Cybersecurity

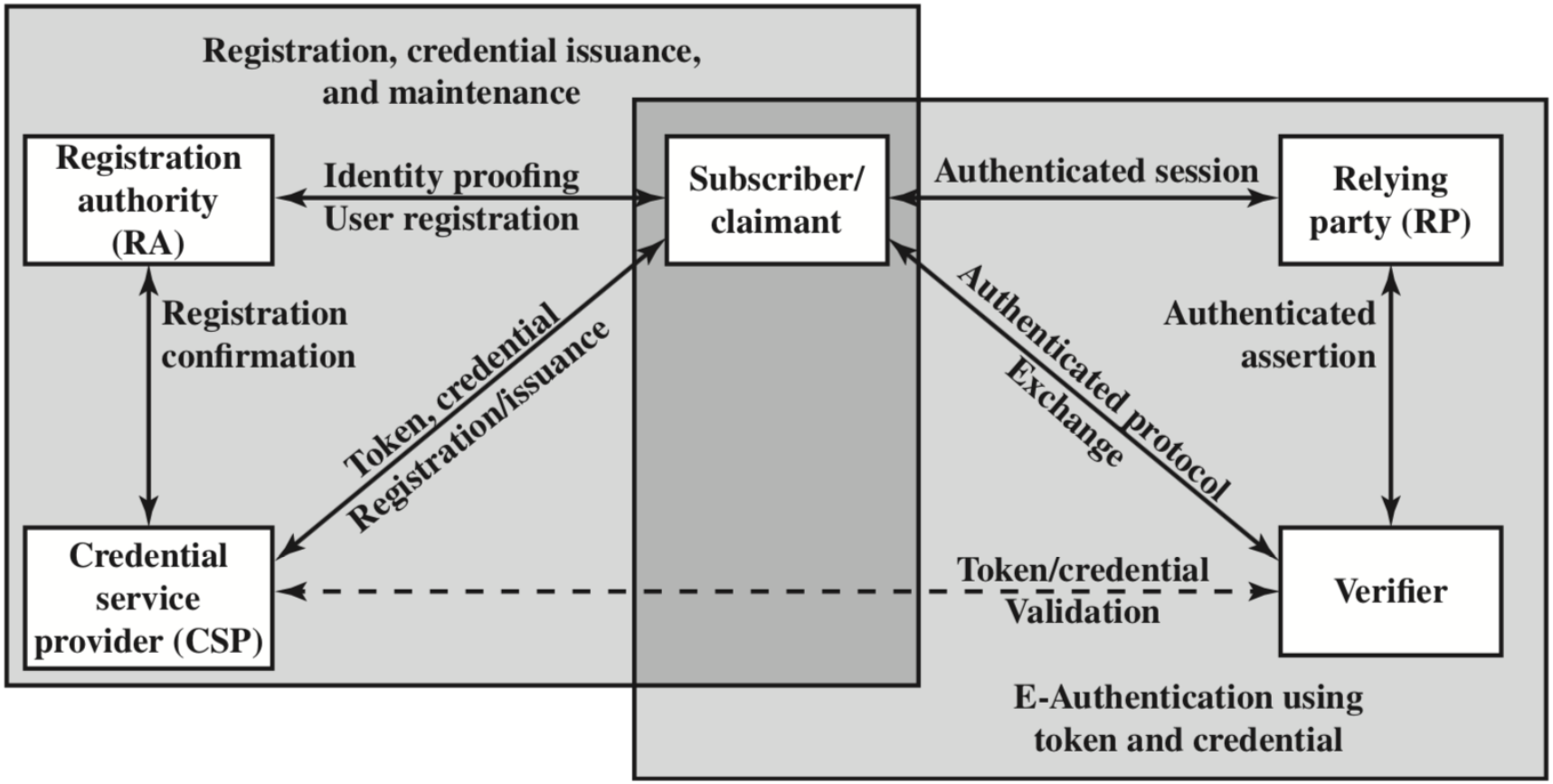
User authentication

Mauro Barni
University of Siena

Most crucial building block

- User authentication lies at the heart of virtually any secure system
- Provides the basis for (but it is distinct from)
  - Access control
  - Accountability
  - Traceability
- Two **distinct** functions
  - identification
  - verification

NISR model for user authentication



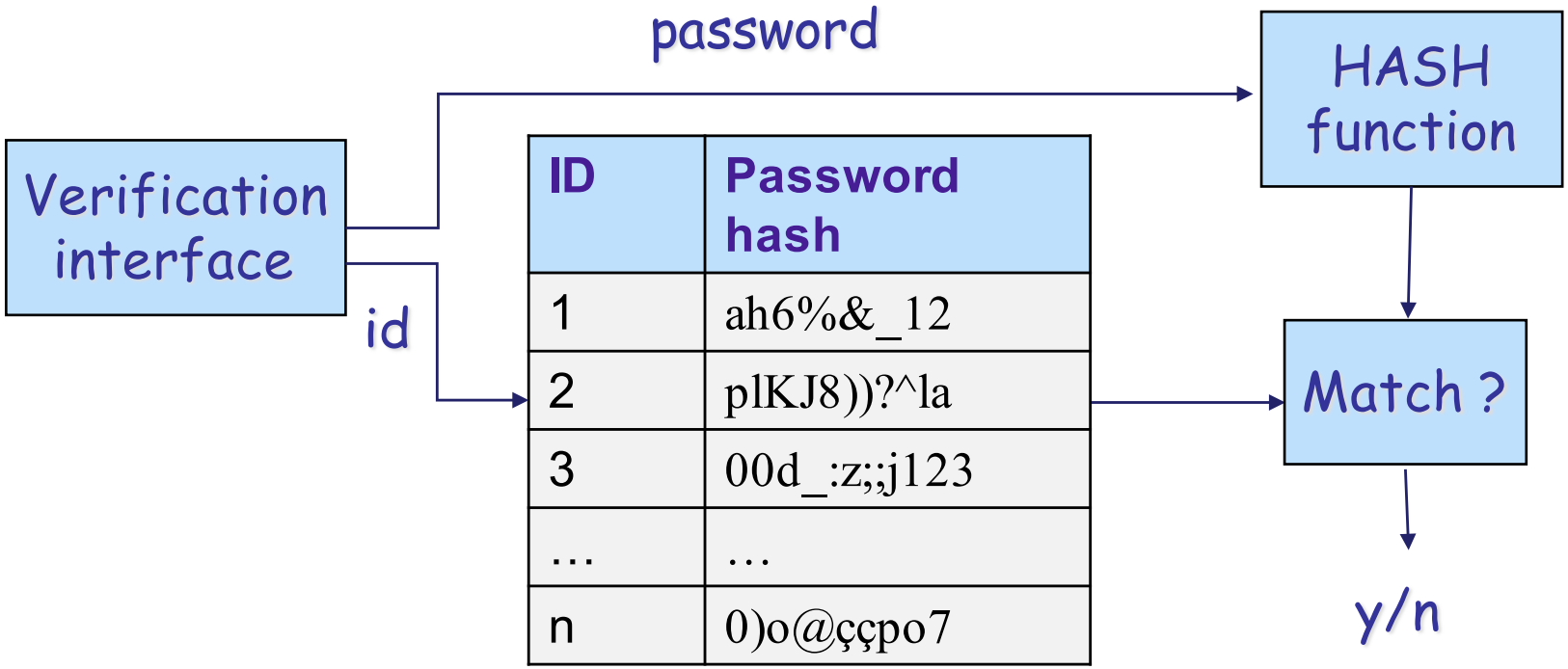
Authentication means

- **Something you know**
  - ID, PIN, passwords, answers to questions
- **Something you have**
  - memory card, smart card, token, electronic keycards
- **Something you are**
  - biometrics
- **Something you do**
  - behavioural biometrics (signatures, gestures)

