Computer and Network Security/ Cyber Physical Systems Security

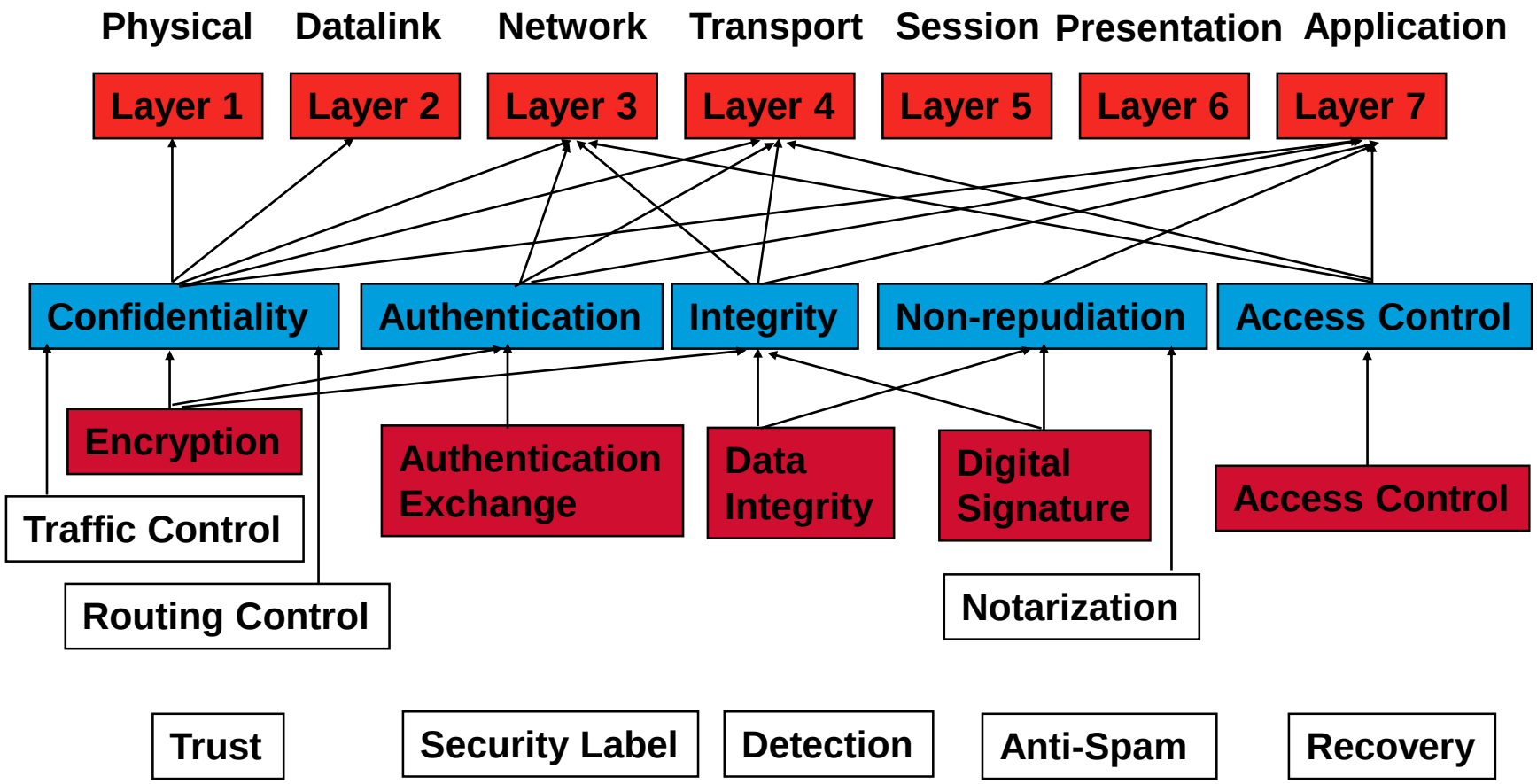
Dr. Chan Yeob Yeun

Week 3 - 4

Weekly Lecture Plan

Wk	Contents	Cmt	Wk	Contents	Cmt
1	**Introduction**		9	Foundations of Network Security II	
2	Foundations of Computer Security	Tutorial Assig Plan	10	Network-Based Threats and Attacks	
3	Identification and Authentication I		11	Network Security Protocols I	
4	Identification and Authentication II	Quiz 1	12	Network Security Protocols II	Quiz 3
5	Access Control		13	Firewalls	
6	Modern Computer Attacks		14	IDS / IPS	Assig Submit
7	Malicious Code	Assig Confirm	15	Revision and Presentation	
8	Foundations of Network Security I	Quiz 2	16	Exam	

What is Computer and Network Security ?



Authorisation and access control

·Authorisation is a term relating to the notion of access control.

·Any system will often need to make a decision about whether another entity should be allowed to perform a particular action.

·This is normally referred to as access control.

Access Controls

- The 1st difficulty an attacker meets is how to access the target system.

- You can limit and prevent access in many ways (e.g. using physical security) but you must allow some access for the system to run successfully.

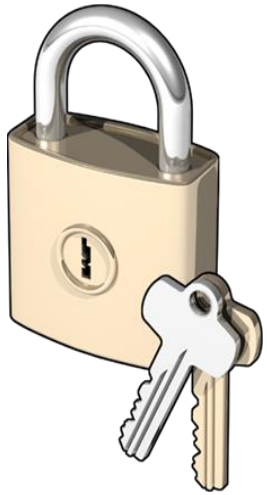
- Access can be controlled using two mechanisms:

1- Authentication:

Proofing the user identity (before accessing the system)

2- Authorisation:

Limiting the user ability (after accessing the system)

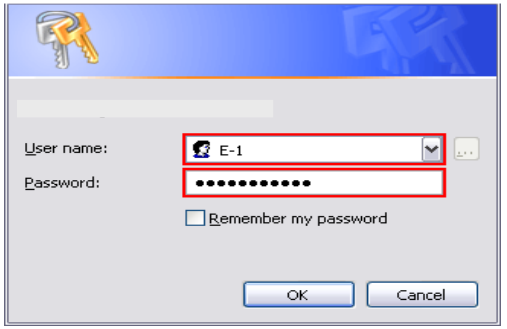


Authentication

- The process by which people prove they are who they say they are.
- A good authentication system is the one which applies more than one technique to proof the user identity.
- e.g. A two-factor authentication system uses at least two of the following:
  - Something you know (e.g. password/PIN)
  - Something you have (e.g. smart card/token)
  - Something you are (e.g. fingerprint/iris)

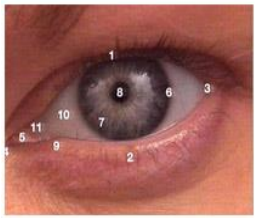
Authentication Controls

- Usernames and Passwords



- Certificates and Tokens

- Biometrics



Usernames and Passwords

The most common form of user authentication

Main Issues:

- Clear-text passwords (as in telnet, FTP, rlogin)
- Weak passwords
- Password file protection
- Password management (length, change, content)
- Failed login attempts controls
- Concurrent sessions
- Inactive/unused accounts

Password Cracking

- Discovering a plain text password given an encrypted password
- Usually done using an automated tool
- Main steps:

