25. Introduction to Networking

25. Introduction to Networking

To discuss network security, first we need to know how the network is designed. This section provides a (simplified) overview of the various Internet layers and how they interact. A video version of this section is available: see Lecture 11, Summer 2020.

25.1. Local Area Networks

The primary goal of the Internet is to move data from one location to another. A good analogy for the Internet is the postal system, which we'll refer to throughout this section.

The first building block we need is something that moves data across space, such as bits on a wire, radio waves, carrier pigeons, etc. Using our first building block, we can connect a group of local machines in a **local area network (LAN)**.

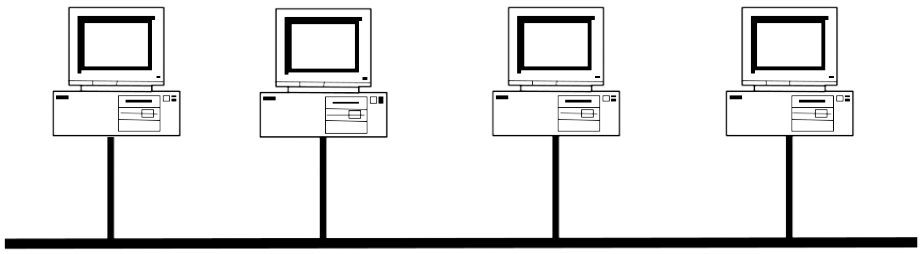

Figure 1: Diagram of a LAN, where computers are directly interconnected

Note that in a LAN, all machines are connected to all other machines. This allows any machine on the LAN to send and receive messages from any other machine on the same LAN. You can think of a LAN as an apartment complex, a local group of nearby apartments that are all connected. However, it would be infeasible to connect every machine in the world to every other machine in the world, so we introduce a **router** to connect multiple LANs.

A router is a machine that is connected to two or more LANs. If a machine wants to send a message to a machine on a different LAN, it sends the message to the router, which forwards the message to the second LAN. You can think of a router as a post office: to send a message somewhere outside of your local apartment complex, you'd take it to the post office, and they would forward your message to the other apartment complex.

With enough routers and LANs, we can connect the entire world in a **wide area network**, which forms the basis of the Internet.

25.2. Internet layering

You may have noticed that this design uses layers of abstraction to build the Internet. The lowest layer (layer 1, also called the physical layer) moves bits across space. Then, layer 2 (the link layer) uses layer 1