- **Multifactor authentication**

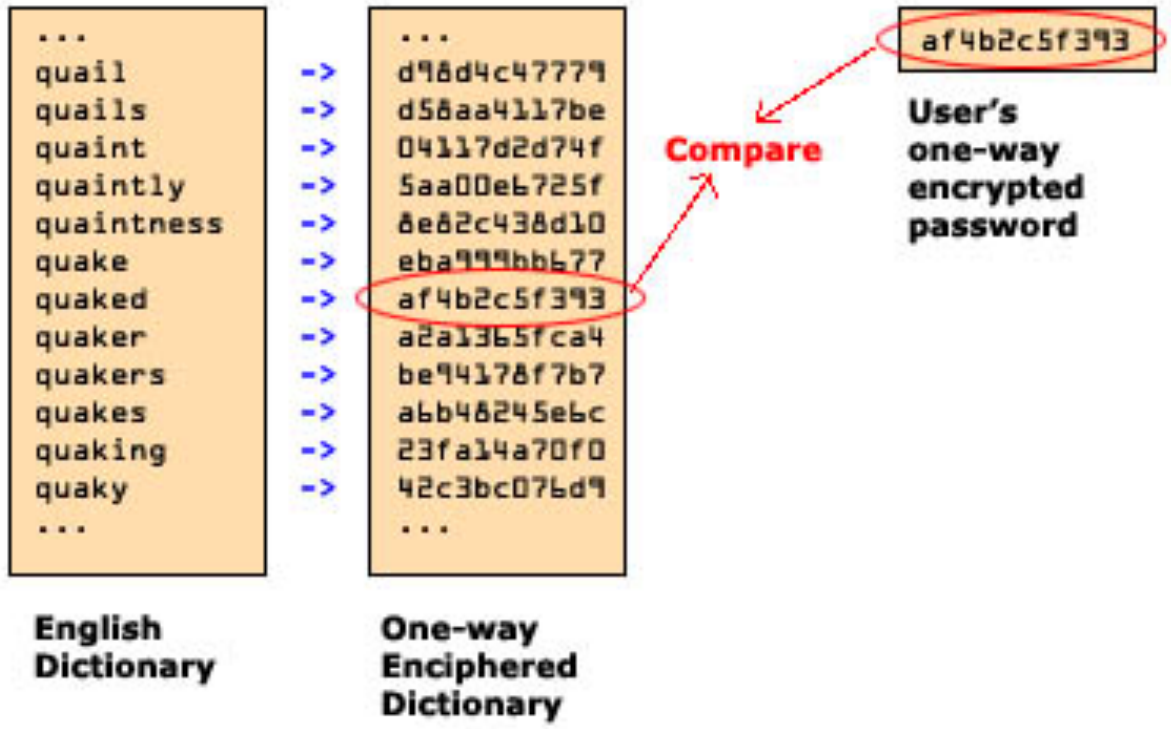
Password-based authentication

- Systems maintain an ID-password file
- Only password hashes are stored



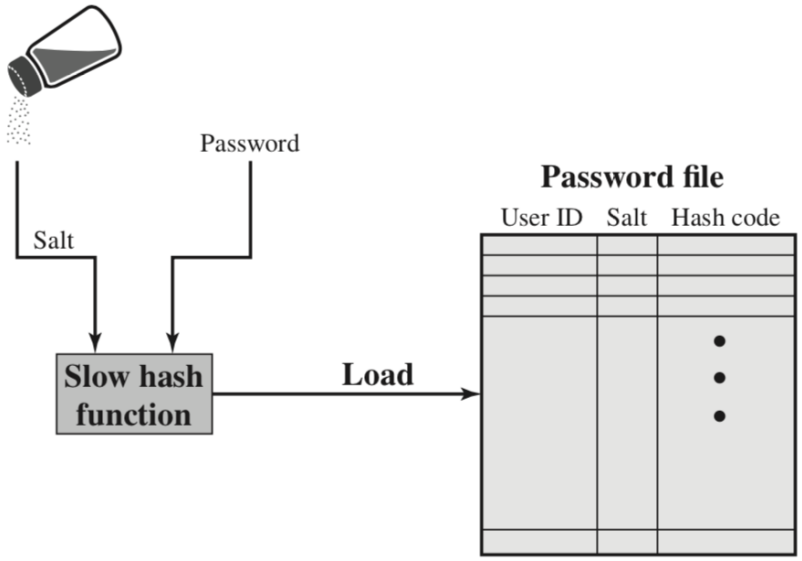
Attacks against password systems

- **Offline dictionary attack**
- If the password file is hacked (always possible) passwords are at risk
- Attack starts with more likely (weak) passwords

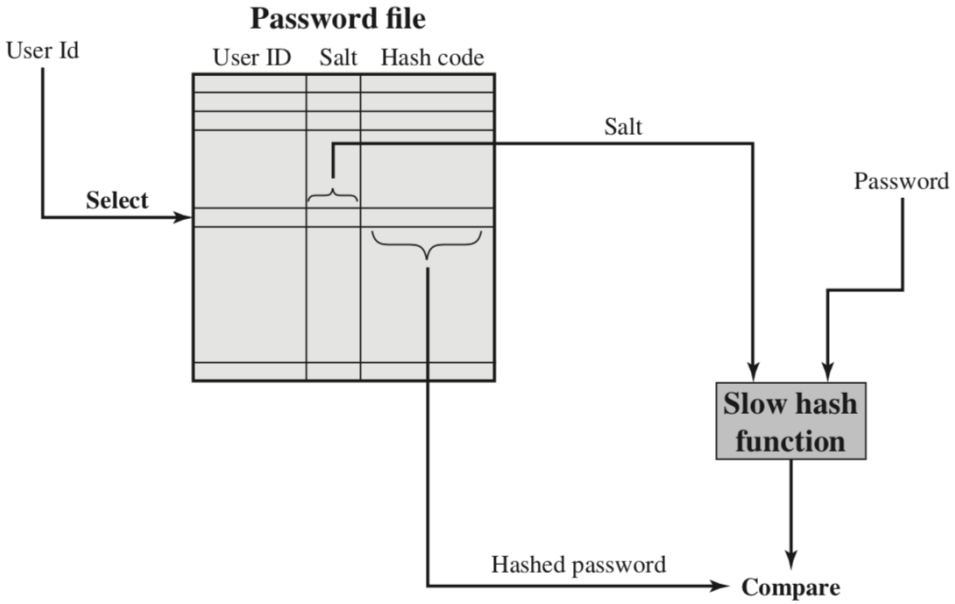


Randomized hashes

- **Use of password and salt values to compute hashes**
  - Used by UNIX



Password storage



Verification

Randomized hashes

- **Randomization serves three purposes**
  - It prevents duplicate passwords from being recognized
  - It greatly increases the complexity of dictionary attacks
  - It becomes nearly impossible to detect if a person has used the same password on multiple systems

Attacks against password systems

- **Rainbow tables**
  - By using 1.4 GB of data, 99.9% of Windows password hashes were guessed in 13.8 seconds
- **Exploit tendency towards short and easy password**
  - Try user's name, initials, account name, under several permutations
  - Try words from various dictionaries.
  - Try permutations on the words from previous step.
  - Try various capitalization permutations

