Module 1.1 - Networking Review.

For textbooks and additional resources used in this module see the Sources section at the end of this document.

Networking Review

This module serves as a review/brief introduction to the OSI and TCP reference models. We'll briefly cover each of the layers for each model and provide resources for further reading.

Before we cover each of the reference models, let's define what a protocol is.

What is a protocol?

"In networking, a protocol is a set of rules for formatting and processing data. Network protocols are like a common language for computers. The computers within a network may use vastly different software and hardware; however, the use of protocols enables them to communicate with each other regardless." https://www.cloudflare.com/learning/network-layer/what-is-a-protocol/

Another description of a protocol is:

"Communication and language have greatly enhanced the abilities of the human race. By using a common language, humans are able to transfer knowledge, coordinate actions, and share experiences. Similarly, programs can become much more powerful when they have the ability to communicate with other programs via a network. The real utility of a web browser isn't in the program itself, but in its ability to communicate with webservers." [2]

The OSI Model

What is the OSI model?

The Open Systems Interconnection (OSI) model is an abstract representation of how the Internet works. It contains 7 layers, with each layer representing a different category of networking functions. https://www.cloudflare.com/learning/network-layer/what-is-a-protocol

Physical layer This layer deals with the physical connection between two points. This is the lowest layer, whose primary role is communicating raw bit streams. This layer is also responsible for activating, maintaining, and deactivating these bit-stream communications.

Data-link layer This layer deals with actually transferring data between two points. In contrast with the physical layer, which takes care of sending the raw bits, this layer provides high-level functions, such as error correction and flow control. This layer also provides procedures for activating, maintaining, and deactivating data-link connections.

Network layer This layer works as a middle ground; its primary role is to pass information between the lower and the higher layers. It provides addressing and routing.

Transport layer This layer provides transparent transfer of data between systems. By providing reliable data communication, this layer allows the higher layers to never worry about reliability or cost-effectiveness of data transmission.

Session layer This layer is responsible for establishing and maintaining connections between network applications.

Presentation layer This layer is responsible for presenting the data to applications in a syntax or language they understand. This allows for things like encryption and data compression.

Application layer This layer is concerned with keeping track of the requirements of the application.

[Image source: Hacking the Art of Exploitation 2nd ed, ch4. Jon Erickson. No starch press, 2008]

OSI Layers - Additional information:

- Physical (1): This is the level of physical communication in the real world. At this level, we have specifications for things such as the voltage levels on an Ethernet cable, what each pin on a connector is for, the radio frequency of Wi-Fi, and the light flashes over an optic fiber.

- Data Link (2): This level builds on the physical layer. It deals with protocols for directly communicating between two nodes. It defines how a direct message between nodes starts and ends (framing), error detection and correction, and flow control.

- Network layer (3): The network layer provides the methods to transmit data sequences (called packets) between nodes in different networks. It provides methods to route packets from one node to another (without a direct physical connection) by transferring through many

intermediate nodes. This is the layer that the Internet Protocol is defined on, which we will go into in some depth later.

- Transport layer (4): At this layer, we have methods to reliably deliver variable length data between hosts. These methods deal with splitting up data, recombining it, ensuring data arrives in order, and so on. The Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are commonly said to exist on this layer.

- Session layer (5): This layer builds on the transport layer by adding methods to establish, checkpoint, suspend, resume, and terminate dialogs.

- Presentation layer (6): This is the lowest layer at which data structure and presentation for an application are defined. Concerns such as data encoding, serialization, and encryption are handled here.

- Application layer (7): The applications that the user interfaces with (for example, web browsers and email clients) exist here. These applications make use of the services provided by the six lower layers.

[1, pg. 11]

Here are the common Internet Protocols we'll cover as we journey through socket programming:

- TCP: As described above, TCP is a transport layer protocol that ensures reliable data delivery. TCP is meant to be used with IP, and the two protocols are often referenced together as TCP/IP.

- HTTP: The Hypertext Transfer Protocol (HTTP) is the foundation of the World Wide Web, the Internet that most users interact with. It is used for transferring data between devices. HTTP belongs to the application layer (layer 7), because it puts data into a format that applications (e.g., a browser) can use directly, without further interpretation. The lower layers of the OSI model are handled by a computer's operating system, not applications.

- HTTPS: The problem with HTTP is that it is not encrypted — any attacker who intercepts an HTTP message can read it. HTTPS (HTTP Secure) corrects this by encrypting HTTP messages.

- TLS/SSL: Transport Layer Security (TLS) is the protocol HTTPS uses for encryption. TLS used to be called Secure Sockets Layer (SSL).

- UDP: The User Datagram Protocol (UDP) is a faster but less reliable alternative to TCP at the transport layer. It is often used in services like video streaming and gaming, where fast data delivery is paramount.

For more information you can visit the RFC for each protocol. Cloudflare Some of the most important protocols to know are (https://www.cloudflare.com/learning/network-layer/what-is-a-protocol/):

The TCP/IP Layer model

Another model, more commonly used in practice, is the TCP/IP model.

