

Introduction to Computer Security

Chapter 3: User Authentication

Chi-Yu Li (2020 Spring)
Computer Science Department
National Chiao Tung University

User Authentication

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

Definition of User Authentication (RFC 4949)

- The process of verifying an identity claimed by or for a system entity
- Two steps
 - 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

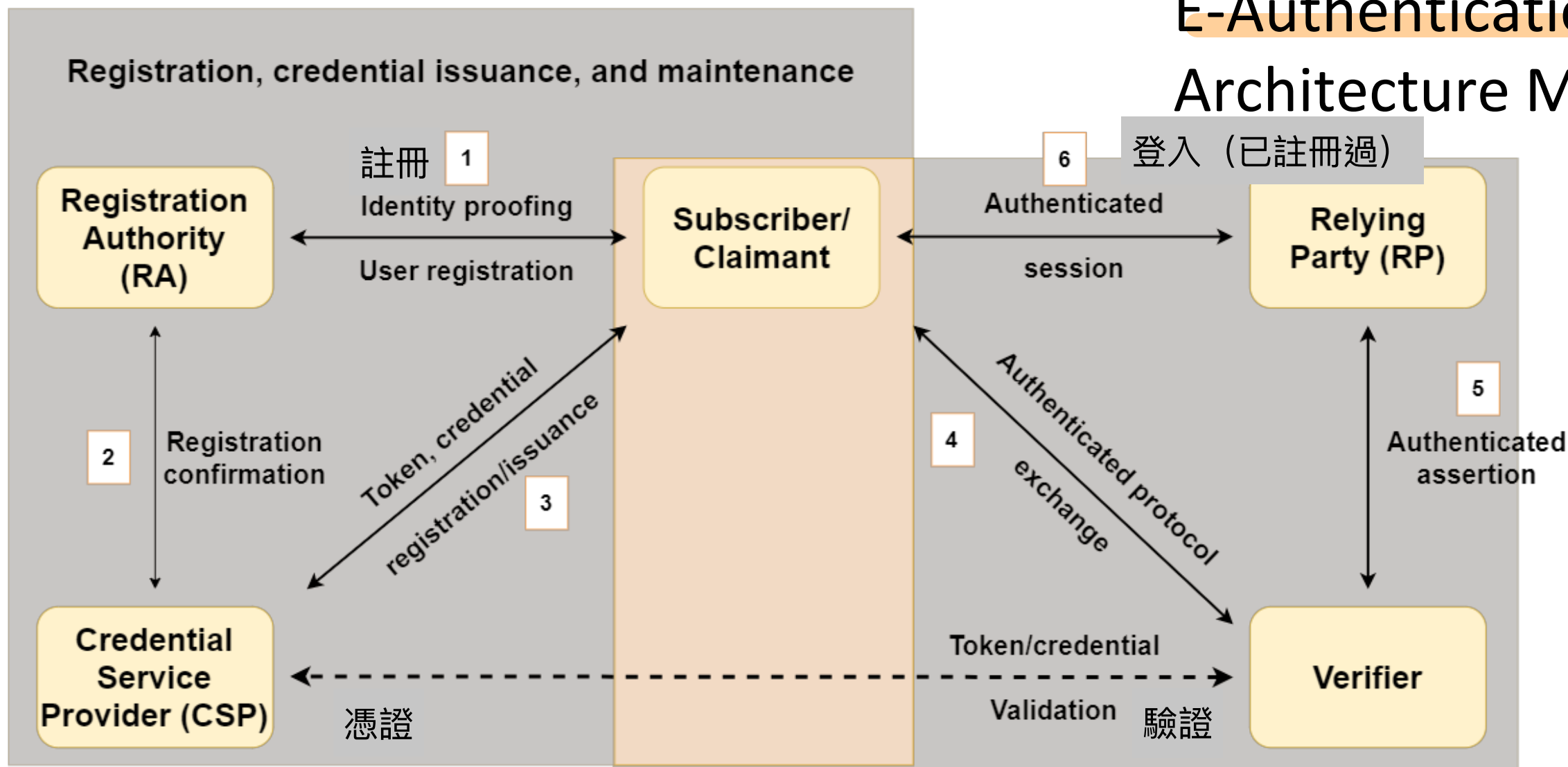
出示識別證

驗證

Outline

- Electronic User Authentication Model
- Password-based Authentication
- Token-based Authentication
- Biometric Authentication
- Remote user Authentication
- Security Issues for User Authentication

E-Authentication Architecture Model



Defined by NIST SP 800-63-2 (Electronic Authentication Guideline, August 2013)

E-Authentication using token and credential

Cornerstone: Credential and Token

● Credential

只要確認你是不是，可不可以允許你

- Paper credentials: documents that attest to the identity
 - e.g., passports, driver's licenses, and student ID cards
 - Contain the subject's description, a picture of the subject or a signature of the subject

- E-authentication credential: an object or data structure
 - Authoritatively binds an identity (via an identifier) and (optionally) additional attributes, to at least one token (or authenticator) possessed and controlled by a subscriber

Cornerstone: Credential and Token (Cont.)

確認你的身分

- Token: something that the Claimant possesses and controls is used to authenticate the Claimant's identity

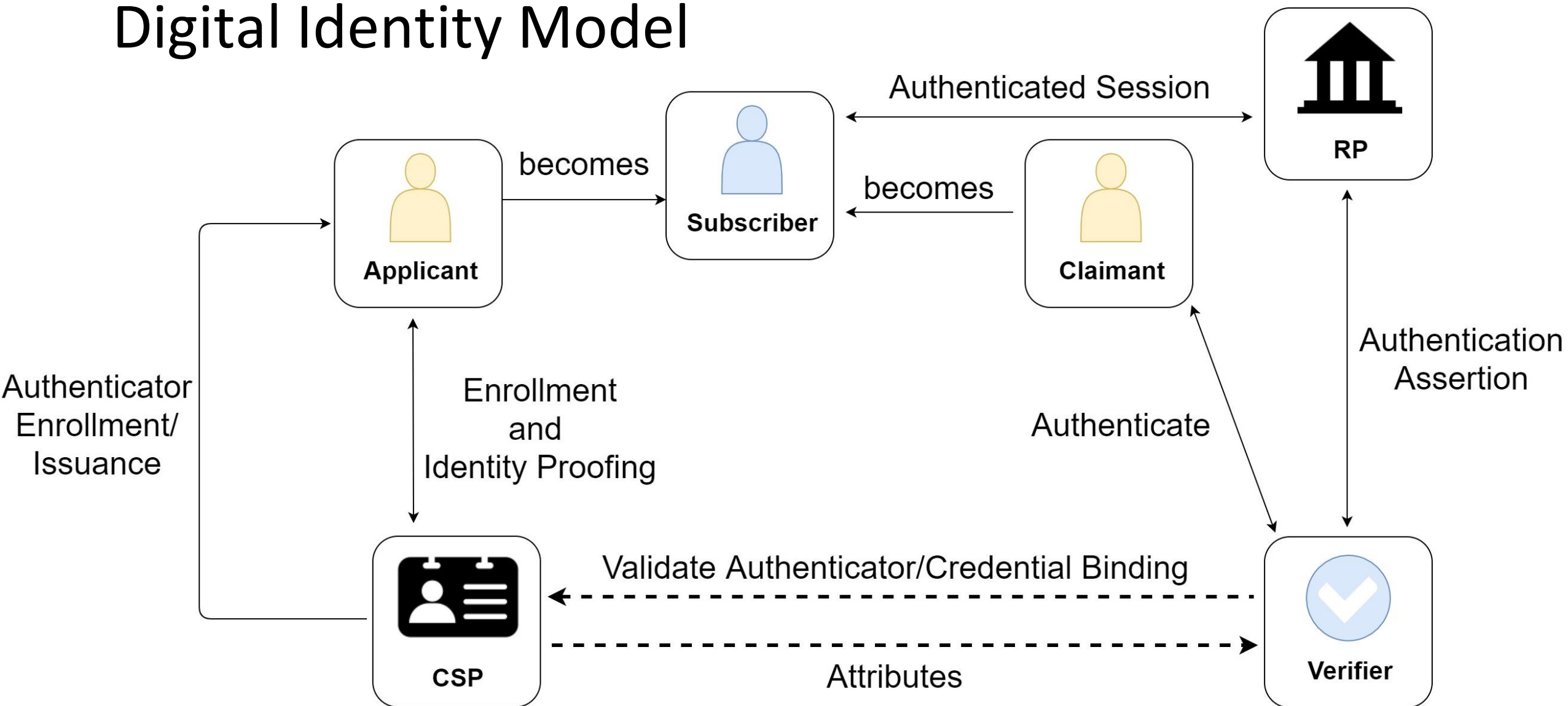
- Typically a cryptographic module or password
- Also named as authenticator