  Choose a ***guess word*** randomly from a dictionary file

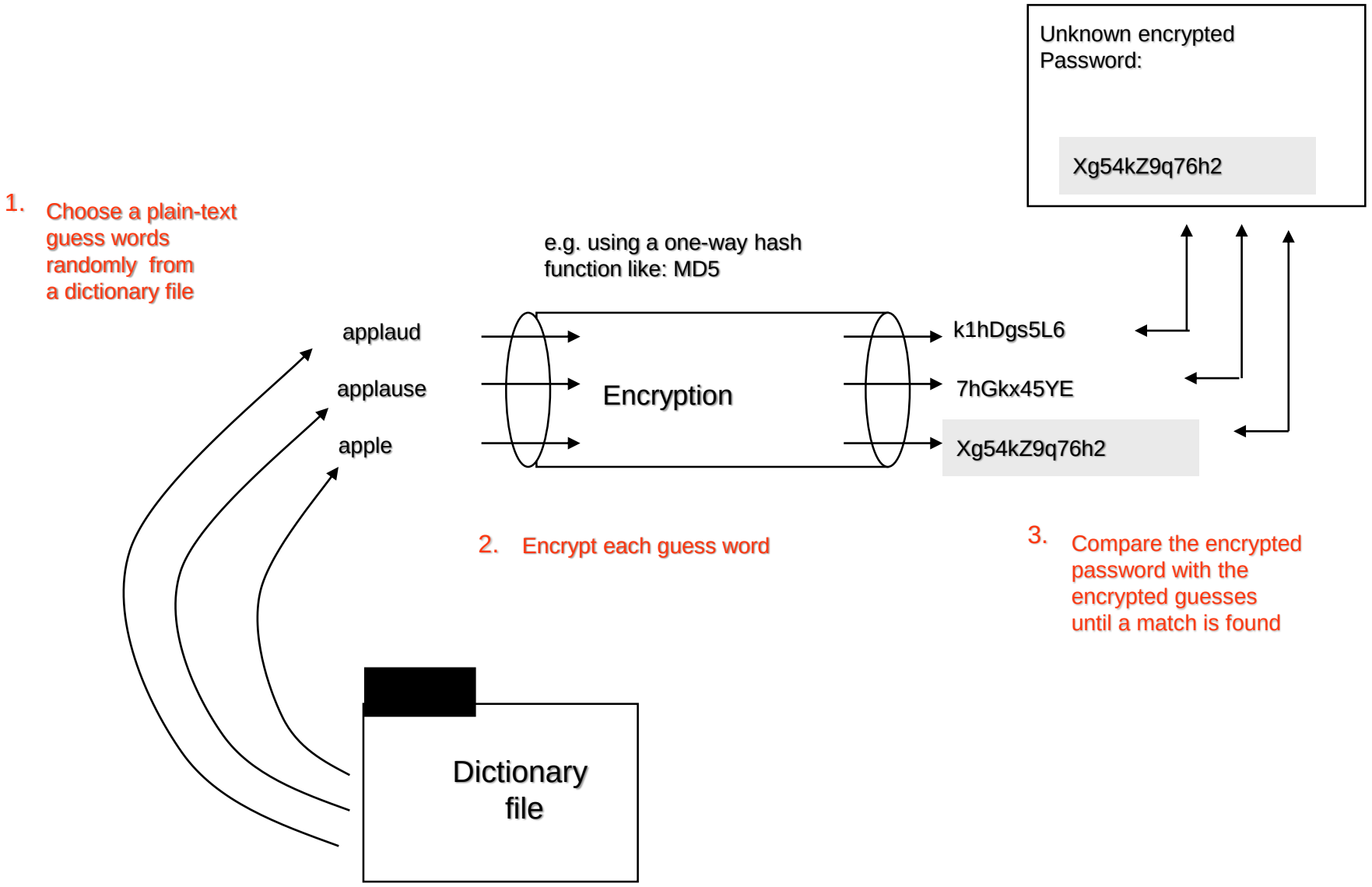
  Encrypt the ***guess word*** using the same encryption algorithm used to encrypt the unknown ***password***

  Compare the unknown encrypted ***password*** with the encrypted ***guess word***, if they match then the plain text ***guess word*** is the ***password***

  - Repeat the above steps using more guess words

- Main Types:
  - Dictionary Attack: Uses a large list of dictionary words to crack passwords.
  - Hybrid Attack: Adds numerals and symbols to dictionary words.
  - Brute-Force Attack: Tries all possibilities exhaustively.

How Password Cracking is done?



One-Way Hash Function

- Irreversible function that takes a variable-length input string and outputs a fixed-length smaller string.
- Uses a cryptographic checksum to ensure integrity of associated data
- Applications: authentication, integrity checking
- Examples:
    - Message Digest (MD5)
    - Secure Hash Algorithm (SHA)
    - Blowfish

Password Cracking Tools

Many free tools are available.

•These tools are good for administrators and web masters but they can be misused by attackers.

•By default HTTP will not terminate the session after a number of failed login attempts (which leaves it open for automated tools).

•So other measures need to be taken, for example:

- Number of failed login attempts
- CAPTCHA: Completely Automated Public Turing test to tell Computers and Humans Apart

Stopping Password Cracking

- Users should choose strong non-dictionary passwords.
- Users should change their passwords regularly.
- Accounts should be locked after a reasonable number of failed login attempts.

BUT BE AWARE:

- This might enable someone to lock valid accounts on your system i.e a Denial Of Service (DOS) to your users.

Some Password Controls

- Password length
- Password complexity
- Password history (reuse)
- Maximum password age
- Minimum password age
- Account Expiration (guests, ex-employees, contractors)
- Account Restriction (by time, machine, .etc)
- Account lockout (can lead to a DoS)

Note:
Use both Policy and Filters

Account Harvesting

- Collecting valid user logins, to start guessing their passwords.
- Can be obtained from bad error messages. e.g.: after a failed login attempt the user gets the following:

  `"Incorrect password"` or `"Invalid password"`.

- This is a confirmation that the `username` is valid.
- So one can run a script to collect all valid accounts and start cracking their passwords.

Stopping Account Harvesting

- Make the error message as general as possible, for example, this is a good error message:

  “Invalid user name and / or password”

  OR

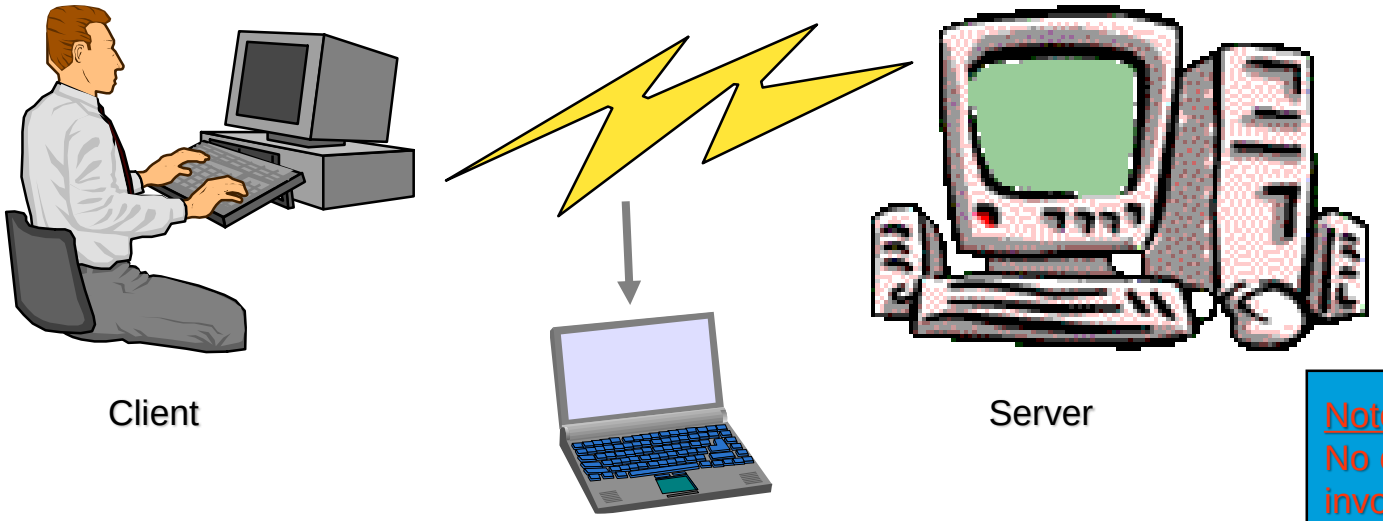
  “Invalid user name and password combination”

- Make sure to maintain this in case of a multilingual web site. One might look for slight variations between error messages of different languages.

Password Authentication Protocol (PAP)

- Most commonly used form of dial-up user authentication
- PAP is a simple but weak authentication mechanism.
- The user enters his password, then it is sent in clear-text to the authentication server for validation.
- Anyone sniffing the network can discover the password easily.
- In some PAP implementations the client hashes (encrypts) the password before it is sent to the server, where it is compared to a stored hash of the password.
- If the hashes are identical, the server grants access, otherwise not.
- This protection doesn't improve PAP security, since an attacker can still sniff the hashed password and send it to the PAP server to authenticate (a replay attack)

Password Authentication Protocol (PAP)



Note:
No challenge is involved in this approach

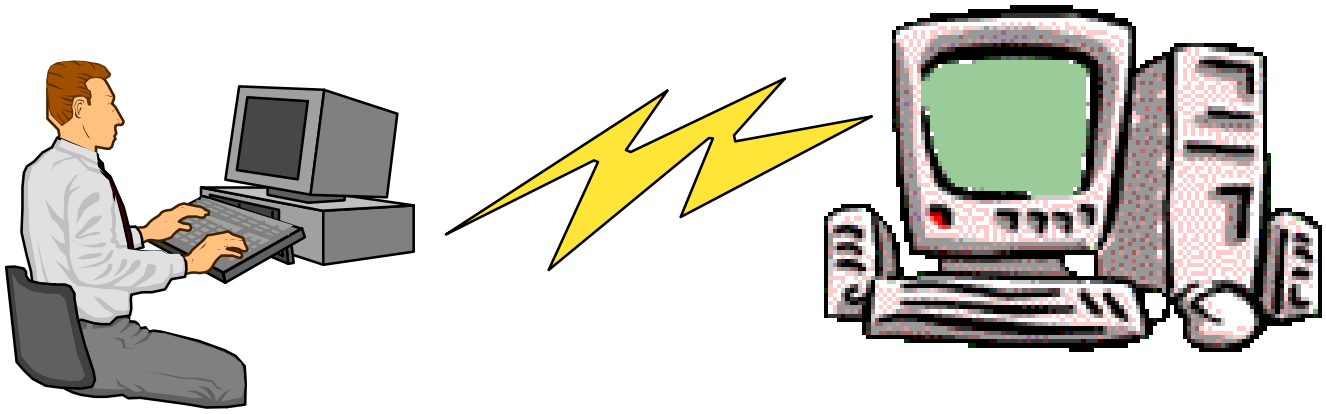
1. User sends username and password either in clear-text (or hashed), this can easily be sniffed by a network attacker

