

Module 3: The Application Layer

Compiler: Sarah Meng Li

Acknowledgement: This summary is made based on lecture notes of CSCI 3171 Network Computing taught by Dr. Srinivas Sampalli in fall 2019.

Objectives and Topics

Having obtained an overview of the TCP/IP protocol suite, we will now focus on the top layer, namely the Application Layer.

1. Principles of Network Applications
2. Analysis of popular protocols
3. Peer-to-peer applications
4. Video streaming and content distribution networks
5. Socket programming

Principle of Network Applications

1. Network applications (e.g., web browsing)
2. Applications of layered protocols (e.g., HTTP, like a contract)
3. Client server architecture
4. Peer-to-peer architecture
5. Processes and sockets

Definition 3.1: A **network application** is any program that resides on end-systems and is able to communicate over a network.

Remark: Interconnection devices (e.g., repeaters, bridges/switches, routers) do not run network applications.

Example 3.2:

1. Web browsing: a web browser communicates with the web server.
2. Email: an email client communicates with a mail server.
3. **DNS:** a **host** queries a **domain** name server for the IP address of a remote host.
4. Video streaming: Your laptop streams a Netflix video from the Netflix server.

Definition 3.3: An **application layer protocol** provides the interface between the network application and the TCP/IP protocol suite.

It defines the following:

1. What types of messages are exchanged (e.g., request, response, etc)
2. What is the message syntax (the fields and their formats in each message)
3. What is the message semantics (what does each field mean)
4. What are the rules for when and how messages are sent

Example 3.4:

1. Open protocols (defined by standards by the Internet Engineering Task Force)
 - a. **Hyper Text Transfer Protocol (HTTP):** Web browsing

Commented [SL1]: What is DNS?

[source: <https://www.cloudflare.com/learning/dns/what-is-dns/>]

The Domain Name Systems (DNS) is the phonebook of the Internet. Humans access information online through domain names, like nytimes.com or espn.com. Web browsers interact through Internet Protocol (IP) addresses. DNS translates domain names to IP addresses so browsers can load Internet resources. Each device connected to the Internet has a unique IP address which other machines use to find the device. DNS servers eliminate the need for humans to memorize IP addresses such as 192.168.1.1 (in IPv4), or more complex newer alphanumeric IP addresses such as 2400:cb00:2048:1::c629:d7a2 (in IPv6).

Commented [SL2]: What's a host?

Website hosts are companies that house, serve, and maintain files for one or more websites. Think of a website hosting company like a shopping center that contains several individual stores. If you want to open a store at the shopping center, you can lease space and set up shop.

Just like a shopping center, website hosting companies enable you to lease space on their web servers where you can store your website files and make them available for visitors to view.

Because you have registered your domain, you can use it at any web hosting company you want. Remember your store in the shopping center? Suppose you decided to move your store and lease space at a different shopping center. If you packed up your products and moved to a new location, you would still be able to use the same name for your store. ... [1]

Commented [SL3]: What is a domain? And how is it different from a website?

[source: <https://ca.godaddy.com/help/whats-the-difference-between-a-domain-and-a-website-16574>]

After registering a domain, you need to do something with it, put it to work for you. Your domain can be used for a website, for an email account, or for both. You can, however, register a domain and choose not to use it for a website or an email account but you can't have a website or an email account without a domain.

When used for a website, your domain name is what people type into their browser to find your site — this is why they are sometimes referred to as web addresses. So godaddy.com is the domain used for GoDaddy's website. Registering a ... [2]

- Wireshark
 - i. Client HTTP version
 - ii. Language and encoding accepted by the client
 - iii. User-agent client details
- b. Simple Mail Transfer Protocol (SMTP): Email
- c. Domain Name Service (DNS): IP address lookup
- d. Secure Shell (SSH): Secure remote connection
- 2. Proprietary protocols
 - a. Skype
 - b. FaceTime
 - c. Kazaa
 - d. BitTorrent
 - e. Tor

Two types of application architecture

Definition 3.5 Client-server architecture

1. Server is “always-on” and reachable.
2. Server has a permanent IP address.
3. Servers are usually located in data centers for scaling, load balancing and reliability.
4. Clients do not communicate with each other
5. Clients only communicate with servers.
6. Client may have dynamic IP addresses (when the destination is hidden behind a firewall)

Definition 3.6 Peer-to-peer (P2P) architecture

1. There is no one set of “servers” that is always on.
2. Peers may be intermittently connected and may not always be reachable.
3. Peers have dynamic IP addresses.
4. Each host, called a peer, can be a client or a server.
5. Peers request service from other peers, and provide/transfer service to other peers.

