



Cybersecurity: An Introduction

Algorithms, Data and Security
A.Y. 2025/26

Valeria Cardellini

Global Governance, 3rd year
Science and Technology Major

Computer security & co.



- **Computer security, cybersecurity** or information technology (IT) **security** is the protection of computer systems (including interconnecting networks) from theft or damage to their hardware, software or electronic data, as well as from disruption or misdirection of the services they provide
- Growing importance due to:
 - Increasing reliance on computer systems, Internet and wireless networks (e.g., Bluetooth and Wi-Fi)
 - Exponential growth of smart devices (smartphones, smart TVs, Internet of Things)

Examples of cybersecurity attacks

- Hackers breached Sony PlayStation network, potentially stealing **personal information** of 77 million gamers (2011)
 - Including names, birthdates, addresses, passwords, logins, online IDs, etc.

<https://www.theguardian.com/technology/2011/apr/26/playstation-network-hackers-data>



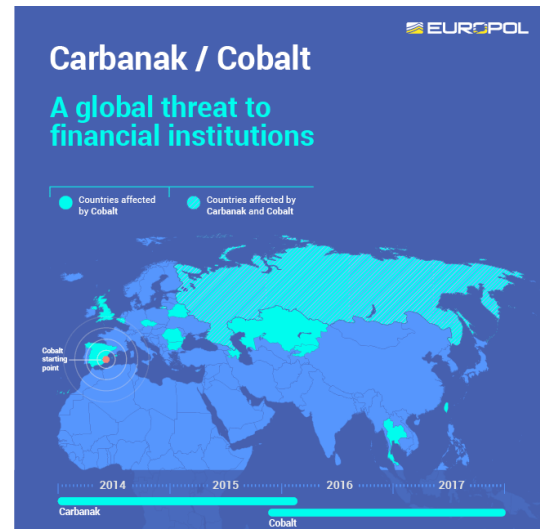
Examples of cybersecurity attacks

- Attackers do not only aim to steal personal data: **impersonation** is also an attacker goal
- In 2011 hackers breached the certificate authority DigiNotar and **stole digital certificates** that were used to impersonate major websites, including Google, Microsoft, Skype, and Yahoo! <https://www.nytimes.com/2011/04/07/technology/07hack.html>
 - This allowed attackers to intercept communications and potentially access sensitive information
- Takeaway: attackers can gain access indirectly by exploiting weak points in a business ecosystem, especially trusted third parties

Examples of cybersecurity attacks

- From access to control: attackers can directly manipulate systems to steal money
- In 2014, the Carbanak *malware* targeted financial institutions via phishing emails and **over \$900 millions were stolen**
 - Attackers were able to manipulate their access to banking networks to steal money in a variety of ways
 - E.g., ATMs were instructed to dispense cash remotely
 - Takeaway: once attackers gain access, they can operate like insiders and control critical systems

Valeria Cardellini - ADS 2025/26



Examples of cybersecurity attacks

- In 2017 a *ransomware* worm named WannaCry spread rapidly through across global networks
 - It infected over 200,000 computers in more than 150 countries by exploiting a vulnerability in Microsoft Windows
 - Once inside a system, it encrypted files and demanded payment in Bitcoin to restore access
 - The attack used an exploit allegedly developed by the NSA and later leaked by a hacking group
 - Takeaway: unpatched systems can enable large-scale, automated attacks



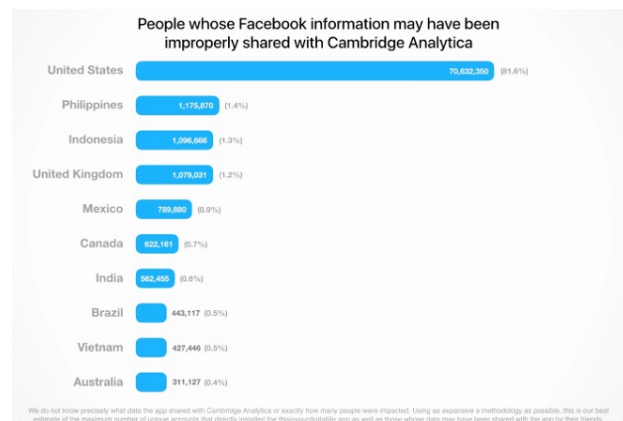
Valeria Cardellini - ADS 2025/26

Examples of cybersecurity attacks

- In 2018 it was revealed that **Cambridge Analytica** had harvested personal data of 87 million Facebook users without their consent
 - Data was collected through a third-party app and used for political profiling and targeted advertising during the 2016 U.S. presidential campaign

https://en.wikipedia.org/wiki/Facebook–Cambridge_Analytica_data_scandal

- Takeaway: personal data can be exploited to influence opinions and behavior at scale



Valeria Cardellini - ADS 2025/26

Examples of cybersecurity attacks

- In 2019, a massive dataset known as “Collection #1” **exposed over 1 billion email–password combinations**. Rather than a single breach, it compiled data from thousands of previous breaches and was shared on hacking forums and the dark web

<https://www.theguardian.com/technology/2019/jan/17/breached-data-largest-collection-ever-seen-email-password-hacking>
- Sites to check if your email was exposed in this or some other known **data breach**
 - E.g., <https://haveibeenpwned.com>
 - If your email appears in a breach, change your passwords immediately and avoid reusing them across sites

Valeria Cardellini - ADS 2025/26

Examples of cybersecurity attacks

- Facebook (Meta) **data breach** (2019-2021)
 - Personal data of over 530 million users was leaked on hacker forum online after being scraped via platform vulnerability
 - Exposed data included phone numbers, email addresses, locations, birthdates, and user IDs
 - Dataset exposed publicly in 2021, highlighting the risks of previously unpatched or misused features
- Impact: fueled phishing and social engineering attacks
- Takeaway: even without a direct breach, exposed data can have long-term security consequences

Valeria Cardellini - ADS 2025/26

8

Data breach

- **Data breach**: security incident in which sensitive, confidential or protected data has been accessed and disclosed (or lost) in an unauthorized fashion
 - May involve **personally identifiable information** (PII), personal health information, trade secrets or intellectual property
 - Most common form is about PII (e.g., names, credit card numbers)
 - Larger in number and impact in the last years
 - Huge cost
 - Average cost of a data breach: \$4.45 million in 2023
 - Global annual cost: \$10 trillion per year by 2025

Vulnerabilities and attacks

- A **vulnerability** is a weakness in design, implementation, operation or internal control
- Vulnerabilities are often hunted or exploited with the aid of automated tools using customized scripts
- To secure a computer system, it is important to understand which **cybersecurity attacks** can be made against it

