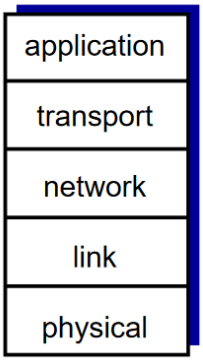
Joyce Yeo 5 Feb 2020

The Internet

What happens when you send an email to a friend? How is the data delivered reliably? The Internet is a complicated network of networks that connect people around the world.

Suppose you want to send a message ("Hello") to a friend. To ensure that the message is correctly delivered to your friend, the message should include the correct name and address, above many other details. To standardise the communication process, **protocols** have been designed. These protocols specify message formats that must be adhered to, and how to respond to different types of messages received.



Many protocols operate to send just one message to your friend. These protocols operate at different **conceptual layers**, which can be visualised using the Internet Protocol stack. Each protocol and layer is responsible for providing a **service**.

An analogy for layered service would be in a business. At the highest level, the director is concerned with business growth and profits. However, the manager must plan for the project progress. In turn, each staff member is responsible for a specific part of the project, like writing a document or signing a form. The director sees a business, the manager sees a project, the staff sees a specific task.

In the same way, the internet functions across many layers, each having a different concern and providing a different service.

Application Layer

When you click "Send" in your email, the email application formats the message. An example:

Application Packet = [sender, receiver, data, title, message].

There are many applications, each having their own unique message formats. For instance, web applications like google.com work on Hypertext Transfer Protocol (HTTP). There could also be proprietary protocols.

Transport Layer – process-to-process delivery, reliable data transfer services

The transport layer then adds a **header** to the existing application packet, forming a transport packet.

Transport Packet = [Transport Header , Payload = Application Packet]

An element in the transport header is the **port** number. A computer can run many applications, for instance email, social media and gaming. When the transport packet arrives at the receiving host, the port number is used to send the packet to the correct **process**.

Besides the port number, the transport packet can also contain a **checksum**, which protects the application packet. A checksum adds all the bytes in the application packet and saves that number. A receiver then re-computes and adds all the bytes in the application packet, verifying the checksum. Sometimes, the packet needs to be re-transmitted. The re-transmission is managed by a transport-level protocol, Transmission Control Protocol (TCP). TCP is a transport protocol that provides reliable data transfer. TCP will label packets with sequence numbers and ensure that the packets are all received in order. For example, if a receiver TCP gets packet 1 and 3, it will wait for packet 2 before sending the packets 1, 2, 3 to the application email, in the correct order.

However, such re-transmission services are time-consuming. In some cases, reliable service is not as important, compared to other factors like speed. An example would be in video calls – the video need not have a 100% quality, but the delay between speaking and receiving should not be too long. In such cases, an alternative to TCP can be used, the User Datagram Protocol (UDP). UDP does not re-transmit lost messages.

Depending on what services are needed, applications select the most appropriate protocols. The transport layer provides process-to-process delivery, reliable data transfer services.

Network Layer – host-to-host delivery

Network Packet = [Network Header , Payload = Transport Packet]

The network header contains the **IP address**, a **unique number to address a host**. Every computer has an IP address assigned to them. This can be determined by searching "what is my ip" on Google. The network layer is concerned with sending the packets to the correct host. At this layer, the contents of the transport and application packets are not important.

Routers operate at the network layer. They read the network header of incoming packets and send them to different destinations based on **routing protocols**. For instance, a group of 5 routers could communicate and share information with each other on which IP addresses are located at which areas.

Overall, the network layer is concerned with routing a packet from host to host.

Link Layer – sending data across a wire

Link Layer Packet = [Link Layer Header , Payload = Network Packet]

Due to the limitations of physical media the link layer breaks down the large network packets into smaller **frames**. Common link layer protocols include WiFi (wireless) and Ethernet (wired). Fundamentally, these protocols aim to ensure that bits sent on one end of the wire can be received. This task is problematic when both ends of the wire attempt to transmit data at the same time. The data is then lost as the electrical signals or waves collide. The link layer controls how fast and how much data is sent through the wire.

Different physical media pose different physical challenges. In wired media, it is always possible to detect collisions. However, laying wires everywhere is expensive and troublesome. It is more common now to have wireless connections but detecting collisions in wireless media is not always possible.

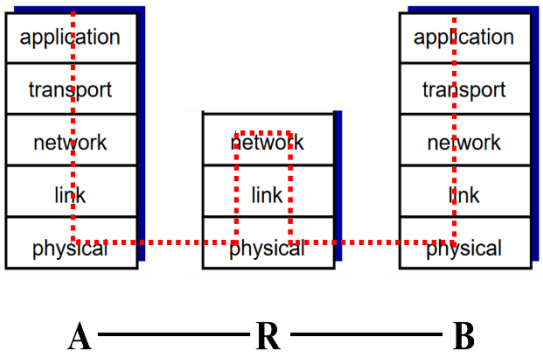
The link layer manages the transfer of data over a link (could be wired or wireless).

Physical Layer

At the physical layer, the actual bits are transmitted by sending electrical signals and waves. Many bits can be sent at once by varying the amplitude and phase of the waves. With 2 different amplitudes and 2 phases, there can be 4 unique signals (00, 01, 10, 11). Each wave sent will represent 2 bits. However, more amplitudes and phases means that it is harder to differentiate between two signals that arrive, creating room for error.

Overview

Once the packet reaches the destination, the network layer checks the network header, that the IP address matches the host IP address. Then, the network header is stripped. The transport layer then verifies the checksum of the packet, before stripping the transport header and sending the data to the application layer. The application then formats the bytes to display the message to the receiver.



A packet can go through many layers as it reaches its destination. Consider a packet that moves from A → Router → B. In moving between the wires from A to R, the packet traverses from the Application to Physical layer in A, then from the Physical to Network layer in R. Routers only operate at the network level. Then, the router sends the packet to B. The packet again traverses to the Physical layer to B, then back to the Application layer in B. At the end, B can see a "Hello" from A.