Authentication Mechanisms

Dr Zia ur Rehman

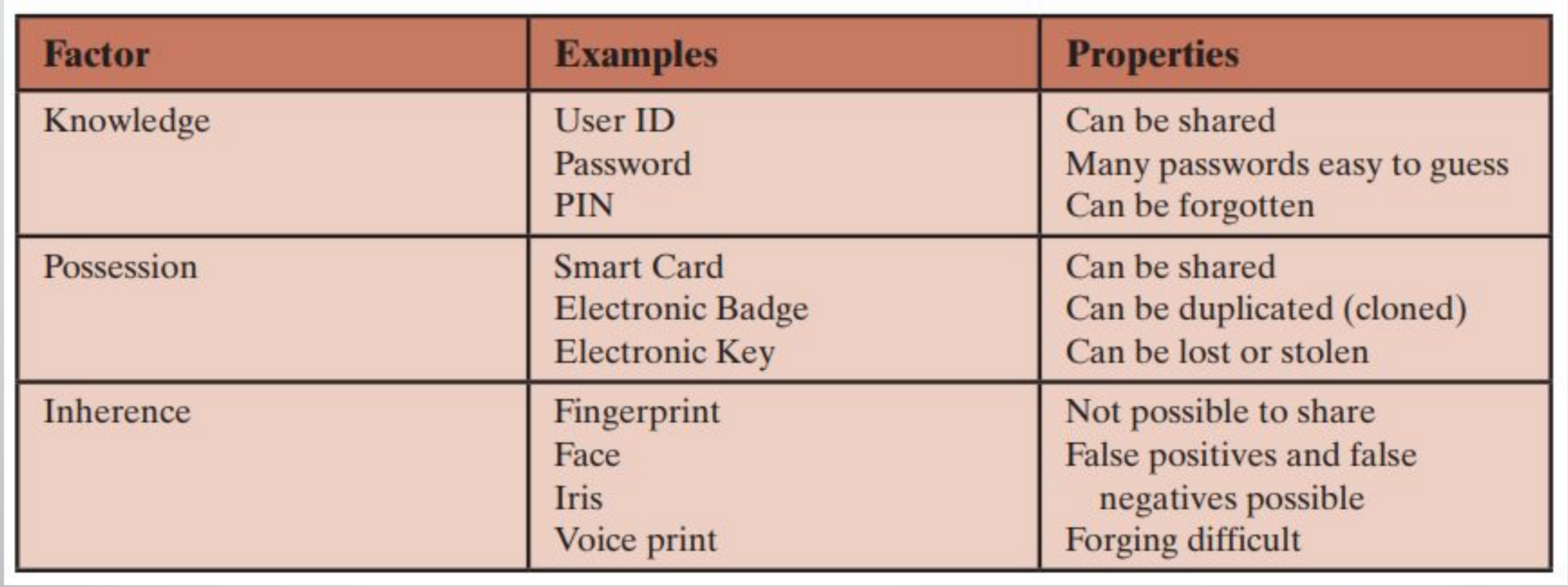
Authentication

- Authentication is the process of verifying the identity of a user or system.
- An authentication mechanism is a method or process used to confirm the identity of a user or system, and it ensures that only authorized individuals or systems have access to specific resources.
- In other words, it is process of determining whether some user or some application or process acting on behalf of a user is, in fact, who or what it declares itself to be.

Authentication Mechanisms / Means

- There are three general means, or authentication factors, of authenticating a user’s identity, which can be used alone or in combination.
- **Knowledge factor (something the individual knows):** knowledge factors can come in the form of passwords, passphrases, personal identification numbers (PINs), or answers to secret questions.
- **Possession factor (something the individual possesses):** Physical entity possessed by the authorized user to connect to the client computer or portal, referred as token.
- **Inherence factor (something the individual is or does):** Refers to characteristics, called biometrics, that are unique or almost unique to the individual.