- **40% of passwords (UNIX) guessed in < 1 hour**

Attacks against password systems

- **Attacks targeting a specific user**
  - The attacker targets a specific account and submits password guesses until the correct password is discovered.
  - Use of lockout to avoid it
- **Use of popular passwords**
- **Users errors**
- **Multiple password use**

- **New trend: build statistical or data driven models of users' generate passwords**

Defenses: prevent access to password file

- **Separate ID file and file with password hashes**
- Useful and good, but we can not rely on this kind of protection only
  - Unexpected software vulnerabilities
  - Multiple use of the same passwords
  - Physical attacks: emergency disks, back-ups, boot with different operating systems …
  - Password sniffing

Defenses: password selection strategies

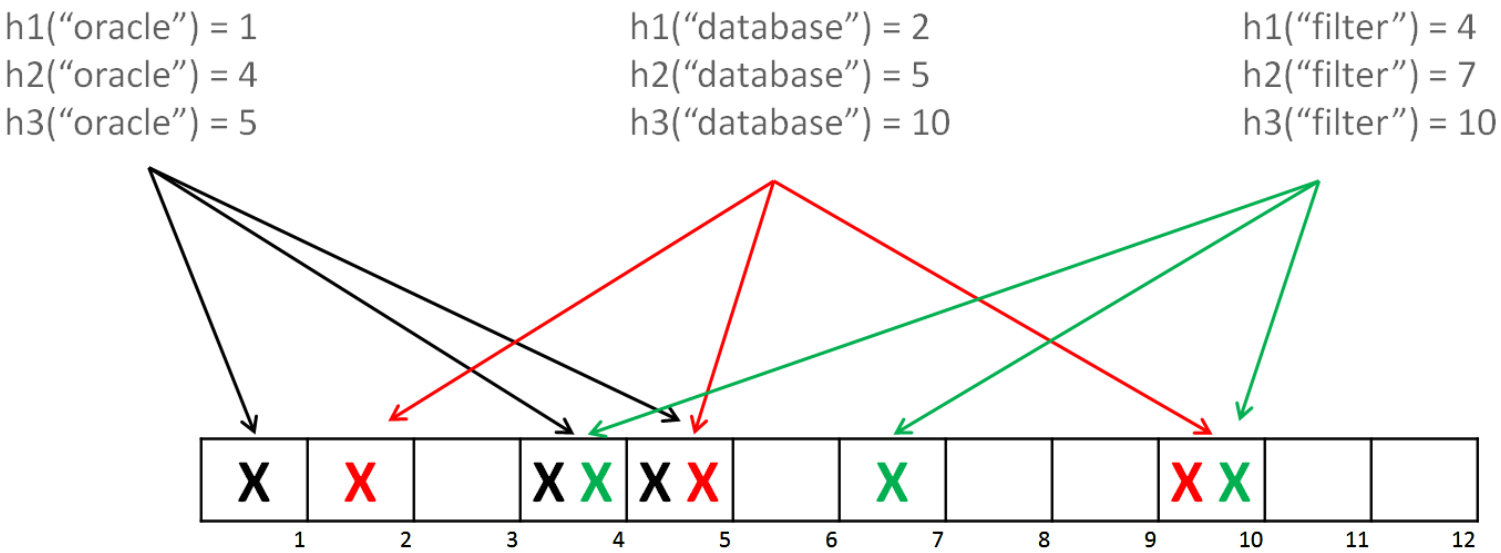
- User education
  - many ignore recommendations
  - many are not capable to judge
  - an easy to remember trick (initial letters of sentence)
- Computer-generated password
  - Difficult to memorize, not accepted
- Reactive password checking
  - Expensive
- Proactive password checker
  - Tradeoff between complexity and acceptance

Rule enforcement

- NIST SP 800-63-2 suggests the following rules:
  - Password must have at least sixteen characters (basic16)
  - Password must have at least eight characters including an uppercase and lowercase letter, a symbol, and a digit. It may not contain a dictionary word (comprehensive8).
- Password checker
  - Build a dictionary with bad passwords and check that password chosen by the users are not contained in it
  - Time and space complexity

Rule enforcement with Bloom filter

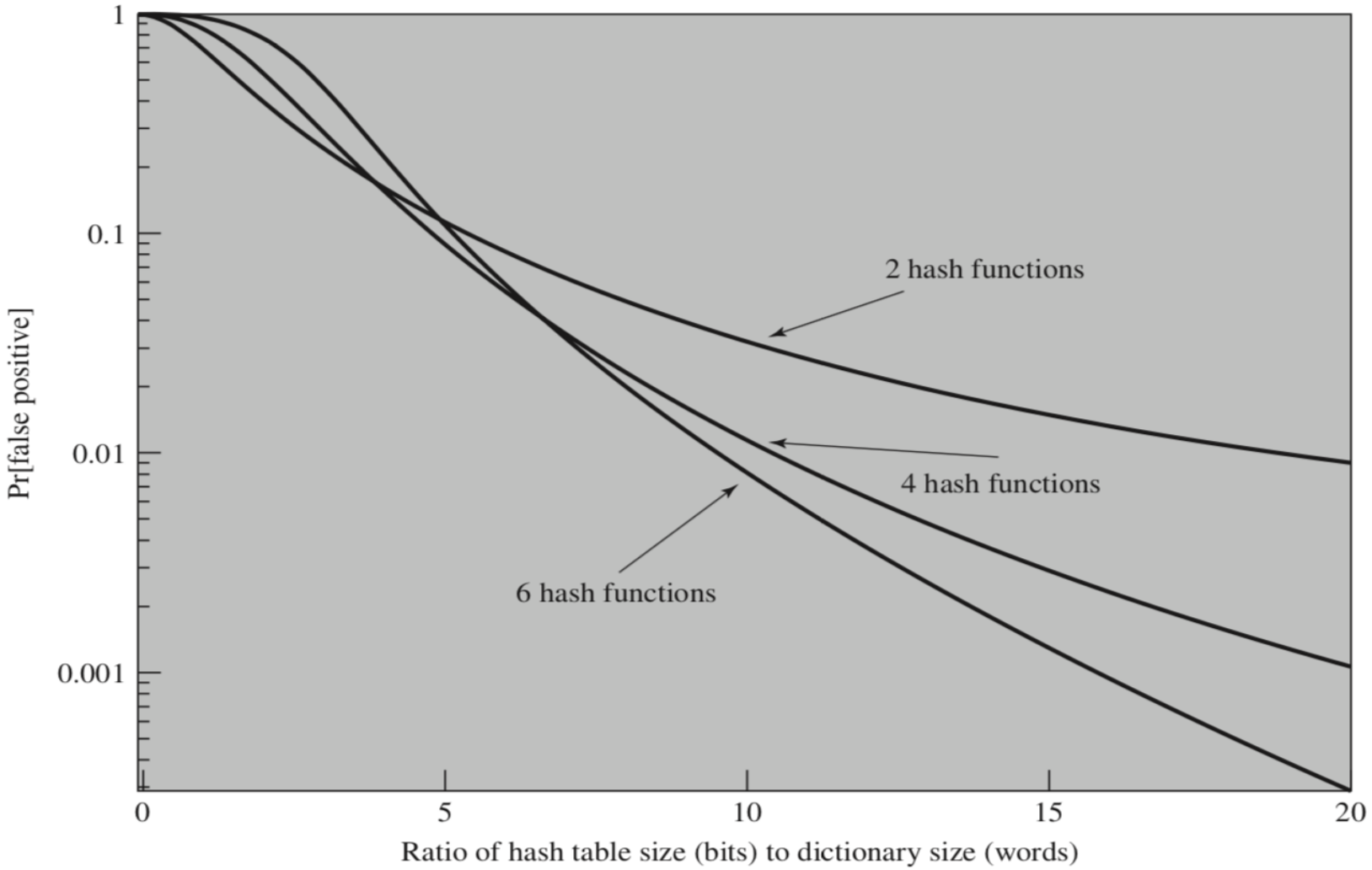
- For each password apply K hash functions: $H_1(x)$ … $H_k(x)$
- Initialize a hash table with N entries ($H(x) \in [0, N-1]$)
- If $H_j(x_i) = m$ for any i,j, let $T(m) = 1$



