CVS8301:Cryptography Theory and Applications - Lecture 1

Habeebah Adamu Kakudi (Mrs) PhD

2025-07-28

Table of contents

1 Lecture Notes: Foundations and Purpose of Cryptography

1.1 Introduction to Cryptography

Cryptography is the science of securing communication and data from adversaries. Cryptography is a method of protecting information by transforming it into an unreadable format. Only those who possess a special knowledge – typically referred to as a 'key' or 'cipher' – can decipher this information back into its original form. This method of encoding and decoding information is known as encryption and decryption respectively.

- Etymology: From Greek "kryptos" meaning hidden/secret.
- Purpose: Confidentiality, integrity, authenticity, and non-repudiation in digital communication.

1.2 History of Cryptography

Cryptography is an ancient mathematical science that was originally used for military communications, and designed to conceal the contents of a message should it fall into the hands of the enemy. Recent developments in cryptography have added additional uses, including mechanisms for authenticating users on a network, ensuring the integrity of transmitted information and preventing users from repudiating (i.e. rejecting ownership of) their transmitted messages.

1.2.1 Ancient Era

- **Caesar Cipher**: Julius Caesar's monoalphabetic substitution cipher.
- **Scytale** (Sparta): Transposition using a cylinder.
- **Vigenère Cipher**: More secure polyalphabetic substitution.

1.2.2 Medieval & Early Modern Period

- Al-Kindi's cryptanalysis of substitution ciphers.
- Cipher evolution in European diplomacy and military operations.

1.2.3 Modern Era

- **WWII breakthroughs**: Enigma machine and Bletchley Park.
- Emergence of digital cryptography in the 1970s (DES, RSA).

1.3 Cryptographic Security Goals**

Goal	Description
Confidentiality	Ensuring only intended recipients can access data
Integrity	Ensuring data is not altered in transit
Authentication	Confirming the identity of sender/receiver
Non-repudiation	Ensuring parties can't deny their involvement once a transaction is completed

1.4 Cryptographic Paradigms

Cryptography is the backbone of modern information security. Symmetric-key encryption, one of its oldest and most trusted forms, offers speed and simplicity—but also comes with unique challenges. This chapter explores the mechanisms, uses, limitations, and real-world case studies of symmetric encryption.

1.4.1 Symmetric Cryptography

a. Concept and Terminology

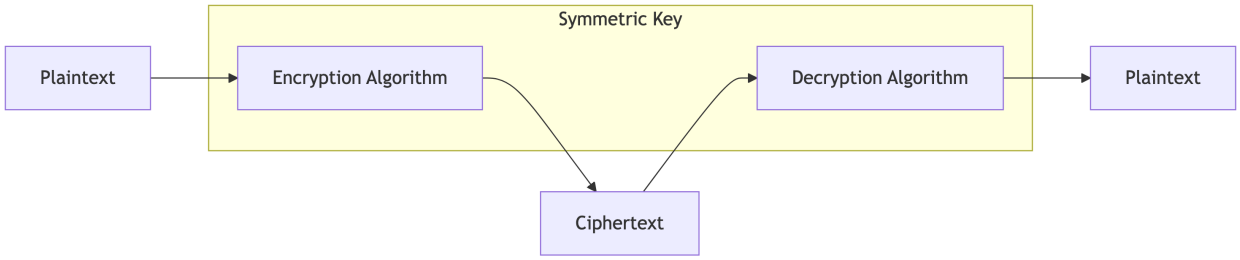
Symmetric Cryptography is a cryptographic method in which the same key is used for both encryption and decryption.

An encryption algorithm is a computational procedure that uses the key to transform plaintext into ciphertext, where a Cipher text is the transformed message.

A decryption is the reverse process, restoring ciphertext into readable plaintext using the same key.

To unscramble the encrypted data, you will need an encryption "key."(kind of like a Password) The key is a very large number that an encryption algorithm uses to change the data back into a readable form. Without the key, no one but the owner of the encrypted data will be able to access a readable version. This unscrambling process is called "decryption."This is what's known as symmetric-key encryption.