Processes and Sockets

Definition 3.7

A **process** is program running within a host. Each network application requires a process for its operation.

1. Processes on different hosts communicate by exchanging messages.
2. A client process (residing on a client) initiates a communication.
3. A server process (residing on a server) waits for communication to be initiated.
4. In P2P, peers can run both client and server processes.

Commented [SL4]: What are the differences between open/standard and closed/proprietary protocols?

[source: <http://www.omnisecu.com/basic-networking/difference-between-proprietary-and-standard-protocols.php>]

Proprietary protocols are usually developed by a single company for the devices (or Operating System) which they manufacture. AppleTalk is a proprietary protocol developed by Apple Inc. Appletalk protocol may work well in network environments consisting only Apple devices. But other vendors may not support Appletalk protocol. Proprietary protocols will not scale well in network environments consisting of multi-vendor equipment.

Standard protocols are agreed and accepted by whole industry. Standard protocols are not vendor specific. Standard protocols are often developed by collaborative effort of experts from different organizations. Examples of standard protocols are IP, TCP, UDP etc. RFC (Request for Comments) is an IETF platform to develop Standard Protocols.

Commented [SL5]: What happens when the destination hides behind a firewall?

- Each process has a socket.

Definition 3.9

A process sends and receives messages through their sockets. Think of socket as a door or an interface between the process and the protocol stack.

- The sending process pushes the message through its sockets to the Transport layer.
- The receiving process receives the message through its sockets from the Transport layer.

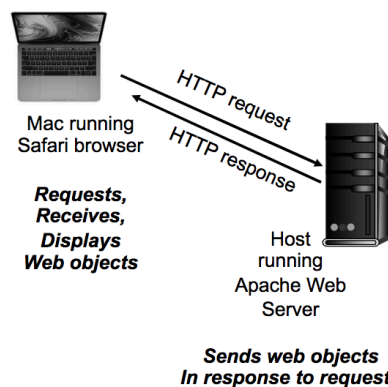
How does socket work? 3.10

- Sockets are identified by two addresses, namely the port address and the IP address
- Sockets on the client side <Client port address, client IP address>
- Sockets on the server side <Server port address, server IP address>

Remark: In Wireshark, port addresses can be found in Transmission Control Protocol part; IP addresses can be found in the Internet Protocol part.

I. HTTP Overview

- It is based on the client-server architecture.
- It is connection-oriented.
- It runs on top of TCP.
- The standard port number for server is 80. (the port on the server side)
- It is a stateless protocol → The statelessness leads to potential security issues such as DoS attacks. Amazon uses web cookies to remember what did the client buy last time.
- A web page consists of objects. Each object can be an HTML file, audio file, video file, image, Java applet, etc.
- A web page has a base HTML file and a number of referenced objects.
- Each object is addressable by a URL.



Commented [SL6]: Why a socket is needed between the Application and the Transport layer?

Commented [SL7]: Version 1.1 most common

Commented [SL8]: It has a connection set-up phase, message exchange, and a connection tear-down phase.

Commented [SL9]: The server contains no info about the client's past requests.

http://gaia.cs.umass.edu	wireshark-labs/intro-wireshark-file1.html
Web server (host) name	Object path name

HTTP Messages

1. Messages are mainly requests and responses.

Definition 3.11

Request messages are always sent from the client to the server (VERBS and NOUNDS)

- GET: retrieve data (web object) from the server
- POST: send client-generated data to the server. The POST method creates a new thread and a new path.
- PUT: also sends client-generated data to the server, but the difference is that this method overwrites previous data on that thread.
- DELETE: asks the server to delete the client-generated data stored in the server.
- TRACE: Asks the server to return a diagnostic trace.
- HEAD: Similar to GET requests but gets only the header in response.
- OPTIONS: Asks the server to return the list of request methods that it supports.