Cybersecurity attacks: backdoor

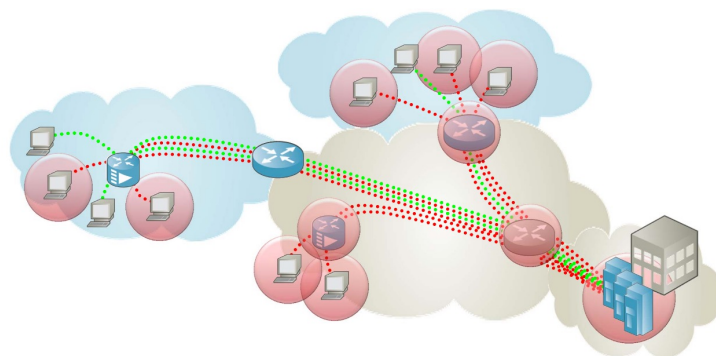
- **Backdoor**: any secret method of bypassing normal authentication or security controls
 - May exist for a number of reasons, including by original design or from poor configuration
 - May have been added by an authorized party to allow some legitimate access, or by an attacker for malicious reasons
 - In any case, it is a vulnerability

Cybersecurity attacks: DoS

- **Denial-of-service attack (DoS)**: designed to make a machine or network resource unavailable to its intended users.
 - Attackers can deny service to individual victims, e.g., by entering a wrong password enough consecutive times to cause the victims account to be locked, or by overloading the capabilities of a machine or network and block all users at once
 - Distributed denial of service (DDoS) attacks: the attack comes from a large number of points and defending is much more difficult

Cybersecurity attacks: DDoS

- How does a DDoS attack work?



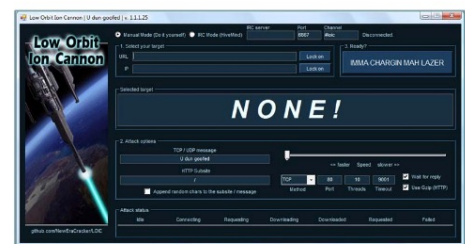
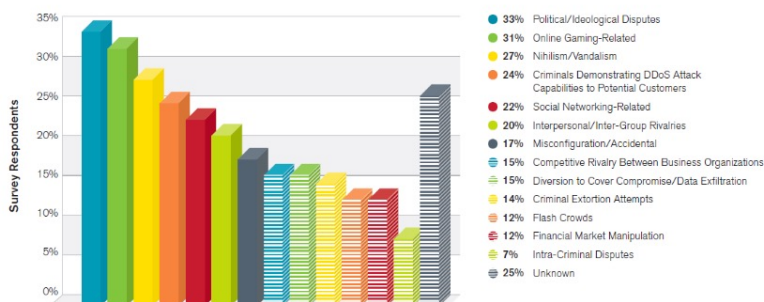
- During a DDoS attack, compromised hosts or bots coming from distributed sources overwhelm the target with illegitimate traffic so that the servers can not respond to legitimate clients

Cybersecurity attacks: DDoS

- Why are DDoS attacks happening?
 - Hacktivism: volunteer botnets
 - Extortion: “commercial” botnets
 - Online demonstrations
 - Cyberwar



Most Common Motivations Behind DDoS Attacks



Low Orbit Ion Cannon (LOIC)

Cybersecurity attacks: direct-access

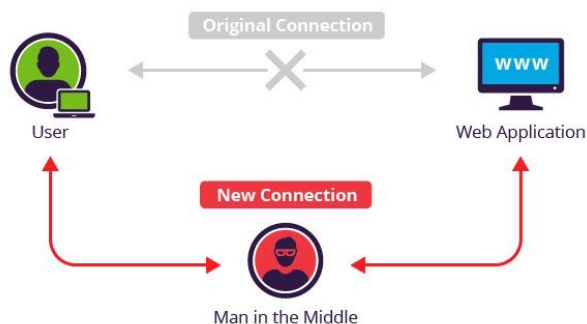
- **Direct-access attack:** unauthorized user gaining physical access to a computer or a device
 - Can directly copy data from it
 - May also compromise security by making operating system modifications, installing malware, ...
 - Disk encryption is a method to prevent it
 - Technology which protects information by converting it into unreadable code that cannot be deciphered easily by unauthorized people
 - Impacts on performance (less evident in recent disks)

Cybersecurity attacks: eavesdropping

- **Eavesdropping**: act of surreptitiously listening to a private conversation, typically between hosts on a network
 - E.g., programs such as Carnivore and NarusInSight used by FBI and NSA to eavesdrop on Internet service providers
 - E.g., an attacker within reception range of an unencrypted Wi-Fi access point
 - Even machines with no contact to outside world can be eavesdropped upon via monitoring the faint electromagnetic transmissions generated by hardware
 - More generally: **man-in-the-middle** attack

Cybersecurity attacks: MITM

- **Man-in-the-middle attack** (MITM): the attacker secretly relays and possibly alters communications between two parties who believe they are directly communicating with each other

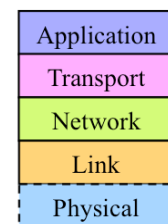


Cybersecurity attacks: MITM

- Two phases to perform a MITM attack:
 - [Interception](#)
 - [Decryption](#)
- *Interception*: the attacker stays in between the data stream, captures and collects the data received from the sender to later manipulate, reuse or sell it
- *Decryption*: the attacker analyzes the used data encryption, try to decrypt and reuse

Cybersecurity attacks: MITM

- Networking stack
 - Network communication protocols are organized in a stack
 - Application protocols (e.g., HTTP)
 - Transport protocols (e.g., TCP)
 - Network protocols (e.g., IP)
 - Link protocols (e.g., Ethernet)
 - Different addresses are used at the various layers to identify communicating entities
 - E.g., IP addresses, MAC addresses
- How can the attacker intercept user traffic?
 - [Rogue access point](#)
 - [DNS spoofing](#)
 - [ARP spoofing](#)

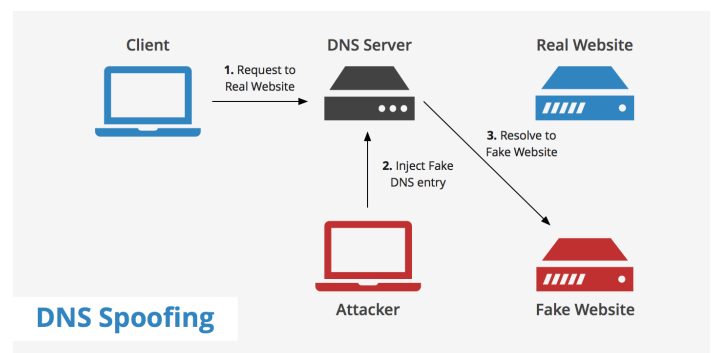


Cybersecurity attacks: MITM

- Rogue access point
 - Wireless access point (AP): networking hardware device that allows other Wi-Fi devices to connect to a wired network or wireless network
 - **Rogue AP**: wireless AP installed on a secure network without explicit authorization from a local network administrator, whether added by a well-meaning employee or by a malicious attacker