學生證是credential

刷了之後知道你0716000(Great!)，學號就是一個token

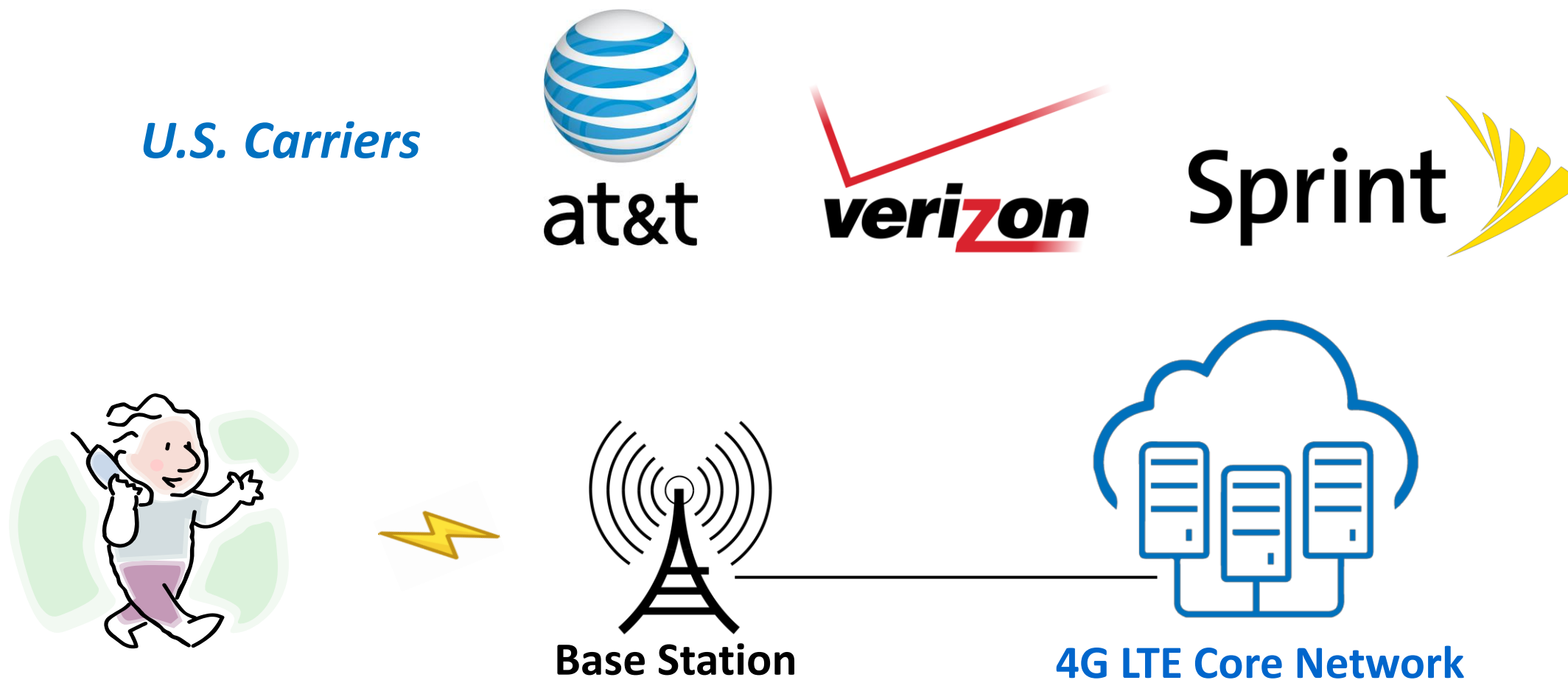
- In other words, authentication establishes confidence that
 - The Claimant has possession of an authenticator(s) bound to the credential, and (optional) the attribute values of the subscriber
 - Attribute values: s(he) is a Taiwan citizen, or a student at NCTU

Digital Identity Model

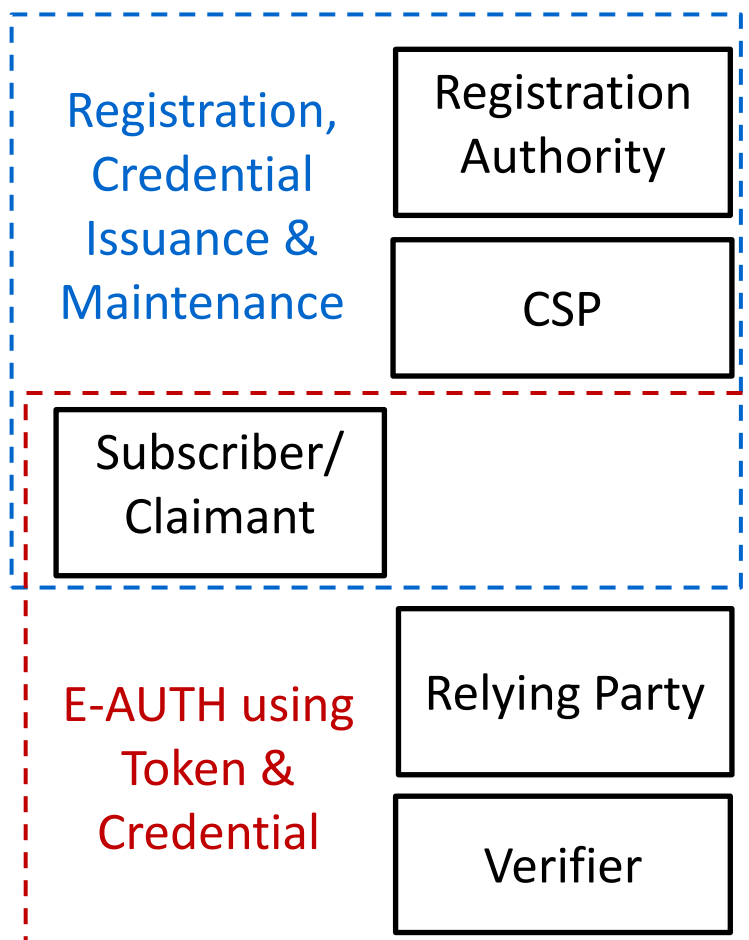


Digital Authentication

Example: Do Cellular Services Follow the E-Auth?



Role for Each Entity



人工註冊一個號碼



給你sim卡用一波她基地台



at&t Store



4G LTE Core Network

CSP: Credential Service Provider

Tokens (Authenticators)

- Something the individual knows
 - e.g., password, answers to prearranged questions
- Something the individual possesses
 - e.g., electronic keycards, smart cards
- Something the individual is (static biometrics)
 - e.g., fingerprint, retina, face
- Something the individual does (dynamic biometrics)
 - e.g., voice pattern, handwriting

Password-based Authentication

- Widely used line of defense against intruders
 - ❑ User provides name or identifier (ID) and password
 - ❑ System compares password with the stored one
 - A password file indexed by user ID: store usernames or hash values of passwords
- User ID
 - ❑ Determines that the user is authorized to access the system
 - ❑ Determines the user's privileges
 - ❑ Used in discretionary access control



Local Login

登入 Facebook

電子郵件或電話號碼

密碼

登入

Remote Login

離線字典攻擊

1. 離線拿到系統hash密碼檔
2. 破解hash value
3. 得到常見passwords

Attacks and Countermeasures

● Offline dictionary attack

- ❑ Obtain the system's password file (passwords stored in hash values)
 - CVE cases: <https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=password>
- ❑ Search for valid passwords with hash values of commonly used passwords
 - Tools: <http://www.openwall.com/john/>, <http://project-rainbowcrack.com/>
- ❑ Countermeasure: prevent unauthorized access to the password file, intrusion detection measures to identify a compromise, etc.

● Specific account attack

對一個帳號窮舉密碼

- ❑ Submit password guesses until the correct password is discovered or the account is blocked (more than 5 failure times)

Attacks and Countermeasures (Cont.)

● Popular password attack 試常見的密碼

- ❑ Try popular passwords, e.g., 123456, against a wide range of user IDs
 - Assume that adversary obtains user IDs in advance
- ❑ Countermeasure: inhibiting the selection by users of common passwords, scanning the IP addresses of auth requests and client cookies for submission patterns

● Password guessing against single user 你夠了解他，就能猜出他密碼

- ❑ Gain knowledge about the account holder and system password policies, and use that knowledge to guess the password
- ❑ Countermeasure: enforcement of password policies that make passwords difficult to guess (e.g., minimum length of the password)

Attacks and Countermeasures(Cont.)

● Workstation hijacking

工作站如果不會自動登出User，那會滿危險的喔

- ❑ Wait until a logged-in workstation is unattended
- ❑ Countermeasure: automatically logging the workstation out after a period of inactivity

● Exploiting user mistakes

就很智障的給密碼

- ❑ Mistakes: write it down, share it via any ways, keep preconfigured passwords, etc.
- ❑ Countermeasure: user training, intrusion detection, simpler passwords combined with another auth. mechanism, etc.