2. Server authenticates the password associated with a given username and grants or denies access accordingly

Challenge Handshake Authentication Protocol (CHAP)

- CHAP can be used instead of PAP wherever possible.
- It guards against password thefts using a challenge/response mechanism, and varies the challenge for every login attempt to protect from replay attacks.
- A typical CHAP session follows these steps.
    - The client initiates communication with the server by sending his user ID.
    - The server sends a challenge back to the client
    - The user enters his password
    - The client uses the challenge and the password to create an encrypted response
    - The client sends the response to the server
    - The server determines what should be the response using the challenge and the locally stored password
    - If the responses are identical the server grants access, otherwise access is denied

Challenge Handshake Authentication Protocol (CHAP)



1. Client sends userID to server

2. Server sends a random challenge to client

3. Client uses the password and the challenge to generate an encrypted response and sends it to the server

4. Server verifies the response by a similar process using the challenge and the stored user password

Advantages of CHAP Vs. PAP

- In CHAP the password, encrypted or not, never traverse the network.
- In CHAP even if an attacker gets the encrypted response he can't use it in a replay attack since the challenge is changing for each user login attempt.
- CHAP however requires that the password is stored in a reversibly encrypted text (i.e possible to decrypt). Which is considered to be less secure than a one-way hash encryption.
- The more advanced versions of CHAP are: MS-CHAPv1 and MS-CHAPv2.
    - MS-CHAPv1 uses irreversible one-way hash.
    - MS-CHAPv2 provides mutual authentication between both client and server.

Single Sign-On (SSO)

- A process by which the user is granted access to many systems after authentication with a single username/password.
- A typical example is the Microsoft Passport.
- SSO systems provide convenience to users. However, compromising one SSO account means gaining access to all other resources using that account.
- Therefore, consider providing separate accounts to protect more sensitive data and resources.

One-Time Password

- Each password is used only once.
- This prevents reuse of the password if it is intercepted by an attacker.
- Two current implementations of one-time password systems are:
  - RSA SecurID:
    - H/W or S/W based authenticators (tokens)
    - H/W tokens (e.g. a Key fob)
    - S/W tokens are available for PDAs and mobile phones.
  - S/Key:
    - Uses a passphrase to generate one-time passwords.

RSA SecurID

Basic Operation:

RSA Tokens

•The authenticator (token) generates a simple one-time code every 60 sec.

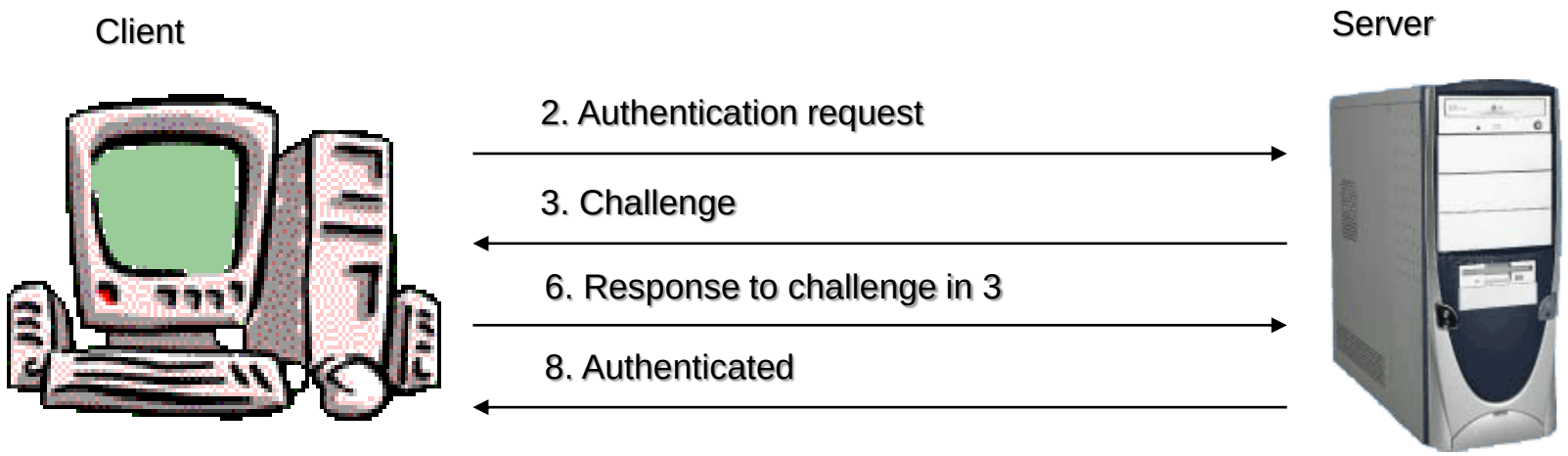
•The user combines this password with his PIN to generate a one-time a password.

•The RSA Authentication Server can validate this password since its clock is synchronised with the token and it knows the user PIN number from the registration stage.

•The authentication code changes every 60 sec, therefore password will change every time its used.

•This is a two-factor authentication system as it combines something you know (PIN), with something you have (token)

S/Key



1. User enters passphrase

4. Generator generates password

5. Entry of password into system, ecnryption

4. Generator generates password

7. Evaluation of response

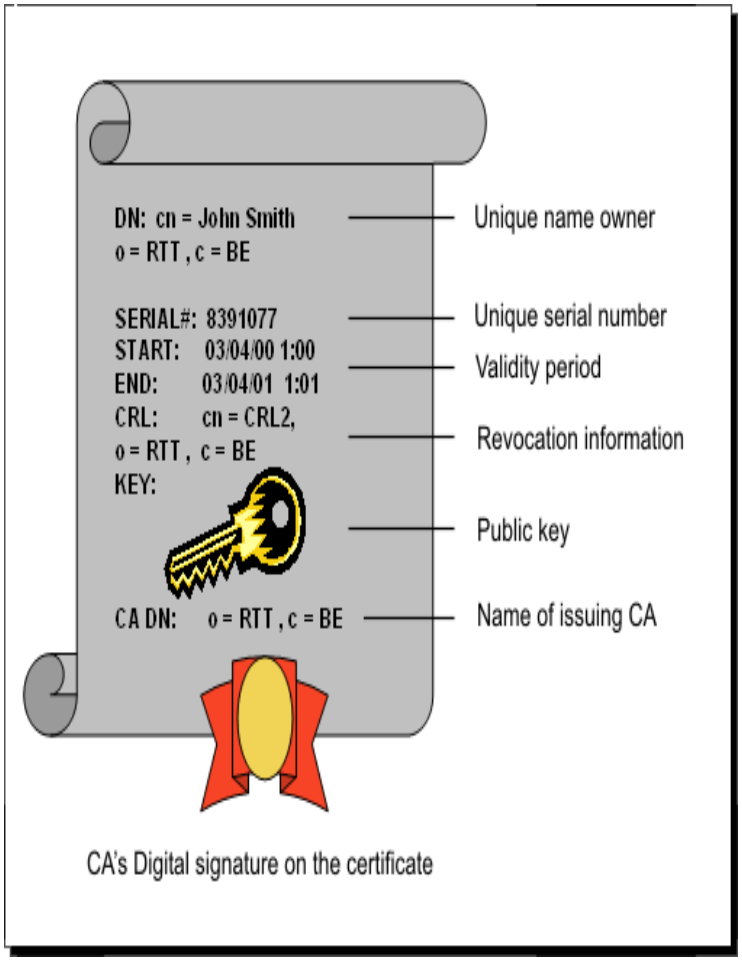
The S/Key one-time password process

Notes on one-time password systems

- S/Key and other one-time password systems provide defense against passive eavesdropping and replay attacks.
- There is no privacy of transmitted data, nor any protection from session hijacking.
- Additional secure channel protection should be provided through e.g. IPSec and SSH.

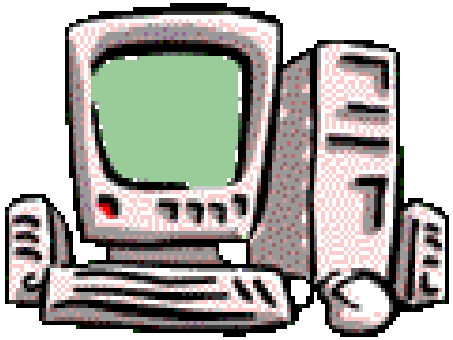
Certificate-based Authentication

- A certificate is a collection of information that binds an *identity* (user, computer, service or device) to the public key of a public/private key pair.
- The certificate is digitally signed by its issuing authority which is a Certificate Authority (CA) e.g. VeriSign
- The infrastructure that supports the use of certificates is known as the Public Key Infrastructure (PKI)
- Two common systems that use certificates for authentication are SSL/TLS and smart cards



Certificate-based Authentication

Client



Server

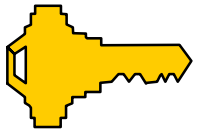
1. Authentication request

2. Challenge

4. Response

7. Authenticated

3. Client's private key used to encrypt / digitally sign the challenge and produce the response



Private key

5. Public Key of client used to decrypt /verify the response

Public key

6. Result compared to challenge

Scenario 1: Client-Authentication

Notes on Certificate-based Authentication

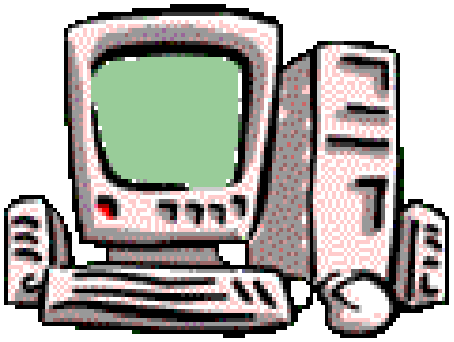
- In Symmetric encryption the same key is used for encryption and decryption.
- However; the public/private key algorithm is Asymmetric, i.e. it uses two different keys: one for encryption and the other for decryption.
- If a private key encrypts only the related public key decrypts and vise versa.
- Both the private/public keys are generated by the client and the public key is sent to the Certificate Authority (CA).
- The CA generates the certificate and signs it using its own private key and returns a copy of the certificate to the client and stores it in its database.
- The strength of the certificate-based authentication relies on protecting the private key. If the private key is known by someone then he can impersonate the client.