What is the TCP/IP layer?

The TCP/IP protocol suite is the most common network communication model in use today. The TCP/IP reference model differs a bit from the OSI model, as it has only four layers instead of seven. The TCP/IP reference model was developed after the TCP/IP protocol was already in common use. It differs from the OSI model by subscribing a less rigid, although still hierarchical, model. For this reason, the OSI model is sometimes better for understanding and reasoning about networking concerns, but the TCP/IP model reflects a more realistic view of how networking is commonly implemented today. [1, pgs.: 12-13]

The TCP/IP Layers:

- Network Access layer (1): On this layer, physical connections and data framing happen. Sending an Ethernet or Wi-Fi packet are examples of layer 1 concerns. Internet layer (2): This layer deals with the concerns of addressing packets and routing them over multiple interconnection networks. It's at this layer that an IP address is defined.

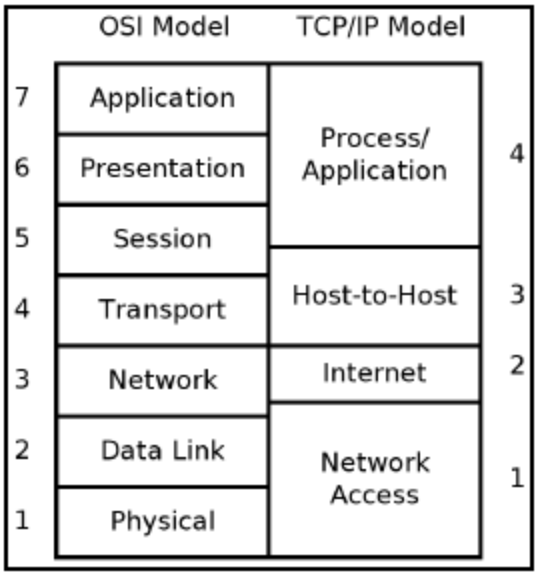
- Host-to-Host layer (3): The host-to-host layer provides two protocols, TCP and UDP, which we will discuss in the next few chapters. These protocols address concerns such as data order, data segmentation, network congestion, and error correction.

- Process/Application layer (4): The process/application layer is where protocols such as HTTP, SMTP, and FTP are implemented. Most of the programs that feature in this book could be considered to take place on this layer while consuming functionality provided by our operating system's implementation of the lower layers.

Why do we need a layered model?

When two computers talk to each other, they need to speak the same language. The structure of this language is described in layers by the OSI or TCP/IP model. The structure of this language is described in layers and these layers/model provide standards that allow hardware, such as routers and firewalls, to focus on one aspect of communication that applies to them and ignore others. [2, pg:196]

OSI and TCP/IP layer mapping is illustrated by the following diagram:



The Internet Protocol

How do computers communicate across a network?

Computers and networking devices use the Internet Protocol versions 4 and 6 to communicate across networks.

- IPv4 uses 32-bit addresses, which limits it to addressing no more than 232 or 4,294,967,296 systems.
- IPv6 was designed to replace IPv4 and has been standardized by the Internet Engineering Task Force (IETF) since 1998. It uses a 128-bit address, which allows it to address a theoretical 2128 = 340,282,366,920,938,463,463,374,607,431,768,211,456, or about a 3.4 x 1038 addresses.

  [1, pg:17]

What is an IP address?

All Internet Protocol traffic routes to an address. This is similar to how phone calls must be dialed to phone numbers. IPv4 addresses are 32 bits long. They are commonly divided into four 8-bit sections. Each section is displayed as a decimal number between 0 and 255 inclusive and is delineated by a period.

* Here are some examples of IPv4 addresses:

0.0.0.0

127.0.0.1

10.0.0.0

172.16.0.5

192.168.0.1

192.168.50.1

255.255.255.255

IPv6 addresses are 128 bits long. They are written as eight groups of four hexadecimal characters delineated by colons. A hexadecimal character can be from 0-9 or from a-f. Here are some examples of IPv6 addresses:

0000:0000:0000:0000:0000:0000:0000:0001

2001:0db8:0000:0000:0000:ff00:0042:8329

fe80:0000:0000:0000:75f4:ac69:5fa7:67f9

ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

[1]

Sources, textbooks, and Additional materials

[1] Lewis Van Winkle, “Hands-On Network Programming with C". Packt Publishing. May 2019. ISBN: 9781789349863. Chapter 1. https://learning.oreilly.com/library/view/hands-on-network-programming/9781789349863/

[2] Jon Erickson, “Hacking the Art of Exploitation 2nd ed”. No Starch Press. February 2008. ISBN: 978-1593271442. Chapter 4, sections 0x400 – 0x430. https://learning.oreilly.com/library/view/hacking-the-art/9781593271442/

[3] https://www.cloudflare.com/learning/network-layer/what-is-a-protocol