Attacks and Countermeasures(Cont.)

● Exploiting multiple password uses

不要都用一樣的密碼

- ❑ Different network devices share the same or similar password for a given user
- ❑ Countermeasure: a policy that forbids it

● Password sniffing/phishing

攔截傳送中的未加密密碼

- ❑ Passwords are transmitted without encryption, e.g., http or ftp
- ❑ Phishing web pages
- ❑ Countermeasure: encryption, inputting passwords with trusted devices and environments, etc.

Still: Most Commonly Used User Authentication

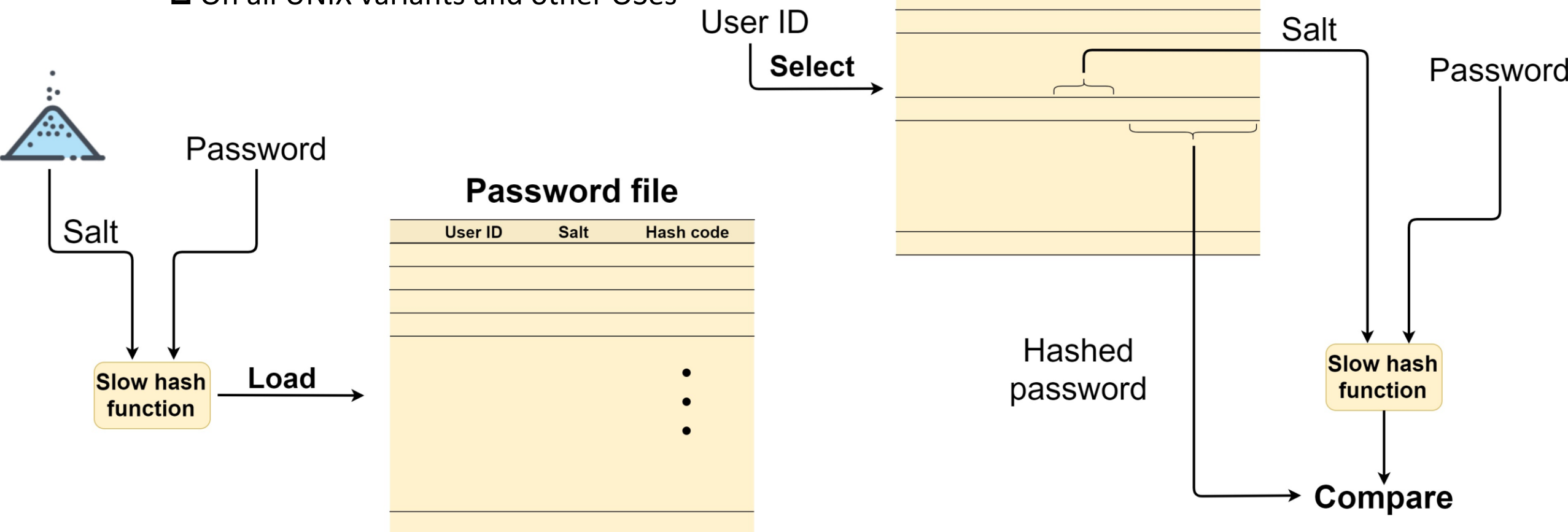
- Reasons for the persistent popularity of passwords

- ❑ Cheap, convenient for use, and easy to implement
- ❑ Other techniques based on client-side hardware require the implementation of the software on both client and server
 - e.g., fingerprint scanners and smart card readers
- ❑ Physical tokens are expensive and/or inconvenient
 - e.g., smart cards
- ❑ Biometric tokens are expensive and/or not sufficiently accurate

Use of Hashed Passwords

- A widely used password security technique

- ▣ On all UNIX variants and other OSes



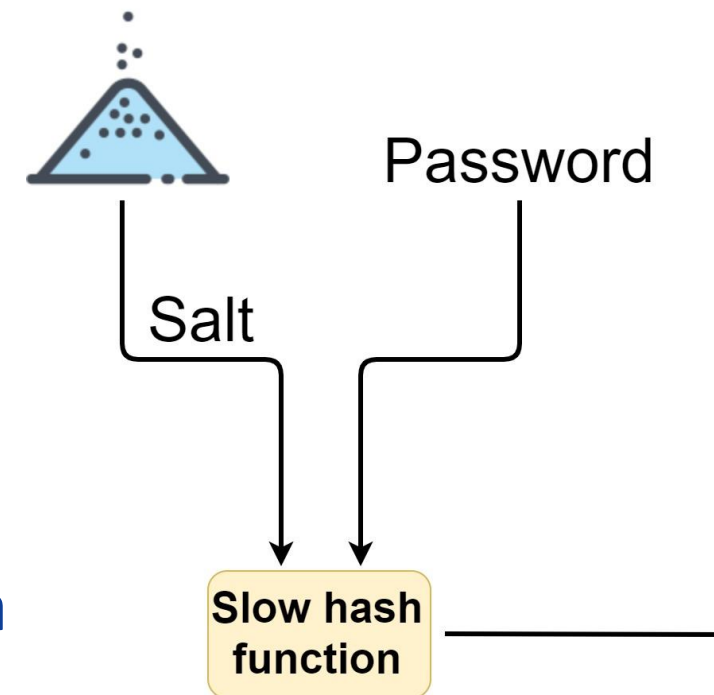
(a) Loading a new password

(b) Verifying a password

Why Salt?

用來和pswd一起hash，你想破解？很難！

- Purpose I: prevents duplicate passwords from being visible in the password file
- Purpose II: greatly increases the difficulty of offline dictionary attacks
 - For a salt of length b bits, the number of possible passwords is increased by a factor of 2^b
- Purpose III: Nearly impossible to find out whether a person has used the same password on multiple systems



Two Threats

- Threat I: password guessing on the machine

- ❑ Attackers gain access on a machine using a guest account or by some other means
- ❑ Run a password guessing program, called a password cracker

- Threat II: password guessing on another machine

- ❑ They can have a copy of the password file on another machine, and then run the cracker
- ❑ Run through millions of possible passwords in a reasonable period

Old Implementation of UNIX Password Scheme

- Password: up to 8 characters in length (56-bit value using 7-bit ASCII)
 - Serving as the key input to DES
- Modified DES encryption
 - An one-way hash function with a data input of a 64-bit block of zeros
- Repeated for a total of 25 encryptions
- Has been regarded as inadequate (50 million password guesses in about 80 minutes)
 - But, still often required for compatibility with existing account management software or multivendor environments

KEY : 密碼

INPUT : 第一輪用全ZERO
做25輪DES加密

Improved Implementations

- Recommended hash function is based on MD5
 - ❑ Salt: up to 48-bit
 - ❑ Password length is unlimited
 - ❑ A 128-bit hash value
 - ❑ Slowdown: an inner loop with 1000 iterations
- Bcrypt: developed for OpenBSD based on the Blowfish symmetric block cipher
 - ❑ Most secure version of Unix hash/salt scheme
 - ❑ A 128-bit salt and a 192-bit hash value
 - ❑ Configurable cost variables (number of iterations)

Password Cracking of User-chosen Passwords

● Traditional approaches

□ Dictionary attack

- Prepare a large dictionary of possible password and try each
- Each password must be hashed using each salt value and then compared to stored hash values
- Countermeasure: slow hash functions

□ Password crackers exploit that fact that people choose easily guessable passwords

□ Rainbow table attacks

- Pre-compute tables of hash values for all salts (a mammoth table)
- Example: using 1.4GB rainbow table to crack 99.9% of all alphanumeric Windows password hashes in 13.8s (<http://lasecwww.epfl.ch/~oechslin/publications/crypto03.pdf>)
- Countermeasure: a sufficiently large salt value and a large hash length
- Notable open-source password cracker (developed in 1996 and still in use): John the Ripper
 - A combination of brute-force and dictionary techniques

Modern Approaches for Password Cracking

- Complex password policy

- Forcing users to pick stronger passwords

- However, password-cracking techniques have also improved

- The processing capacity available for password cracking has increased dramatically
- The use of sophisticated algorithm (e.g., Markov modeling + natural language) to generate better password candidates