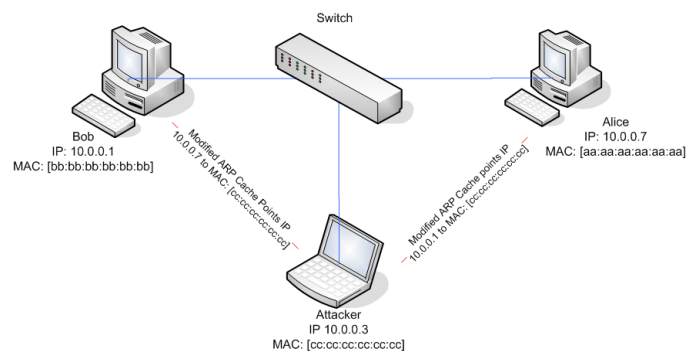
Cybersecurity attacks: MITM

- DNS spoofing
 - **Domain Name System** (DNS) resolves domain names (e.g., web.uniroma2.it) to IP addresses (e.g., 160.80.1.246)
 - When using a DNS spoofing attack, the attacker attempts to corrupt DNS information, altering the IP address of the destination website with that of the malicious host, so that data is sent to attacker's website



Cybersecurity attacks: MITM

- ARP spoofing
 - [Address Resolution Protocol](#) (ARP) resolves the IP address to corresponding MAC address of a device in a local network
 - The attacker links its MAC address to the IP of a legitimate user so that data sent by the user to the host IP address (e.g., to Alice) is instead transmitted to the attacker



Valeria Cardellini - ADS 2025/26

22

Cybersecurity attacks: MITM

- How to prevent MITM attacks?
 - Exchange messages over a secure channel
 - Use some method of authentication for messages that requires an exchange of information (such as public keys, see later)
 - Most cryptographic protocols include some form of [endpoint authentication](#)
 - E.g., Transport Layer Security (TLS) can authenticate one or both parties using a mutually trusted third party called a [certificate authority](#)

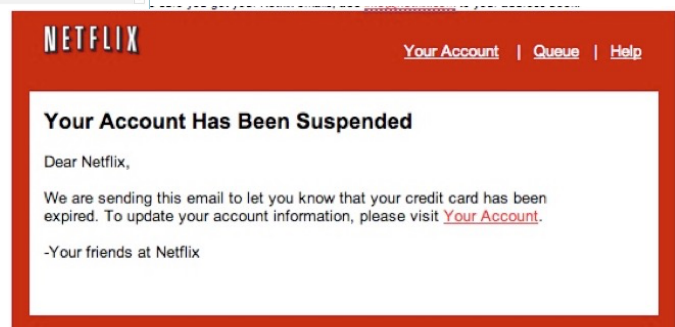
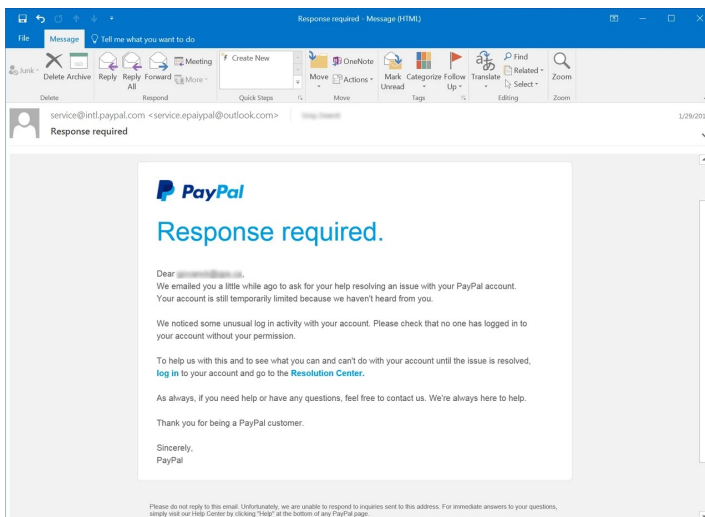
Valeria Cardellini - ADS 2025/26

23

Cybersecurity attacks: phishing

- **Phishing**: fraudulent attempt to steal **sensitive information** such as usernames, passwords, and credit card details directly from users
 - Typically carried out by **email spoofing** or instant messaging
 - Directs users to enter details at a fake website whose look and feel are almost identical to the legitimate one
 - Can be classified as a form of social engineering

Phishing: examples



Phishing: how to avoid

- Some useful suggestions
 - Think before clicking!
 - Install anti-phishing toolbar in your browser
 - Verify web site's security
 - Make sure site's URL begins with "https" and there should be a closed lock icon near the address bar
 - Keep your browser up to date
 - Be wary of pop-ups
 - Never give out personal information
 - Use antivirus software

<https://www.phishing.org/10-ways-to-avoid-phishing-scams>

Cybersecurity attacks: spear phishing

- **Spear phishing**: targeted phishing campaign that appears more credible to its victims by gathering specific information about the target, and thus has a higher probability of success
 - A spear phishing e-mail may spoof an organization (such as a financial institution) or individual that the recipient actually knows and does business with, and may contain very specific information (e.g., recipient's first name rather than just "Dear user")

Cybersecurity attacks: privilege escalation

- **Privilege escalation**: an attacker with limited or restricted access is able to elevate their privileges or access level without authorization
 - E.g., a standard user might exploit a vulnerability to gain access to restricted data or even obtain “root” access, giving them full, unrestricted control over the system

Cybersecurity attacks: social engineering

- **Social engineering**: involves manipulating individuals into revealing confidential information, such as passwords or credit card numbers, often by impersonating trusted entities like banks, contractors, or customers
 - E.g., email phishing
- Evolving nature of social engineering attacks
- Critical need for organizations to implement robust cybersecurity
 - Including employee training, threat detection systems, multi-factor authentication

Cybersecurity attacks: social engineering

- Some trends in 2025
 - **Credential theft** via social engineering largely increased because of the **use of generative AI**, which enables attackers to craft more convincing phishing messages, achieving higher click-through rates compared to human-written messages
 - **Business Email Compromise (BEC)** attacks, where attackers impersonate executives to trick employees into transferring funds
 - **Smishing (SMS phishing)** and **quishing (QR code phishing)** attacks have seen substantial increased
 - **Voice phishing (vishing)** attacks surged, with threat actors impersonating IT support staff to gain unauthorized access to systems

Cybersecurity attacks: spoofing

- **Spoofing**: act of masquerading as a valid entity through falsification of data, in order to gain access to information or resources that one is otherwise unauthorized to obtain
- Several types of spoofing, including:
 - **Email spoofing**, where an attacker forges the sender (e.g., *From* field) email address
 - **IP address spoofing**, where an attacker alters the source IP address in a network packet to hide their identity or impersonate another computing system
 - **Biometric spoofing**, where an attacker produces a fake biometric sample to pose as another user