Diagram: The Symmetric Encryption Process



Term	Meaning
Plaintext	Original readable message
Ciphertext	Encrypted, unreadable data
Key	Secret code used for encrypting and decrypting
Encryption	Process of transforming plaintext into ciphertext
Decryption	Reversing ciphertext back into plaintext
Cipher	Algorithm used for transformation

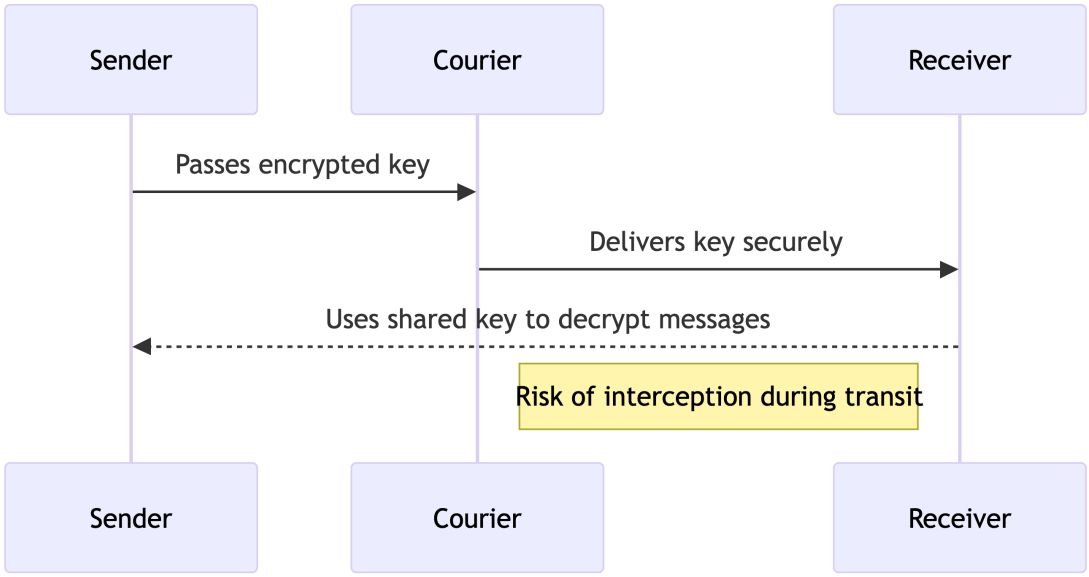
b. Core Properties

Feature	Description
Speed	Much faster than asymmetric encryption; ideal for large data volumes
Efficiency	Less computational overhead, good for devices with limited resources
Simplicity	Relies on fewer operations compared to public-key systems
Security Risk	Key must remain secret—if intercepted, entire system is compromised

c. Key Distribution Challenge

- **Problem**: Securely exchanging the key between sender and receiver without interception.

- **Why Is It Difficult?**
- The key must be **shared securely** between sender and receiver.
- If intercepted, the key can be **used to decrypt all communication**.
- Manual methods (like couriers) aren't scalable; digital channels may be insecure.
- **Solutions Explored**:
  - **Manual Exchange**: Trusted couriers (e.g., locked briefcase scenario)
  - **Secure Channels**: Encrypted networks, dedicated lines (e.g., military-grade comms)
  - **Hybrid Systems**: Using asymmetric cryptography (e.g. Diffie-Hellman) to share the symmetric key safely



d. Spy Movie Analogy: Why It Works

- The metaphor highlights the asymmetry between data security and key vulnerability.
- **Real-life parallel**: In many systems, the most sensitive part isn't the message but the key itself.
- **Illustration**: Think of ransomware—data is encrypted and inaccessible, but the unlock key is sold or extorted.

e. Prominent Symmetric Algorithms

Algorithm	Key Size	Notes
Data Encryption Standard(DES)	Operates on 64-bit blocks with a 56-bit key	Deprecated due to vulnerability to brute-force. In 1999, DES was cracked in 22 hours using distributed computing.
Triple DES (3DES)	Extension of DES that applies encryption three times with key sizes 112/168-bit	An improvement, but now considered outdated. Slower than AES. Considered secure until ~2017, now phased out due to meet-in-the-middle attacks.