Authentication Factors

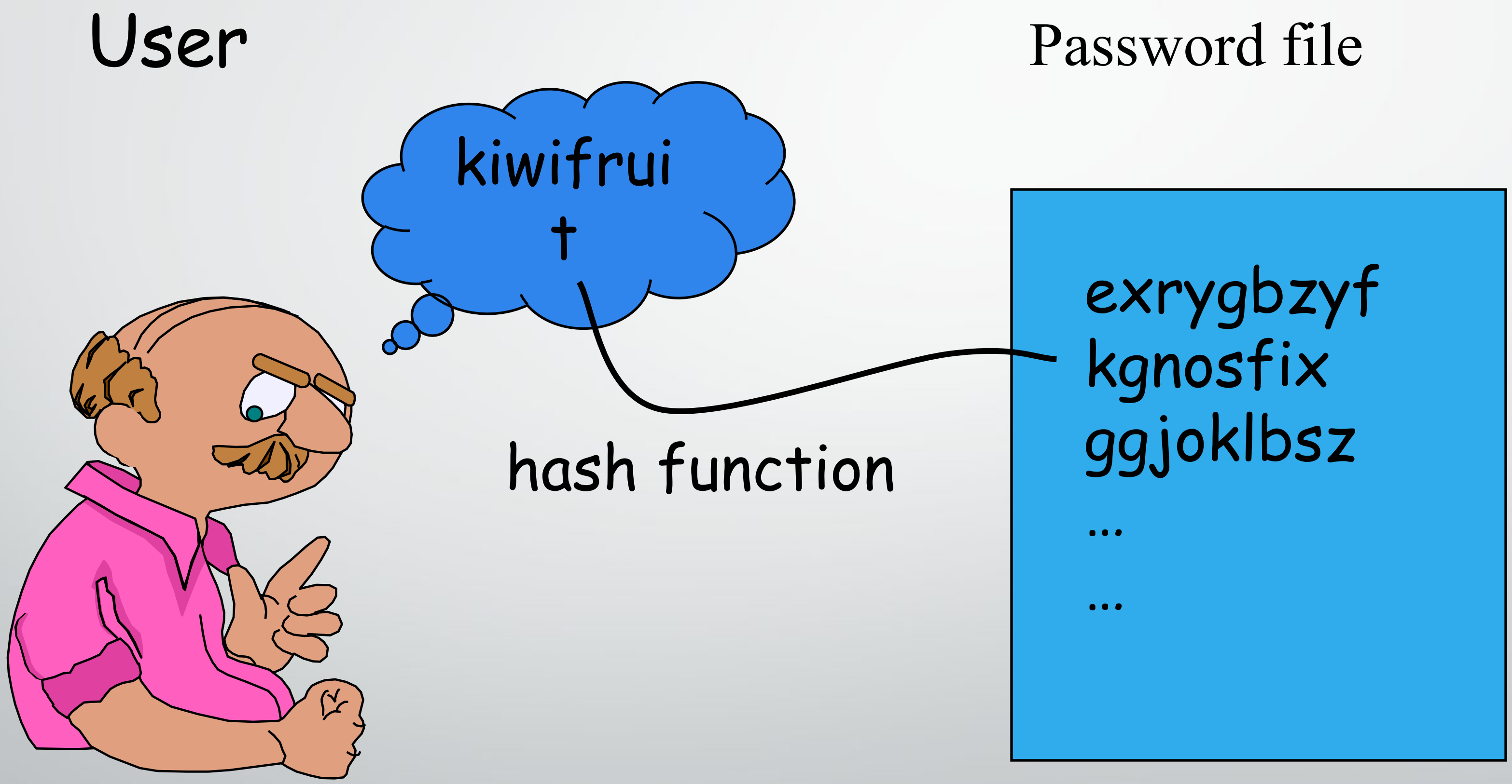


Factor	Examples	Properties
Knowledge	User ID Password PIN	Can be shared Many passwords easy to guess Can be forgotten
Possession	Smart Card Electronic Badge Electronic Key	Can be shared Can be duplicated (cloned) Can be lost or stolen
Inherence	Fingerprint Face Iris Voice print	Not possible to share False positives and false negatives possible Forging difficult

Password Authentication

- Basic idea
  - User has a secret password
  - System checks password to authenticate user
- Issues
  - How is password stored?
  - How does system check password?
  - How easy is it to guess a password?
    - Difficult to keep password file secret, so best if it is hard to guess password even if you have the password file