SSL/TLS

- Secure Socket Layer (SSL) is a certificate-based system developed by Netscape.
- SSL is most commonly used for server authentication and encrypted communication.
- Transport Layer Security (TLS) is the Internet Standard version of SSL.
- SSL and TLS provide similar functionality, however; they are not compatible with each other.
- SSL is most commonly used for server authentication from a client.

SSL – A Server Authentication Scenario

Client



Server

2. Request a Web Page from Server

3. Send Server certificate to client

8. Send symmetric encryption key to server

1. URL entered in browser

4. Browser looks for certificate in its store of public CA certificates

5. Browser checks the signature on server certificate using Public key of CA

6. If correct signature then Accept

7. Generate a symmetric encryption key using the server's public key

9. Symmetric key is decrypted using server's own private key

10. The symmetric encryption key is now shared between client and server and can be used for secure communication

Notes on SSL Authentication

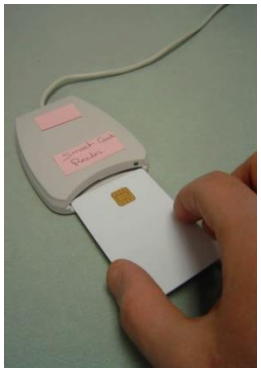
- Unless server is properly configured to use SSL, the server is not authenticated and unprotected communication can take place.
- SSL Security relies on the client using https:// instead of http:// in URL entry.
- If client does not have a copy of the CA's certificate, the server will offer to provide one. This ensures secure communication but does not provide server authentication.
- Security of SSL authentication relies on client refusing to connect to a server that can not be identified by a third party CA. Experience shows that many users just ignore such browser warnings and continue on.

Smart Cards and other H/W-Based Authentication

- Protecting the private key is highly important on certificate-based authentication systems.
- In most cases the private key is stored on the computer, i.e. there is a potential for compromise
- Smart cards and tokens enable storing the private key away from the computer.
- A typical smart card that is used for authentication has a computer chip that stores the private key and a copy of the certificate as well as to provide processing.
- When the user enters the smart card he will be asked to provide a PIN before the computer can communicate with the card and use data.

USB token

Smart Card



Authorisation

Authorisation determines what an authenticated user can do on the system or network

Authorisation can be achieved in a number of ways, e.g:

- User Rights:
  - E.g.: create groups, assign users to groups, logo on to system
- Role-Based Authorisation:
  - E.g.: administrators, users, guests
- Access Control Lists (ACLs):
  - E.g.: blocking certain communication on a given port from passing through the firewall.
- File Access Permissions:
  - E.g.: read, write, execute permissions in Unix
- Rule-Based Authorisation:

  E.g.: user x can access file y from any computer in Unix lab but not in PC lab.

Biometrics

Something that you know?

Something that you have?

Something that you are

Simple:

- Verification – Is this who he claims to be?
- Identification – who is this?

Advanced:

- Detecting multiple identities
- Patrolling public spaces

2 Categories of Biometrics

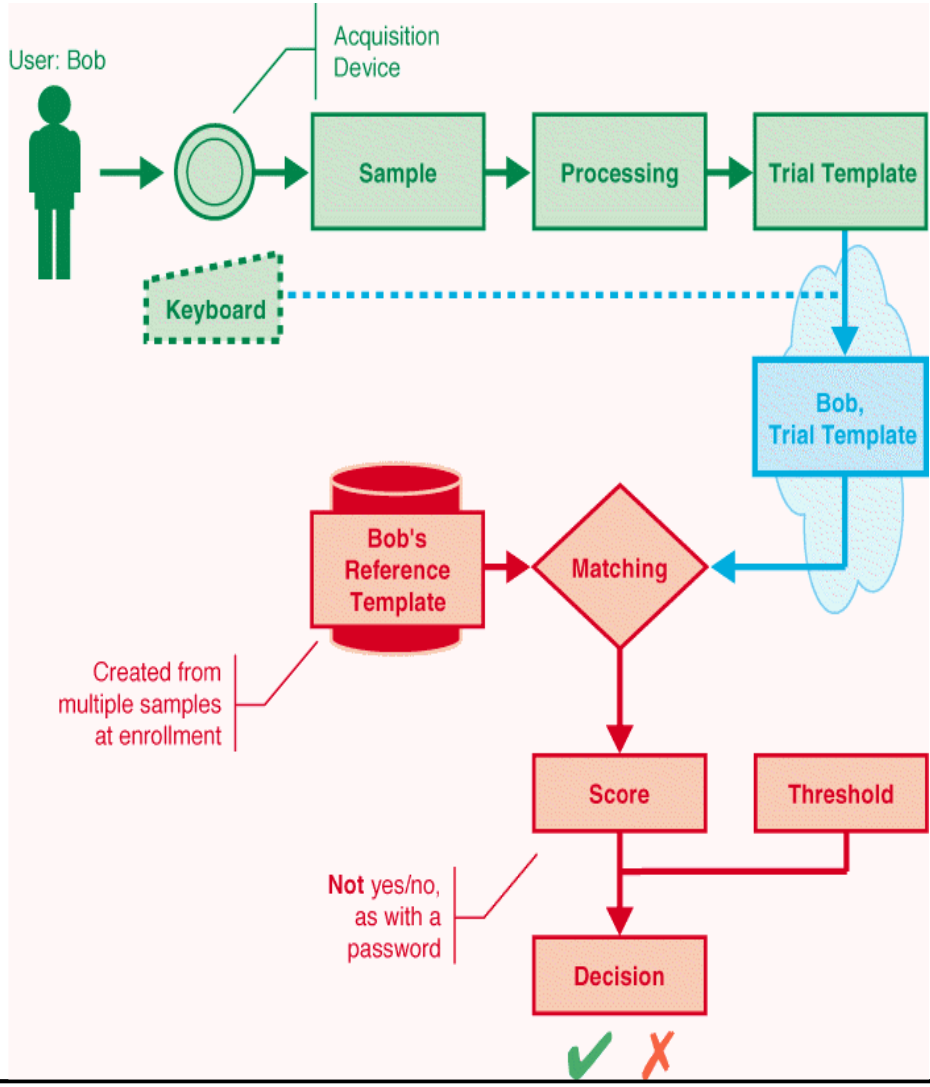
Physiological – also known as static biometrics: Biometrics based on data derived from the measurement of a part of a person's anatomy. For example, fingerprints and iris patterns, as well as facial features, hand geometry and retinal blood vessels

Behavioral – biometrics based on data derived from measurement of an action performed by a person and, distinctively, incorporating time as a metric, that is, the measured action. For example, voice (speaker verification)

Biometrics – How do they work?

Although biometric technologies differ, they all work in a similar fashion:

- The user submits a sample that is an identifiable, unprocessed image or recording of the physiological or behavioral biometric via an acquisition device (for example, a scanner or camera)
- This biometric is then processed to extract information about distinctive features to create a trial template or verification template
- Templates are large number sequences. The trial template is the user's "password."



Biometrics

Definition

Biometrics is the science of verifying and establishing the identity of an individual through physiological features or behavioral traits.

