Intro to Networks

Quick intro to computer networks

Nomenclature

- Data
- Bits
- Node
- Link
- Network
- Protocol
- Protocol Stack

OSI vs TCP/IP

LAYERS	OSI	TCP/IP
L7	Application	Application
L6	Presentation	↑
L5	Session	↑
L4	Transport	TCP/UDP
L3	Network	IP/ICMP
L2	Data Link	Network access
L1	Psysical	-

- OSI (Open Systems Interconnection) model: De jure / TCP/IP model: De facto

Protocol Layers

Layers	CLIENT	SERVER
App	show www.google.com	<h1>Google</h1>
L7	GET / HTTP/1.1 Host: www.google.com User-Agent: curl/8.9.1 Accept: /	HTTP/1.0 200 OK Date: Mon, 25 Nov 2024 06:57:24 GMT Cache-Control: private, max-age=0
L4	Source port: 31232 Destination port: 443 SEQ: 1	Source port: 443 Destination port: 31232 SEQ: 1
L3	Source IP: 172.31.3.4 Dest IP: 142.250.185.196	Source IP: 142.250.185.196 Dest IP: 172.31.3.4

IP

IP (Internet Protocol) is the fundamental communication protocol used to send data across a network. It ensures data packets are addressed, routed, and delivered between devices.

- Packet-Based: Data is broken into packets, which are sent independently and reassembled at the destination.
- Routing: Routers direct packets to their destination based on IP addresses.
- Stateless: Each packet is treated independently, without information about previous packets.
- Layer: Operates at the network layer (Layer 3) in the OSI model.

IP address

An IP address (Internet Protocol address) is a unique identifier assigned to a device on a network, allowing it to communicate with other devices. It acts as a virtual address, enabling data to be sent to the correct destination.

- IPv4: 32-bit address, written as four decimal numbers separated by dots (e.g., 192.168.1.1).
- IPv6: 128-bit address, written as eight groups of hexadecimal numbers separated by colons (e.g., 2001:0db8:85a3::8a2e:0370:7334).
- Public IP: Used for devices directly accessible on the internet.
- Private IP: Used within private networks (e.g., homes or businesses) and not routable on the internet.
- Static IP: Manually assigned, does not change.
- Dynamic IP: Assigned by DHCP, changes over time.

DHCP (Dynamic Host Configuration Protocol)

DHCP is a network protocol used to automatically assign IP addresses to devices (hosts) on a network.

- Simplified Management: Eliminates manual IP address configuration.
- Address Reuse: IP addresses are leased for a certain period and can be reused when no longer needed.
- Avoids Conflicts: Prevents duplicate IP addresses on the network, ensuring each device gets a unique address.

TCP

TCP (Transmission Control Protocol) provides reliable, ordered, and error-checked delivery of data between applications. Transport layer (Layer 4) of the OSI model.

- Connection-Oriented: TCP establishes a connection between the sender and receiver
- Reliable Delivery: It ensures that data is delivered accurately and in the correct order, using acknowledgments and retransmissions if packets are lost or corrupted.
- Flow Control: TCP manages data flow to prevent the sender from overwhelming the receiver.
- Error Detection: TCP uses checksums to detect and handle data corruption during transmission.
- Segmentation and Reassembly: It breaks large data into smaller packets (segments) for transmission and reassembles them at the destination.

TCP ports

TCP ports are a fundamental part of the TCP/IP protocol suite, used to identify specific processes or services on a device for communication.

- A port is a 16-bit number (0–65535) that serves as an endpoint for network communication.
- It allows a device to differentiate between multiple applications or services running on the same IP address.
- Source Port: The port number of the application initiating the connection.
- Destination Port: The port number of the service or application being accessed.

