

Introduction to Cryptography

Lecture 10: User Authentication

CYB 650

Introduction to Cryptography

Presented by
Dr. Mohammad Alkhatib

Module 3

Topics to be covered

- User Authentication Methods
- Password Authentication & Vulnerabilities
- Countermeasures
- Password Cracking or Guessing
- Password Selection Techniques
- Token-Based Authentication
- Biometric Authentication
- Remote User Authentication

• Importance

- Fundamental building block and primary line of defense
- Basis for access control and user accountability

• Authentication Process

- Identification step
 - Presenting an identifier to the security system
- Verification step
 - Presenting or generating authentication information that corroborates the binding between the entity and the identifier

The four means of authenticating user identity are based on

something the individual knows

- password, PIN, answers to prearranged questions

something the individual possesses (token)

- smartcard, electronic keycard, physical key

something the individual is (static biometrics)

- fingerprint, retina, face

something the individual does (dynamic biometrics)

- voice pattern, handwriting, typing rhythm

Widely used line of defense against intruders:

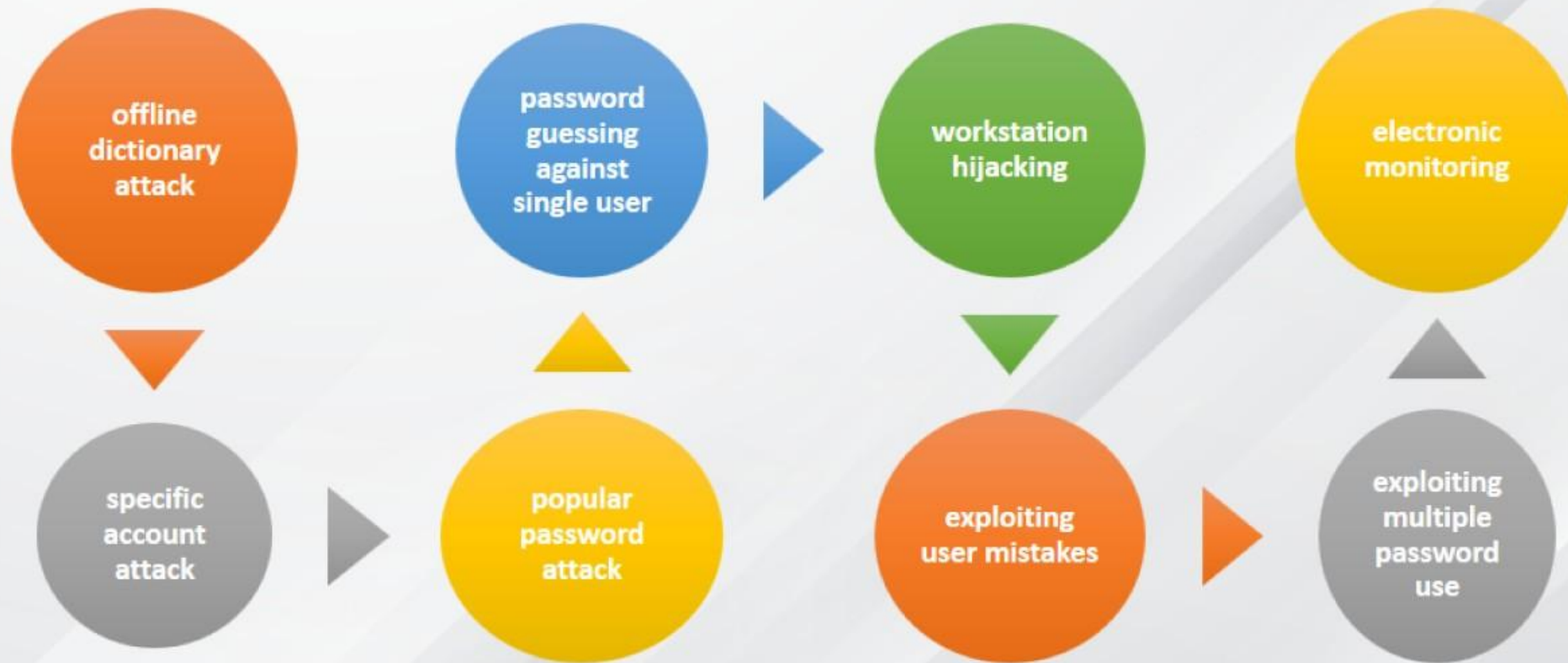
- User provides name/login and password
- System compares password with the one stored for that specified login