Cybersecurity attacks: tampering

- **Tampering**: malicious modification of products, some examples:
 - "Evil Maid" attack: attack on an unattended device, in which an attacker with physical access alters it in some undetectable way so that they can later access the device or data on it
 - Hardware trojan: malicious modification of the circuitry of an integrated circuit

Cybersecurity attacks: Malware

- **Malware** is malicious software or code that typically damages or disables, takes control of, or steals information from a computer system
 - Includes botnets, viruses, worms, Trojan horses, backdoors, spyware, adware, ransomware
 - **Botnet**: network of devices infected with malware
 - **Ransomware**: type of malware that threatens to publish the victim's data or perpetually block access to it unless a ransom is paid
 - Many malware attacks rely on social engineering (e.g., phishing) to trick users into opening malicious files or links
 - Zero-day vulnerabilities (previously unknown weaknesses) are often exploited in sophisticated attacks like WannaCry
 - Rise of ransomware has led to massive disruptions, particularly in critical infrastructure sectors

Malware: example

- Flame: malware discovered in 2012 that attacks computers running Windows OS
 - Used for targeted cyber espionage in Middle Eastern countries

```
if not _params.STD then
  assert(loadstring(config.get("LUA.LIBS.STD"))())
  if not _params.table_ext then
    assert(loadstring(config.get("LUA.LIBS.table_ext"))())
  end
  if not LIB_FLAME_PROPS_LOADED then
    LIB_FLAME_PROPS_LOADED = true
    flame_props = {}
    flame_props.FLAME_ID_CONFIG_KEY = "MANAGER.FLAME_ID"
    flame_props.FLAME_TIME_CONFIG_KEY = "TIMER.NUM_OF_SECS"
    flame_props.FLAME_LOG_PERCENTAGE = "LEAK.LOG_PERCENTAGE"
    flame_props.FLAME_VERSION_CONFIG_KEY = "MANAGER.FLAME_VERSION"
    flame_props.SUCCESSFUL_INTERNET_TIMES_CONFIG = "GATOR.INTERNET_CHECK"
    flame_props.INTERNET_CHECK_KEY = "CONNECTION.TIME"
    flame_props.BPS_CONFIG = "GATOR.LEAK.BANDWIDTH_CALCULATOR.BPS_QUEUE"
    flame_props.BPS_KEY = "BPS"
    flame_props.PROXY_SERVER_KEY = "GATOR.PROXY_DATA.PROXY_SERVER"
    flame_props.getFlameId = function()
      if config.hasKey(flame_props.FLAME_ID_CONFIG_KEY) then
        local l_1_1_0 = config.get
        local l_1_1_1 = flame_props.FLAME_ID_CONFIG_KEY
        return l_1_1_0(l_1_1_1)
      end
      return nil
    end
  end
end
```



State threats?

Multi-vector and polymorphic attacks

- **Multi-vector and polymorphic attacks**: combine several types of attacks and changed form to avoid cybersecurity controls as they spread
- Multi-vector has multiple entry points (*vector* refers to entry point used by hackers to infiltrate in a network)
 - Example: attackers first gain access through a **phishing email** (social engineering), then exploit a system **vulnerability** to escalate privileges, and finally deploy **ransomware** or **steal data** through a backdoor
- Polymorphic attacks involve malware that is designed to change its code or behavior every time it is executed in order to avoid detection by security systems
- Can cause more damage and are much harder to defend against

Multi-vector and polymorphic attacks

- How to defend against multi-vector and polymorphic attacks?
- Use behavioral detection systems
 - Security systems should analyze behavioral patterns of malware to detect malicious activity, e.g., if a file is changing or trying to encrypt files
 - Detecting anomalies can exploit AI and ML techniques
- Regular software patching
 - Keep all systems and software up-to-date
- Layered security (defense in depth)
 - Implement multiple layers of security defenses (firewalls, intrusion detection systems, endpoint protection, network segmentation)
- User education
 - Train users to spot suspicious emails and links

Toolbox for security: Cryptography

- Protect data
 1. Stored somewhere
 2. In transit (communication)

Basic concepts in information security

- **Confidentiality/privacy**: third parties should not be able to read private information
- **Integrity**: data should be protected against (accidental or malicious) tampering
- **Availability**: access to services/resources should be available to legitimate users
- **Non-repudiation**: prove integrity and origin of data
- **Authentication**: prove something/someone to be true/valid
- **Access control**: prevent unauthorized access to resources

Authentication

- Something you know (password)
- Something you have (token)
- Something you are (biometry)
- Something you do

Authentication

- **Multi-factor** (or n-factor) **authentication**: combine different pieces of evidence (or factors)
 - The larger n, the stronger the authentication (but more complicated as well...)
- Example of 2-factor authentication: withdrawing of money from an ATM
 - Requires the correct combination of a bank card (something the user possesses) and a PIN (something the user knows)

Attacking passwords

- Brute force: try all possible combinations
- Dictionary: try all words in a dictionary (or their combinations)
- Default passwords (e.g., admin/admin)
- Password linked to user (relative's name, pet name, birthdate, phone number, ...)
- Password resetting
- Password sniffing
- Cracking a hashed password: get hashed password (from data stored in or transmitted by a computer system) and try

Data breaches: what you can do

- As number and scale of data breaches increase, be careful to your [password practices](#)
- Use [at least 2-factor authentication](#)
- Use [strong passwords](#), especially for important accounts, and update them regularly
 - E.g., at least 12 characters, mix uppercase and lowercase letters, numbers and symbols
 - The longer the password, the more difficult to attack the password by brute force
- Security experts don't rule out analogue books containing passwords
 - As long as these are not stored on your device or with it!

Cryptography

- Practice and study of techniques for secure communication in the presence of third parties called adversaries
 - From Greek words [crypto](#) (meaning hidden), and [graphein](#) (meaning writing)
- Includes everything from hiding messages in plain sight to scrambling message content so that no one without the secret key can understand the message

Two examples of cryptography from history

- Histiaeus encouraged Aristagoras of Miletus to revolt against the Persians
 - Wrote message on shaved head of the messenger, and sent him after hair growth
- Scytale from Sparta
 - Wrap a strip of paper around a tube of specific size, then write the message sideways (generally one letter per strip). Only someone with same size tube can read the message

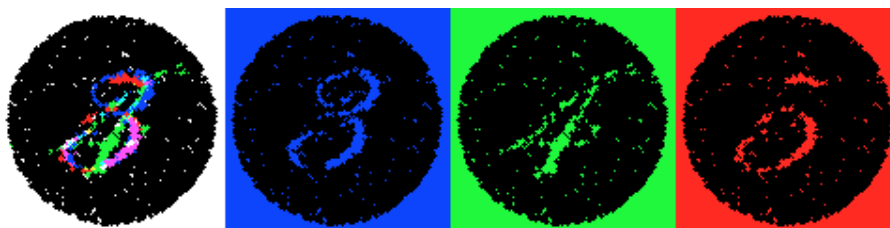


Valeria Cardellini - ADS 2025/26

44

Steganography

- One aspect of cryptography
 - See Histiaeus example
- Practice of concealing a file, message, image, or video within another file, message, image, or video
 - From Greek words *steganos* (meaning covered, concealed, or protected), and *graphein* (meaning writing)

