to a local DHCP server. The DHCP server can reply with all of the above configuration values. The DHCP may give out an arbitrary IP address from its supply, or it may use a specific setup based on the LAN address of the sender. This is a much more convenient way to do configuration for the end user.

### 4.2 TCP

TCP is another crucial protocol in the Internet protocol stack besides IP. It belongs to the transport layer and is built on IP protocol. We say TCP provides end-to-end reliability, resequencing, and flow control.

TCP enables two hosts to establish a connection and exchange streams of data, which are treated in bytes. TCP can detect errors or lost data and can trigger retransmission until the data is received, complete and without errors. The delivery of data in the proper order is guaranteed.

### 4.2.1 TCP 3-Way Handshake

A TCP connection is done with a 3-way handshake between a client and a server. The following is a simplified explanation of this process.

- The client asks for a connection by sending a TCP segment with the `SYN` control bit set and a random sequence number that is used by both parties to identify the conversation.
- The server responds with its own `SYN` segment that includes identifying information that was sent by the client in the initial `SYN` segment.
- The client acknowledges the server's `SYN` segment.

The connection is then established and is uniquely identified by a 4-tuple called a socket or socket pair:

```
(destination IP address, destination port number)
(source IP address, source port number)
```

Note that it takes 3 one-way trips before the client may send anything to the server. Therefore, in TCP, we will have at least 3-single-trip latency. The packets involved are tiny, so the latency of the network in between dominates.

### 4.2.2 Virtual Circuit

We say TCP is end-to-end and stream-oriented, meaning that to both parties it seems there is a pipe in between. The sender may simply pour the data into its end and do not need to worry about anything that happens behind the scenes. The rest of work is up to the TCP implementation software.

TCP breaks the data stream up into small packets for transmission, numbers them, and includes a checksum with each packet before sending them out. The packets are resembled at the destination and the original data stream is rebuilt and presented to the recipient at the other end of the pipe.

According to what we talked about packet switched technology, several unpleasant things may happen during the trip of packets through the data network:

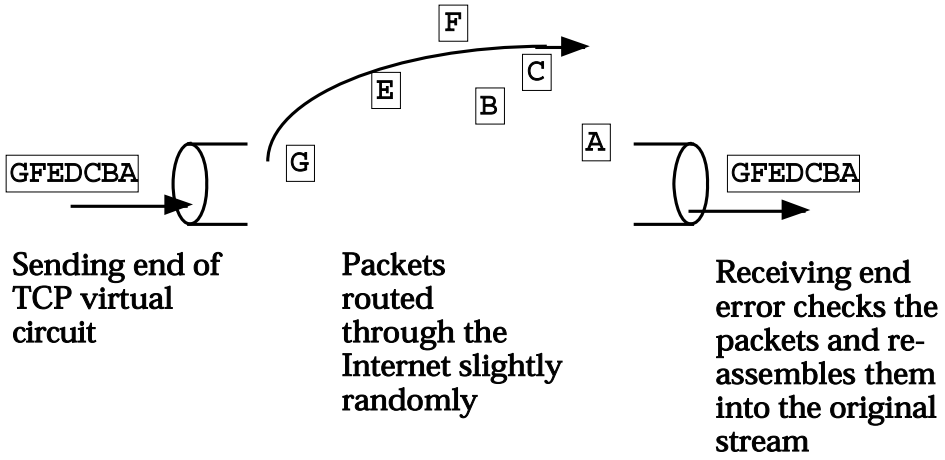


Figure 1: TCP Virtual Circuit

- Packets are corrupted.
- Packets are missing.
- Packets are received duplicately.
- Packets are out-of-order.

TCP detects all these cases and fixes them on its own, requesting that the source re-sends packets as needed. Concretely, TCP uses "acknowledgement" to tell the source which packets have been received correctly. The same mechanism also serves to slow the sender down to a rate that the recipient is capable of accepting. But the most important view of TCP is that TCP gives the appearance of a stream all the way from the sender to the recipient. It hides all the underlying datagram, routing, ACK, re-assembly details. It just looks like a FIFO pipe between the two ends, noting though that the packets arrive irregularly due to the packet switched technology's bursty nature.

### 4.3 UDP

UDP is another protocol in the transport layer, but different from the reliable, pipe-like style of TCP, UDP supports an unreliable transport of a single datagram. No 3-way handshaking is needed to set up a connection in advance in UDP, so it has lower overhead and tends to be faster.

You may doubt the usefulness of UDP. Actually some applications don't require 100% reliability but need fast transmission, say an audio conversation over the Internet. Even though

words may get lost, we still prefer a conversation at the normal speed. That is we prefer 'Helo world' to 'H e l l o wo r l d'.

With UDP, the sender still needs to specify the recipient's IP address and port number in UDP datagram, but the sender will have no way to know if the latter really receives the datagram.