Basic password scheme



User

Password file

kiwifruit

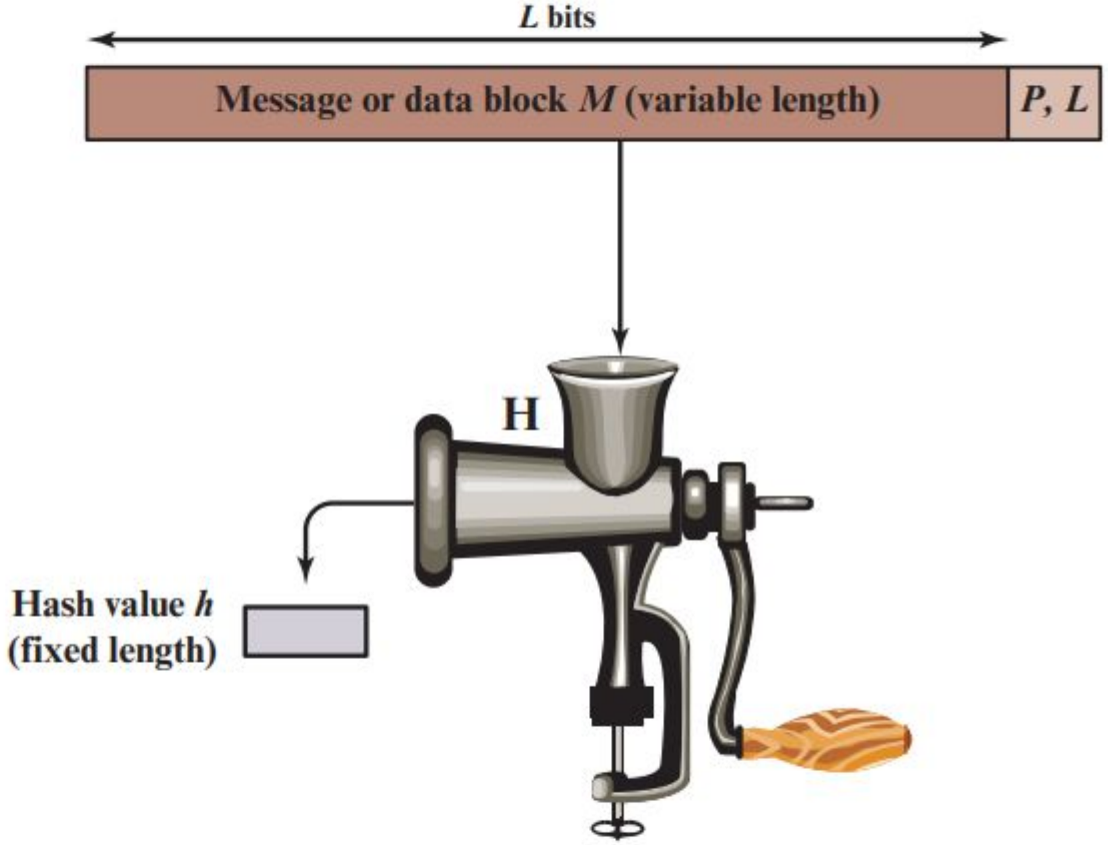
hash function

exrygbzyf
kgnosfix
ggjoklbsz
…
…

Basic Password Scheme

- Hash function h : strings → strings
  - Given h(password), hard to find password
  - No known algorithm better than trial and error
- User password stored as h(password)
- When user enters password
  - System computes h(password)
  - Compares with entry in password file
- No passwords stored on disk