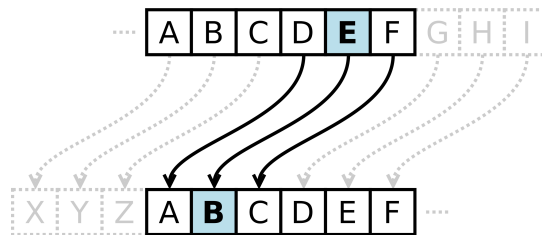


Valeria Cardellini - ADS 2025/26

45

Another example of cryptography from history

- Another famous example of cryptography from history: alphabet shift **ciphers** used by Julius Cesar
 - Example: the letters in the alphabet are shifted 3 in one direction to *encrypt* and 3 in the other direction to *decrypt*



Crypt DCODEX with a shift of 3 -> AZLABU

Decrypt AZLABU with a shift of 3 -> DCODEX

Encryption and decryption

- **Encryption** is the process of encoding information using an algorithm
 - Converts the original representation of the information (**plaintext**), into an alternative form (**cyphertext**)
 - Cyphertext contains a form of the original plaintext that is unreadable by a human or computer without the proper cipher to decrypt it
- **Decryption** is the process of converting cyphertext back to plaintext

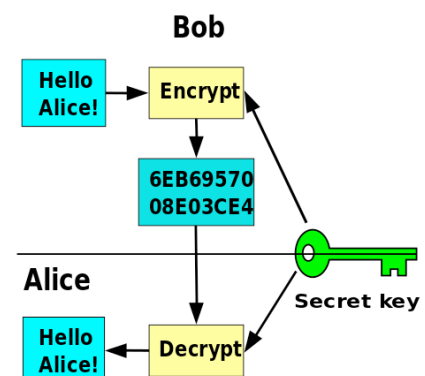
Modern encryption methods

- **Private-key cryptography** (symmetric-key algorithm): the same key is used for encryption and decryption
- **Public-key cryptography** (asymmetric-key algorithm): two different keys are used for encryption and decryption
- The keys represent a shared secret between two or more parties

Private-key cryptography



- Two parties share the **same (secret) key**
 - Single key used both to encrypt/decrypt
- In general (with the same algorithm):
 - The longer the key, the stronger the security
 - The longer the key, the slower the algorithm
 - Key with n bits: brute force attacks will require 2^n attempts
 - With current technology, $n \geq 128$ enough (tradeoffs security vs. performance)

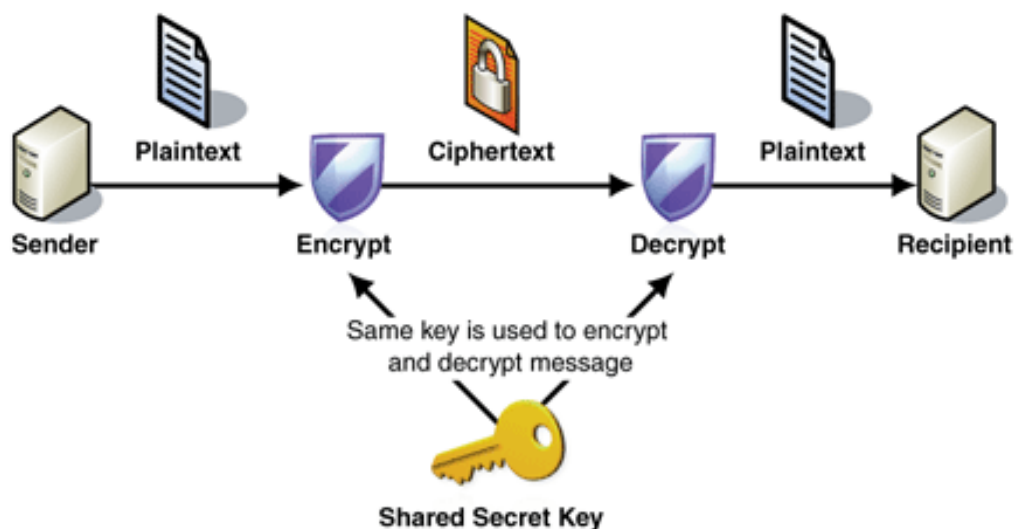


Private-key cryptography

- How does it work?
 1. Sender and receiver share same secret key
 2. Sender encrypts plaintext with secret key
 3. Sender sends cyphertext over insecure channel (Internet)
 4. Receiver decrypts cyphertext and recovers original plaintext
 5. Sniffer in the middle can only access cyphertext

Private-key cryptography

- Also called **symmetric-key cryptography**



Private-key cryptography: pros and cons

- Pros:
 - Usually very fast (good tradeoff between security and performance)
- Cons:
 - Key management: who generates and distributes secret keys?
 - Scalability: N parties imply a total of $O(N^2)$ secret keys...

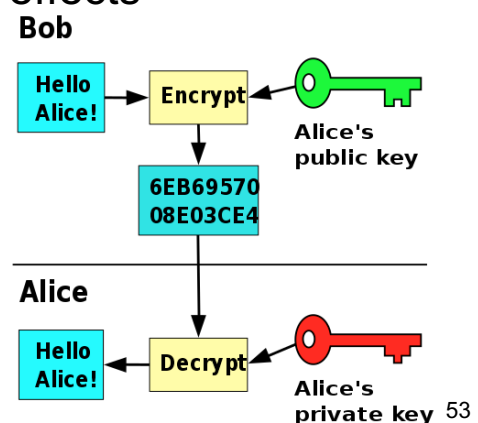
How many links?

$$N(N-1)/2 = O(N^2)$$



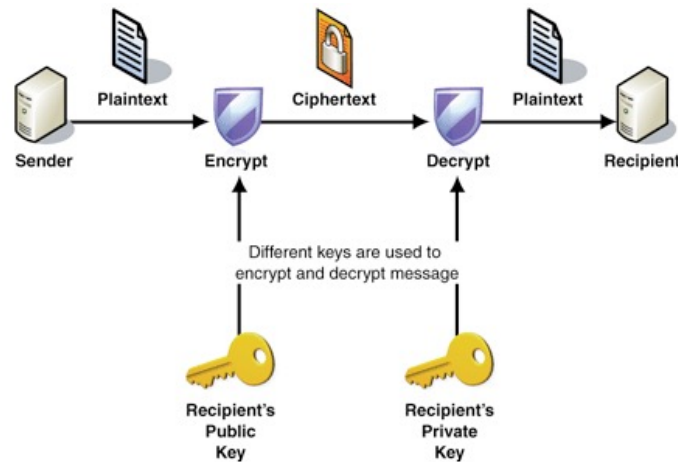
Public-key cryptography

- Two parties use **two (distinct) keys**: one **public key** and one **private key**, which form a key pair
 - Proposed by W. Diffie and M. Hellman in 1976
- The two keys are used for encryption and decryption
 - The keys reverse each other's effects
 - Public key can be freely distributed to communicating parties
 - Private key should be kept secure by its owner