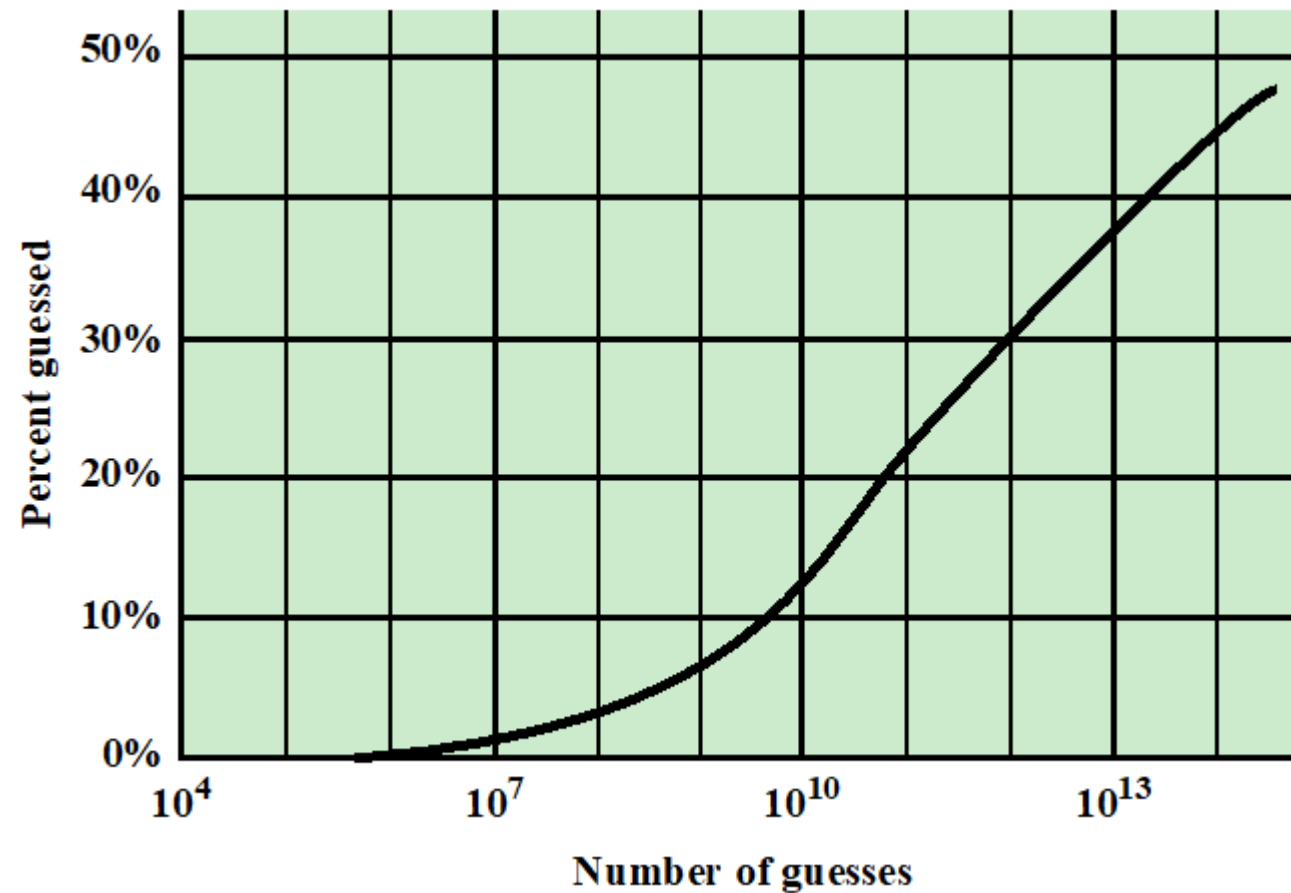
- http://www.cs.cornell.edu/~shmat/shmat_ccs05pwd.pdf

- Studying examples and structures of actual passwords in use

- Apply data mining techniques to studying public password files (leaked by security vulnerability)
- E.g., an SQL injection attack against online games, Rockyou.com

→ a data breach resulting in the exposure of over 32M plaintext passwords in 2009

Percentage of Passwords Recovered



- An analysis of the passwords used by over 25000 students at a research university with a complex password policy [1]
 - Using a database consisting of a collection of leaked password files [2]

[1] http://www.cs.umd.edu/~jkatz/security/downloads/passwords_revealed-weir.pdf

[2] https://www.cylab.cmu.edu/files/pdfs/tech_reports/CMUCyLab13013.pdf

Major Countermeasures

- Password file access control
- Password selection strategies
- Proactive password checking

Password File Access Control

● Two mechanisms

對的人才可以用shadow pswd file

- ❑ Access control: makes the password file available only to privileged users
- ❑ Shadow password file

● Vulnerabilities

- ❑ Weakness in the OS that allows access to the file
- ❑ Accident with permissions making it readable
- ❑ Users with same password on other systems
- ❑ Weakness in physical security may provide access to backup media
- ❑ Sniffing network traffic

Password Selection Strategies

- User education

- Users can be told the importance of using hard-to-guess passwords

- Computer-generated passwords

- FIPS 181 Automated Password Generator: <http://www.fips.gov>

- But, users have trouble remembering them

反應式確認

一直跑cracker，如果哪一天被他破解
他會跟你說：『誒這不行』

- Reactive password checking

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

- Complex password policy or proactive password checker

- Rejecting guessable passwords

Proactive Password Checking

主動式確認：

註冊的時候會要求規則

如果有不太OK或不太安全的密碼
會跟你說：『誒這不行』

- Rule enforcement

- ❑ Specific rules that passwords must adhere to
- ❑ e.g., must be at least eight characters long, must include at least one for each of uppercase and lowercase

- Password checker

- ❑ Compile a large dictionary of “bad” passwords not to use
- ❑ But, it is space-consuming and time-consuming

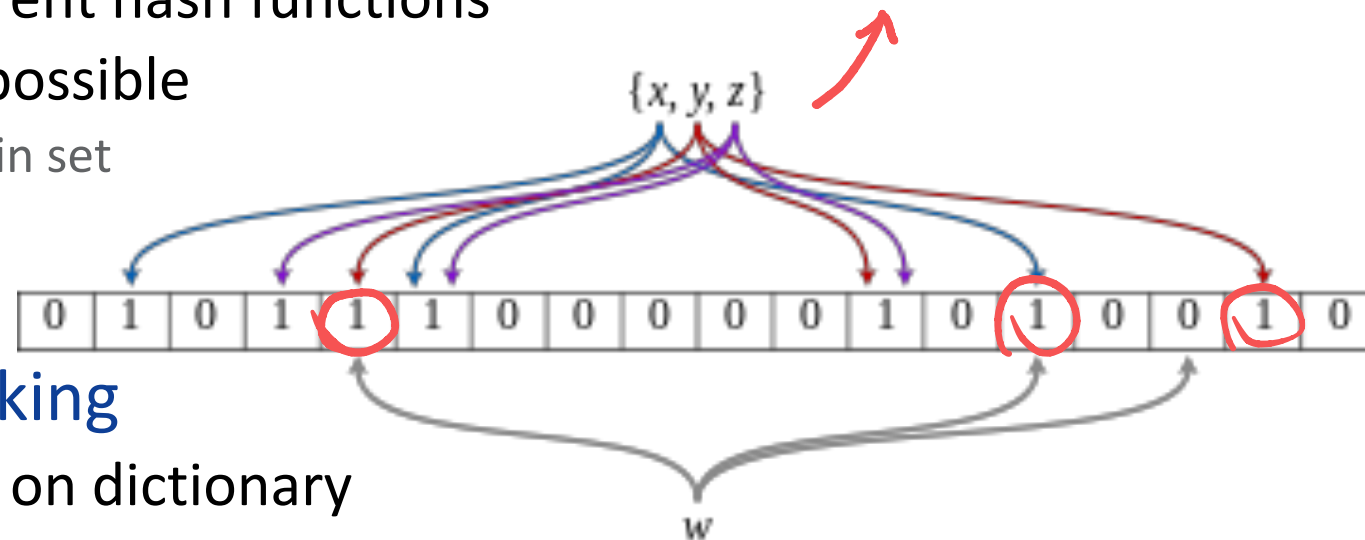
Can we use a hash function to address the issues?

Proactive Password Checking (Cont.)