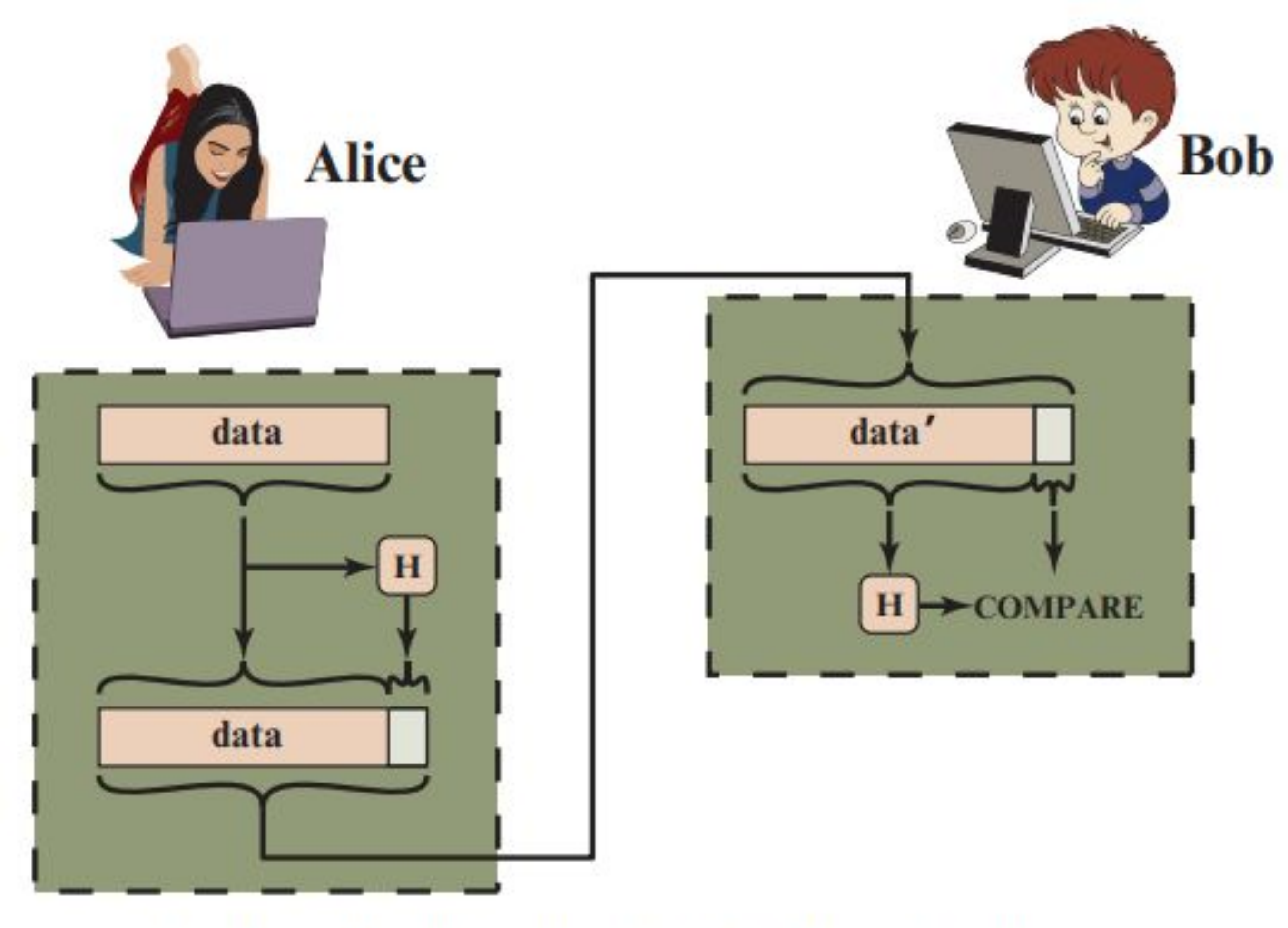
Cryptographic Hash Function



P, *L* = padding plus length field