Public-key cryptography

- Also called **asymmetric-key cryptography**



- In figure: encrypt with public key and decrypt with corresponding private key
- Can also do the reverse process: encrypt with private key and decrypt with corresponding public key

Valeria Cardellini - ADS 2025/26

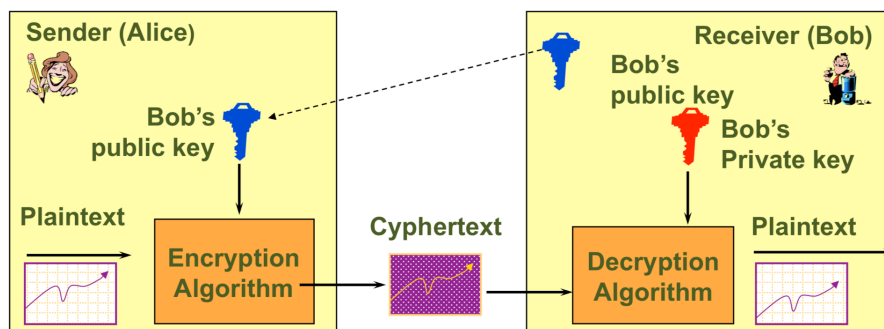
54

Public-key cryptography

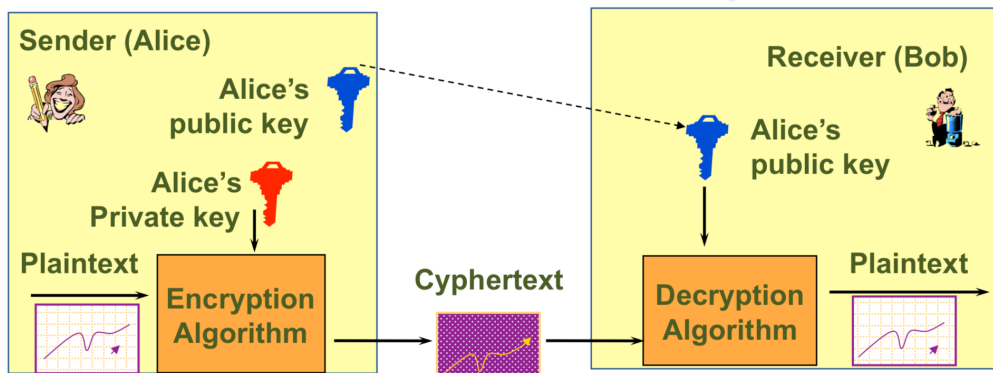
- Properties of public-key cryptography:
 - If you know the public key, it is computationally difficult to guess the private key, even though they are necessarily related
 - A message encrypted with public key can be decrypted **only** with the corresponding private key
 - Likewise, a message encrypted with private key can be decrypted **only** with the corresponding public key

Public-key cryptography: Confidentiality

- Bob makes sure his public key is known
- Alice encrypts plaintext with Bob's public key
- Bob decrypts cyphertext with his own private key. He is the only that knows this private key, so he is the only one who can read the plaintext



Public-key cryptography: Authentication and non-repudiation



- Alice encrypts message to Bob with her private key
- **Authentication:** Bob is sure that the message comes from Alice; indeed, only Alice knows her private key (Alice's public key is the only one that can decrypt this message)
- **Non-repudiation:** Alice cannot claim she didn't send the message (she is the only one the knows her private key)

Public-key cryptography: pros and cons

- Pros:
 - Key exchange is no longer critical
 - No scalability issue: n users, n keys
 - Authenticity and non-repudiation guarantees
- Cons:
 - Much slower (10000x) than private key

Private-key vs. public-key cryptography

- | | |
|--|---|
| <ul style="list-style-type: none">• Private-key<ul style="list-style-type: none">– Same key for encryption and decryption– Algorithms are faster– Key distribution is problematic– One key for each pair of users– Cannot guarantee non-repudiation | <ul style="list-style-type: none">• Public-key<ul style="list-style-type: none">– Two different keys for encryption and decryption– Algorithms are slower– Key distribution is simpler– n keys for n users– Guarantees non-repudiation |
|--|---|

Why complexity matters in cryptography...

- In cryptography, security often relies on computational hardness
- Problems are designed to be easy to compute in one direction
 - But extremely hard to reverse without a secret key
- This leads to the use of complex mathematical algorithms
 - E.g., large prime factorization
- Security depends on classical computers needing impractical time

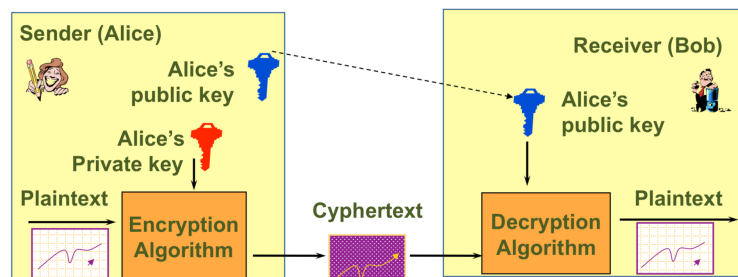
... and the role of Quantum Computing

- Quantum computing
 - Computing paradigm that uses quantum states (such as superposition and entanglement) to represent and manipulate information
- Quantum computing changes the assumption of impractical time
 - Quantum algorithms can solve certain “hard” problems much faster
 - Example: Shor’s algorithm breaks widely used public-key schemes
- Implication:
 - Some cryptographic systems may become insecure in the quantum era
 - Drives research in post-quantum cryptography

Traditional handwritten signature

- Created manually
 - Verified manually (robust?)
 - Non-forgable (robust?)
 - Non-repudiation (robust?)
 - Affixed to document (non-transferable)
-
- Can we do the same in the digital world having a signature that is more difficult to forge than handwritten signature? **Digital signature**

Public-key cryptography and digital signature



- Public-key cryptography is fundamental to implementing digital signature
 - Cyphertext is a sort of digital signature on plaintext
 - If Alice sends both plaintext and cyphertext, Bob would be able to verify Alice's signature (authentication and non-repudiation)
- There is an issue related to performance (not security), especially for large documents