Rule enforcement with Bloom filter

- If user chooses a password y such that $T(H_i(y)) = 1$ for all i, then password is rejected
- False rejections are possible
  - tradeoff between complexity and false alarm probability

Rule enforcement with Bloom filter



Token-based authentication

- **Memory cards**
  - Can store but not process data
  - Bank cards, hotel room keys
  - Often used in conjunction with passwd
- **Smart tokens (cards)**
  - Can store and process data

Smart tokens classification

- **Physical characteristic**
  - Smart tokens include an embedded microprocessor. A smart token that looks like a bank card is called a smart card. Other smart tokens can look like calculators, keys, or other small portable objects
- **User interface**
  - Keypad, display, buttons
- **Electronic interface**
  - contact, contactless
- **Authentication protocol**

Authentication protocols

- **Static**
  - With a static protocol, the user authenticates himself or herself to the token then the token authenticates the user to the computer.
- **Dynamic password generation**
  - Once common for internet banking
- **Challenge response**
  - the computer system generates a challenge and the smart token generates a response based on the challenge. Example: pubkey crypto

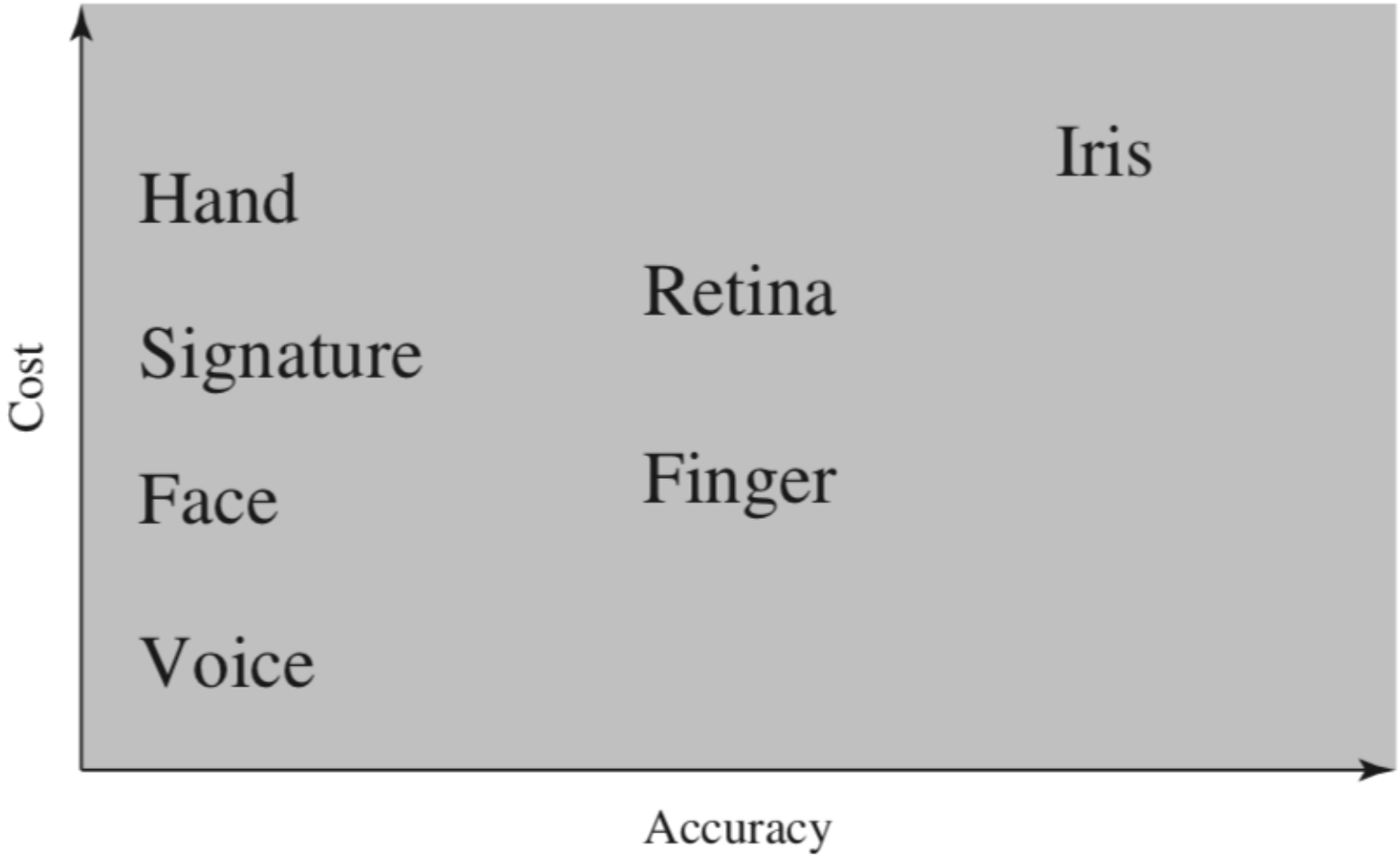
Biometric authentication

- **Based on a biometric trait of the user**
- **The biometric trait must be**
  - Universal
  - Unique (discriminating power)
  - Permanent (space, time, age …)
  - Difficult to spoof
  - Easy to measure (non intrusive)
  - Cheap
  - Non-sensitive

Common (and less common) traits

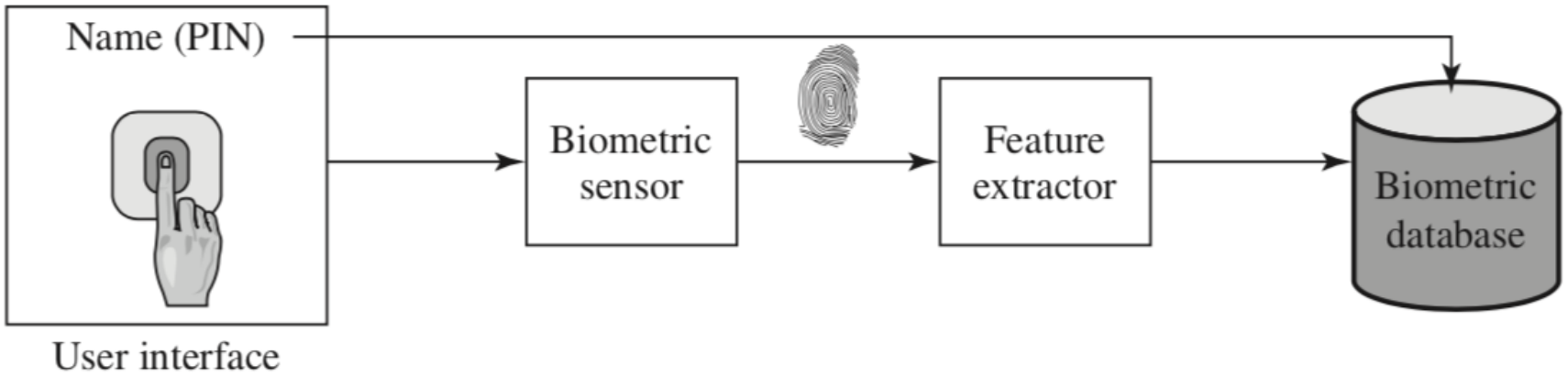
- Face, facial characteristics
- Fingerprints
- Iris
- Retinal pattern
- Hand, ear, foot geometry
- Veins
- ECG, EEG
- Voice
- Signature
- Gait
- ….