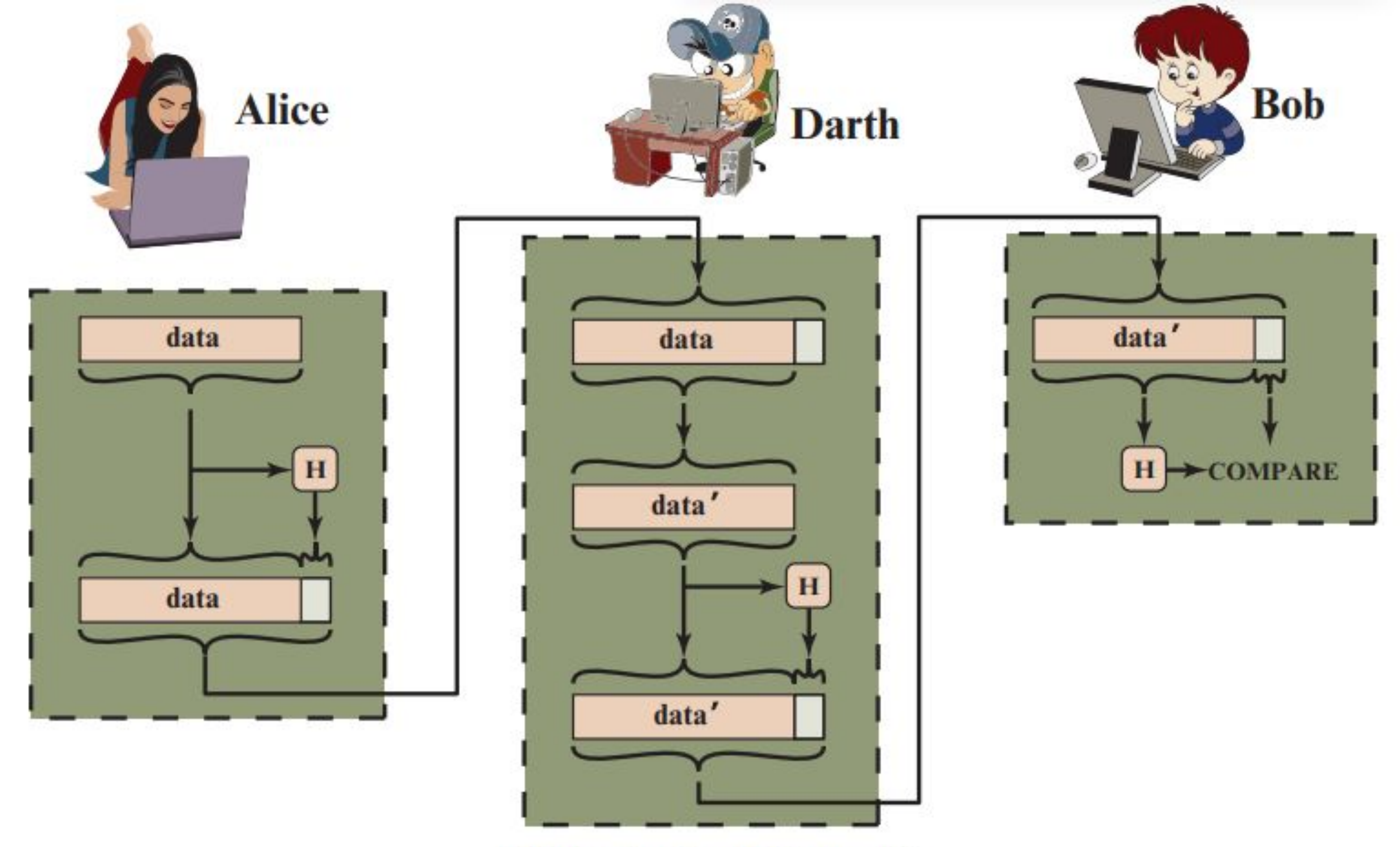
Figure 11.1 Cryptographic Hash Function; $h = \mathrm{H}(M)$