Algorithm	Key Size	Notes
Advanced Encryption Standard (AES)	Official replacement for DES. Fast, secure, scalable to 128, 192, or 256-bit keys. Operates on 128-bit blocks; resistant to known attacks.	Modern standard, fast, and widely adopted. Used in TLS, VPNs, SSH, BitLocker, WhatsApp, etc.
Rivest Cipher 4 (RC4)	Variable Key sizes: 40–2048 bits	Stream cipher, now largely discouraged. Biased output and weak key scheduling. Deprecated in modern TLS due to vulnerability to statistical analysis.

f. Real-World Applications

- **Local File Encryption**: BitLocker, FileVault use AES to protect stored data
- **Disk Encryption**: Encrypting entire storage systems with symmetric keys
- **VPNs**: Tunnels encrypted using symmetric algorithms for performance
- **Messaging Apps**: Often use symmetric keys once secure channels are established

g. Thought Exercise

Imagine an organization needs to send sensitive biometric data between two departments. What encryption method should they use—and how should they distribute the key? Reflect on speed, risk, and practicality.

h. Ethical & Legal Considerations

- Governments may require "lawful access" to keys—creating backdoor risks.
- Should data owners be allowed strong encryption even if it impedes law enforcement?
- Encryption policy debates: balancing civil liberties with national security.

i. Reflection Essay Prompts

1. Why does key distribution remain one of the hardest challenges in symmetric encryption?
2. Is it ever safe to transmit a symmetric key digitally? Under what conditions?
3. Explore how asymmetric encryption helps solve symmetric encryption's biggest weakness.

1.4.2 Asymmetric Cryptography

a. Definition

Asymmetric encryption, also known as **public-key cryptography**, revolutionized secure communication. Unlike symmetric methods, it

allows parties to exchange data securely **without prior key sharing**, solving major scalability and trust issues in open networks like the internet. It uses public/private key pairs and enables secure communication without shared secrets. While symmetric encryption uses the *same* key for encryption and decryption, asymmetric encryption relies on a **key pair**—one public, one private. This unlocks powerful capabilities like digital signatures and secure key exchange.

b. Comparative Overview

Feature	Symmetric	Asymmetric
Key Usage	Same key for both	Public & private key
Speed	Fast	Slower
Scalability	Poor in large networks	Excellent
Use Cases	Bulk data encryption	Digital signatures, key exchange
Algorithms	AES, DES, RC4	RSA, ECC, ElGamal

c. RSA (Rivest-Shamir-Adleman)

- Developed in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman.
- Security rooted in difficulty of factoring large prime numbers.
- Key sizes typically range from 1024 to 4096 bits.

RSA Key Flow

Example Use Case

- **Email Signing**: RSA is used to sign emails via protocols like PGP, verifying sender identity and integrity.

d. ECC (Elliptic Curve Cryptography)

- Relies on algebraic structure of elliptic curves over finite fields.
- Provides same security as RSA with **much smaller key sizes**—ideal for constrained devices.
- Popular curves: secp256k1 (used in Bitcoin), Curve25519.