- **Bloom filter: a space-efficient probabilistic data structure**

- Used to test whether an element is a member of a set
- A bit array of m bits, and k different hash functions
- But, false positive matches are possible
 - Result: possibly in set or definitely in set
- Space advantage: do not store the data items

三組不太安全的密碼
各自指到三個1



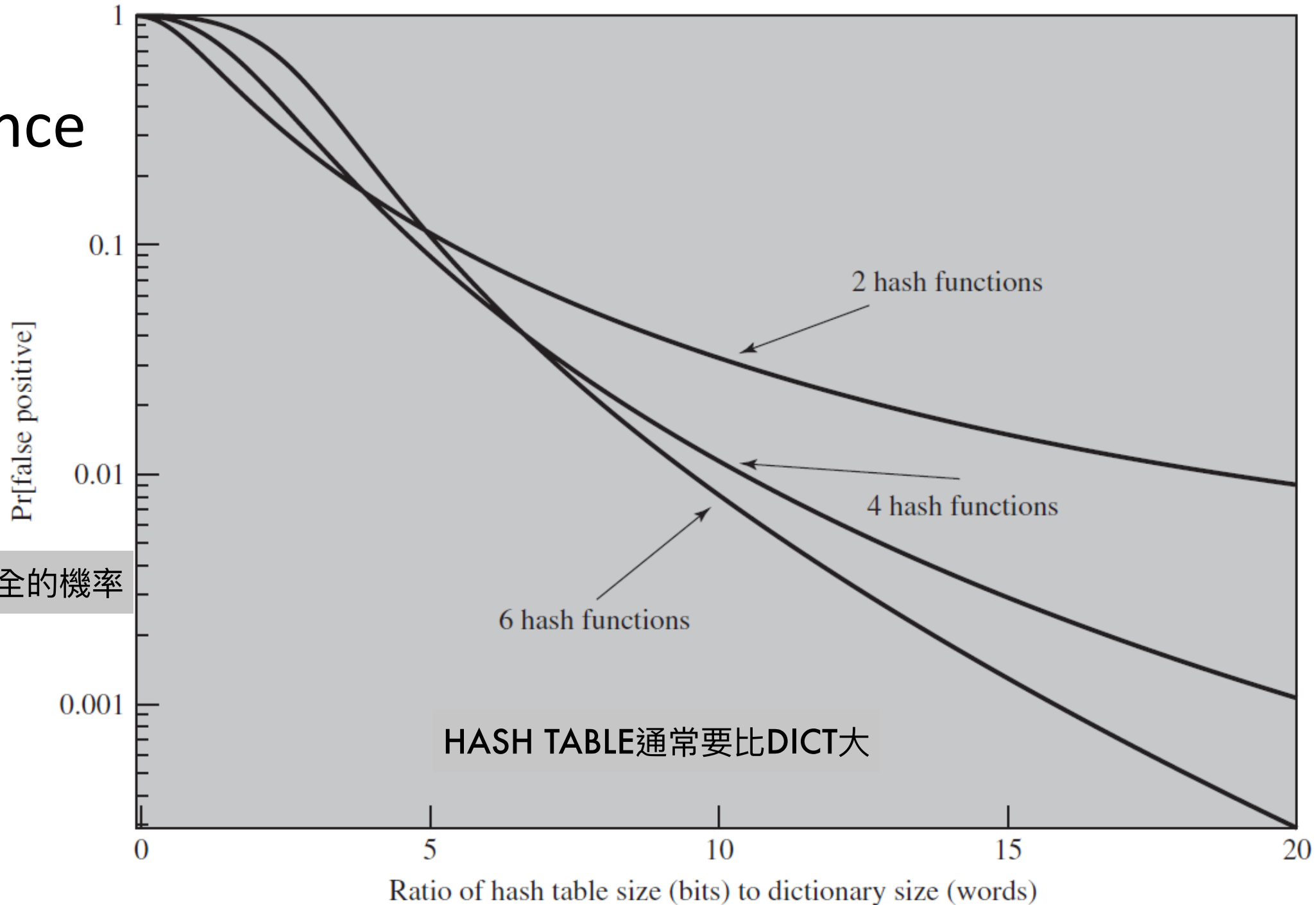
- **Applied to the password checking**

- (Traditional) build a table based on dictionary
- Check desired password against this table

$m = 18$ and $k = 3$ (From Wikipedia)

你輸入的密碼，他拿去比叫
都1：就跟你說不安全

Performance of Bloom Filter



明明安全你說我不安全的機率

HASH TABLE通常要比DICT大

Outline

- Electronic User Authentication Model
- Password-based Authentication
- Token-based Authentication
- Biometric Authentication
- Remote user Authentication
- Security Issues for User Authentication

Token-based Authentication

● Types of Cards used as Tokens

	Card Type	Defining Feature	Example
把字凸起來	Embossed	Raised characters only, on front	Old credit card
磁條	Magnetic stripe	Magnetic bar on back, characters on front	Bank card
晶片	Electronic memory	Electronic memory inside	ATM, credit cards
智慧感應 (很屌)	Smart Contact Contactless	Electronic memory and processor inside Electrical contacts exposed on surface Radio antenna embedded inside	SIM card Biometric ID card

Memory Cards

只能存



● Functions

- ❑ Can store but do not process data
- ❑ Can include an internal electronic memory

● Most common: the bank card with a magnetic stripe on the back

● Alone for physical access (e.g., Hotel room)

- ❑ Combined with a password or PIN: provides significantly greater security

● Drawbacks

- ❑ A special reader is required
- ❑ Token loss
- ❑ User dissatisfaction

Smart Tokens

還可以process data

● Categorized along four dimensions

□ Physical characteristics

- Include an embedded microprocessor
- Like a bank card: smart card
- Others: calculators, keys, small portable objects

□ User interface

- Manual interfaces include a keypad and display for interaction

□ Electronic interface

- Required by a smart card or other token to communicate with a compatible reader/writer
- Contact: direct contact between a card reader and a conductive contact plate on the card
- Contactless: both the reader and the card have an antenna



Smart Tokens (Cont.)

□ Authentication protocol

■ Static

- User authenticates himself or herself to the token
- Token authenticates the user to the computer

TOKEN過去，可以就是可以

■ Dynamic password generator

- Token generates a unique password periodically (e.g., every minute)
- Initialized and synchronized for the token and the computer

TOKEN裡面會週期性產生密碼（一直變）

■ Challenge-response

- Computer generates a challenge
- Token generates a response to the challenge

PC給一個挑戰，OK的TOKEN出來的回應就會OK

Most Important: Smart Cards

卡要變成電腦囉！！！！

- Contains an entire microprocessor

- Including processor, memory, and I/O ports
- (optional) a special co-processing circuit for cryptographic operation

- Three types of memory

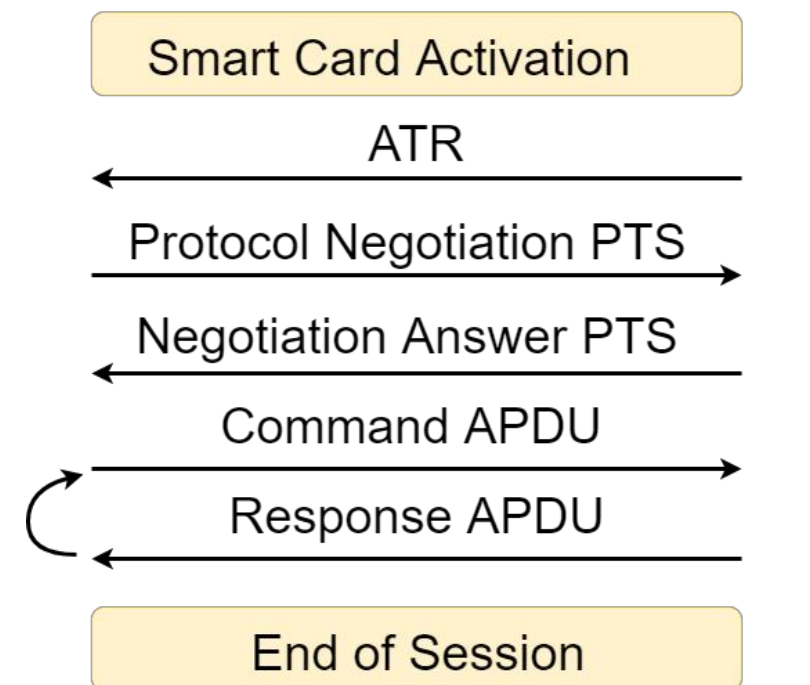
- Read-only memory (ROM)
- Electrically erasable programmable ROM (EEPROM)
- Random access memory (RAM)



Smart card



Card reader



APDU = Application Protocol Data Unit

ATR = Answer To Reset

PTS = Protocol Type Selection

Smart Cards: Electronic Identity (eID) Cards

German

- Verified by the national government as valid and authentic
 - ❑ Most advanced eID: German eID card
- Three eID functions
 - ❑ ePass: government use; offline (e.g., electronic passport)
 - Stores a digital representation of the identity (e.g., face and fingerprint images)
 - ❑ eID: general-purpose use; offline and online
 - Stores an identity record (e.g., name, date of birth, address)
 - ❑ eSign: generating a digital signature
 - Stores a private key and a certificate verifying the key (e.g., X.509 certificate)