Commented [SL10]: User-generated form, profile picture, etc.

Definition 3.12

Response messages are always sent from the server to the client

- Mainly called Response (no sub-types)
- Each response message will contain different information depending on the request.

HTTP Status Codes

1. 200's are used for successful requests
 - a. 200 → Request OK
2. 300's are used for redirections
 - a. 301 → Requested object has been moved.
3. 400's are used to indicate a problem with the request
 - a. 404 → file not found
4. 500's are used to indicate a problem with the server.
 - a. 505 → HTTP Version not supported.

TCP Three-way Handshake

Cookies

Definition 3.13

A cookie is a small text file that is stored by your browser on your machine by the server. A cookie just contains a User ID and the site that you visited.

- The User ID is created when you visit the site for the first time. The same ID is also stored in the server's database.
- The next time you visit the same site, the ID number is sent along with your request. Thus the server is able to recognize you.
- Cookies can also contain more information such as your location, purchase history, session start time, etc.

Usage 3.14

- Session management (logins, shopping carts, game scores)
- Personalization (user preferences, weather reports, local news, suggestions for next purchase)
- Tracking (data analytics or user behaviour)

Proxy Server

The actual **nuts and bolts** of how the internet works is not something a people often stop to consider. The problem with that is the inherent danger of data security breaches and identity theft that come along with the cute dog pictures, 24 hour news updates, and great deals online.

Commented [SL11]: Nuts and bolts: the practical facts about a particular thing, rather than theories or ideas about it.

But what actually happens when you browse the web? You might be using a proxy server at your office, on a Virtual Private Network (VPN) or you could be one of the more tech-savvy who always use a proxy server of some kind or another.

Definition 3.15

A **proxy server** acts as a gateway between you and the internet. It's an intermediary server separating end users from the websites they browse. Proxy servers provide varying levels of functionality, security, and privacy depending on your use case, needs, or company policy.

If you're using a proxy server, internet traffic flows **through** the proxy server on its way to the address you requested. The request then comes back through that same proxy server (there are exceptions to this rule), and then the proxy server forwards the data received from the website to you.

Usage 3.16

Modern proxy servers do much more than forwarding web requests, all in the name of data security and network performance.

- Proxy servers act as a firewall and web filter, provide shared network connections, and cache data to speed up common requests.
- A good proxy server keeps users and the internal network protected from the bad stuff that lives out in the wild internet.
- Lastly, proxy servers can provide a high level of privacy.

Usage 3.16

Modern proxy servers do much more than forwarding web requests, all in the name of data security and network performance.

- Proxy servers act as a firewall and web filter, provide shared network connections, and cache data to speed up common requests.
- A good proxy server keeps users and the internal network protected from the bad stuff that lives out in the wild internet.
- Lastly, proxy servers can provide a high level of privacy.

Scenario 3.17

When too many clients making too many requests to web servers, we can cache the web information in the proxy server so that the original web server does not need to process requests.

- Browsers send all requests to a proxy server
- If object is found in cache, proxy server returns the object.
- If the object is not found in cache, proxy server requests original server, then caches it and returns it to the client.

Usually proxy servers are located in the local LAN or your local machine.

<https://www.varonis.com/blog/what-is-a-proxy-server/>

How does proxy server work? 3.18

Every computer on the internet needs to have a unique Internet Protocol (IP) Address. Think of this IP address as your computer's street address. Just as the post office knows to deliver your mail to your street address, the internet knows how to send the correct data to the correct computer by the IP address.

A proxy server is basically a computer on the internet with its own IP address that your computer knows. When you send a web request, your request goes to the proxy server first. The proxy server then makes your web request on your behalf, collects the response from the web server, and forwards you the web page data so you can see the page in your browser.

When the proxy server forwards your web requests, it can make changes to the data you send and still get you the information that you expect to see.

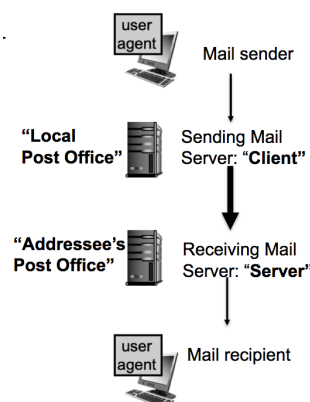
How does proxy server maintain security and privacy? 3.19

- A proxy server can change your IP address, so the web server doesn't know exactly where you are in the world.
- It can encrypt your data, so your data is unreadable in transit.
- And lastly, a proxy server can block access to certain web pages, based on IP address.