ECC Overview

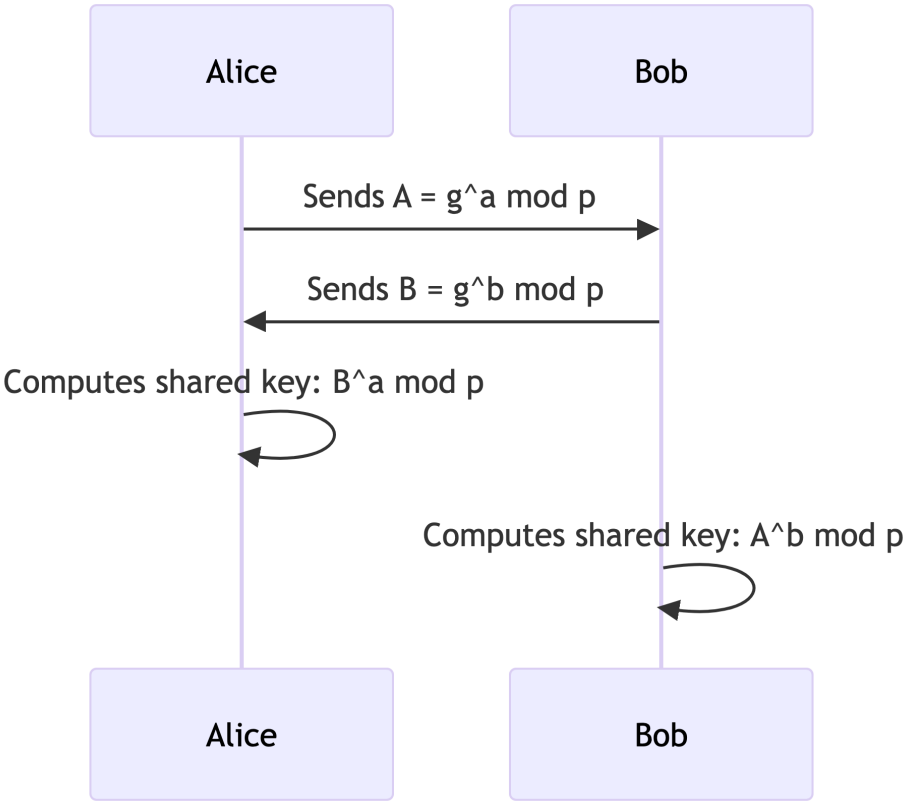
Benefits

- Efficient key generation and encryption.
- Widely used in mobile apps, IoT, cryptocurrencies.

e. Diffie-Hellman Key Exchange

- Allows two parties to securely derive a shared secret over an unsecured channel.
- Foundation of many secure protocols, including TLS.

DH Key Exchange



Real-World Application

Forms basis of **Perfect Forward Secrecy (PFS)** in modern HTTPS connections.

f. Case Study: Securing Telemedicine Platform

A Nigerian startup, *CareBridge*, offers remote consultations and wants to secure doctor-patient communication.

Implementation

- **RSA** used for authentication and digital signing of medical transcripts.
- **ECC** secures communication on mobile devices using end-to-end encryption.

- **Diffie-Hellman** enables dynamic session keys for video calls.

Outcome

- Platform achieved GDPR and NDPR compliance.
- Latency reduced by adopting ECC for mobile endpoints.
- Data tampering attempts declined to zero.

g. Summary Comparison Table

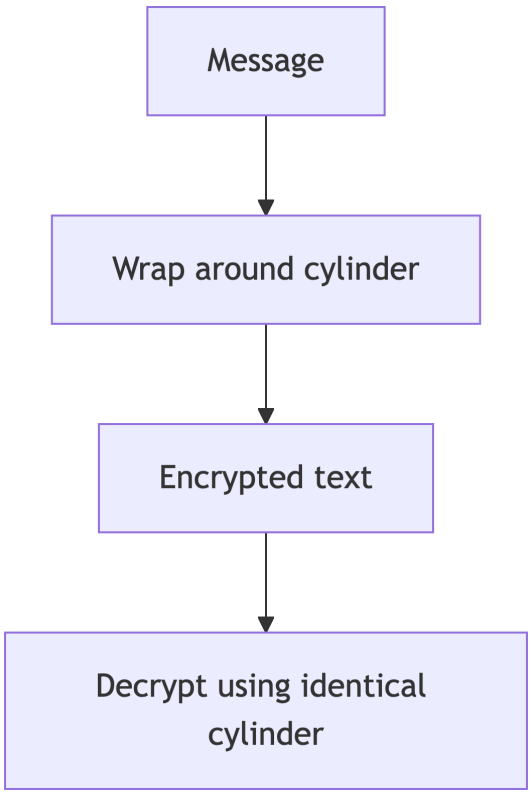
Algorithm	Key Size	Speed	Security Basis	Use Cases
RSA	1024–4096 b	Moderate	Integer factorization	Digital signatures, secure email
ECC	160–512 b	Fast	Elliptic curve math	Mobile security, crypto wallets
Diffie-Hellman	Variable	Moderate	Discrete logs	Secure key exchange

1.5 Case Studies: Ancient and Modern Ciphers

1.5.1 Ancient Ciphers: Foundations of Secrecy

a. Scytale (Spartan Cipher Tool)

- **Origin**: Ancient Sparta (~5th century BCE)
- **Mechanism**: A strip of parchment wound around a cylinder. The message is written along the cylinder's length and appears scrambled when unwound.
- **Security Logic**: Only readable with a matching diameter rod.