Message digest and digital signature

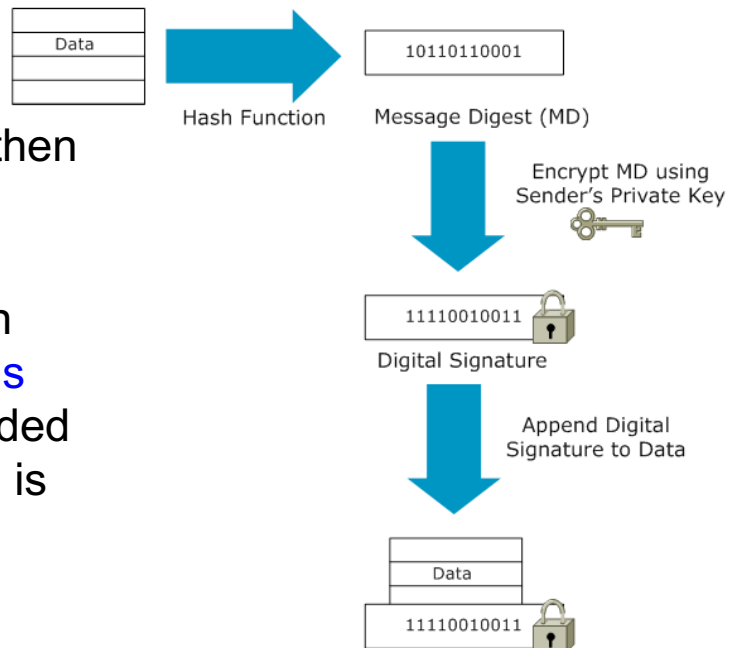
- To avoid performance issue, let's use a fingerprint of the message (or document), called **message digest**
- Message digest is a **fixed-size numeric representation** generated from the original message using a hash function
 - Since digest is much smaller than original message, it speeds up the signing process
- Message digest is then encrypted using the sender's private key, forming a digital signature

Hashing as a message digest

- Message digest is computed by a **hash function**, which is a transformation that meets the following criteria:
 - *Coherent*: same text, same fingerprint (digest)
 - *Random*: to prevent attacks
 - *Collision-free* (i.e., when two messages hash to same digest): collision probability is very small
 - *One-way* (i.e., irreversible): cannot go back from digest to text
 - *Uniform*: evenly distributed hash values

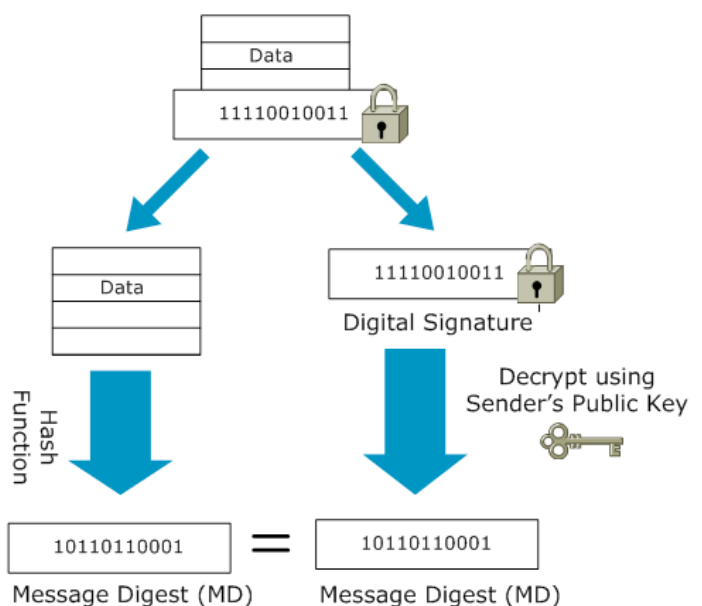
How to sign

- **Message digest** is computed by using a hash function
- If only one bit of the message is changed, then the message digest changes
- Message digest is then **encrypted with sender's private key** and appended to the message (which is plaintext) as digital signature



How to verify

- In order to verify the signature, the receiver:
 1. Applies the same hash function to the message and computes a message digest
 2. Decrypts digital signature using sender's public key and obtains the message digest of the original message
 3. Compares the two message digests. If they are equal, the signature is OK (i.e., message was signed by sender). If they are different, there is a problem



Digital signature scheme

- A digital signature scheme consists of 3 algorithms
 - A **key generation algorithm** that selects a private key uniformly at random from a set of possible private keys. The algorithm outputs the private key and a corresponding public key
 - A **signing algorithm** that, given a message and a private key, produces a signature
 - A **signature verifying algorithm** that, given the message, public key and signature, either accepts or rejects the message's claim to authenticity

New challenges in cybersecurity

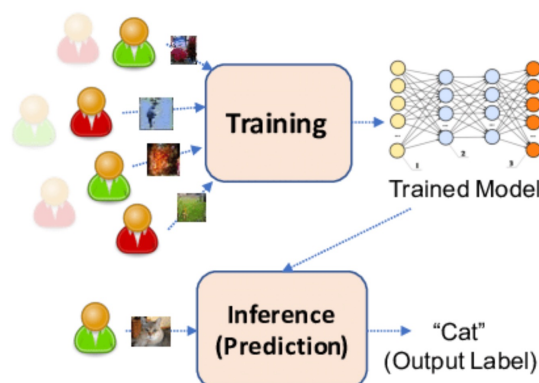
- Different attack scenarios
 - Automotive: the more complex the vehicle IT, the higher the risk of cyberattacks
 - Critical infrastructures (e.g., power grid, water supply, medical services): repercussions of attacks are massive
 - Manufacturing: need to secure industrial IoT devices that communicate with each other
- AI-powered cyberattacks
 - Including AI-generated malware that constantly evolves, deepfake attacks with highly convincing fake audio, video, or images

AI, ML and privacy

- AI and ML require extensive datasets to get accurate predictions
- But ability to collect, analyze, and act on vast amounts of data raises serious **privacy issues**
- **Federated learning (FL)**: novel approach that addresses fundamental privacy issues of traditional ML

ML training and privacy

- **ML model**: what we get from the output of *training* on data; once trained, a model is then used for *inference* (e.g., prediction)
- Traditional ML training: **collect training data centrally**, on a centralized server or data center, where the data is cleaned, preprocessed, and used to train the model



What is federated learning?

- Scenario: many clients (e.g., mobile devices, organizations) act as data sources
- Federated learning (FL) is a **privacy-preserving model training** that addresses privacy issue of traditional ML by **avoiding to directly share data**
 - Introduced by Google in 2017
- Goal: **train collaboratively** a ML model on **multiple clients located at the network edge**, while keeping the training data locally on clients
 - Data is not independently or identically distributed

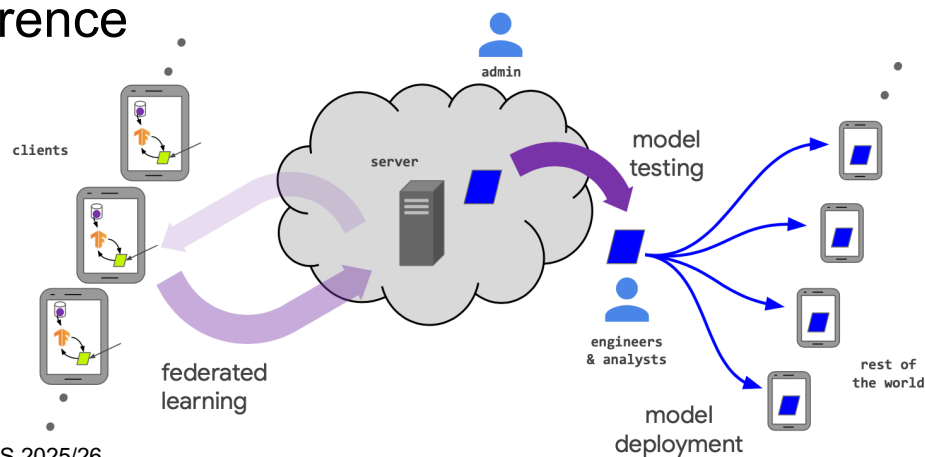
What is federated learning?