The user ID:

- Determines that the user is authorized to access the system
- Determines the user's privileges
- Used in discretionary access control

Module 3

Password Vulnerabilities

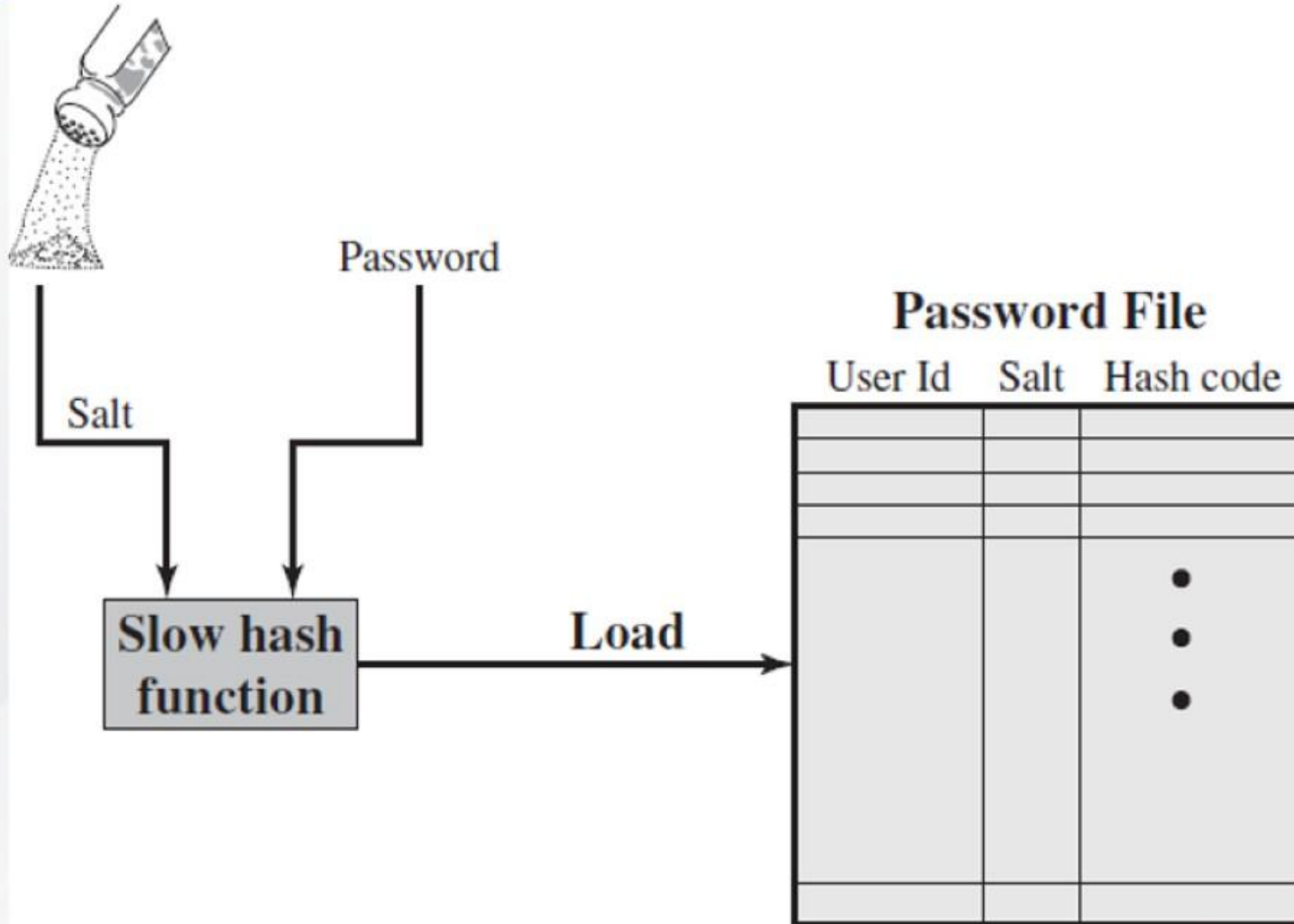


Offline dictionary attack: Typically, strong access controls are used to protect the system's password file. However, experience shows that determined hackers can frequently bypass such controls and gain access to the file. The attacker obtains the system password file and compares the password hashes against hashes of commonly used passwords. If a match is found, the attacker can gain access by that ID/password combination.

- **Specific account attack:** The attacker targets a specific account and submits password guesses until the correct password is discovered.
- **Popular password attack:** A variation of the preceding attack is to use a popular password and try it against a wide range of user IDs. A user's tendency is to choose a password that is easily remembered; this unfortunately makes the password easy to guess.
- **Password guessing against single user:** The attacker attempts to gain knowledge about the account holder and system password policies and uses that knowledge to guess the password.

- **Workstation hijacking;** The attacker waits until a logged-in workstation is unattended.
- **Exploiting user mistakes:** If the system assigns a password, then the user is more likely to write it down because it is difficult to remember. This situation creates the potential for an adversary to read the written password. A user may intentionally share a password, to enable a colleague to share files, for example.
- **Exploiting multiple password use.** Attacks can also become much more effective or damaging if different network devices share the same or a similar password for a given user.
- **Electronic monitoring:** If a password is communicated across a network to log on to a remote system, it is vulnerable to eavesdropping. Simple encryption will not fix this problem, because the encrypted password is, in effect, the password and can be observed and reused by an adversary