Historical Insight

- Used for wartime correspondence to ensure messages couldn't be understood if intercepted.

b. Caesar Cipher

- **Origin**: Named after Julius Caesar, ~1st century BCE
- **Method**: Each letter is shifted by a fixed number (typically 3).
- **Example**:
  - **Plaintext**: "HELLO"
  - **Ciphertext**: "KHOOR" (H+3 = K, E+3 = H…)

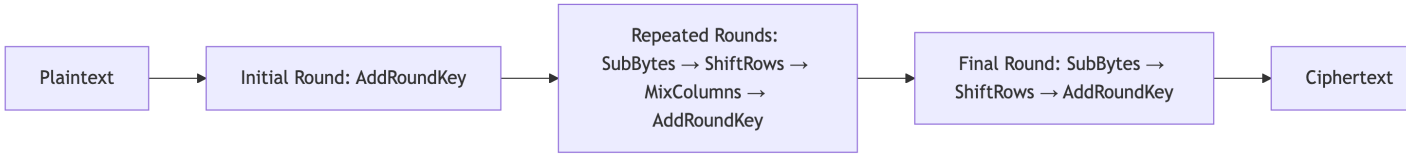
Weakness

- Easily broken via frequency analysis or brute force due to limited shift options (only 25).

1.5.2 Modern Ciphers: Encryption in the Digital Age

a. AES (Advanced Encryption Standard)

- **Adopted**: By NIST in 2001 to replace DES.
- **Block Size**: 128 bits
- **Key Sizes**: 128, 192, or 256 bits
- **Rounds**: 10, 12, or 14 (depending on key length)

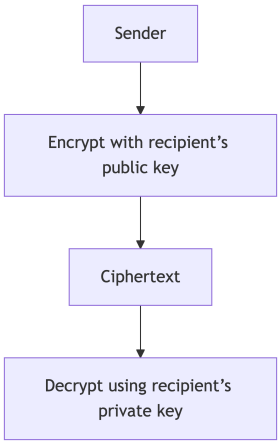


Case Study: WhatsApp Encryption

- Uses AES along with Signal Protocol for end-to-end security.
- Ensures even WhatsApp servers cannot read message contents.

b. RSA (Rivest-Shamir-Adleman)

- **Public-key encryption algorithm** developed in 1977.
- **Security Root**: Difficulty of factoring large prime numbers.
- **Key Size**: Typically 2048 bits and above for strong security.



Case Study: Digital Signatures in Email

- RSA is used in PGP/GPG for signing and encrypting emails.
- Ensures confidentiality, authenticity, and integrity.

1.5.3 Real-World Applications Across Sectors

Sector	Cipher Used	Implementation Notes
Banking Systems	Triple DES	Used for encrypting PINs and card data in legacy ATMs. Slowly phased out due to AES adoption.
Messaging Apps	Signal Protocol	Combines asymmetric key exchange (X3DH, RSA) and symmetric encryption (AES-GCM).
Military Comms	Frequency Hopping, Quantum Key Distribution	Utilized to prevent jamming and interception; quantum methods explore unbreakable key sharing.

1.5.4 Comparative Snapshot: Ancient vs. Modern

Feature	Ancient Cipher	Modern Cipher
Encryption Method	Manual	Algorithmic/Digital
Key Distribution	Physical (e.g., rod)	Secure key exchange
Vulnerability	Obvious patterns	Sophisticated attacks (e.g., side-channel)
Use Case	Military messaging	Internet, finance, apps

1.5.5 Extended Case Study: National e-Health Systems in Africa

Scenario

Nigeria's Ministry of Health implements an encrypted national patient database.

Cryptography Stack

- **AES-256** used for encrypting health records.
- **RSA** secures user authentication and digital consent forms.
- **Diffie-Hellman** negotiates ephemeral session keys for database queries.

Result

- Improved data privacy compliance under NDPR and HIPAA.
- Enabled secure teleconsultation across remote rural clinics.
- System integrated with biometric verification devices.