II. Understanding Electronic Mail and SMTP

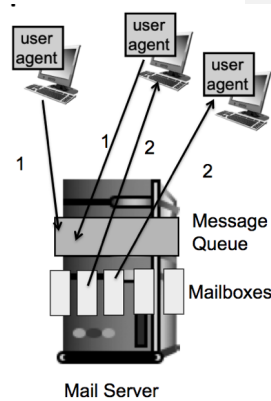
Definition 3.20 The components of Electronic Mail

- What is a **user agent**?
 - This is also known as the **mail composer** and **mail reader**. Software that sits on the user's machine.
 - Examples: Outlook, Eudora, Claris, Thunderbird, Apple Mail, Pine, Gmail.
- What is a **mail server**?
 - Server responsible for receiving, storing, and sending email messages. Typically



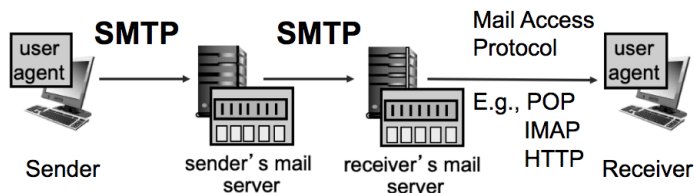
maintained by user's university, organization, ISP, or could be centrally maintained in larger servers.

- Example: Google servers for email.
- What are the components of the mail server?
 - **Mailbox** for each user: Buffer that contains incoming messages for that user.
Receiving user agent collects mail from its mailbox.
 - **Message queue**: buffer that contains all the outgoing messages from different users.
Sending user agent deposits mail into the queue.



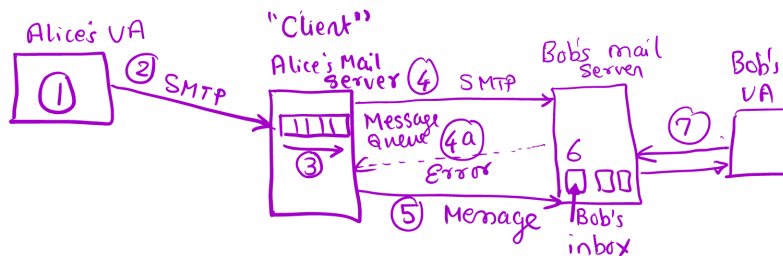
Definition 3.21 Protocols in Electronic Mail

- **SMTP (Simple Mail Transfer Protocol)**: This is the protocol for transfer
 - Between the sending user agent and sender's mail server.
 - Between the sender's mail server and the receiver's mail server.
- **Mail Access Protocol**: This is the protocol for retrieving messages from the receiver's mail server. Some examples are as follow.
 - Post Office Protocol (POP)
 - Internet Mail Access Protocol (IMAP)
 - Hyper Text Transfer Protocol (HTTP) (for Gmail, Hotmail. Etc.)



3.22 Anatomy of an Email Transfer

- Scenario: Allice (alice@cs.dal.ca) sends a mail to Bob (bob@stanford.edu)
 1. Alice's user agent composes message to bob@stanford.edu
 2. Alice's user agent sends message to Alice's mail server using SMTP. Message gets stored in the Message Queue.
 3. Message reaches front of the queue.
 4. Alice's mail server (the "client") opens SMTP connections to Bob's mail server (the "server").
 - a. If there is an error in the email ID, error message is sent back.
 5. Client sends Alice's message over a TCP connection to Bob's mail server.
 6. Bob's mail server receives message and place it in Bob's mailbox.
 7. Bob invokes his user agent to read the message from the mailbox.



3.23 SMTP Features

- Runs on TCP: connection-oriented protocol
- **Three phases:**
 - **Greeting or handshakes (open connections)**
 - **Transfer of message**
 - **Closure of connection**
- Server (receiving mail server) uses port number 25.
- Direct transfer from sending to receiving mail server.
- All messages are command/response like HTTP.
- But unlike "HTTP" which is a "pull" protocol, SMTP is a "push" protocol.