Message Authentication



(a) Use of hash function to check data integrity

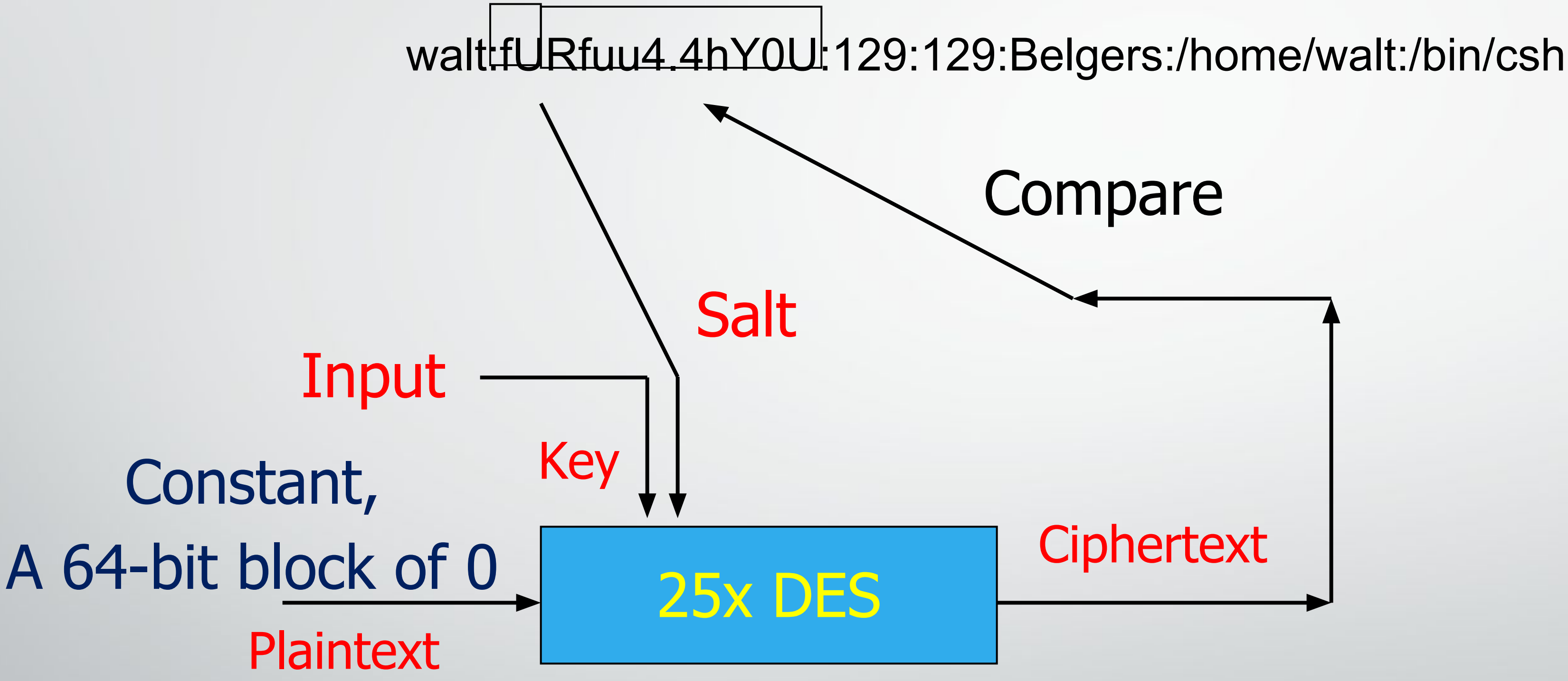
Message Authentication



(b) Man-in-the-middle attack

Salt

- Password line



walt:fURfuu4.4hY0U:129:129:Belgers:/home/walt:/bin/csh

When password is set, salt is chosen randomly
12-bit salt slows dictionary attack by factor of 2^{12}

Dictionary Attack – some numbers

- Typical password dictionary
  - 1,000,000 entries of common passwords
    - people's names, common pet names, and ordinary words.
  - Suppose you generate and analyze 10 guesses per second
    - This may be reasonable for a web site; offline is *much* faster
  - Dictionary attack in at most 100,000 seconds = 28 hours, or 14 hours on average
- If passwords were random
  - Assume six-character password
    - Upper- and lowercase letters, digits, 32 punctuation characters
    - 689,869,781,056 password combinations.
    - Exhaustive search requires 1,093 years on average