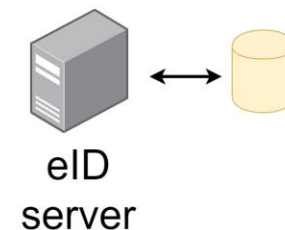


Taiwan

Online User Authentication with the eID Function

6. User enter PIN
- 
1. User requests service (e.g., via Web browser)

4. Authentication request
5. PIN request
7. Authentication protocol exchange
8. Authentication result for redirect



2. Service request
3. Redirect to eID message
3. Redirect to eID message
10. Service granted

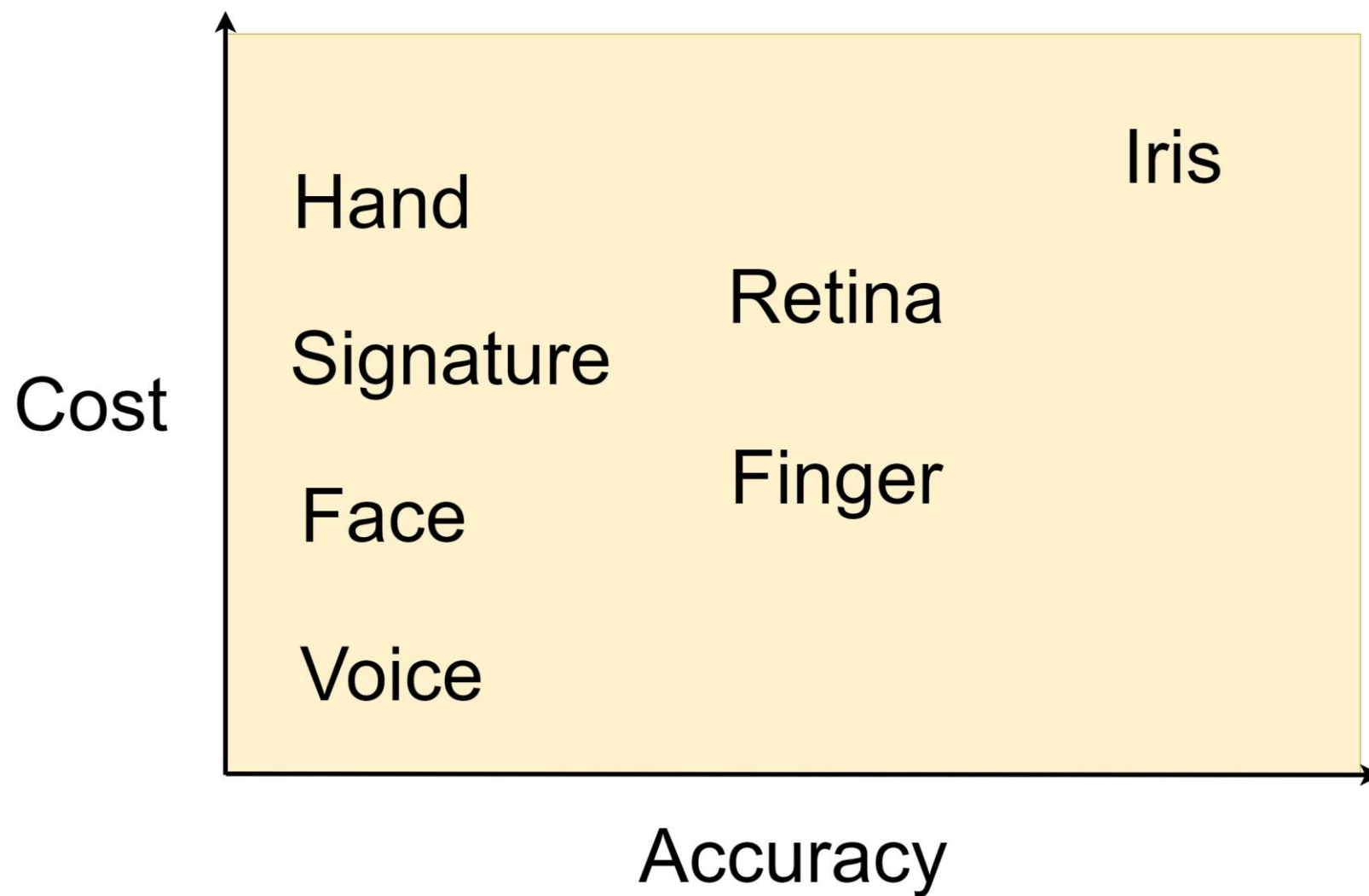


Biometric Authentication

- Authentication based on unique physical characteristics
 - ❑ Static: facial characteristics, fingerprints, hand geometry
 - ❑ Dynamic: signature, voice
- Relies on pattern recognition technologies
 - ❑ More complex and expensive than passwords and tokens
 - ❑ Not yet to mature as a standard tool



Cost vs. Accuracy of Various Biometric Characteristics



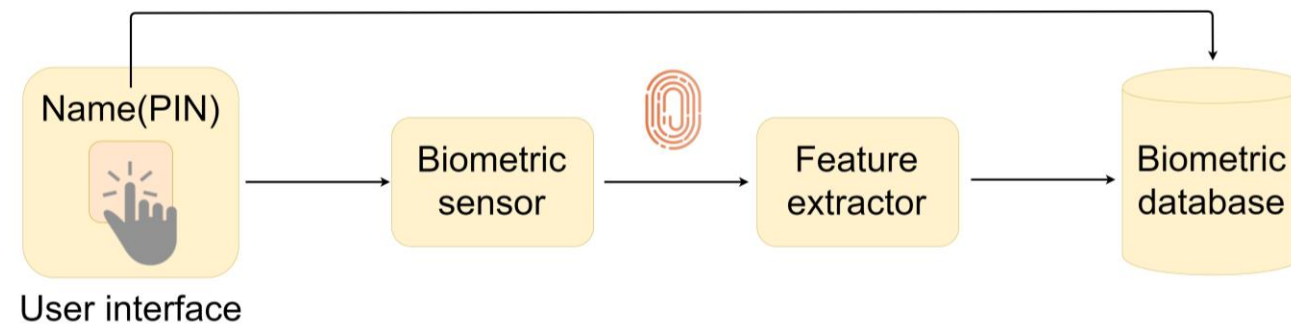
Biometric Auth System Operation

他知道你是WATERSO，他就去找WATERSO所在的地方，看你的指紋有沒有符合

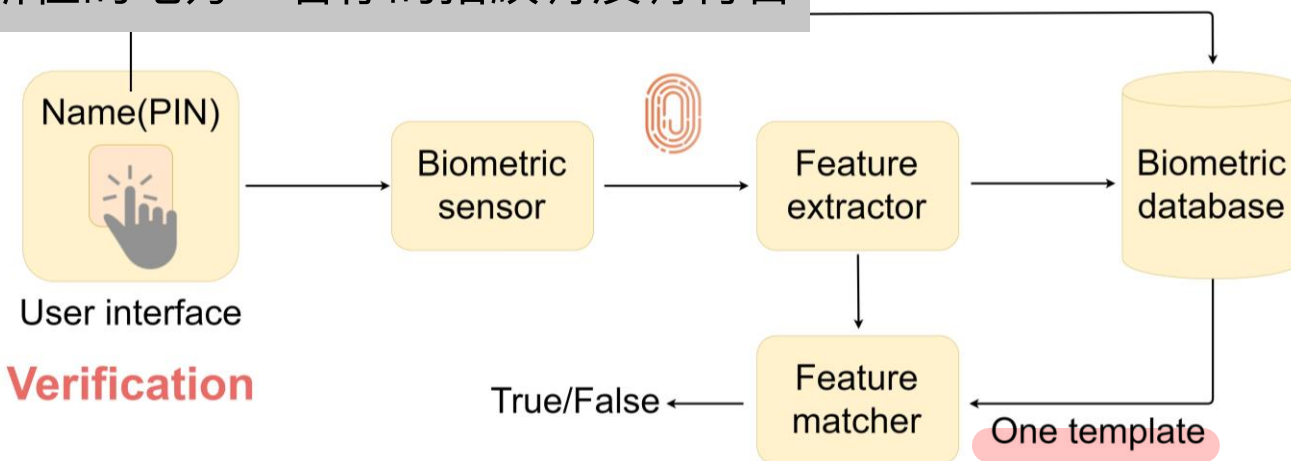
- **Verification**: analogous to a user logging on to a system by using a smart card and a PIN

- **Identification**: user presents biometric info without other info; system compares it with stored templates