Examples

Physical Biometrics

- Fingerprint
- Hand Geometry
- Iris patterns

Behavioral Biometrics

- Handwriting
- Signature
- Speech
- Gait

Chemical/Biological Biometrics

- Perspiration
- Skin composition(spectroscopy)

Overview of Biometrics

Biometric	Acquisition Device	Sample	Feature Extracted
Iris	Infrared-enabled video camera, PC camera	Black and white iris image	Furrows and striations of iris
Fingerprint	Desktop peripheral, PC card, mouse chip or reader embedded in keyboard	Fingerprint image (optical, silicon, ultrasound or touchless)	Location and direction of ridge endings and bifurcations on fingerprint, minutiae
Voice	Microphone, telephone	Voice Recording	Frequency, cadence and duration of vocal pattern
Signature	Signature Tablet, Motion-sensitive stylus	Image of Signature and record of related dynamics measurement	Speed, stroke order, pressure and appearance of signature
Face	Video Camera, PC camera, single-image camera	Facial image (optical or thermal)	Relative position and shape of nose, position of cheekbones
Hand	Proprietary Wall-mounted unit	3-D image of top and sides of hand	Height and width of bones and joints in hands and fingers
Retina	Proprietary desktop or wall mountable unit	Retina Image	Blood vessel patterns and retina

Strengths, Weaknesses and Usability of Biometrics

Biometric	Strengths	Weakness	Usability
Iris	• Very stable over time • Uniqueness	• Potential user resistance • Requires user training • Dependant on a single vendor's technology	• Information security access control, especially for Federal Institutions and government agencies • Physical access control (FIs and government) • Kiosks (ATMs and airline tickets)
Fingerprint	• Most mature biometric technology • Accepted reliability • Many vendors • Small template (less than 500 bytes) • Small sensors that can be built into mice, keyboards or portable devices	• Physical contact required (a problem in some cultures) • Association with criminal justice • Vendor incompatibility • Hampered by temporary physical injury	• IS access control • Physical access control • Automotive
Optical	• Most proven over time • Temperature stable	• Large physical size • Latent prints • CCD coating erodes with age • Durability unproven	

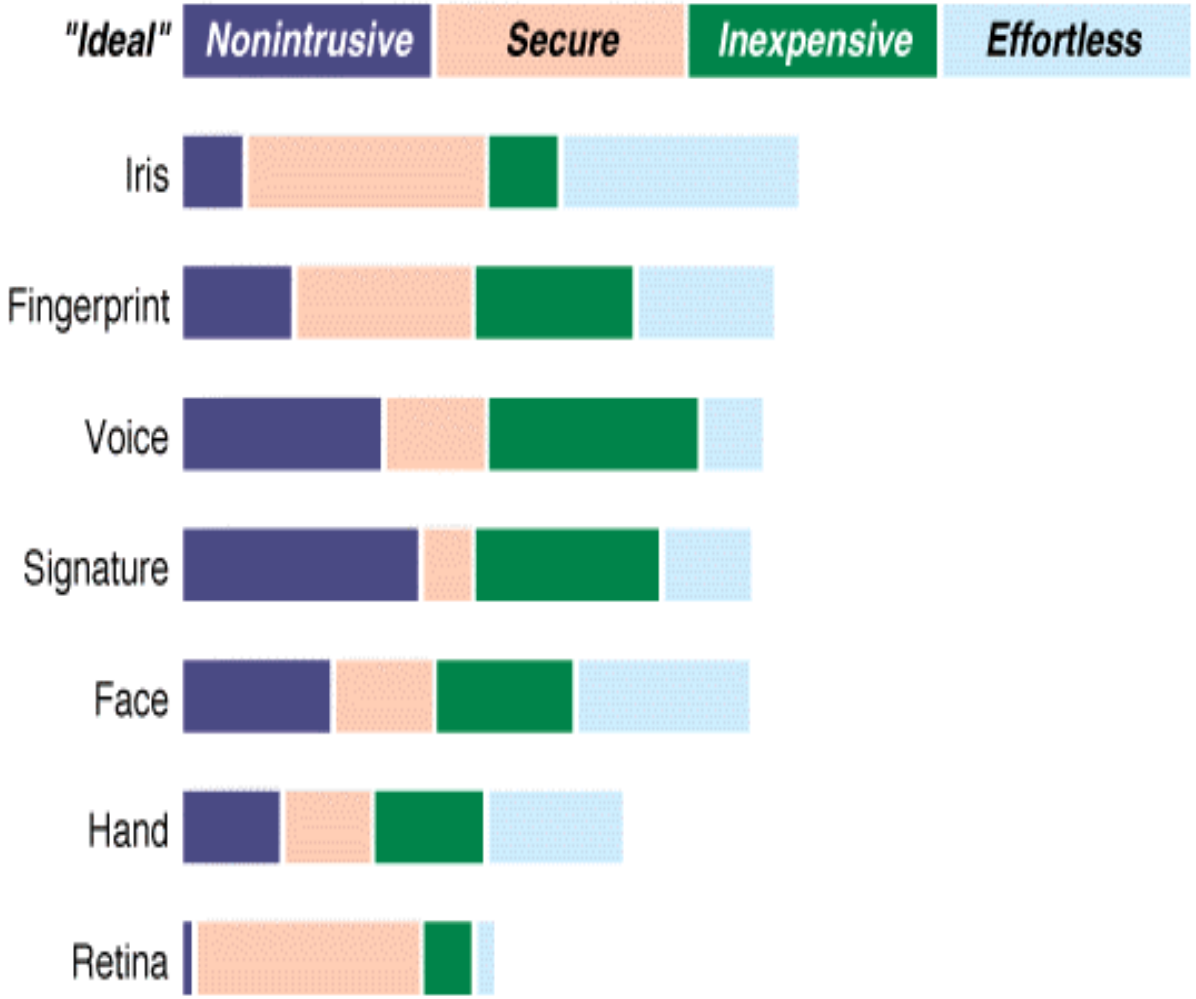
Strengths, Weaknesses and Usability of Biometrics

Biometrics	Strengths	Weakness	Usability
Silicon	• Small physical size • Cost is declining	• Requires careful enrollment • Unproven in sub optimal conditions	
Ultrasound	• Most accurate in sub optimal conditions	• New technology, few implementations • Unproven long term performance	
Voice	• Good user acceptance • Low training • Microphone can be built into PC or mobile device	• Unstable over time • Changes with time, illness stress or injury • Different microphones generate different samples • Large template unsuitable for recognition	• Mobile phones • Telephone banking and other automated call centers
Signatures	• High user acceptance • Minimal training	• Unstable over time • Occasional erratic variability • Changes with illness, stress or injury • Enrollment takes times	• Portable devices with stylus input • Applications where a “wet signature” ordinarily would be used.

Strengths, Weaknesses and Usability of Biometrics

Biometrics	Strengths	Weakness	Usability
Face	• Universally present	• Cannot distinguish identical siblings • Religious or cultural prohibitions	• Physical access control
Hand	• Small template (approximately 10 bytes) • Low failure to enroll rate • Unaffected by skin condition	• Physical size of acquisition device • Physical contact required • Juvenile finger growth • Hampered by temporary physical injury	• Physical access control • Time and attendance
Retina	• Stable over time • Uniqueness	• Requires user training and cooperation • High user resistance • Slow read time • Dependent on a single vendor's technology	• IS access control, especially for high security government agencies • Physical access control (same as IS access control)

Comparison of Different Biometrics Technology



Promise that Biometrics hold for Privacy

Increased Security

- Biometric cannot be lost, stolen or forgotten; it cannot be written down and stolen by social re-engineering
- By implementing biometrics organizations can positively verify users' identities, improving personal accountability
- In conjunction with smart cards biometrics can provide strong security for Public Key Infrastructure (PKI)

Why Biometrics?

- With numerous devices, traditional paradigm of user name and password based scenarios are not practical
- Only authorized users should have access to data and services
- Biometrics provide an unobtrusive and convenient authentication mechanism
- Advantages of biometrics
    - Uniqueness
    - No need to remember passwords or carry tokens
    - Biometrics cannot be lost, stolen or forgotten
    - More secure than a long password
    - Solves repudiation problem
    - Not susceptible to traditional dictionary attacks

Biometrics

False Rejection Rate (FRR)

- Percentage of rejected access attempts of authorised users

False Acceptance Rate (FAR)

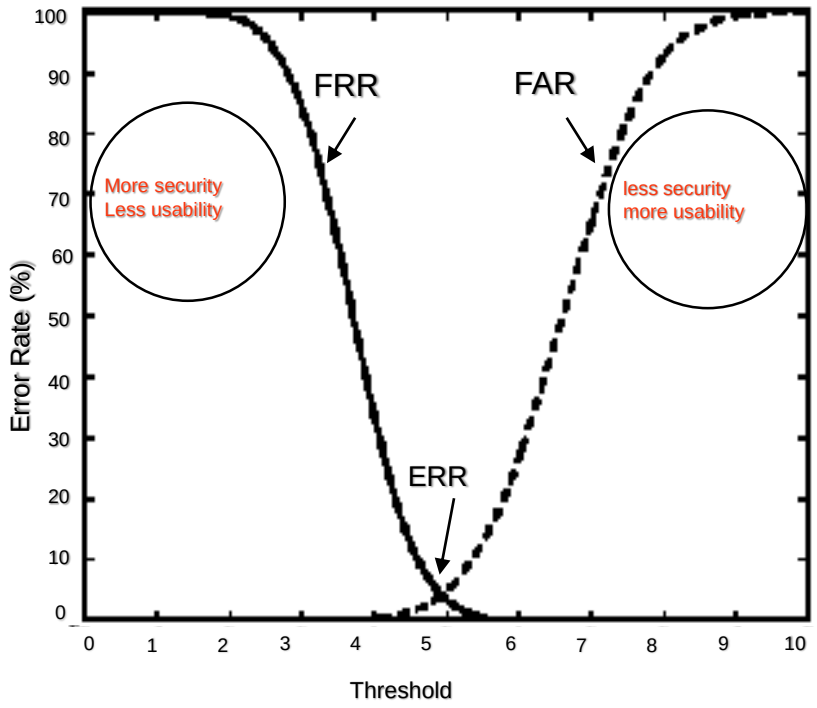
- Percentage of successful access attempts of unauthorised users