- A distributed ML paradigm where **multiple clients collaborate** in solving a ML problem while maintaining privacy
 - Each client's data remains local, without being transferred or exchanged
 - Clients only **share model updates** (gradients or weights) to improve the global model
 - These updates are then aggregated by **a central server** and used to achieve the learning objective



FL system

- Actors in a FL system
 - Clients (data holders): responsible for local training and model updates
 - Central server (aggregator): responsible for training coordination, never sees raw data
- After training, trained model is used for inference



Valeria Cardellini - ADS 2025/26

74

FL training process

- A central server orchestrates the training process, by repeating the following steps until training is stopped (convergence or max number of rounds)
 1. **Client selection**: The server samples from a set of clients meeting eligibility requirements (e.g., in order to avoid impacting device users)
 2. **Broadcast**: The server broadcasts the current model parameters and the training program to the selected clients
 3. **Client computation**: Each selected client locally computes an update to the model by executing the training (e.g., running stochastic gradient descent on its local data) and sends back to the server the newly updated model parameters: **no private data is shared**

Valeria Cardellini - ADS 2025/26

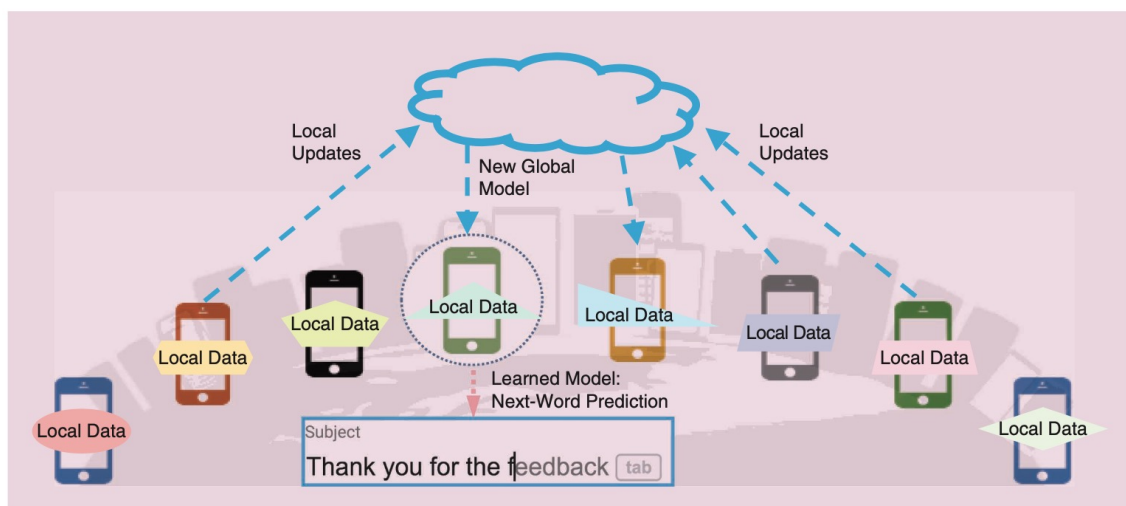
75

FL training process

4. **Aggregation**: The server collects an aggregate of the client updates. For efficiency, stragglers might be dropped
5. **Model update**: The server updates the model based on the aggregated update computed from the clients that participated in the current round (e.g., by performing weighted average) and thus constructs an improved model, which is used in the next round of the model training

FL application: next-word prediction

- Next-word prediction on mobile phones, while preserving privacy of data and reducing strain on network

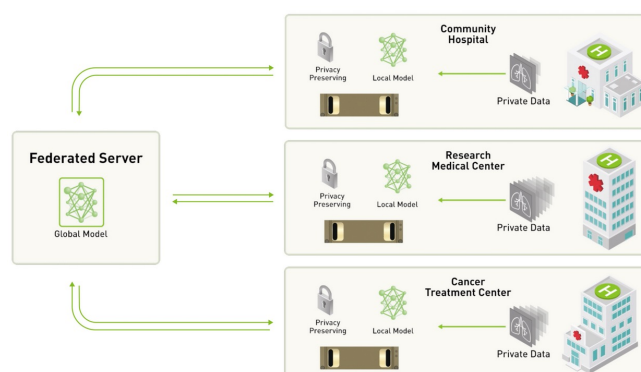


FL example: next-word prediction

- Goal: train a predictor (e.g., a recurrent neural network model) in a decentralized fashion, rather than sending raw data to a central server
- How it works
 - Mobile devices communicate with a central server periodically to learn a global model
 - At each round, a subset of selected devices performs **local training** on their non-identically distributed data, and sends these **local updates** to the server
 - After aggregating the updates, the server sends back the new global model to another subset of devices
 - Iterative training process continues until convergence is reached or some stopping criterion is met

FL application: health images analysis

- Healthcare institutes have archives containing many images, but health data is private and cannot be used by other institutes without the necessary patient consent and ethical approval
- Using FL learning, the ML model can be trained with higher accuracy by using a large number of images but keeping the health data of each institute within its own secure infrastructure



FL challenges

- Communication efficiency
- Heterogeneity of systems and data
- Privacy concerns (new kinds of attacks!)
 - Attacks on model updates: attackers could *recover information about the training data* from the model updates
 - Attacks on trained model: since the trained model also reflects training data, attackers could *infer information about the training data* from the trained model, whether or not FL was used to train it
- Adversarial attacks designed to degrade model performance (affect not only FL)
 - E.g., *model update poisoning*

References

- Computer security
https://en.wikipedia.org/wiki/Computer_security
- Public-key cryptography
https://en.wikipedia.org/wiki/Public-key_cryptography
- Digital signature
https://en.wikipedia.org/wiki/Digital_signature
- McMahan and Ramage, Federated Learning: Collaborative Machine Learning without Centralized Training Data, 2017 <https://research.google/blog/federated-learning-collaborative-machine-learning-without-centralized-training-data>
- 5-minute video: What is Federated Learning?
<https://www.youtube.com/watch?v=X8YYWunttOY>