3.24 SMTP Commands and Status Codes

- Commands are sent from client to server
 - HELO: indicates the message is about to be sent.
 - MAIL FROM: contains sender's email address
 - RCPT TO: contains recipient's email address
 - DATA: contains message
 - QUIT: indicates closure of message transfer
- Status codes are sent from server to client
 - 220: for opening connection
 - 221: for closing connection
 - 250: for acknowledging HELO, MAIL FROM, and RCPT TO commands.
 - 354: for indicating readiness to receive message.

Definition 3.25 POP3 Protocol

- Post Office Protocol version 3 is a protocol for retrieving messages from the recipient's mail server by the recipient's user agent.
- There are two phases: authorization phase and transaction phase.

Definition 3.26 Authorization Phase

- Client commands
 - User: provide username
 - Pass: provide password
- Server responses
 - +OK

- -ERR

Authorization Phase	S:	+OK (POP3 server ready)
	C:	User bob
	S:	+OK
	C:	Pass1234
	S:	+OK (successful login)

Definition 3.27 Transaction Phase

- Client commands
 - list: list the messages in the mailbox by numbers
 - retr: receive message by number
 - dele: delete message
 - quit: quit connection

Transaction Phase	C:	list
	S:	1 498
	S:	2 514
	S:	. (end of listing for new messages)
	C:	retr 1
	S:	<Message 1 contents>
	S:	. (end of the message)
	C:	dele 1
	S:	+OK
	C:	quit
	S:	+OK

Definition 3.28 POP3 modes

POP3 can operate in two modes:

- **Download and Delete (Default mode):**
 - This will delete the mail from the recipient's mail server **once it is downloaded to Bob's machine.**
 - Bob cannot re-read his messages if he connects from a different machine.
- **Download and Keep:**
 - By changing the default setting, Bob can keep the message on the mail server.
 - This will enable re-reading of messages from multiple machines.

Remark: POP3 is stateless across sessions, because the names of folders and mappings are not maintained across sessions.

Definition 3.29 IMAP

IMAP is short for "Internet Mail Access Protocol". It is another protocol for retrieval of messages by the recipient's user agent.

- By default, IMAP stores all messages on the mail server by default.
- It allows users to organize messages into folders.

- IMAP's stateful property will allow user to **maintain** names of folders and mappings between folder names and message IDs across multiple sessions.

Remark: IMAP is a stateful protocol because it maintains state across user sessions.

III. Understanding the Domain Name System (DNS)

What's a host?

Website hosts are companies that house, serve, and maintain files for one or more websites. Think of a website hosting company like a shopping center that contains several individual stores. If you want to open a store at the shopping center, you can lease space and set up shop.

Just like a shopping center, website hosting companies enable you to lease space on their web servers where you can store your website files and make them available for visitors to view.

Because you have registered your domain, you can use it at any web hosting company you want. Remember your store in the shopping center? Suppose you decided to move your store and lease space at a different shopping center. If you packed up your products and moved to a new location, you would still be able to use the same name for your store.

The same is true for your domain — regardless of where you decided to host your website's files, your domain remains yours for the duration of your registration.

For example, GoDaddy is both a domain registrar AND a website hosting company — and we have the tools to help you build your website. This means you can register your domain name, build a website for it, and get it on the web - all in one place.

What is a domain? And how is it different from a website?

[source: <https://ca.godaddy.com/help/whats-the-difference-between-a-domain-and-a-website-16574>]

After registering a domain, you need to do something with it, put it to work for you. Your domain can be used for a website, for an email account, or for both. You can, however, register a domain and choose not to use it for a website or an email account but you can't have a website or an email account without a domain.

When used for a website, your domain name is what people type into their browser to find your site — this is why they are sometimes referred to as web addresses. So godaddy.com is the domain used for GoDaddy's website. Registering a domain, however, doesn't automatically create a website — it only means that you've registered the domain.

For example, if you decided to start your own business and you thought of great name for it, you might copyright the name to protect it. However, simply giving your business a name is only one of many steps that go into building your business.

In the same manner, when you register a domain and want to use it for a website, you'll need to first build the website, and then find a host for it, so it can be visible on the Web.