TCP ports

- Well-Known Ports (0–1023) Reserved for standard services (e.g., HTTP, FTP, SMTP).
- Multiple Services: Ports allow a single device to host multiple networked services simultaneously.
- Routing Traffic: They ensure the correct application receives the transmitted data.
- Security: Firewalls and intrusion detection systems can monitor or block traffic based on ports.
- For example, when you visit a website, your browser (source port) connects to the server's port 443 (HTTPS). After establishing the connection, data flows between these ports.

TCP well-known ports

- 80: HTTP (Hypertext Transfer Protocol)
- 21: FTP (File Transfer Protocol)
- 22: SSH (Secure Shell)
- 23: Telnet
- 53: DNS (Domain Name System)
- 25: SMTP (Simple Mail Transfer Protocol)
- 110: POP3 (Post Office Protocol)
- 143: IMAP (Internet Message Access Protocol)
- 443: HTTPS (HTTP Secure)
- 465: SMTPS (SMTP Secure)

HTTP

The HTTP (Hypertext Transfer Protocol) is the foundation of data communication on the World Wide Web. It is a request-response protocol used for transferring data between clients (such as web browsers) and servers.

- Client: A user’s web browser or any other application that makes requests for resources. Chrome, Edge, Firefox, IE, Opera, Safari etc
- Server: A machine or application that stores resources (like HTML files, images, data) and responds to client requests. Apache HTTP Server, Nginx, IIS, Tomcat etc
- Text based protocol

HTTP structure

- Request:
  - Contains the HTTP method, the path (URL), version. Example: `GET /index.html HTTP/1.1`
  - Headers: Provide additional information about the request or client (e.g., User-Agent, Accept, Host, etc.).
  - Body: (optional) Contains data sent with the request (mostly with POST).

- Reponse:
  - Status Line: Contains the HTTP version, a status code, and a reason phrase (e.g., HTTP/1.1 200 OK).
  - Headers: Provide metadata about the response (e.g., Content-Type, Date, Server).
  - Body: The actual content returned from the server (HTML, JSON, etc.).

HTTP methods and responses

- Methods
  - GET
  - POST
  - PUT/DELETE/HEAD/OPTIONS/PATCH
- Responses
  - 1xx: Informational
  - 2xx: Success
    - 200: OK
  - 3xx: Redirection
    - 301: Moved Permanently
    - 302: Found
    - 304: Not Modified
  - 4xx: Client Error
    - 400: Bad Request
    - 401: Unauthorized
    - 404: Not Found
  - 5xx: Server Error
    - 500: Internal Server Error
    - 503: Service Unavailable

State

- Stateless protocol: Each request is independent and does not store information about previous requests.
- However we need state (i.e a user logs in, adds items to a shopping cart, etc)
- Remember the OSI layer 5 (Session layer) is missing in the TCP/IP model
- Cookies and Sessions: Used to maintain state between requests (e.g., user authentication, shopping carts).
  - From server:
    - Set-Cookie: session_id=312893123789
  - From client:
    - Cookie: session_id=312893123789

HTML

- Markup language used to create web pages.
- Structure: Elements (tags) define the content and layout of a page.
- Example: `<h1>Hello, World!</h1>`
- Nesting: Elements can be nested inside other elements.
  - `<body><div><p>Paragraph 1</p><p>Paragraph 2</p></div></body>`
- Elements can have attributes that provide additional information (e.g., `<img src="image.jpg" alt="Description">`).

Email Protocols

- SMTP (Simple Mail Transfer Protocol): SMTP is used to send emails from a client to the server and from one server to another. It’s primarily responsible for sending and relaying outgoing email messages.
- POP3 (Post Office Protocol, Version 3): POP3 is used by email clients to retrieve emails from a mail server to the client’s device. It's typically used for downloading emails, and once an email is downloaded, it is typically deleted from the server (unless configured otherwise).
- IMAP (Internet Message Access Protocol): IMAP is a more advanced protocol for retrieving emails compared to POP3. It allows users to synchronize their emails across multiple devices and leave messages stored on the server.
- All all layer 7.