Equal Error Rate (ERR)

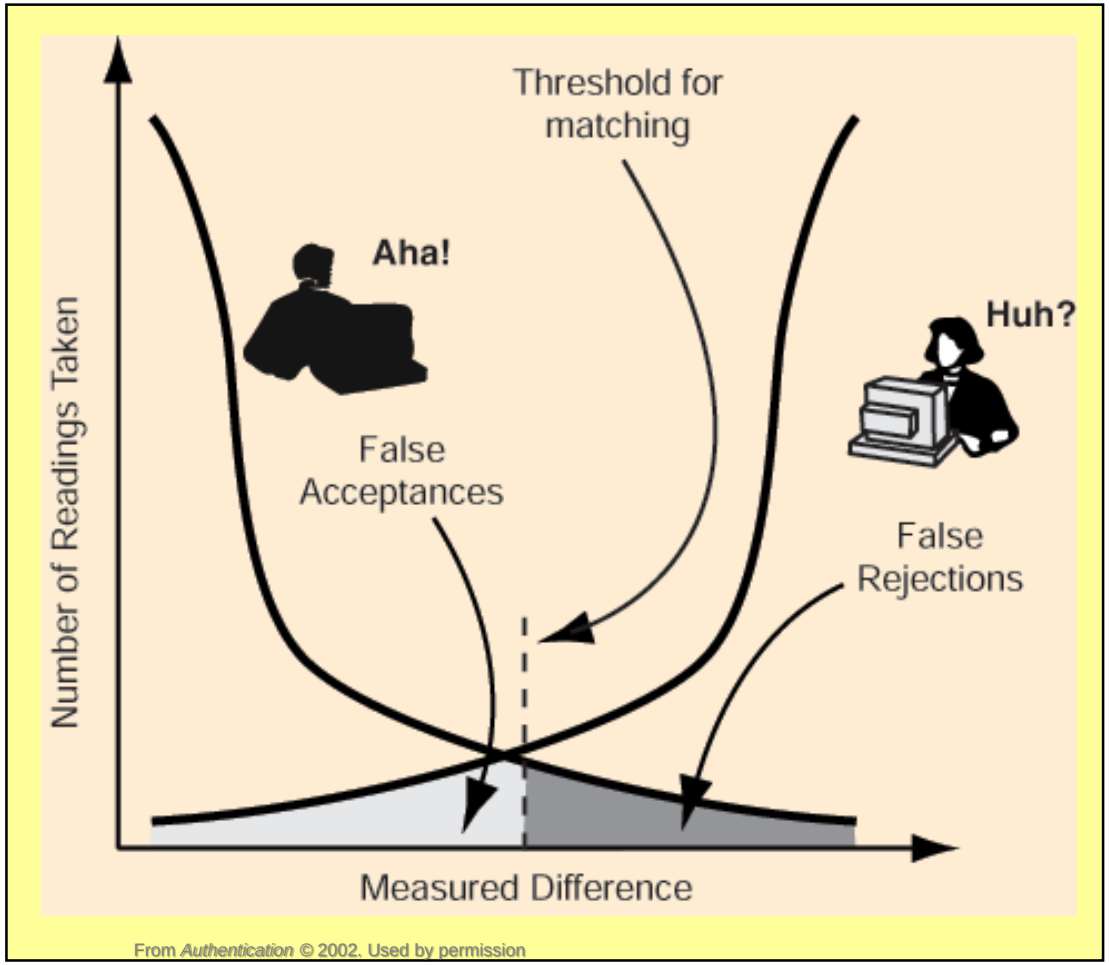
- FAR = FRR

The aim is to minimise

- both FRR and FAR



Matching in Practice



FAR = recognized Bob instead; FRR = doesn't recognize me

False Accept Rate (FAR), False Reject Rate (FRR)

Measurement Trade-Offs

We must balance the FAR and the FRR

- If the tolerance threshold is set too large, then cause Type II Errors (Imposter admiited)
- If the tolerance threshold is too small, then it cause Type I Errors (legitimate users are rejected)

- Lower FAR = Fewer successful attacks
  - Less tolerant of close matches by attackers
  - Also less tolerant of authentic matches
  - Therefore – increases the FRR

- Lower FRR = Easier to use
  - Recognizes a legitimate user the first time
  - More tolerant of poor matches
  - Also more tolerant of matches by attackers
  - Therefore – increases the FAR

•Equal error rate = point where FAR = FAR

Perils that Biometrics hold for Privacy

Privacy is one of the leading inhibitor for biometrics technology.

Main issues:

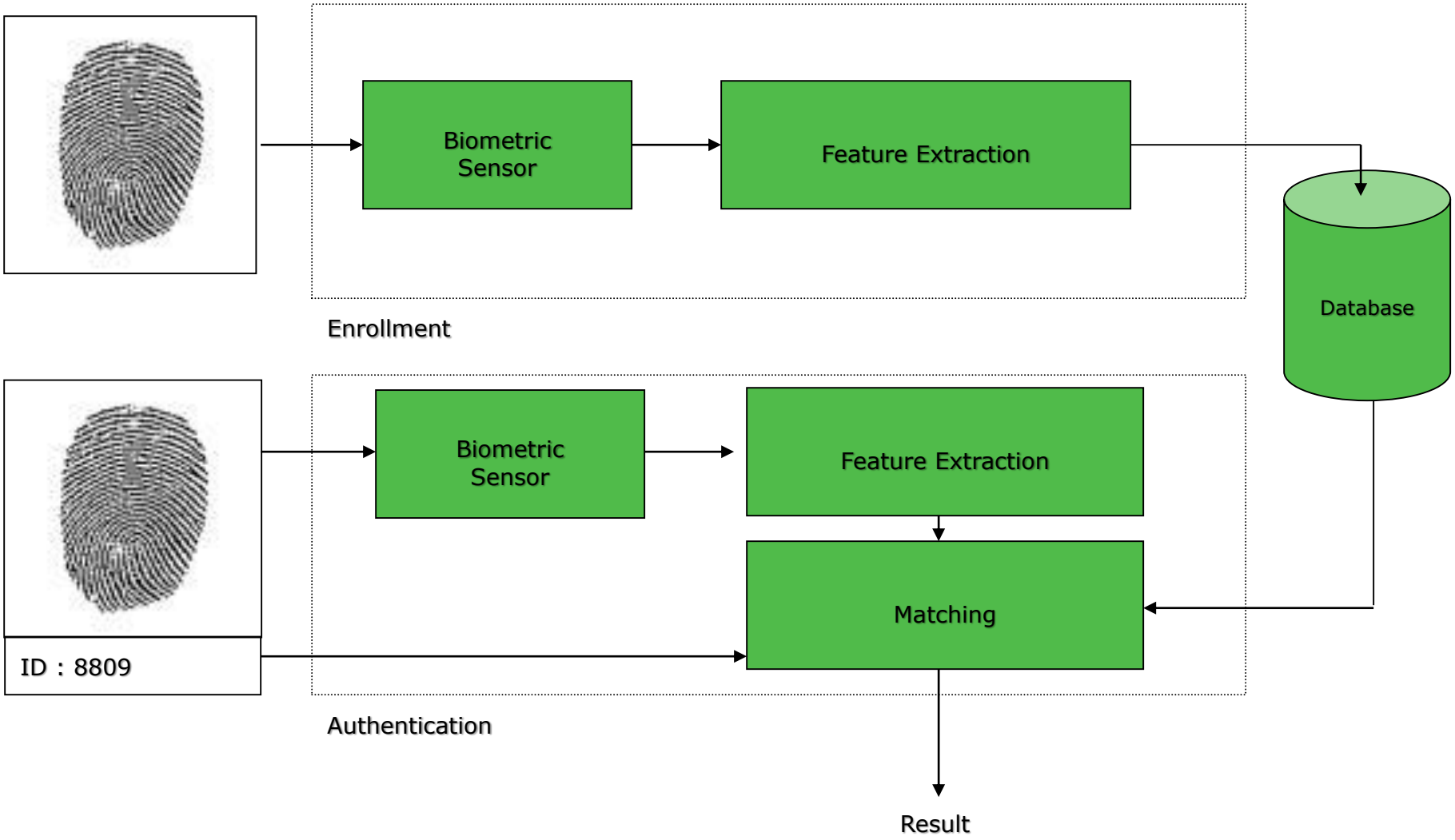
Misuse of Data

- Health/Lifestyle – Specific biometric data has been linked with the information beyond which it is set out to be used for such as AIDS. Is a person able to control the information gathered on himself/herself?