Common (and less common) traits



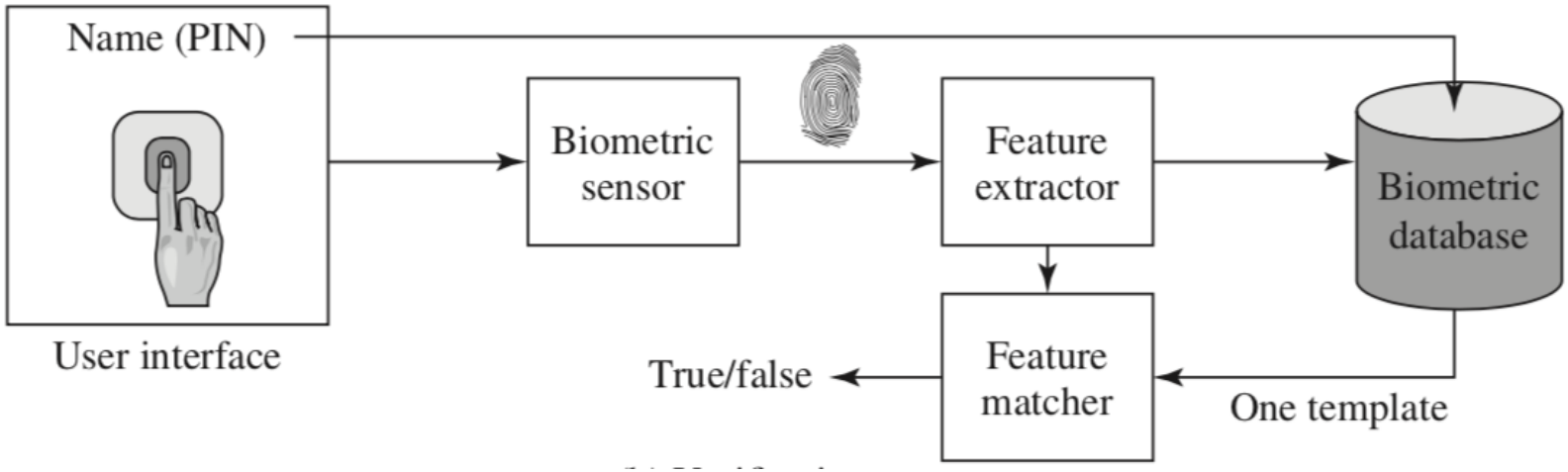
Enrollment phase

- Enrollment may or may not require physical presence of an enrolling agent
- Single or multiple acquisitions to cope with lack of stability



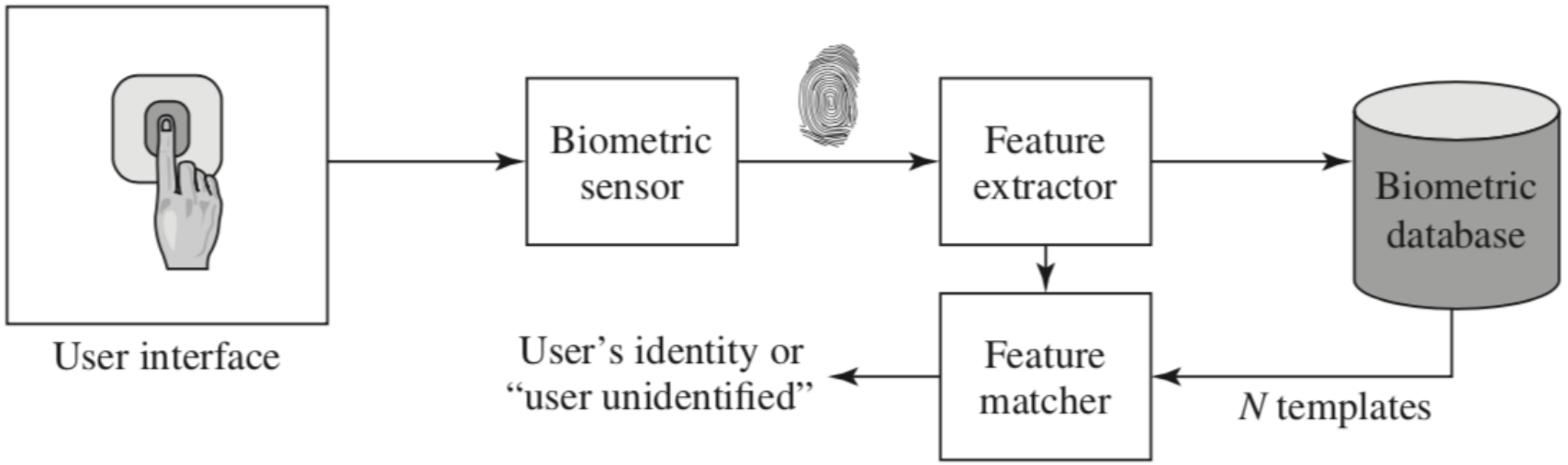
Verification

- A verification protocol verifies that the user is who he/she claims to be
- Most common situation