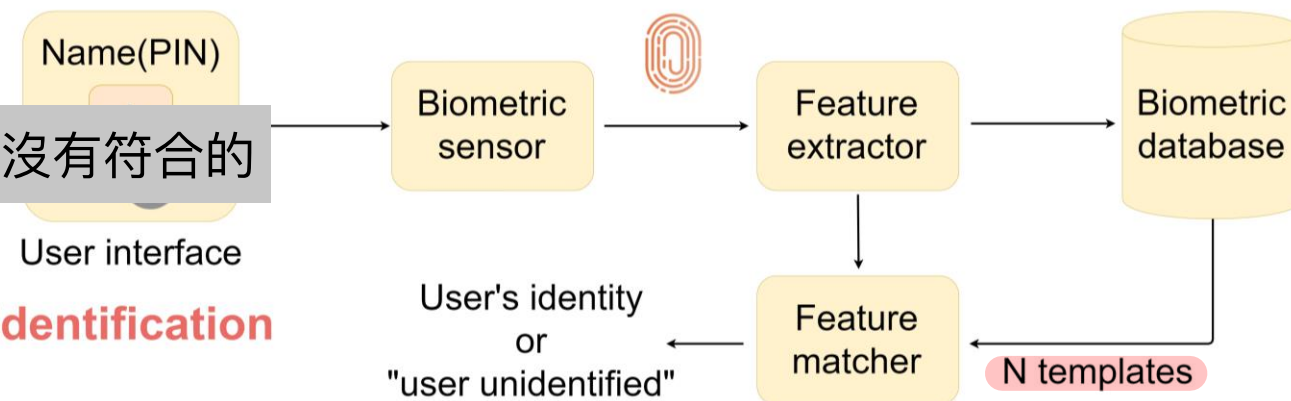
他只拿到你的指紋，掃過整個DATABASE，看有沒有符合的



Enrollment



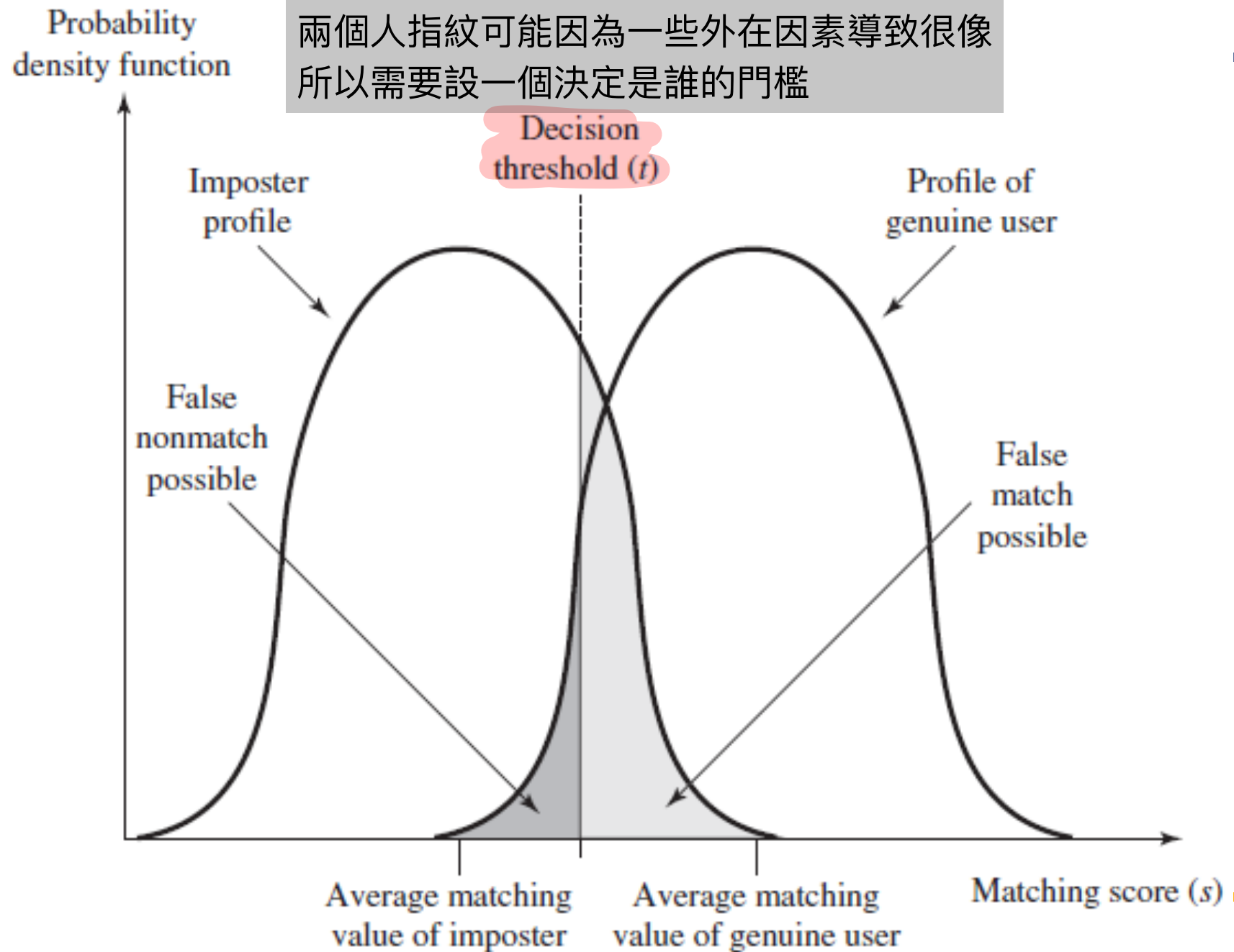
Verification



Identification

Profiles of a Biometric Characteristic of an Imposter and an Authorized User

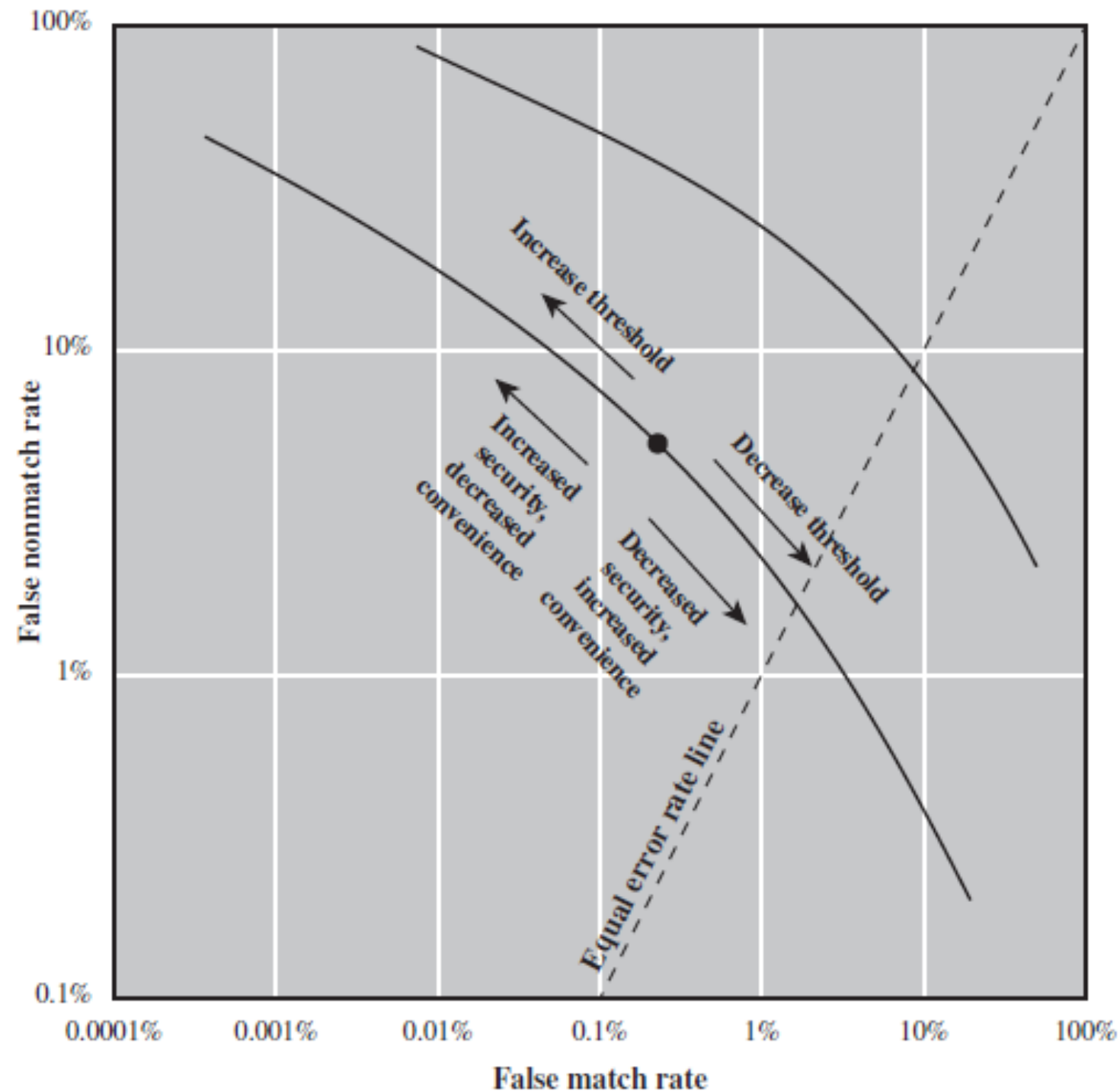
- Dilemma: matching score would vary for a single user
 - e.g., fingerprint: due to sensor noise