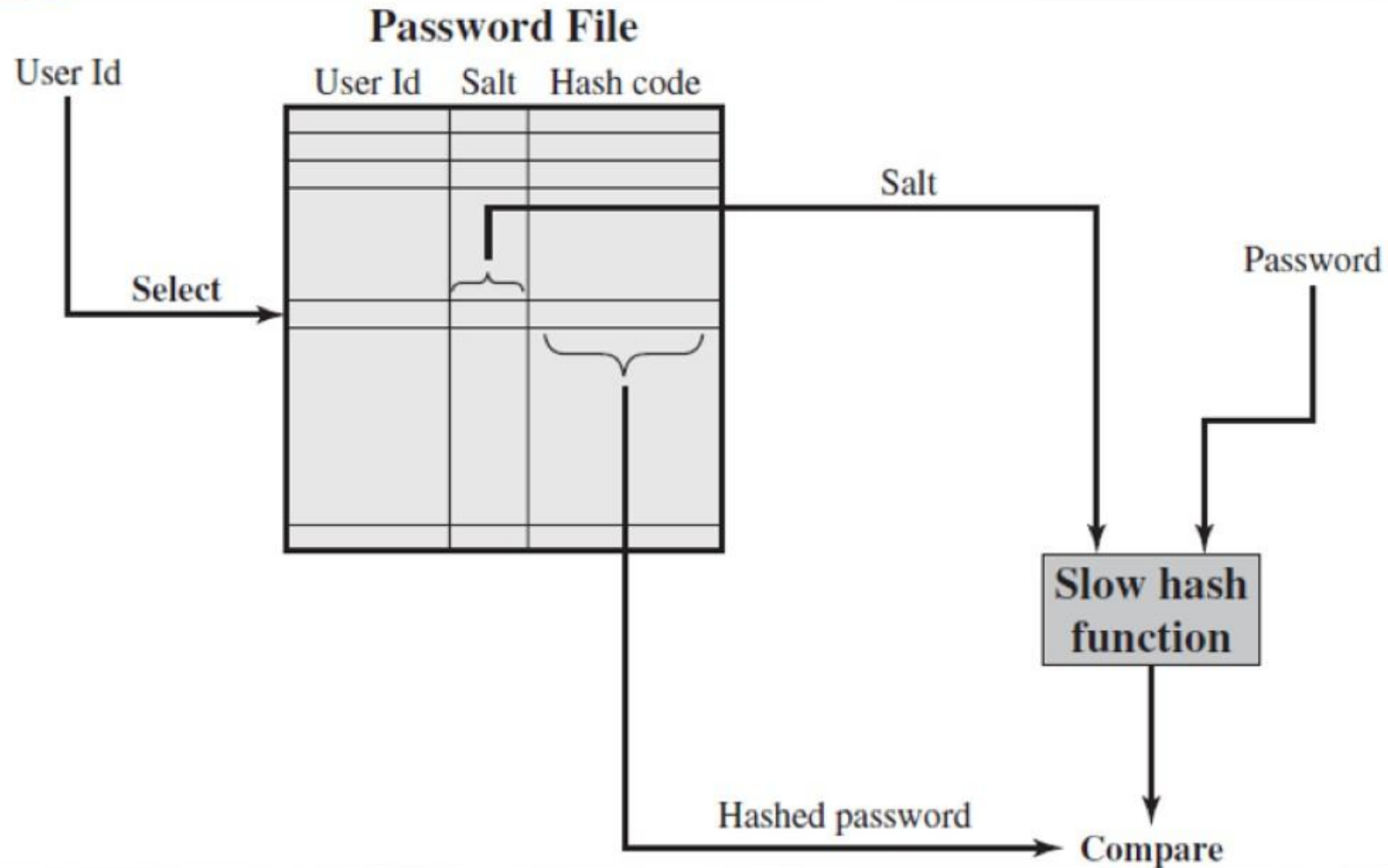
- Controls to prevent unauthorized access to password file
- Intrusion detection measures
- Rapid reissuance of compromised passwords
- Account lockout mechanisms
- Policies to inhibit users from selecting common passwords
- Training in and enforcement of password policies
- Automatic workstation logout
- Policies against similar passwords on network devices



Module 3

Use of Hashed Passwords

Verifying a password



Use of Hashed Passwords

- ❑ Password is combined with a fixed-length **salt value**
- ❑ Password and salt value serve as inputs to a hashing algorithm to produce a **fixed-length hash code**
- ❑ **Hashed password** is then stored, together with a plaintext copy of the salt
- ❑ Why the salt value?
 - ❑ Prevents duplicate passwords from being visible
 - ❑ Increases the difficulty of offline dictionary attacks
 - ❑ Nearly impossible to find out the use of same password on many systems

❑ Dictionary attacks

- ❑ Develop a large dictionary of possible passwords and try each against the password file
- ❑ Each password must be hashed using each salt value and then compared to stored hash values

❑ Rainbow table attacks

- ❑ Pre-compute tables of hash values for all salts
- ❑ A mammoth table of hash values
- ❑ Can be countered by using a sufficiently large salt value and a sufficiently large hash length

- ❑ It can block offline guessing attacks by denying access to encrypted passwords
 - ❑ Make available only to privileged users
 - ❑ Shadow password file
 - ❑ A separate file from the user IDs where the hashed passwords are kept
- ❑ Vulnerabilities
 - ❑ Weakness in the OS that allows access to the file
 - ❑ Accident with permissions making it readable
 - ❑ Users with same password on other systems
 - ❑ Access from backup media
 - ❑ Sniff passwords in network traffic

☐ Short passwords

- ☐ A study on 54 machines reveals that almost **3%** of the passwords were three characters or fewer in length

☐ Guessable passwords

- ☐ A study shows that after collecting 14000 encrypted passwords, nearly one-fourth of the passwords were guessed

Module 3

Password Selection Techniques

User education

- Importance of using hard to guess passwords
- Provide guidelines for selecting strong passwords