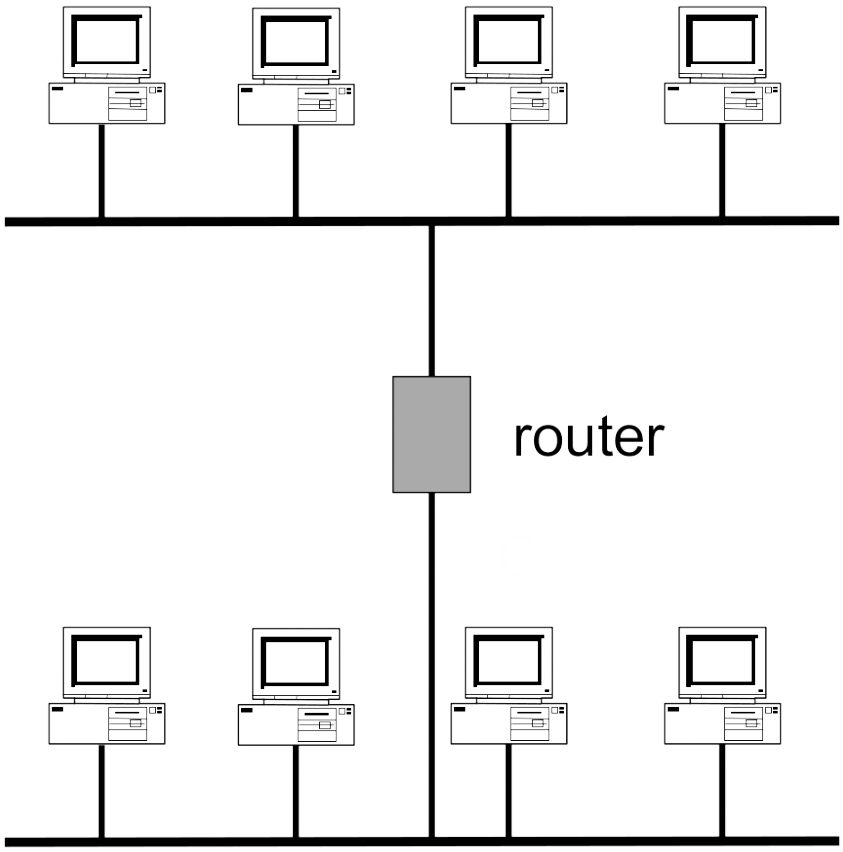

Figure 2: Diagram of a WAN, where two LANs are connected by a router

as a building block to connect local machines in a LAN. Finally, layer 3 (the internetwork layer) connects many layer 2 LANs. Each layer relies on services from a lower layer and provides services to a higher layer. Higher layers contain richer information, while lower layers provide the support necessary to send the richer information at the higher layers.

This design provides a clean abstraction barrier for implementation. For example, a network can choose to use wired or wireless communication at Layer 1, and the Layer 1 implementation does not affect any protocols at the other layers.

In total, there are 7 layers of the Internet, as defined by the OSI 7-layer model. However, this model is a little outdated, so some layers are obsolete, and additional layers for security have been added since then. We will see these higher layers later.

Layer	Name
7	Application
6.5	Secure Transport
6	*obsolete*
5	*obsolete*
4	Transport
3	(Inter)Network
2	Link
1	Physical

25.3. Protocols and Headers

Each layer has its own set of **protocols**, a set of agreements on how to communicate. Each protocol specifies how communication is structured (e.g. message format), how machines should behave while communicating (e.g. what actions are needed to send and receive messages), and how errors should be handled (e.g. a message timing out).

To support protocols, messages are sent with a **header**, which is placed at the beginning of the message and contains some metadata such as the sender and recipient's identities, the length of the message, identification numbers, etc. You can think of headers as the envelope of a letter: it contains the information needed to deliver the letter, and appears before the actual letter.

Because multiple protocols across different layers are needed to send a message, we need multiple headers on each packet. Each message begins as regular human-readable text (the highest layer). As the message is being prepared to get sent, it is passed down the protocol stack to lower layers (similar to how C programs are passed to lower layers to translate C code to RISC-V to machine-readable bits). Each layer adds its own header to the top of the message provided from the layer directly above. When the message reaches the lowest layer, it now has multiple headers, starting with the header for the lowest layer first.

Once the message reaches its destination, the recipient must unpack the message and decode it back into human-readable text. Starting at the lowest layer, the message moves up the protocol stack to higher layers. Each layer removes its header and provides the remaining content to the layer directly above. When the message reaches the highest layer, all headers have been processed, and the recipient sees the regular human-readable text from before.

25.4. Addressing: MAC, IP, Ports

Depending on the layer, a machine can be referred to by several different addresses.

Layer 2 (link layer) uses 48-bit (6-byte) **MAC addresses** to uniquely identify each machine on the LAN. This is not to be confused with MACs (message authentication codes) from the crypto section. Usually it is clear from context which type of MAC we are referring to, although sometimes cryptographic MACs are called MICs (message integrity codes) when discussing networking. MAC addresses are usually written as 6 pairs