Function Creep

- Law Enforcement – The template database may be available for law enforcement
- Credit Reporting – The template database may be cross referenced against other databases including those held in hospitals and the police departments, by a credit reporting agency

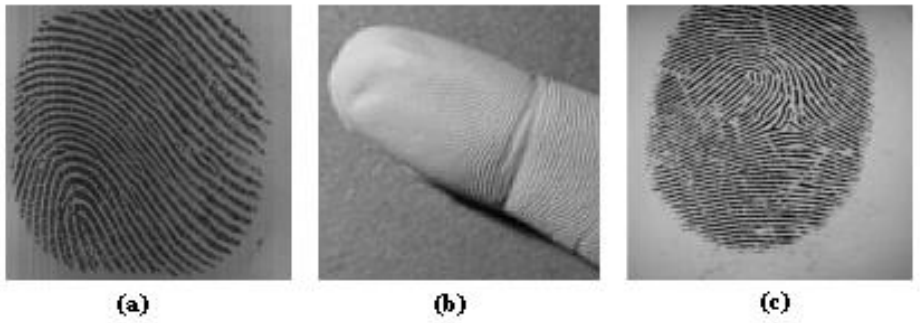
General Biometric System



Security of Biometric Data

Issues in biometrics

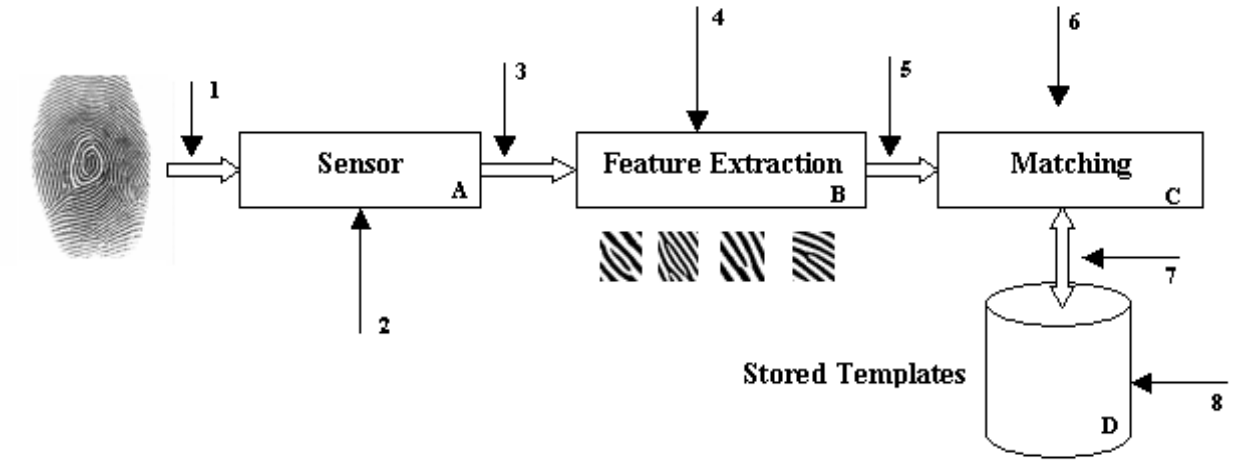
- Biometrics is secure but not secret
- Permanently associated with user
- Used across multiple applications
- Can be covertly captured



Fake Biometrics

Types of circumvention

- Denial of service attacks(1)
- Fake biometrics attack(2)
- Replay and Spoof attacks(3,5)
- Trojan horse attacks(4,6,7)
- Back end attacks(8)
- Collusion
- Coercion



Threats to a Biometric System

Hashing

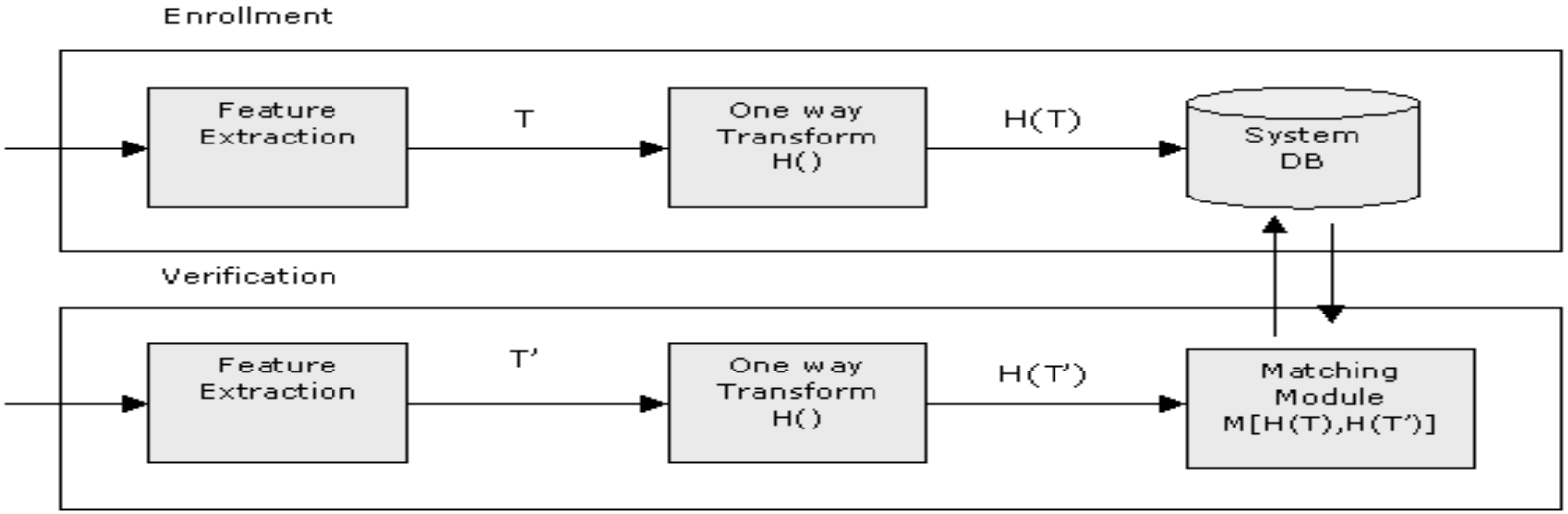
Hashing

- Instead of storing the original password P, a hashed values P′=H(P) is stored instead.
- The user is authenticated if H(password) = P′.
- It is computationally hard to recover P given H(P)
- H() − one way hashing function

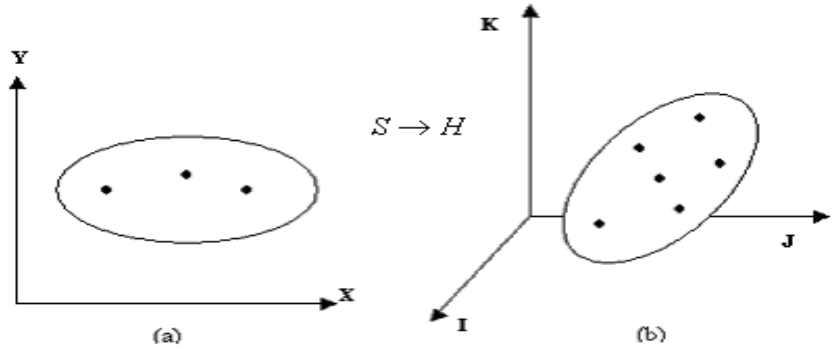
Problem with biometrics

- Biometric data has high uncertainty
- Matching is inexact/probabilistic
- Therefore, hashing function should be error tolerant