Computer generated passwords

- users have trouble remembering them

Reactive password checking

- system periodically runs its own password cracker to find guessable passwords

Proactive password checking

- Eliminate guessable passwords while allowing the user to select a password that is memorable
- System checks if the password is allowable, and if not, rejects it

- ❑ Rule enforcement
 - ❑ Specific rules that passwords must compatible to use
- ❑ Password cracker
 - ❑ Compile a large dictionary of passwords not to use (Two problems: space and time)

Token-Based Authentication

- Objects that a user possesses for the purpose of user authentication are called **tokens**

Card Type	Defining Feature	Example
Embossed	Raised characters only, on front	Old credit card
Magnetic stripe	Magnetic bar on back, characters on front	Bank card
Memory	Electronic memory inside	Prepaid phone card
Smart Contact Contactless	Electronic memory and processor inside Electrical contacts exposed on surface Radio antenna embedded inside	Biometric ID card

Memory Cards

- Store but do not process data
- Most common is the magnetic stripe card
- Include an internal electronic memory
- Used alone for physical access
- Provides significantly greater security when combined with a password or PIN
- Drawbacks:
 - Requires a special reader
 - Loss of token
 - User dissatisfaction



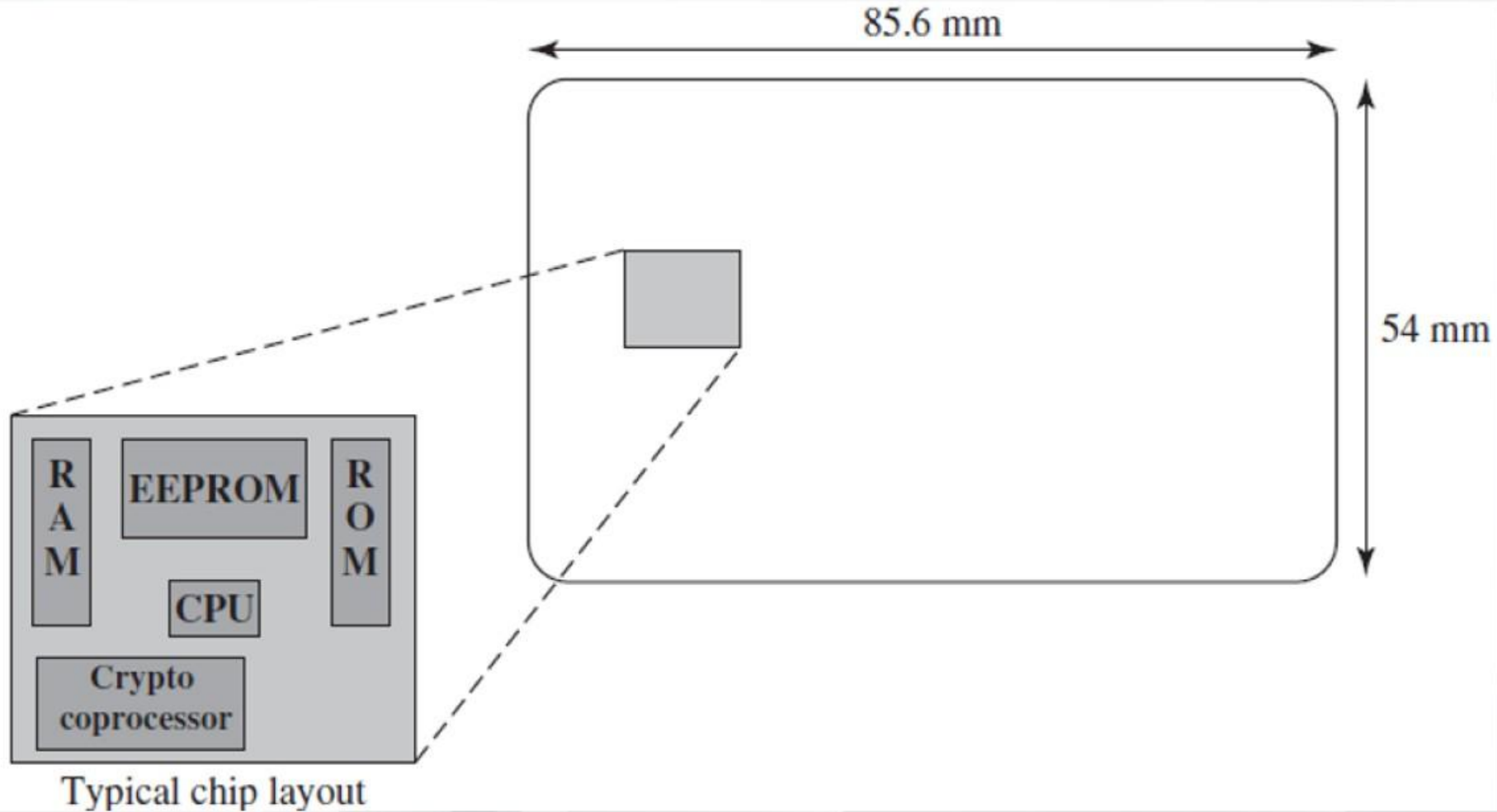
- Physical characteristics:
 - Include an embedded microprocessor
 - A smart token that looks like a bank card
 - Can look like calculators, keys, small portable objects
- Interface:
 - Manual interfaces include a keypad and display for interaction
 - Electronic interfaces communicate with a compatible reader/writer