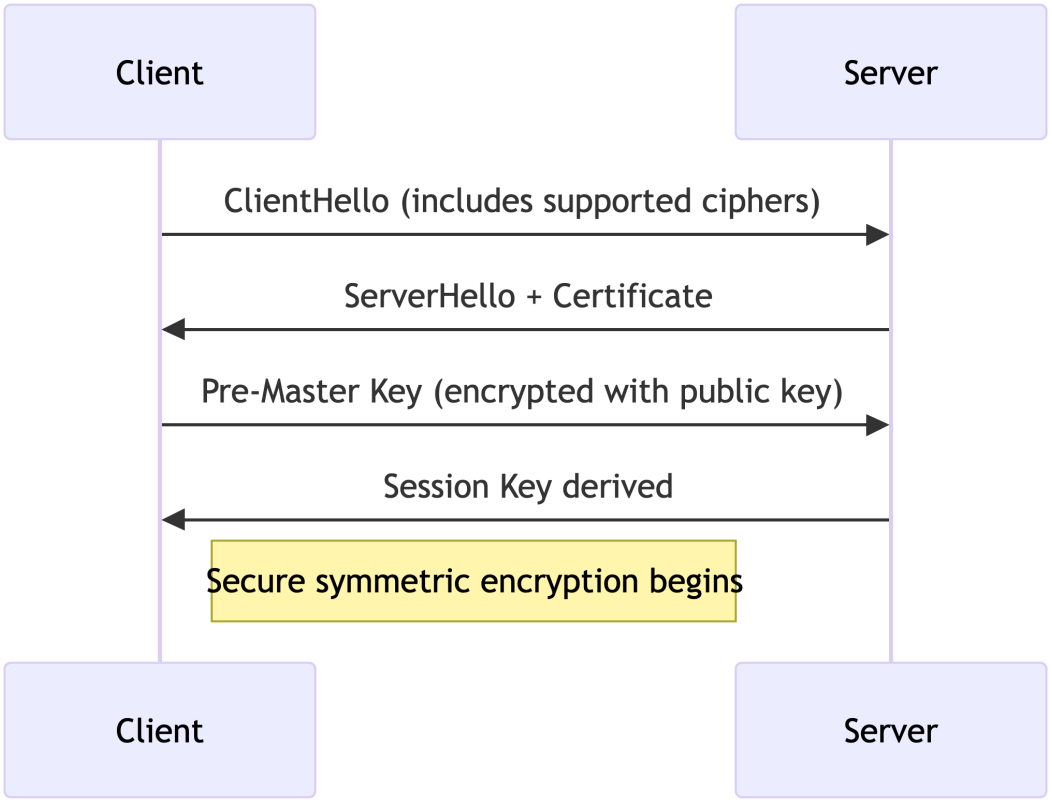
1.6 Applications of Cryptography

1.6.1 E-Commerce Security**

a. SSL/TLS Protocols

- **Purpose**: Secure communication between browsers and servers.
- **Mechanism**: Uses asymmetric encryption for key exchange and symmetric encryption for data transfer.
- **Protocols**: HTTPS is powered by TLS (Transport Layer Security).

b. TLS Handshake



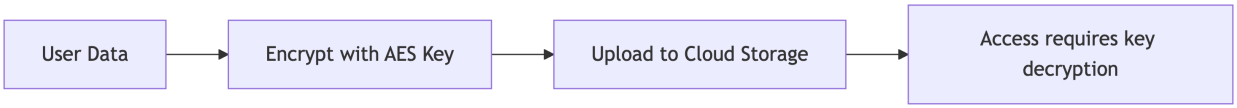
c. Real-World Example - When you enter credit card info on Amazon, TLS ensures it's encrypted in transit, preventing eavesdropping or tampering.

1.6.2 Cloud Security

a. Data-at-Rest Encryption

- **Goal**: Protect stored data from unauthorized access.
- **Method**: AES-256 commonly used to encrypt files before uploading.
- **Key Management**: Often handled via cloud provider's KMS (Key Management Service).