SMTP

- When you send an email from your email client (like Gmail, Outlook, or Thunderbird), the client communicates with an SMTP server.
- The SMTP server then forwards the message to the recipient’s mail server, using DNS (Domain Name System) to find the appropriate destination.
    - A mail exchanger record (MX record) specifies the mail server responsible for accepting email messages on behalf of a domain name. i.e `mail.hcg.gr -> 1.2.3.4`
- Port 25 (original port, commonly blocked by ISPs to reduce spam) / Port 587 & 465 (for secure submission)
- Authentication: SMTP servers often require authentication to prevent unauthorized use (e.g., username and password).
- Security: SMTP by itself does not encrypt the email content, but it's often used with additional security

```
S: 220 smtp.example.com ESMTP Postfix
C: HELO relay.example.org
S: 250 Hello relay.example.org, I am glad to meet you
C: MAIL FROM:<bob@example.org>
S: 250 Ok
C: RCPT TO:<alice@example.com>
S: 250 Ok
C: RCPT TO:<theboss@example.com>
S: 250 Ok
C: DATA
S: 354 End data with <CR><LF>.<CR><LF>
C: Subject: Test message
C:
C: Hello Alice.
C: This is a test message with 5 header fields and 4 lines in the message body.
C: .
S: 250 Ok: queued as 12345
C: QUIT
S: 221 Bye
```

SMTP Message Relaying

- The email client sends a message to the sender’s SMTP server. The message includes the recipient's email address, which the server will use to figure out where to send the email.
- The sender’s SMTP server is responsible for relaying the email to the recipient's mail server. It checks the recipient's email address and uses the DNS (Domain Name System) to determine which server handles emails for the recipient's domain.
- Once the sending SMTP server has the recipient’s mail server’s address, it connects to that server and forwards the email.
- The recipient’s mail server checks whether the message is intended for a valid recipient, and if so, it processes it accordingly.
- The recipient's SMTP server accepts the message (if valid) and stores it, typically in a mailbox managed by IMAP or POP3.

Problems with SMTP

- Lack of Built-in Security
- Email Spoofing
- Open Relays and Spam
- Vulnerability to Spam
- Store-and-Forward Delay
- Email Bounces and Delivery Failures
- Size Limitations
- Solution? **None!**

IMAP (Internet Message Access Protocol)

- IMAP is a protocol used for retrieving and managing emails stored on a mail server. It allows users to access their email from multiple devices while keeping the messages synchronized across them.
- Email Storage: Emails remain on the mail server, and IMAP retrieves a copy for display on the client device.
- Synchronization: Actions performed on one device (e.g., reading, deleting, or organizing emails) are reflected on the server and synchronized with other devices.
- Folders and Labels: IMAP supports server-side folders, allowing users to organize their emails.
- On-Demand Fetching: IMAP fetches email headers first and downloads the full content only when required. This conserves bandwidth and speeds up access on devices with limited storage.

```
S:   * OK IMAP4rev1 Service Ready
C:   a001 login mrc secret
S:   a001 OK LOGIN completed
C:   a002 select inbox
S:   * 18 EXISTS
S:   * FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
S:   * 2 RECENT
S:   * OK [UNSEEN 17] Message 17 is the first unseen message
S:   * OK [UIDVALIDITY 3857529045] UIDs valid
S:   a002 OK [READ-WRITE] SELECT completed
C:   a003 fetch 12 body[header]
S:   * 12 FETCH (BODY[HEADER] {342}
S:   Date: Wed, 17 Jul 1996 02:23:25 -0700 (PDT)
S:   From: Terry Gray <gray@cac.washington.edu>
S:   Subject: IMAP4rev1 WG mtg summary and minutes
S:   To: imap@cac.washington.edu
S:   Message-Id: <B27397-0100000@cac.washington.edu>
S:   MIME-Version: 1.0
S:   Content-Type: TEXT/PLAIN; CHARSET=US-ASCII
S:   )
S:   a003 OK FETCH completed
C    a004 store 12 +flags \deleted
S:   * 12 FETCH (FLAGS (\Seen \Deleted))
S:   a004 OK +FLAGS completed
C:   a005 logout
S:   * BYE IMAP4rev1 server terminating connection
S:   a005 OK LOGOUT completed
```