- Contain:
 - An entire microprocessor
 - Processor
 - Memory
 - I/O ports
- Typically include three types of memory:
 - Read-only memory (ROM)
 - Stores data that does not change during the card's life
 - Electrically erasable programmable ROM (EEPROM)
 - Holds application data and programs
 - Random access memory (RAM)
 - Holds temporary data generated when applications are executed

Module 3

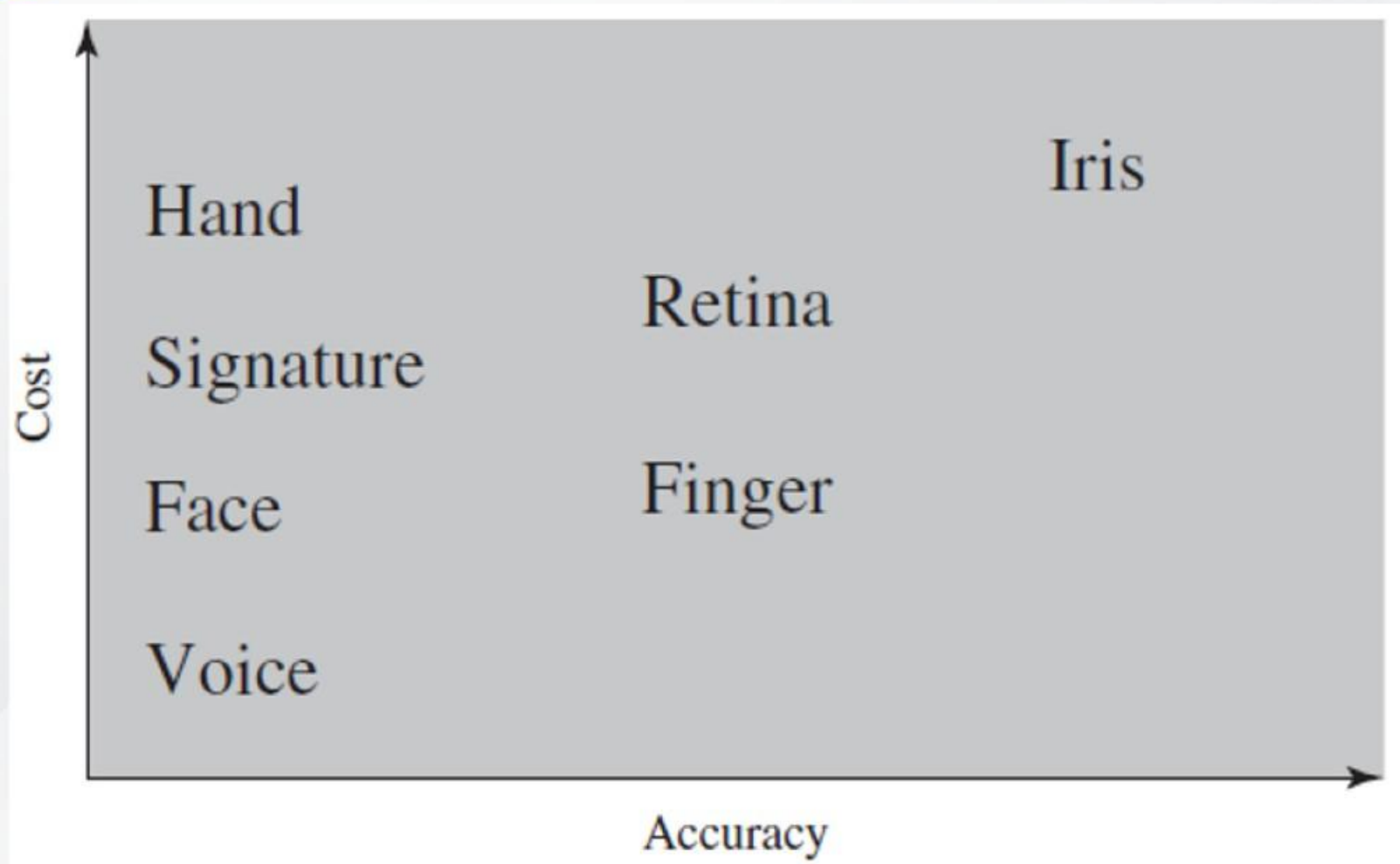
Smart Cards



- Attempts to authenticate an individual based on unique physical characteristics
- Based on pattern recognition
- Is technically complex and expensive when compared to passwords and tokens
- Physical characteristics used include:
 - Facial characteristics
 - Fingerprints
 - Hand geometry
 - Retinal pattern
 - Iris
 - Signature
 - Voice