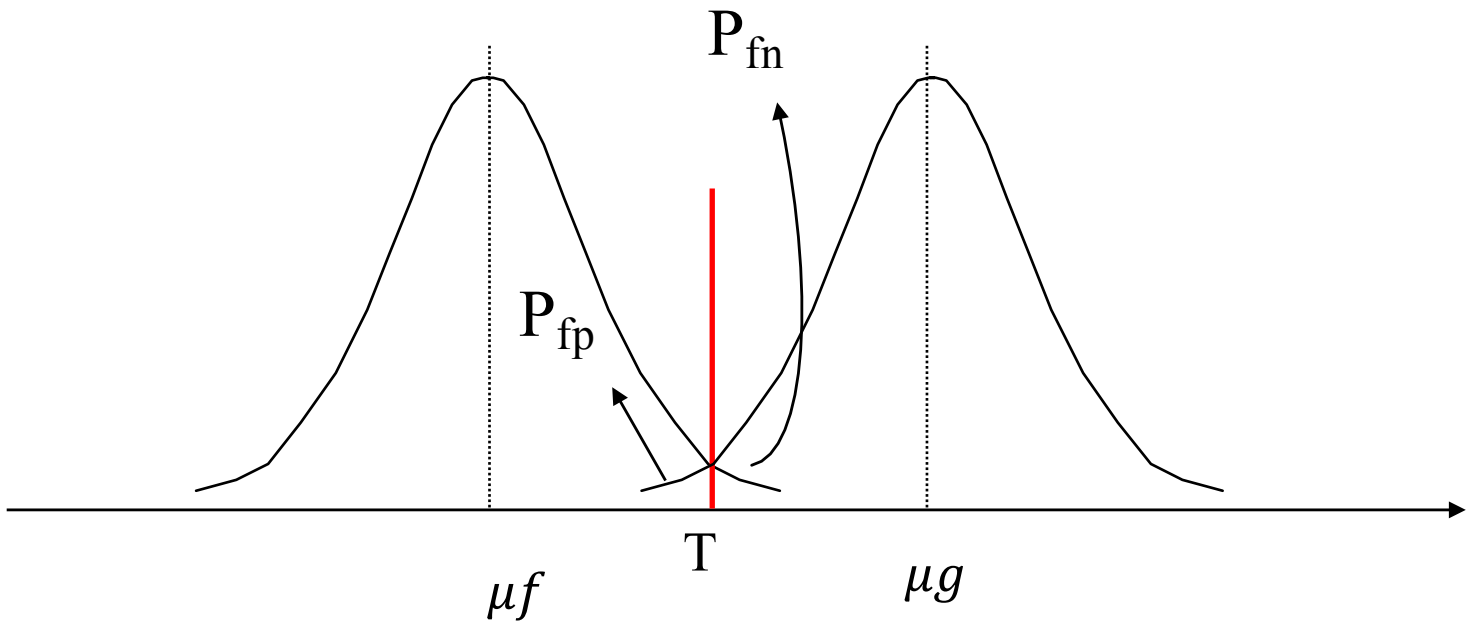
Identification

- An identification protocol must decide if the user is among the enrolled users, OR it identifies who the user is
- Collisions are more problematic than for verification



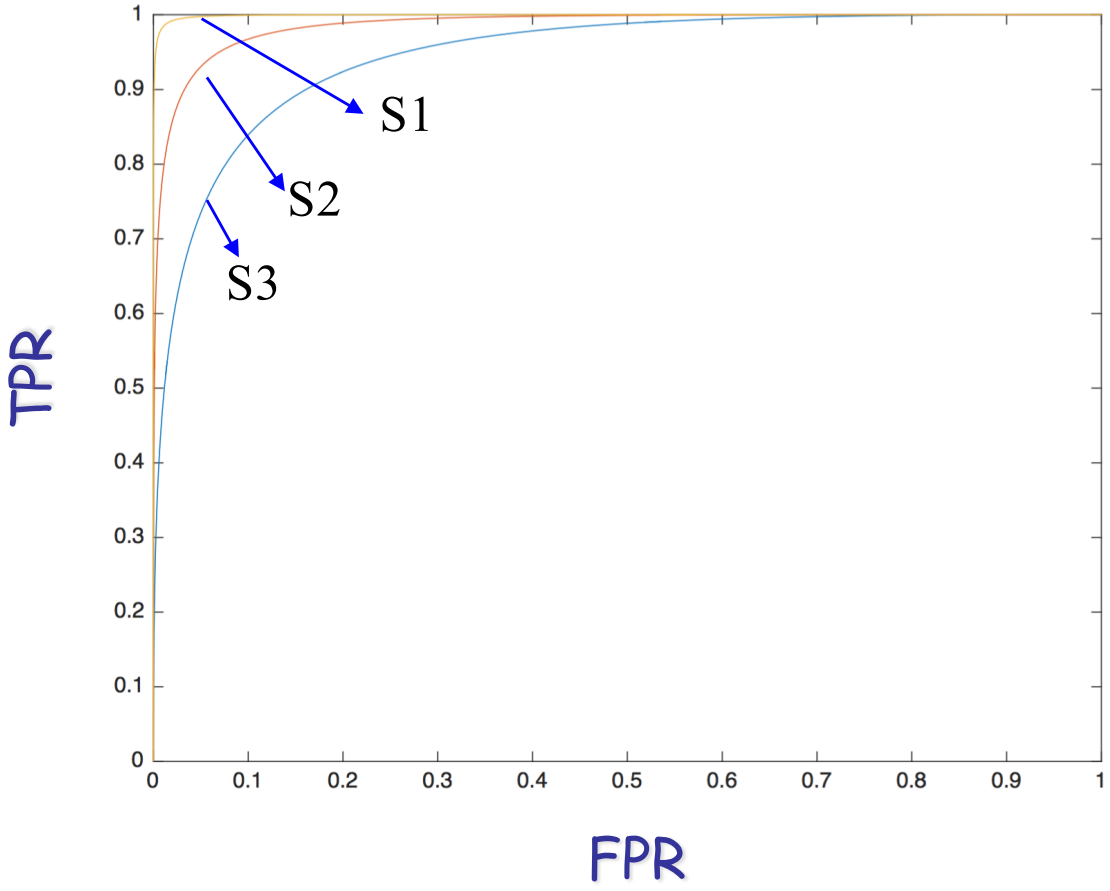
Dealing with errors

- The inexact nature of the acquisition and matching processes causes **unavoidable** errors
- Two types of errors possible: **false positive and false negative**



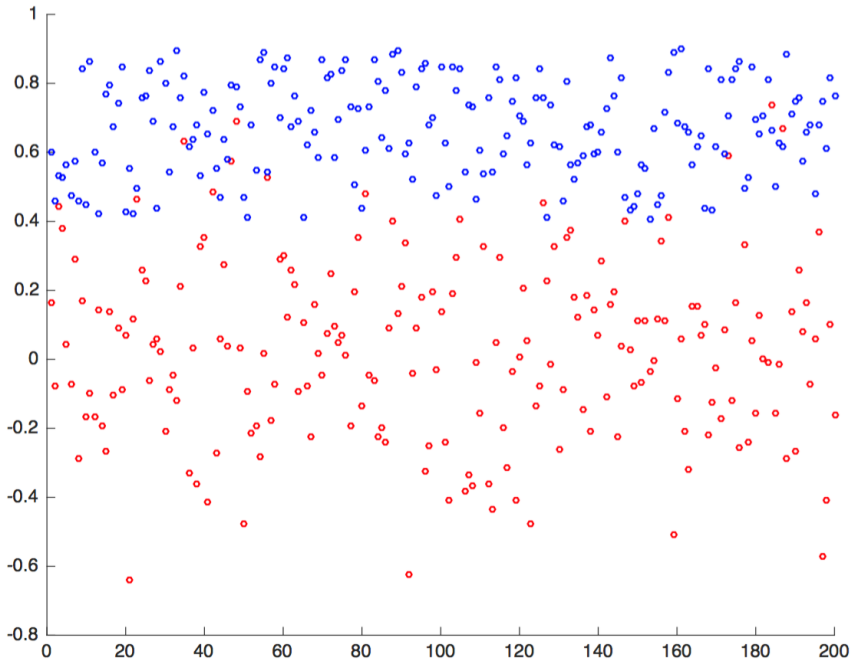
Dealing with errors

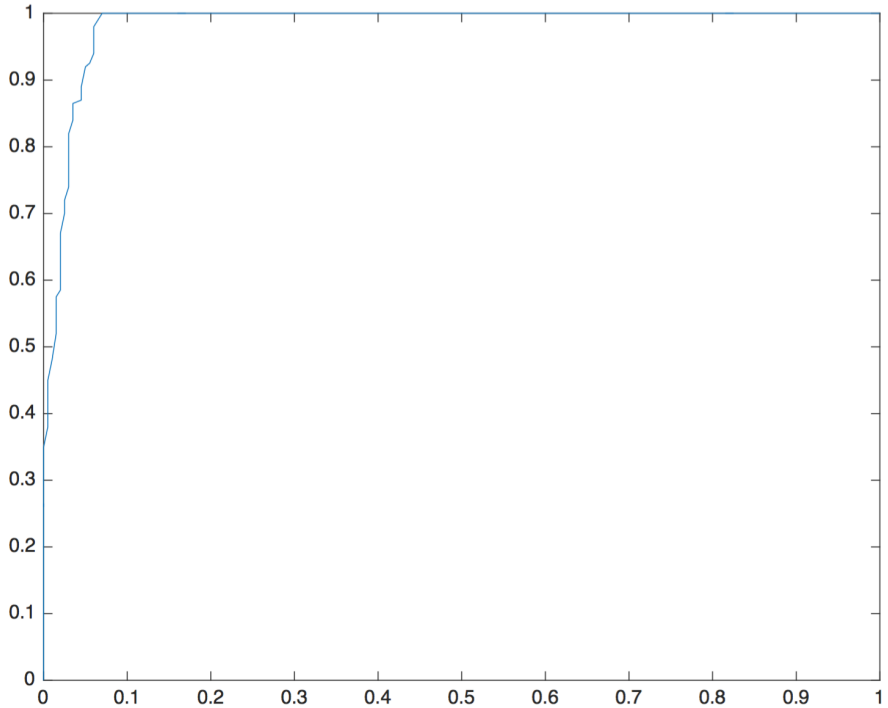
If the distributions of the match-score under the two hypothesis is known the tradeoff between FPR and FNR can be measured exactly: OC curve