POP3 (Post Office Protocol version 3)

- POP3 is a protocol used for retrieving email messages from a mail server to a local device. Unlike IMAP, POP3 is designed for simplicity, focusing on downloading and storing emails locally.
  - Email Download: The client connects to the mail server, retrieves all email messages, and downloads them to the local device.
  - Email Deletion: After downloading, emails are *usually* removed from the server by default, meaning they are only accessible on the device where they were downloaded. **Be careful**
- Local Storage: Once downloaded, the emails are managed and stored locally, independent of the server.
- Offline Access: Since emails are stored locally, users can access them without an internet connection after download.
- No Synchronization: Changes made to emails (like reading or deleting) are not reflected on the server or other devices.

```
S:    +OK POP3 server ready <1896.697170952@dbc.mtview.ca.us>
C:    USER mrose
S:    +OK User accepted
C:    PASS tanstaaf
S:    +OK Pass accepted
S:    +OK mrose's maildrop has 2 messages (320 octets)
C:    STAT
S:    +OK 2 320
C:    LIST
S:    +OK 2 messages (320 octets)
S:    1 120
S:    2 200
S:    .
C:    RETR 1
S:    +OK 120 octets
S:    <the POP3 server sends message 1>
S:    .
C:    DELE 1
S:    +OK message 1 deleted
C:    RETR 2
S:    +OK 200 octets
S:    <the POP3 server sends message 2>
S:    .
C:    DELE 2
S:    +OK message 2 deleted
C:    QUIT
S:    +OK dewey POP3 server signing off (maildrop empty)
```

Email spam/phising

- Email-based attacks are a common vector for cybercrime. They range from mass spam to highly targeted spear-phishing attacks. Learn how they work and how to protect yourself.
- Spam: Unsolicited, unwanted, and often fraudulent emails.
- Phishing: Deceptive emails that attempt to trick recipients into revealing sensitive information, such as passwords, credit card numbers, or other personal data.
- Spear phising: Highly targeted phishing attacks that use specific information about the victim to increase the likelihood of success.

What is Spam?

- Definition: Unsolicited bulk emails, often sent for advertising or malicious purposes.
- Characteristics:
  - Large volume of emails.
  - Sent to random recipients.
  - Low personalization.
- Risks:
  - Waste of resources (time, storage).
  - Potential to deliver malicious content.

What is Phishing?

- Definition: Deceptive emails that attempt to trick recipients into revealing sensitive information, such as passwords, credit card numbers, or other personal data.
- Characteristics:
  - Fake sender addresses or domains.
  - Urgent language to instill panic (e.g., "Your account will be locked").
  - Requests for credentials or financial information.
- Common Targets:
  - Online banking users.
  - E-commerce customers.
  - Social media users.
  - Government agencies.

What is Spear Phishing?

Definition: A targeted phishing attack tailored to a specific individual or organization.

- Characteristics:
  - Personalized to the victim (e.g., using their name or job title).
  - Often references real-world details to gain trust.
- Goals:
  - Gain access to confidential data.
  - Compromise accounts or systems.

How These Attacks Work

- Crafting the Email: Malicious actors use social engineering techniques.
- Delivery: Exploiting email systems to bypass filters.
- Deception: Mimicking trusted entities to deceive users.
- Exploitation: Capturing data or infecting systems upon interaction.

