INB255/INN255 Security

Lecture 8:

Access Control Mechanisms

Outline

- Access control and user authentication
- Authentication mechanisms:
  - Knowledge based
    - Passwords
  - Object-Based
    - Tokens
  - ID-Based
    - Biometrics
- Summary

Access control and user authentication

- Most access control decisions are based on user identity
- Need to have:
  - A unique identifier for each individual
  - A means by which the identity can be verified
    - This process is called user authentication
- User authentication is fundamental for:
  - access control
  - user accountability

Access control and user authentication

- Access control process:
  1. Identification: the entity requesting access presents an identifier to the system
  2. Authentication: the entity requesting access provides information that enables the system to verify the claimed identity
  3. Authorisation: the system checks that the authenticated entity is authorised for the requested access
- If authenticated entity is authorised for type of access requested, access to resource is granted

User authentication

- Knowledge-Based (Something you know):
  - Characterized by secrecy or obscurity
    - something only you would know
  - Examples:
    - memorized passwords
    - responses to questions: your birth date, mother’s maiden name, favourite food, your pet’s name, etc
  - Advantages include:
    - Readily accepted method, low cost implementation
  - Disadvantages include:
    - Can be shared
    - Difficult for user to know if compromised

User authentication

- Object-Based (Something you have):
  - Characterized by physical possession of a token.
  - Examples:
    - Physical key
    - Magnetic swipe card
    - Token used for generating access codes
  - Advantages include:
    - Difficult to share (effort required to make a copy)
    - If lost, the owner may realise - sees evidence of the loss
  - Disadvantages include:
    - If lost, the finder can make use of the token

User authentication

- ID-Based (Something you are):
  - Characterized by uniqueness to one person.
  - Examples include:
    - biometrics such as fingerprint, eye scan, voiceprint, signature
  - Advantages include:
    - Characteristic can't be forgotten or lost
    - May be difficult to copy, share or distribute
    - Should require the person being authenticated to be present at the time and point of authentication
  - Disadvantages include:
    - Harder to replace a compromised biometric authenticator, than to replace passwords or tokens

User authentication

- Location-based (Somewhere you are)
  - Characterized by your location (space and time?)
  - Examples:
    - Use of location and tracking technologies: triangulation of cell-phone signals, or global positioning systems (GPS).
    - Machine IP address and DNS name
    - Link location to time
  - Advantages include:
    - Can improve network security, if access locations are (relatively) local
  - Disadvantages include:
    - Privacy issues – who should know where you are, when?

User authentication

- Multi-factor authentication
  - combines two or more authentication techniques
  - aims to obtain a stronger and more reliable level of authentication than for single factor
- Typical example:
  - Most common two-factor authentication is based on
    - something a user knows (factor one), plus
    - something the user has (factor two).
  - Frequently used combination is password and token
    - Example: ATM card and PIN

Outline

- Access control and user authentication
- Authentication mechanisms:
    - Knowledge based
        - Passwords
    - Object-Based
        - Tokens
    - ID-Based
        - Biometrics
- Summary

User authentication mechanisms
Knowledge based: Passwords

- Reusable passwords
  - the most commonly used authenticator
- **User** provides:
  - *username* or ID, and
  - *password*
- **System** has prior stored value to compare with
  - Successful provision of required value authenticates user to system
- Requirement: system must store the values used to verify the passwords for all system users