Dealing with errors

- Scatterplots and empirical OC curves may help when an exact statistical model is not available
- Operating point is determined by looking at the OC curve

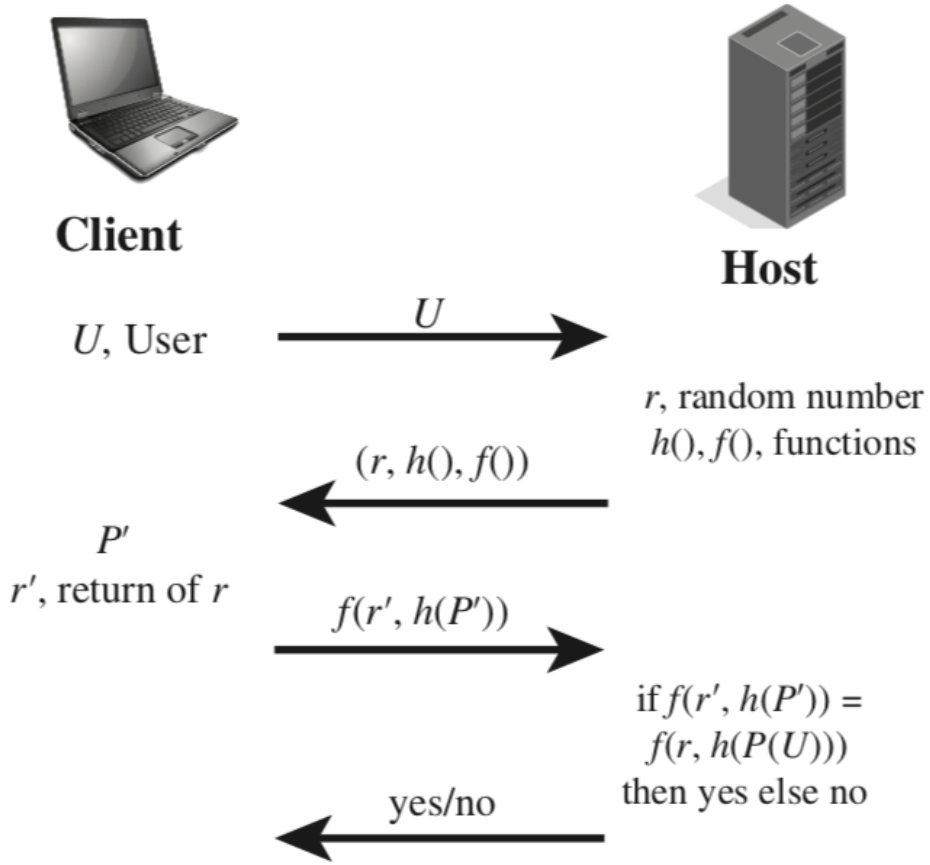




Remote authentication

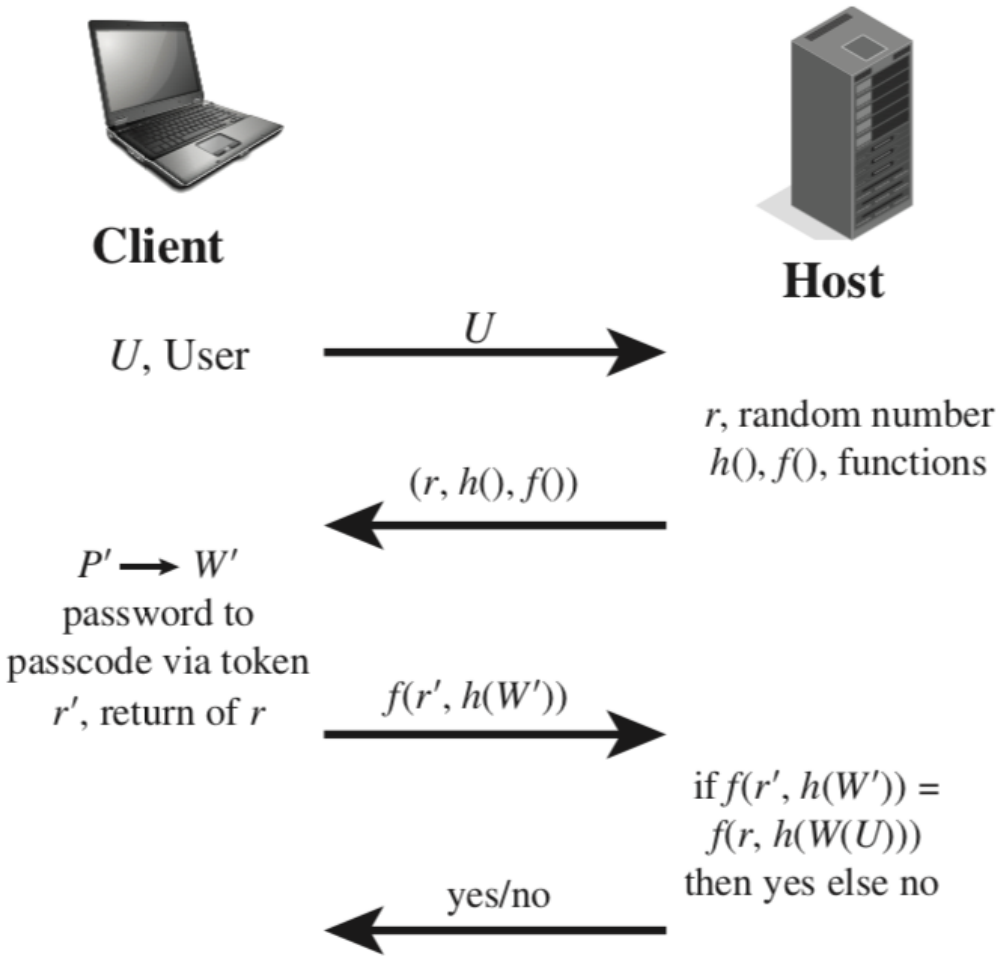
- When authentication is carried out remotely additional threats must be faced with
  - eavesdropping
  - replay attack
- Solutions based on challenge response protocol possibly coupled with cryptography