How to protect yourself

- Use strong, unique passwords for all your accounts.
- Enable two-factor authentication (2FA) for all your accounts.
- Be suspicious of emails that request urgent action, especially if they are from unknown sources.
- Verify the legitimacy of any email before clicking on links or providing personal information.
- Keep your software and email clients up to date with the latest security patches.
- Use spam filters and antivirus software to detect and block spam and phishing emails.
- Learn how to identify phishing emails.
- Be very careful with attachments. Learn how to open them safely.

⚠ Stay Alert. Stay Secure. ⚠

Cryptography: A Brief Introduction

🔑 Securing communication

- Shared key cryptography
- Public key cryptography
- Hashing
- Digital signatures

Shared key cryptography

- Same key for encryption and decryption
- Key distribution problem
- Example: Caesar cipher, Vigenère cipher, Enigma machine, AES

Caesar cipher

- Shift letters in the alphabet by a fixed number of positions
- Example: `HELLO -> IFMMP` (shift by 1)
- ABCDEFGHIJKLMNOPQRSTUVWXYZ
- Very easy to break

Vigenère cipher

- Use a keyword to shift letters in the alphabet
- Example: `HELLO -> IGOPP` (keyword: `ABCD` shift by 1234)
- ABCDEFGHIJKLMNOPQRSTUVWXYZ
- Easy to break if you can do some analysis

One time pad

- Most methods can be broken if you have enough time, resources and access to ciphers
- One time pad is unbreakable
- Uses a random key as long as the message
- Difficult to use in practice

Modern shared key cryptography

- AES (2001): Very complex, very secure

> The design and strength of all key lengths of the AES algorithm (i.e., 128, 192 and 256) are sufficient to protect classified information up to the SECRET level. TOP SECRET information will require use of either the 192 or 256 key lengths. The implementation of AES in products intended to protect national security systems and/or information must be reviewed and certified by NSA prior to their acquisition and use.

- DES (1977): Very simple, very weak; not used anymore
- ChaCha20 (2008): Very fast in software, very secure
- Use AES-256 for all your encryption needs (and use a good implementation)

Introducing Public Key Infrastructure (PKI)

🔑 A system that secures communication over untrusted networks using **public** and **private keys**.

- Certificates: Digital documents that contain public keys and other information.
- Certificate Authorities (CAs): Trusted entities that issue, manage, and revoke certificates.
- Public Key Cryptography: Uses different keys for encryption and decryption.
- Digital Signatures: Verifiable by the receiver.

Alice and Bob Communicate

The Problem:

- Alice wants to send Bob a confidential message.
- They don’t have a shared secret key (like in symmetric cryptography).
- How can they secure their communication?

The Magic of PKI

Solution: Public Key Cryptography (Diffie-Hellman key exchange 1976)

1. **Key Pair**:
   - Each user has:
     - A **public key** (shared with everyone).
     - A **private key** (kept secret).
2. **How It Works**:
   - Alice uses Bob’s **public key** to encrypt the message.
   - Only Bob’s **private key** can decrypt it.

Characteristics of Public Key Cryptography

- private key *cannot* be derived from the public key
- $D_{privkey}(E_{pubkey}(pt)) = pt$
- Public key is shared with everyone / private key is kept secret
- Encryption:
  - $Encrypt_{pubkey}(plaintext) = ciphertext$
  - $Decrypt_{privkey}(ciphertext) = plaintext$
  - *Decryption is only possible with the private key*
- Signing:
  - For signing we also need * $D_{pubkey}(E_{privkey}(pt)) = pt$
  - $Encrypt_{privatekey}(plaintext) = signature$
  - $Decrtypt_{publickey}(signature) == plaintext$
  - *Verifiable signing is only possible with the private key*

How Alice Sends a Message to Bob

1. **Step 1**: Bob generates a key pair.
   - **Public Key**: `pubB`
   - **Private Key**: `privB`
2. **Step 2**: Bob shares `K_pub_B` with Alice.
3. **Step 3**: Alice encrypts her message:
   - $ciphertext = Encrypt_{pubB}(plaintext)$
4. **Step 4**: Bob decrypts it with his private key:
   - $plaintext = Decrypt_{privB}(ciphertext)$

The Role of Trust

The Challenge:

- How does Alice know Bob’s public key is authentic?