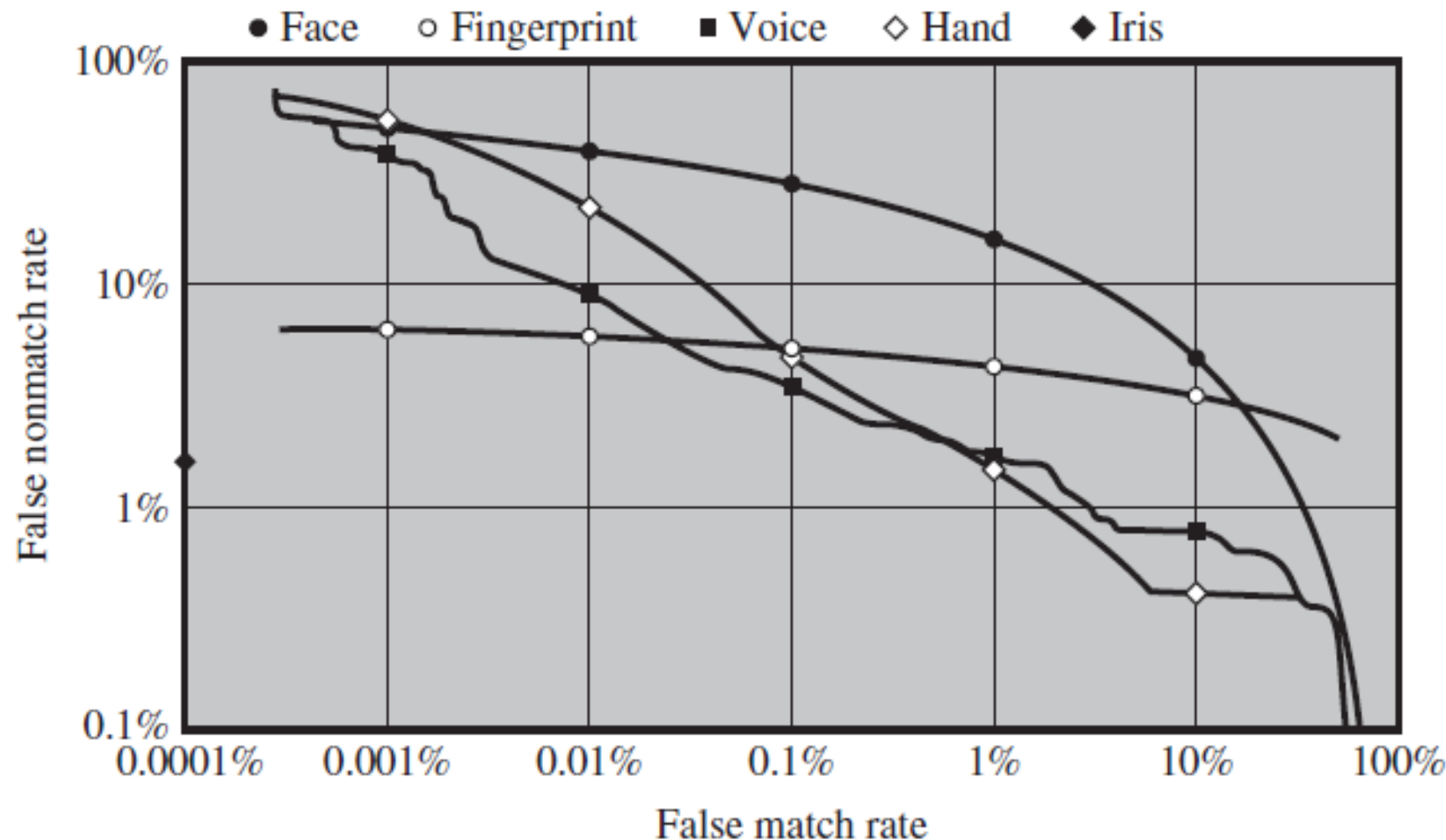
Idealized Biometric Measurement Operating Characteristic Curves (log-log scale)

- Tradeoff between security and convenience

□ Inconvenience: a valid user is denied access



Actual Biometric Measurement Operating Characteristic Curves (log-log scale)

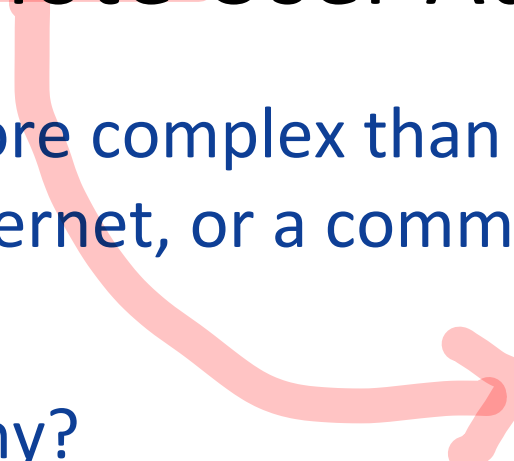


Outline

- Electronic User Authentication Model
- Password-based Authentication
- Token-based Authentication
- Biometric Authentication
- Remote user Authentication
- Security Issues for User Authentication

Remote User Authentication

- More complex than local authentication: over a network, the Internet, or a communication link
- Why?
 - More security threats: eavesdropping, capturing a password, and replaying an authentication sequence that has been observed
- General solution: challenge-response protocols



用網路就有通道
想傳就會被攔就很北藍

Protocols for a Password and a Token

Password

Client	Transmission	Host
U , user	$U \rightarrow$	
	$\leftarrow \{r, h(), f()\}$	random number $h(), f()$, functions
P' password r' , return of r	$f(r', h(P')) \rightarrow$	
	\leftarrow yes/no	if $f(r', h(P')) = f(r, h(P(U)))$ then yes else no

Token

Client	Transmission	Host
U , user	$U \rightarrow$	
	$\leftarrow \{r, h(), f()\}$	r , random number $h(), f()$, functions
$P' \rightarrow W'$ password to passcode via token r' , return of r	$f(r', h(W')) \rightarrow$	
	\leftarrow yes/no	if $f(r', h(W')) = f(r, h(W(U)))$ then yes else no

Protocols for Static and Dynamic Biometric

Static

Client	Transmission	Host
U , user	$U \rightarrow$	
	$\leftarrow \{r, E()\}$	r , random number $E()$, functions
$B' \rightarrow BT'$ biometric D' biometric device r' , return of r	$E(r', D', BT') \rightarrow$	$E^{-1}E(r', P', BT') =$ (r', P', BT')
	$\leftarrow \text{yes/no}$	if $r' = r$ and $D' = D$ and $BT' = BT(U)$ then yes else no

Dynamic

Client	Transmission	Host
U , user	$U \rightarrow$	
	$\leftarrow \{r, x, E()\}$	r , random number x , random sequence challenge $E()$, function
$B', x' \rightarrow BS'(x')$ r' , return of r	$E(r', BS'(x')) \rightarrow$	$E^{-1}E(r', BS'(x')) =$ $(r', BS'(x'))$ extract B' from $BS'(x')$
	$\leftarrow \text{yes/no}$	if $r' = r$ and $x' = x$ and $B' = B(U)$ then yes else no

Security Issues for User Authentication

- **Client attacks: masquerade as a legitimate user**

假裝自己是合法的USER

- ❑ Guessing, exhaustive search, and false match
- ❑ Countermeasures: strong passwords and limited attempts

- **Host attacks: steals the user file where passwords, token passcodes, or biometric templates are stored**

偷USER FILE

- ❑ Theft of plaintext, passcode, and template
- ❑ Countermeasures: strong access control

- **Eavesdropping**

攻擊界的作弊（偷看答案）

- ❑ Shoulder surfing, keystroke logging, copying biometric
- ❑ Countermeasures: multifactor authentication and anomaly detection

Security Issues for User Authentication (Cont.)

登入上一個登入的人

- Relay: repeats a previously captured user response

- Replays stolen password, passcode, and biometric template
- Countermeasures: a random number in challenge-response protocols

用隨機產生驗證碼

- Trojan horse: installation of rogue client or capture device

- e.g., rogue ATM or credit card scanner
- Countermeasures: authentication of client or capture device within trusted security perimeter

讓你以為是官方，你就傻傻給他密碼，她就爽爽登入假裝是你

- Denial of service: lockout by multiple failed authentications

- Countermeasures: multifactor with token

一直登入錯的讓它爆掉

Questions?