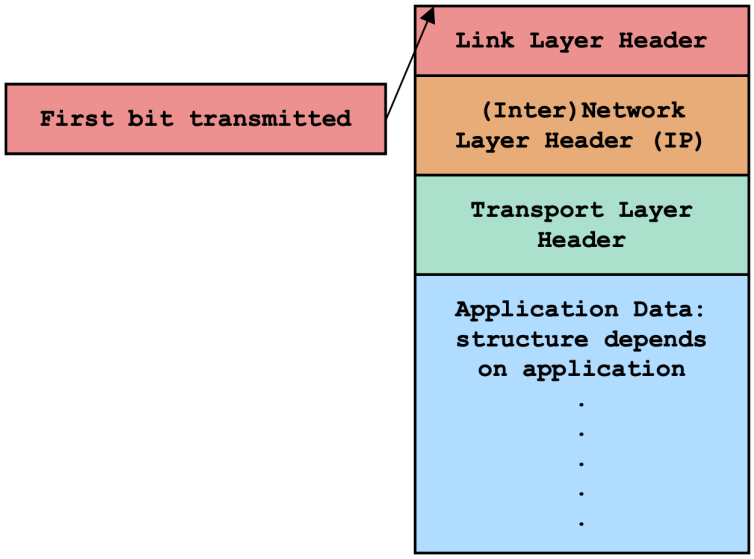

Figure 3: A diagram of a network packet structure, with the link layer header first, then the IP header, then the transport layer header, then the application data

of hex numbers, such as `ca:fe:f0:0d:be:ef`. There is also a special MAC address, the broadcast address of `ff:ff:ff:ff:ff:ff`, that says "send this message to everyone on the local network." You can think of MAC addresses as apartment numbers: they are used to uniquely identify people within one apartment complex, but are useless for uniquely identifying one person in the world. (Imagine sending a letter addressed to"Apartment 5." This might work if you're delivering letters within your own apartment complex, but how many Apartment 5s exist in the entire world?)

Layer 3 (IP layer) uses 32-bit (4-byte) **IP addresses** to uniquely identify each machine globally. IP addresses are usually written as 4 integers between 0 and 255, such as `128.32.131.10`. Because the Internet has grown so quickly, the most recent version of the layer 3 protocol, IPv6, uses 128-bit IP addresses, which are written as 8 2-byte hex values separated by colons, such as `cafe:f00d:d00d:1401:2414:1248:1281:8712`. However, for this class, you only need to know about IPv4, which uses 32-bit IP addresses.

Higher layers are designed to allow each machine to have multiple processes communicating across the network. For example, your computer only has one IP address, but it may have multiple browser tabs and applications open that all want to communicate over the network. To distinguish each process, higher layers assign each process on a machine a unique 16-bit **port number**. You can think of port numbers as room numbers: they are used to uniquely identify one person in a building.

The source and destination addresses are contained in the header of a message. For example, the Layer 2 header contains MAC addresses, the Layer 3 header contains IP addresses, and higher layer headers will contain port numbers.

25.5. Packets vs. Connections

Notice that in the postal system example, the post office has no idea if you and your pen pal are having a conversation through letters. The Internet is the same: at the physical, link, and internetwork layers, there is no concept of a connection. A router at the link layer only needs to consider each individual packet and send it to its destination (or, in the case of a long-distance message, forward it to another router somewhere closer to the destination). At the lower layers, we call individual messages **packets**. Packets are usually limited to a fixed length.

In order to actually create a two-way connection, we rely on higher layers, which maintain a connection by breaking up longer messages into individual packets and sending them through the lower layer protocols.

Higher-layer connections can also implement cryptographic protocols for additional security, as we'll see in the TLS section.

Note that so far, the Internet design has not guaranteed any correctness or security. Packets can be corrupted in transit or even fail to send entirely. The IP (Internet Protocol) at layer 3 only guarantees *best-effort delivery*, and does not handle any errors. Instead, we rely on higher layers for correctness and security.

25.6. Network Adversaries

Network adversaries can be sorted into 3 general categories. They are, from weakest to strongest:

Off-path Adversaries: cannot read or modify any packets sent over the connection.

On-path Adversaries: can read, but not modify packets.

In-path Adversaries: can read, modify, and block packets. Also known as a **man-in-the-middle**.

Note that all adversaries can send packets of their own, including faking or **spoofing** the packet headers to appear like the message is coming from somebody else. This is often as simple as setting the "source" field on the packet header to somebody else's address.