Solution: Certificates and Certificate Authorities (CAs).

1. **Certificate**:
    - A digital document that binds a public key to its owner.
    - Example: Bob gets a certificate for `K_pub_B` from a trusted CA.
2. **Certificate Authority**:
    - A trusted third party that verifies identities and issues certificates.
    - Example: `Let's Encrypt`, `DigiCert`.

Introducing Eve (The Attacker)

The Problem:

- Eve might try to impersonate Bob and trick Alice into using Eve’s public key.

How PKI Solves This:

1. Alice verifies Bob’s certificate.
2. The certificate confirms that `K_pub_B` truly belongs to Bob.
3. Eve’s attack fails because she cannot forge Bob’s private key.

Real-World Example: HTTPS

1. When you visit `https://example.com`:
    - The server provides its public key with a certificate.
2. Your browser:
    - Verifies the certificate using a CA.
    - Establishes a secure connection using the server's public key.

Result:

- Your communication is encrypted and safe from eavesdroppers.

Summary of PKI

Key Concept	Explanation
Public Key	Shared with everyone; used for encryption.
Private Key	Kept secret; used for decryption.
Certificate	Binds a public key to its owner.
Certificate Authority	Verifies identities and issues certificates.

PKI provides a scalable and secure way to enable trust in an untrusted world.

PGP/GPG web of trust

- PGP uses a web of trust to verify the authenticity of public keys
- You can verify the authenticity of a public key by trusting the keys of the people who trust the key you want to verify
- Not used that much anymore; but you can still use it to verify the authenticity of public keys
- PGP is a good way to secure your communication

How Alice verifies Bob's signed message

- Bob signs the message with his private key
- Alice verifies the signature with Bob's public key
- If the signature is valid, Alice knows that the message was signed by Bob
- If the signature is invalid, Alice knows that the message was not signed by Bob and that the message is not authentic

Mallory tries to impersonate Bob

- Mallory signs the message with his private key and sends it to Alice claiming it's from Bob
- Alice verifies the signature with Bob's public key
- Mallory's signature will not be validated with the public key of Bob
- Alice knows that the message was not signed by Bob and that the message is not authentic

Digital signatures

- Remember PKI signing!
- We mainly sign PDF documents; PDFs have support for embedding digital signatures and make it easy to verify them
- We could sign any text document but it's not easy for non-technical users
- How signing works:
    - The contents of the PDF are *hashed* (using a cryptographic hash function)
    - The hash is encrypted with the signer's private key
    - The encrypted hash is embedded in the PDF
    - When verifying, the recipient decrypts the hash with the signer's public key and compares it with the hash of the received PDF
    - If the hashes match, the PDF is authentic and the signature is valid

Digital signatures in Greece

- Greece uses a central system for digital signatures (by the Ministry of Digital Governance)
- Public servants create a digital certificate which is stored on that central system
- The digital certificate is secured with OTP (either by email or by app)
- A valid signed document *also* has an embedded timestamp from a TSA (timestamp authority)
- Learn to identify a signed message
- Learn to differentiate between a valid and an invalid signature
- Learn to sign a message
- Understand the gov.gr document signing and verification process

Electronic Document Management System (EDMS)

- Basic requirements:
  - Verify delivery
  - Verify authenticity
  - Verify timestamp
  - Verify integrity
- HCG uses signed documents delivered by email
- Which requirements are met? Which are not? What can be done to improve?
- Other requirements: Full text search, Document retention, Document classification, RBAC, Audit trail

HCG Apps

- Some useful apps
  - Email
  - nextcloud
  - HRMS
  - Protocol
  - portal
  - ETSD
  - pitheas