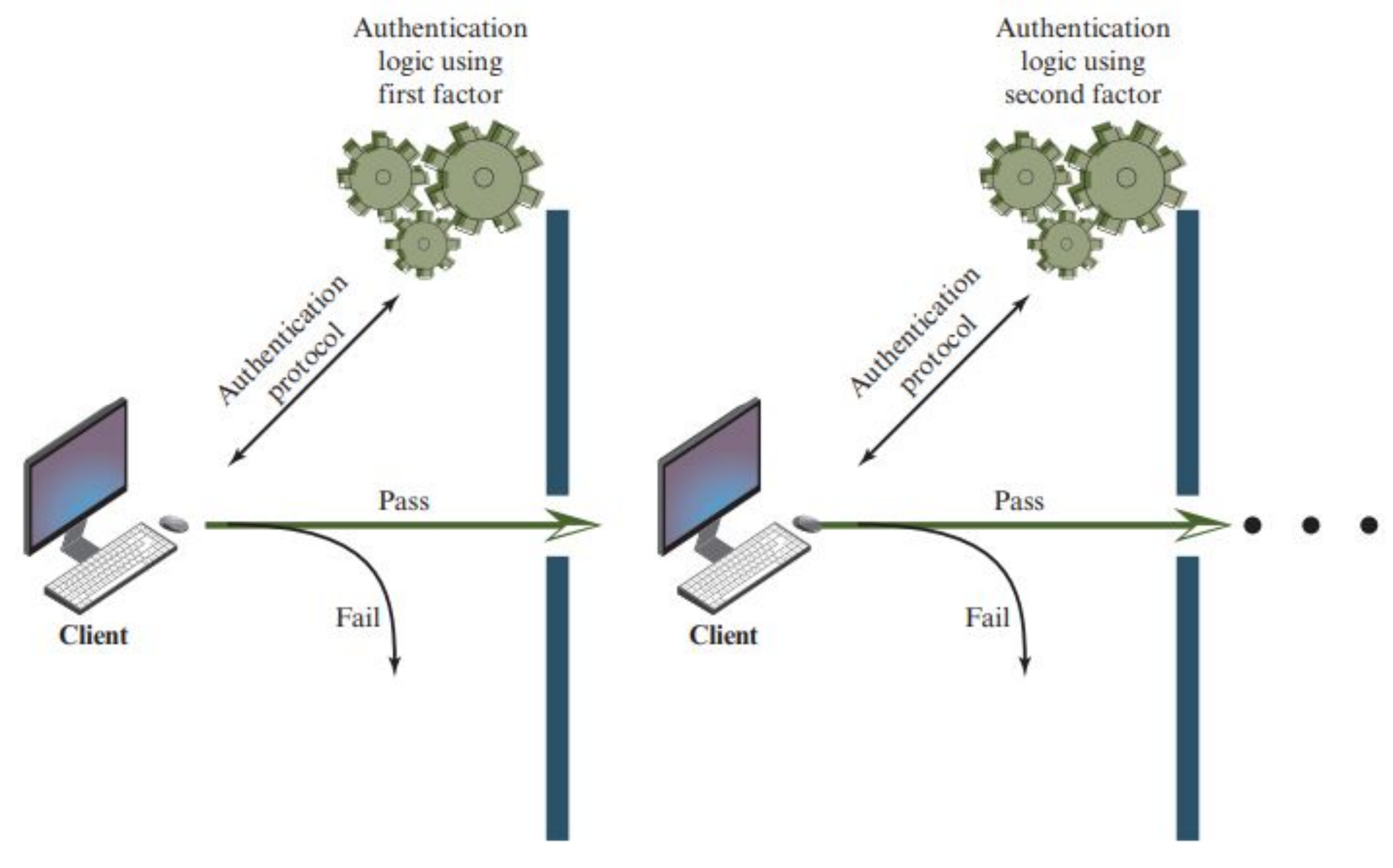
Biometrics

- Use a person’s physical characteristics
  - fingerprint, voice, face, keyboard timing, …
- Advantages
  - Cannot be disclosed, lost, forgotten
- Disadvantages
  - Cost, installation, maintenance
  - Reliability of comparison algorithms
    - False positive: Allow access to unauthorized person
    - False negative: Disallow access to authorized person
  - Privacy?
  - If forged, how do you revoke?

Biometrics

- Common uses
  - Specialized situations, physical security
  - Combine
    - Multiple biometrics
    - Biometric and PIN
    - Biometric and token