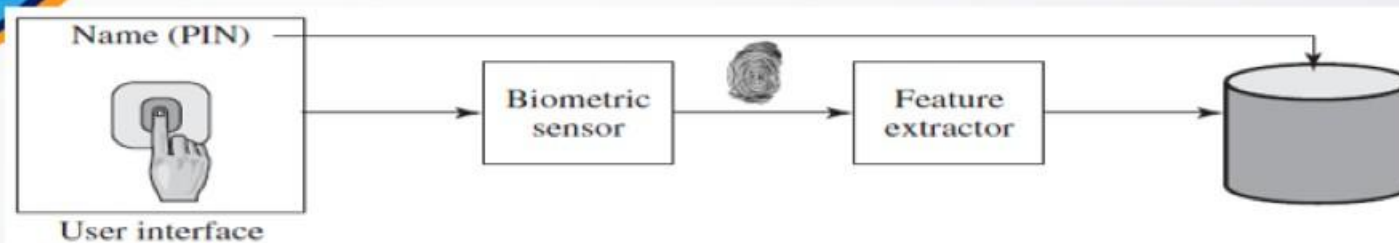
Module 3

Cost and Accuracy

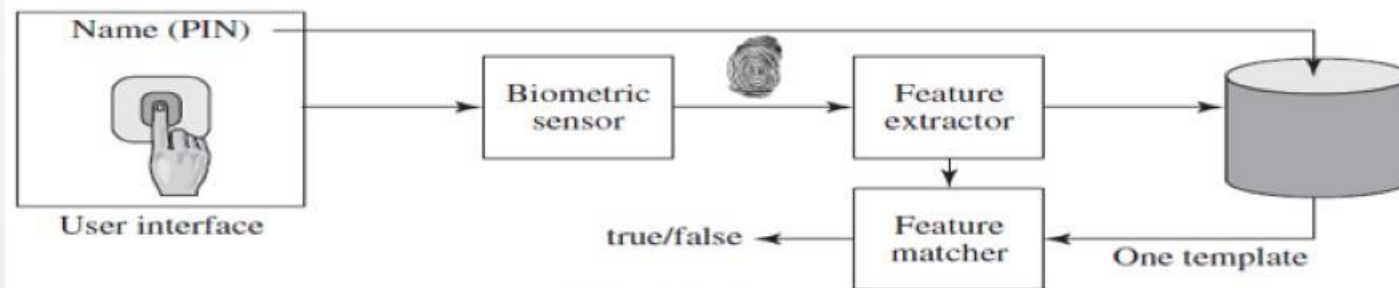


Module 3

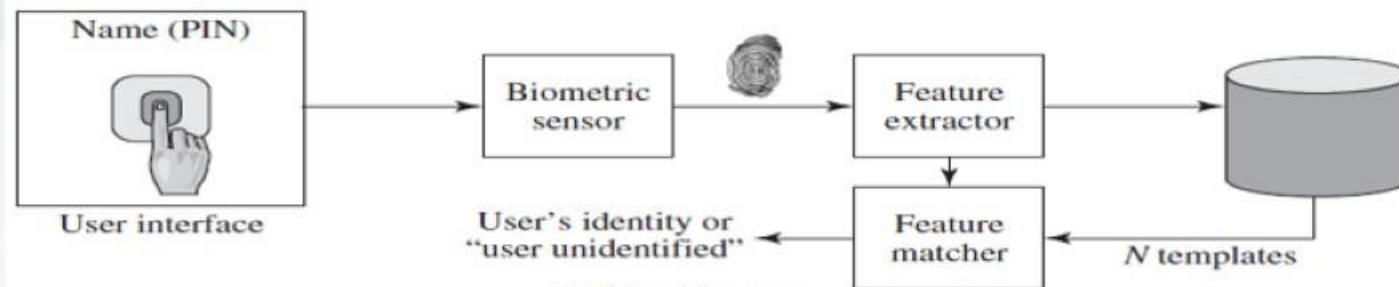
Operations



(a) Enrollment



(b) Verification



(c) Identification

- Authentication over a network, the Internet, or a communications link is more complex
 - Additional security threats such as:
 - Eavesdropping, capturing a password, replaying an authentication sequence that has been observed
- Generally rely on some form of a challenge-response protocol to counter threats



Introduction to Blockchain

- Blockchain is a decentralized, distributed digital ledger that records transactions securely across a network of computers. Each record (block) is linked using cryptography, making it nearly impossible to alter past data.
- Key principles: Decentralization, Transparency, Security, and Immutability.



How Blockchain Works

- 1. A transaction is initiated.
- 2. The transaction is verified by multiple nodes using consensus algorithms.
- 3. The verified transaction is grouped into a block.
- 4. The block is added to the chain using cryptographic hashes.
- 5. The ledger updates across all network participants.

Core Features of Blockchain

- Decentralization: No central authority; distributed control.
- Transparency: Transactions are visible to authorized participants.
- Immutability: Once added, data cannot be changed.
- Security: Cryptographic techniques ensure data integrity.

Role of Cryptography in Blockchain

- Cryptography provides the foundation for blockchain security.
- Hashing (e.g., SHA-256): Generates unique digital fingerprints for data.
- Public and Private Keys: Enable secure identity verification.
- Digital Signatures: Authenticate transactions and prevent forgery.
- Together, these mechanisms protect data integrity and ensure trustless transactions.

Types of Blockchain

- 1. Public Blockchain: Open to all (e.g., Bitcoin, Ethereum).
- 2. Private Blockchain: Access restricted to specific users (e.g., Hyperledger Fabric).
- 3. Consortium Blockchain: Controlled by a group of organizations.
- 4. Hybrid Blockchain: Combines public transparency with private control.

Applications of Blockchain – Overview

- Blockchain technology is transforming multiple industries by offering trust, efficiency, and transparency.
- Key sectors:
 - Finance & Banking
 - Supply Chain & Logistics
 - Healthcare
 - Voting Systems
 - Real Estate
 - Education
 - Government & Public Services



Thank You