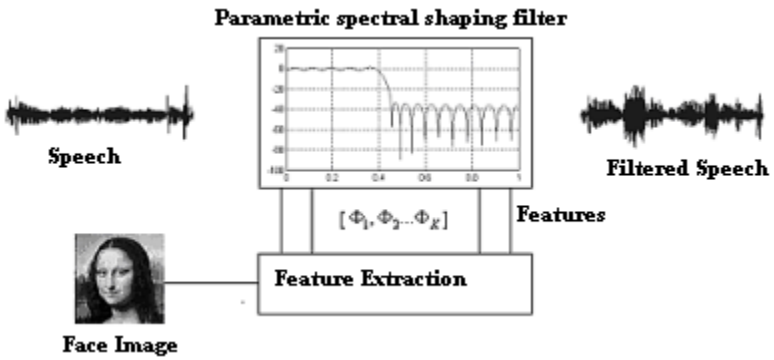
Biometric Hashing



Hashing Schema

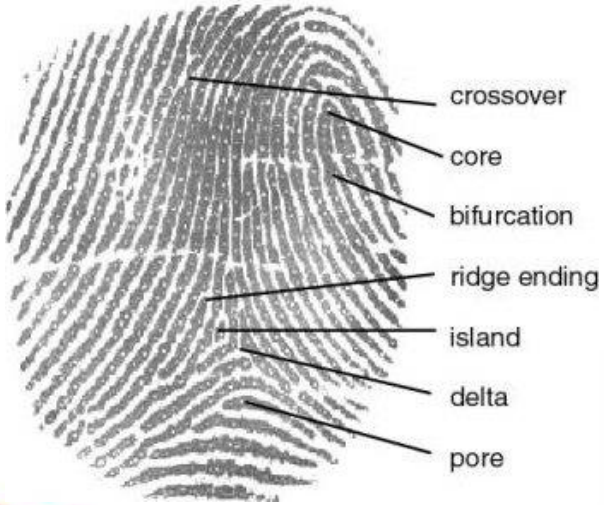


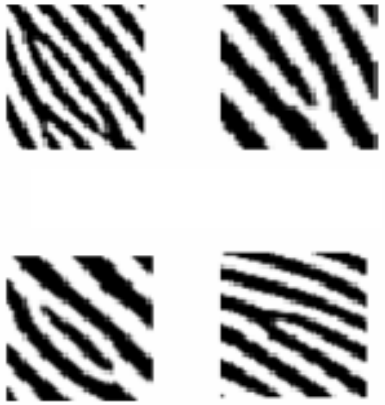
Hashing



Personalized Hashing

Fingerprints 101

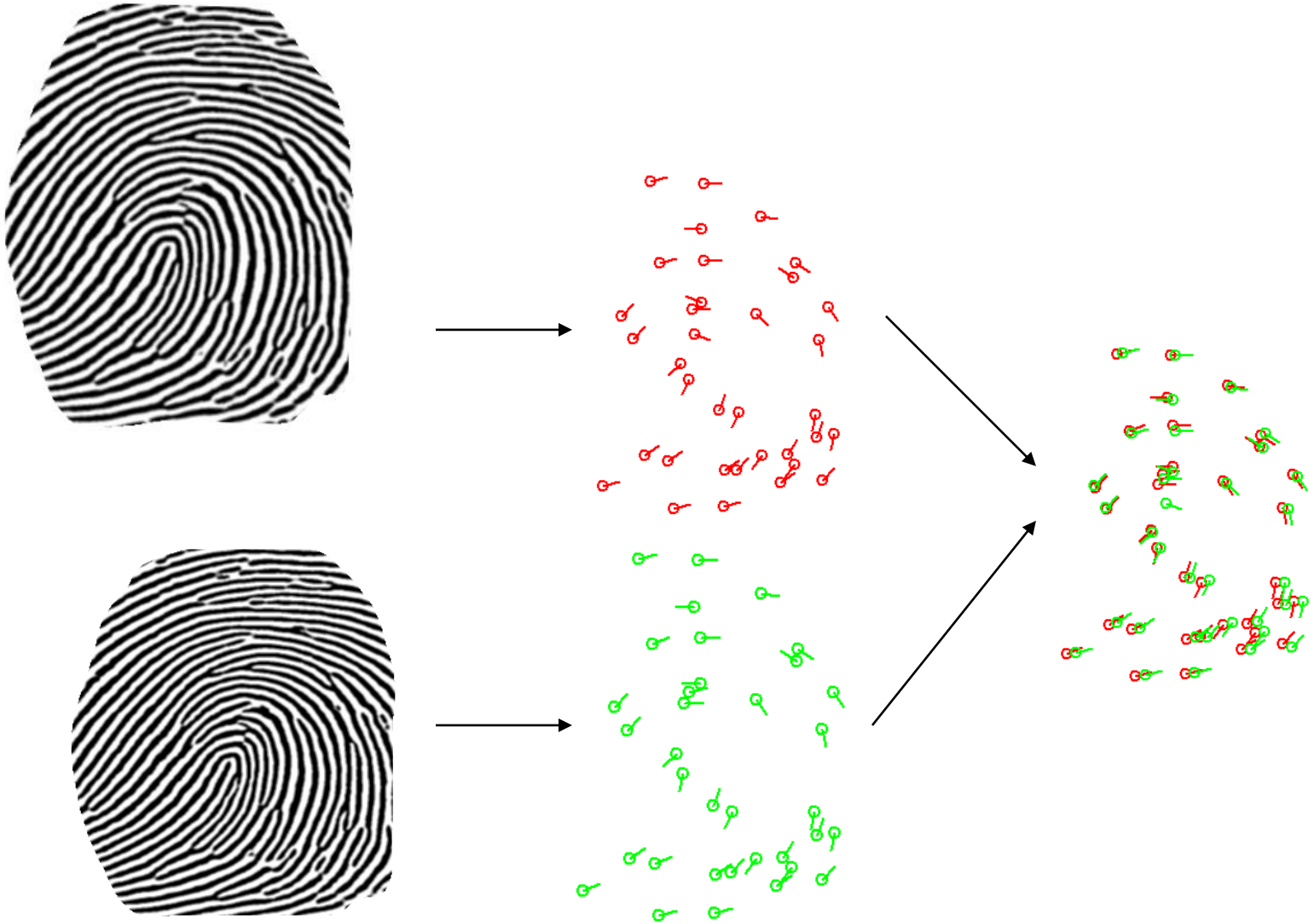




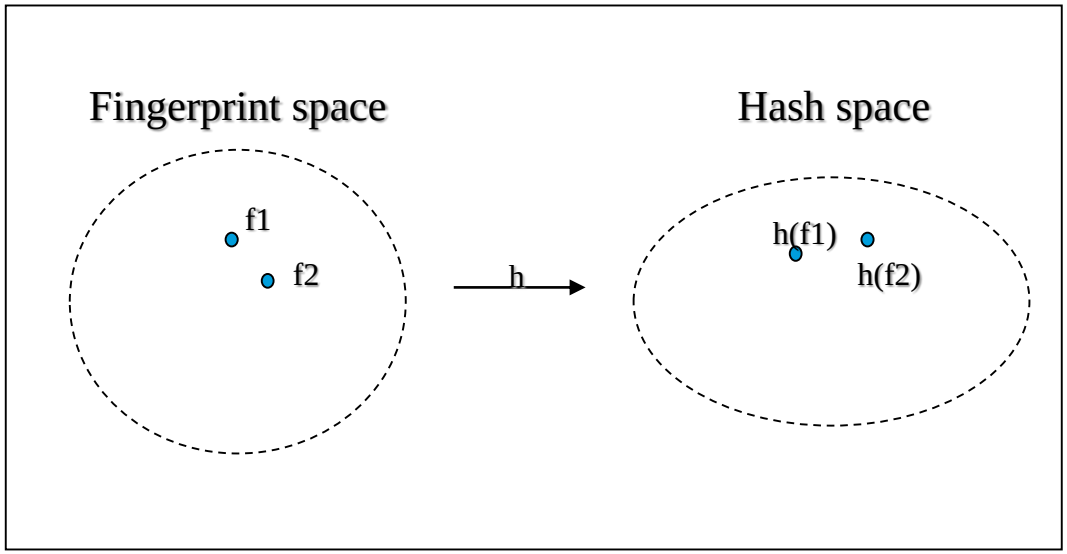
Minutiae: Local anomalies in the ridge flow
Pattern of minutiae are unique to each individual

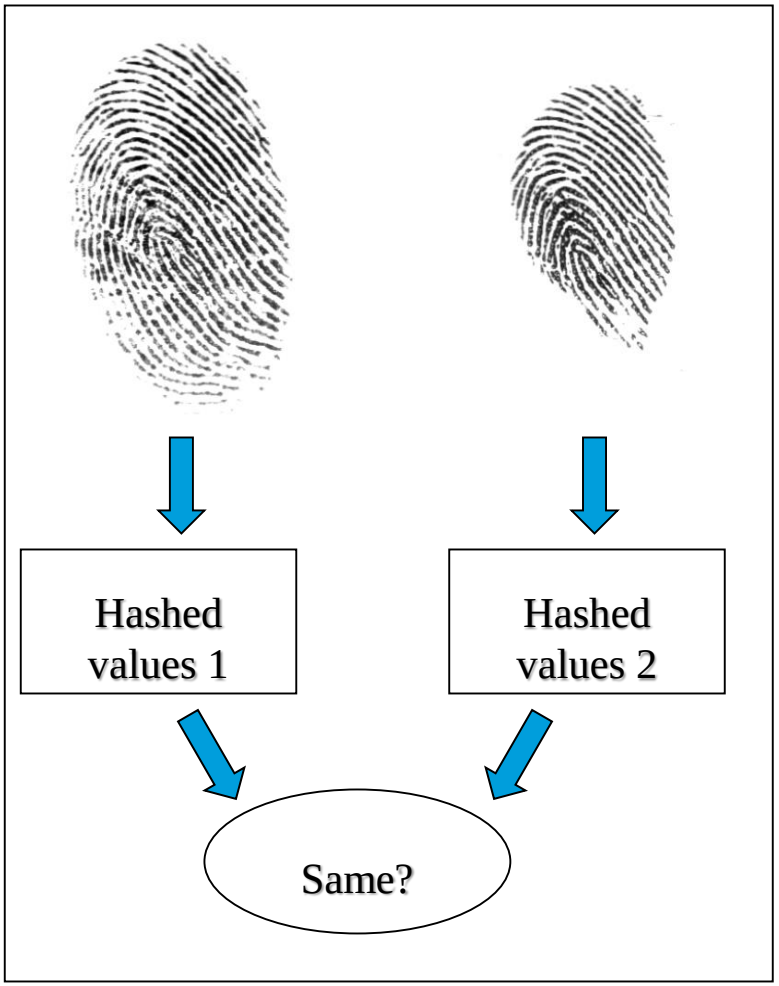
X	Y	θ	T
106	26	320	R
153	50	335	R
255	81	215	B

Fingerprint Verification



Research Challenges





Images include different scanned area.

Set of features is different for two different fingerprints of the same finger.

Similar fingerprints should have similar hash values
Hash values should be invariant to rotation/translation