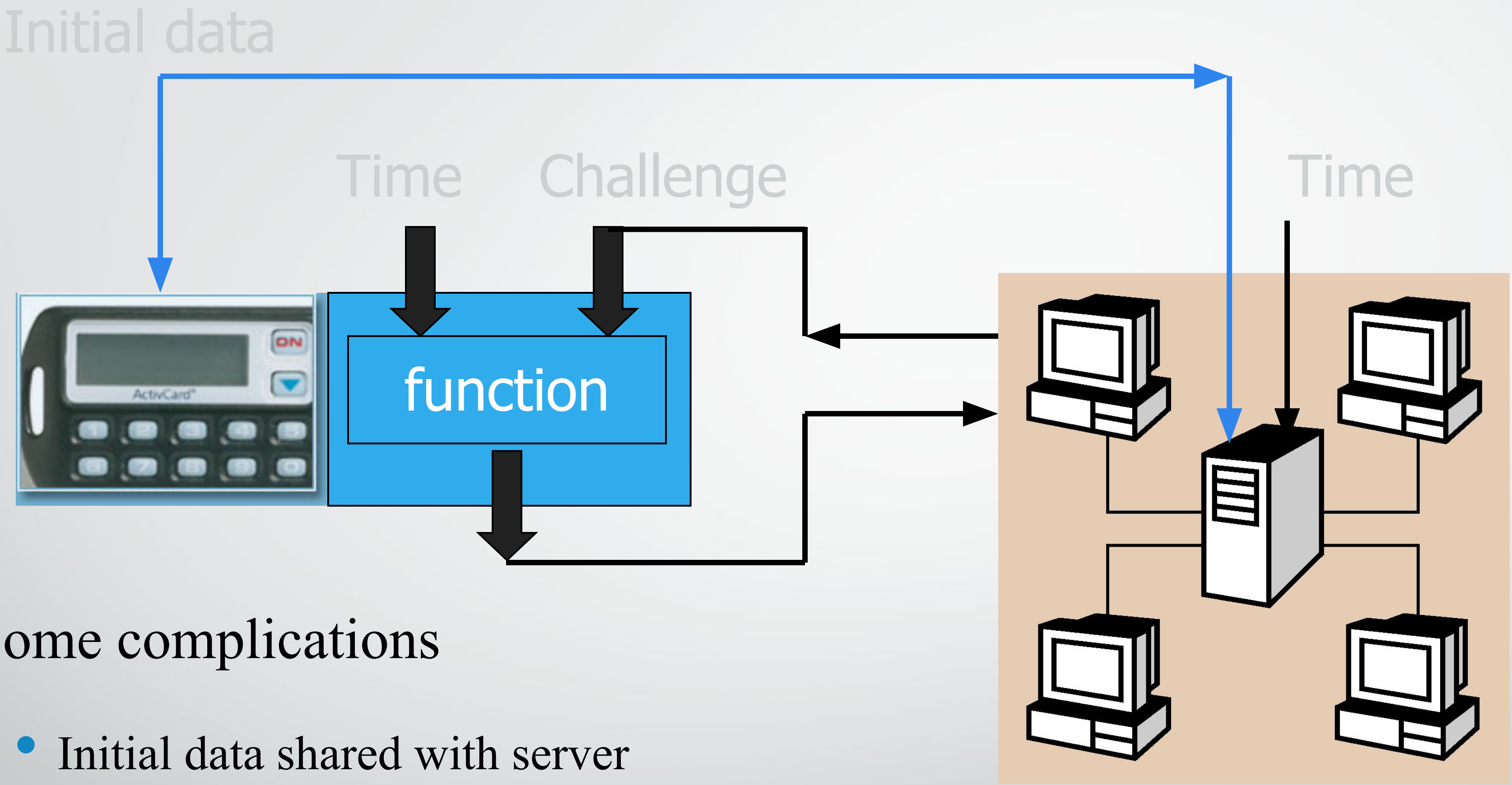
Multifactor Authentication


Authentication logic using first factor
Authentication logic using second factor
Authentication protocol
Authentication protocol
Pass
Pass
Fail
Fail
Client
Client

Token-based Authentication
Smart Card

- With embedded CPU and memory
  - Carries conversation w/ a small card reader
- Various forms
  - PIN protected memory card
    - Enter PIN to get the password
  - Cryptographic challenge/response cards
    - A cryptographic key in memory
    - Computer create a random challenge
    - Enter PIN to encrypt/decrypt the challenge w/ the card
  - Cryptographic Calculator (readerless smart card)
    - Simulating a smartcard: user enter the encrypted result

Smart Card Example



- Some complications
  - Initial data shared with server
    - Need to set this up securely
    - Shared database for many sites
  - Clock skew