Password-based C-R protocol



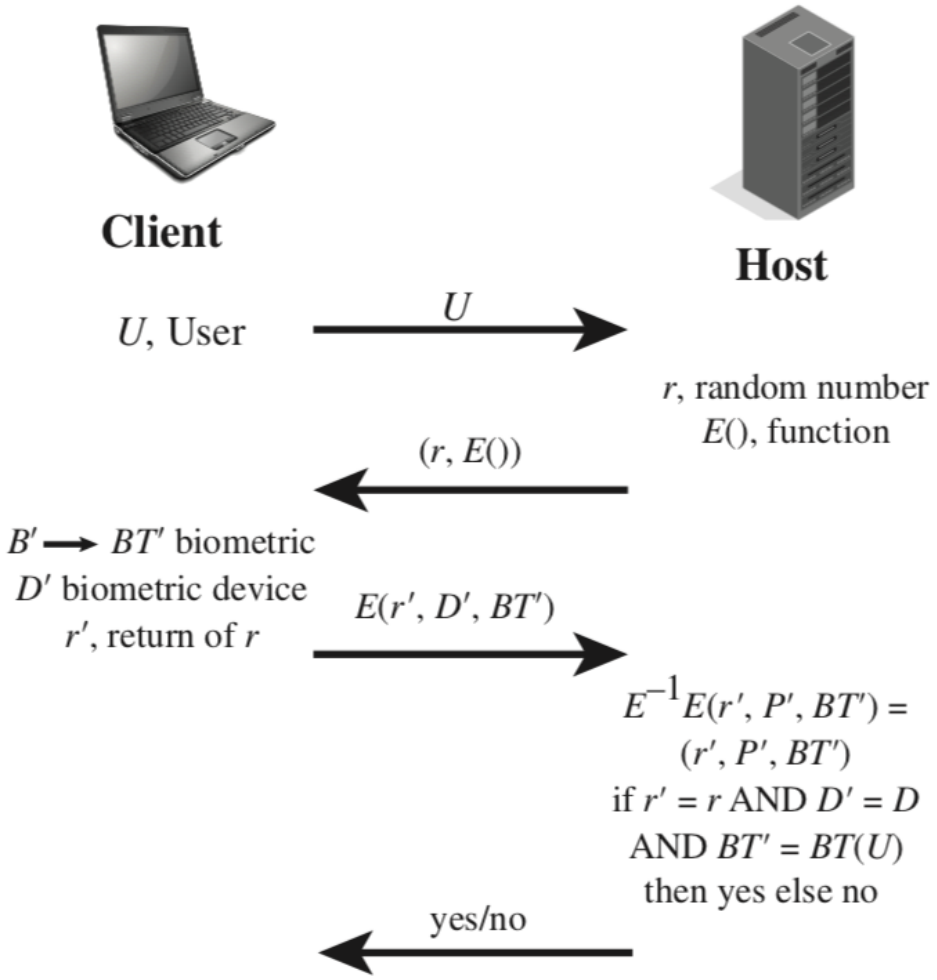
- Neither the password nor the hash of the password are transmitted in plain
- h is a hash function
- f is such that h(P) can not be recovered by observing f(r,h(P))

Token-based C-R protocol



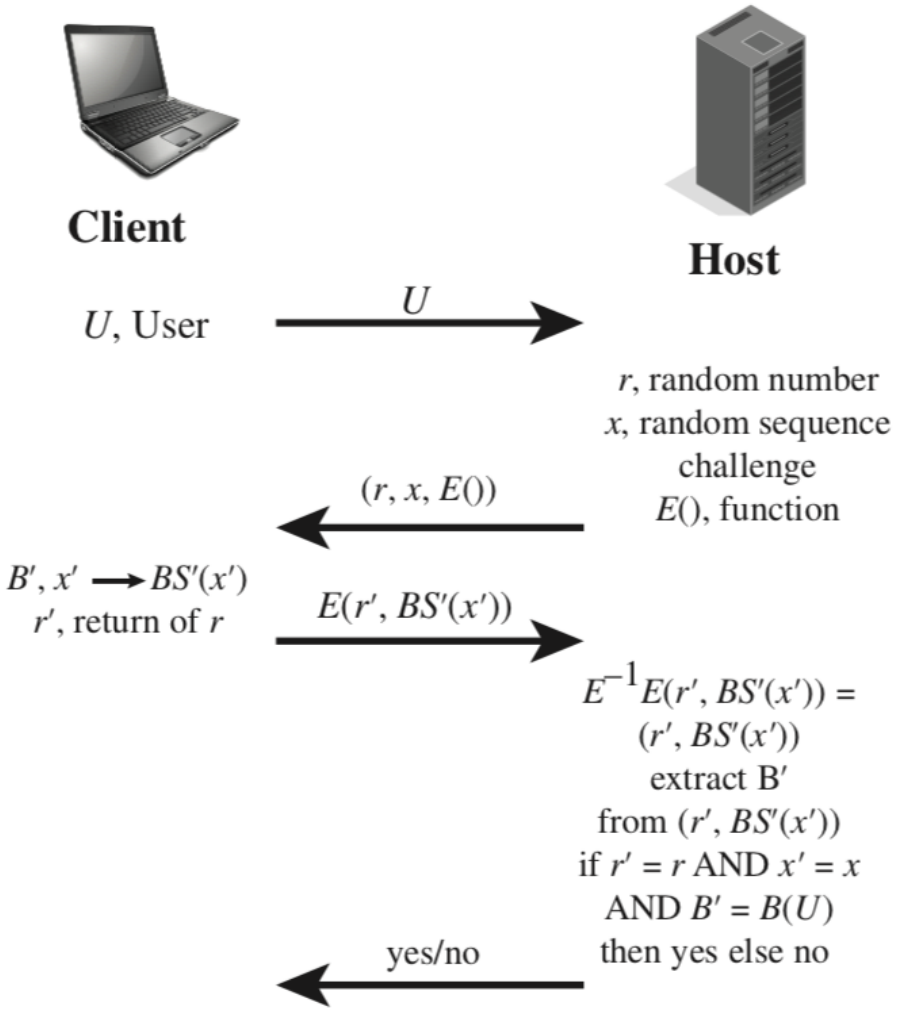
- Password P' is only used by the user to access the token
- In a static system W' is stored in the token
- In a dynamic system W' is generated on the fly by the token and the host

Biometric-based C-R protocol (static)



- E() is an encryption function
- BT' is a biometric template captured by a device at client's side
- D' identifies the biometric device
- BT' = BT(U) means match is above verification threshold

Biometric-based C-R protocol (dynamic)



- The biometric template is also generated based on a challenge
- For instance the user may be asked to type or utter some letters

Summary of attacks

Attacks	Authenticators	Examples	Typical Defenses
Client attack	Password	Guessing, exhaustive search	Large entropy; limited attempts
	Token	Exhaustive search	Large entropy; limited attempts; theft of object requires presence
	Biometric	False match	Large entropy; limited attempts
Host attack	Password	Plaintext theft, dictionary/exhaustive search	Hashing; large entropy; protection of password database
	Token	Passcode theft	Same as password; 1-time passcode
	Biometric	Template theft	Capture device authentication; challenge response

Summary of attacks

Eavesdropping, theft, and copying	Password	"Shoulder surfing"	User diligence to keep secret; administrator diligence to quickly revoke compromised passwords; multifactor authentication
	Token	Theft, counterfeiting hardware	Multifactor authentication; tamper resistant/evident token
	Biometric	Copying (spoofing) biometric	Copy detection at capture device and capture device authentication
Replay	Password	Replay stolen password response	Challenge-response protocol
	Token	Replay stolen passcode response	Challenge-response protocol; 1-time passcode
	Biometric	Replay stolen biometric template response	Copy detection at capture device and capture device authentication via challenge-response protocol

References

- W. Stallings, L. Brown, "*Computer security: principles and practices",* Pearson, 4-th edition. Chapter 3.
- Lectures notes (these slides)