Cloud Encryption Flow



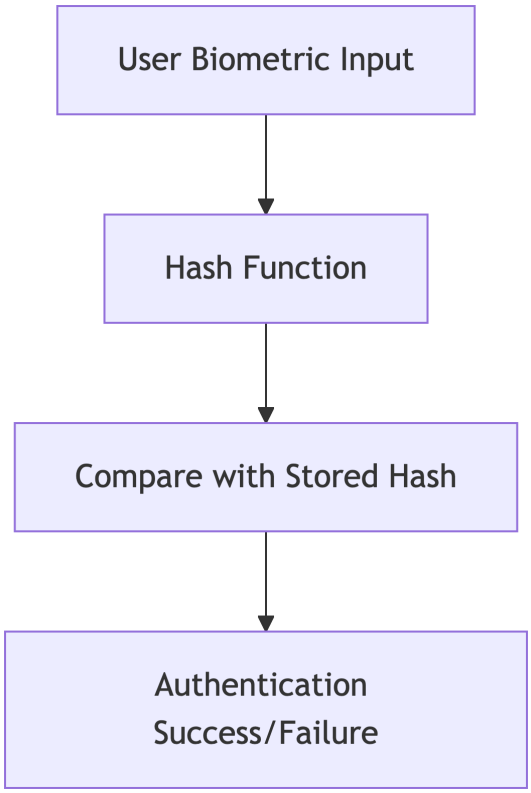
b. Real-World Example - Microsoft Azure and AWS encrypt customer data at rest and in transit using AES and TLS.

1.6.3 Digital Identity

a. Cryptographic Authentication

- **Biometrics**: Fingerprints, facial recognition, iris scans.
- **Protocols**: Combine biometric input with cryptographic hashes and digital signatures.
- **Security**: Prevents spoofing and ensures identity integrity.

Biometric Authentication Flow



b. Real-World Example

- Nigeria's National Identity Management Commission (NIMC) uses biometric data secured with cryptographic protocols for e-ID verification.

1.6.4 Blockchain and Cryptography

a. Core Mechanisms

- **Hash Functions**: SHA-256 used to link blocks securely.
- **Digital Signatures**: Verify transaction authenticity.

- **Zero-Knowledge Proofs**: Prove possession of data without revealing it.

Blockchain Transaction Flow

b. Real-World Example - Bitcoin uses ECDSA (Elliptic Curve Digital Signature Algorithm) to sign transactions. - Zcash uses zero-knowledge proofs to enable private transactions.

c. Summary Table

Application	Cryptographic Tool	Purpose
E-Commerce	TLS, RSA, AES	Secure online transactions
Cloud Storage	AES, KMS	Protect data at rest
Digital Identity	Hashing, Signatures	Authenticate users securely
Blockchain	Hashes, Signatures, ZKPs	Ensure integrity and privacy

1.7 Real-World Challenges in Cryptography

- **Key distribution**: Ensuring secure exchange of keys.
- **Algorithm vulnerabilities**: Side-channel attacks, quantum threats.
- **Human error**: Poor password policies, leaked credentials.
- **Legal vs. privacy**: State surveillance vs. individual rights.

1.8 Critical Reflection Essay Guide

Prompt: *Why does the digital world need cryptography?*
Objectives: - Connect cryptographic goals to real-world needs. - Consider legal, ethical, and privacy implications. - Reflect on personal or societal examples of data compromise.

Suggested Structure:

1. Introduction: Context-setting (importance of digital security).
2. Main Body:
    - Security goals applied to personal data.
    - Cryptography as public infrastructure.
    - Ethical dilemmas (e.g., backdoors, surveillance).
3. Conclusion: Personal stance and future outlook.

1.9 Quick Quiz Questions

Multiple Choice:

1. Which goal does cryptography achieve by ensuring a message has not been altered?
    - a. Confidentiality
    - b. Integrity
    - c. Authentication
    - d. Repudiation

Short Answer:

- Differentiate symmetric and asymmetric encryption with examples.
- Explain how RSA supports secure digital signatures.

True/False:

- AES is an asymmetric cipher.

- Diffie-Hellman allows secure key exchange over insecure channels.

1.10 Practical Lab Activities

- **Build a Caesar Cipher tool**: Encrypt/decrypt text using substitution.
- **Implement AES Encryption using Python**: `PyCryptodome` or `cryptography` library.
- **Protocol Analysis with Wireshark**: Examine TLS handshakes.
- **Create digital signature demo**: Sign and verify using RSA keys.