User authentication mechanisms

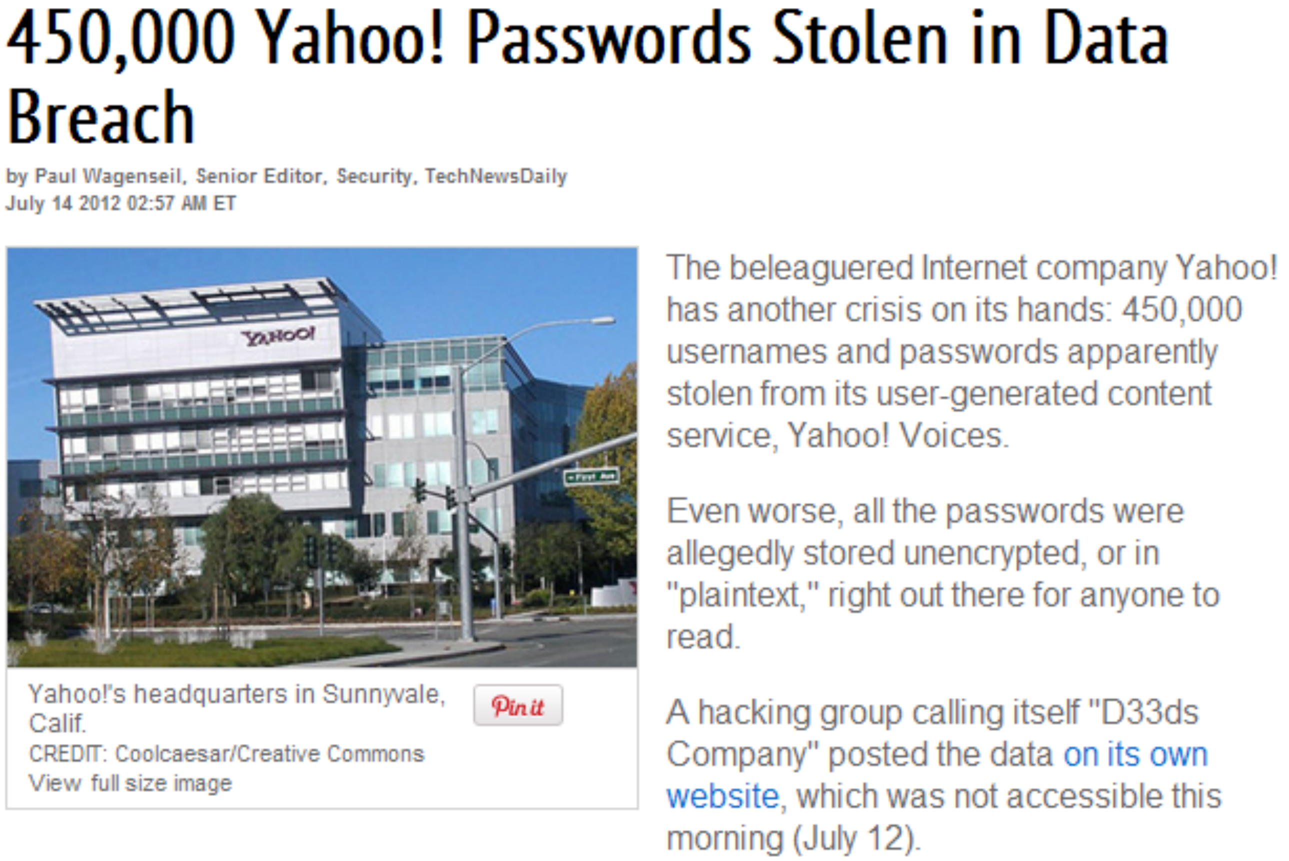
Knowledge based: Passwords

- Reusable passwords
- Problems:
  - Can be easy for user to
    - share (intentionally or not), or
    - forget
  - Can be written down (post-it note on monitor?)
  - Can be easy for attacker to guess
  - No *non-repudiation* if password is known to system
  - System password files are valuable information assets, so password files need to be protected
    - should not be stored with passwords as plaintext

User authentication mechanisms

Knowledge based: Passwords

- Reusable passwords: problems



450,000 Yahoo! Passwords Stolen in Data Breach

by Paul Wagenseil, Senior Editor, Security, TechNewsDaily
July 14 2012 02:57 AM ET

Yahoo!'s headquarters in Sunnyvale, Calif.
CREDIT: Coolcaesar/Creative Commons
View full size image

Pin it

The beleaguered Internet company Yahoo! has another crisis on its hands: 450,000 usernames and passwords apparently stolen from its user-generated content service, Yahoo! Voices.

Even worse, all the passwords were allegedly stored unencrypted, or in "plaintext," right out there for anyone to read.

A hacking group calling itself "D33ds Company" posted the data on its own website, which was not accessible this morning (July 12).

User authentication mechanisms

Knowledge based: Passwords

- Password selection strategies
  - User selected, or
  - Computer generated

- Password checking
  - Reactive password checking
  - Proactive password checking

- Protecting Passwords

User authentication mechanisms
Knowledge based: Passwords

- For user selected passwords, organisation’s security policy should include:
  - User training
    - Explain to users the importance of choosing ‘strong’ passwords.
  - Password selection guidelines
    - What are the characteristics of ‘good’ passwords?
    - AS27002:2006 Clause11.3.1 has guidelines for password use.
- Unlikely to be effective in most organisations
  - particularly with large user population or a high turnover of users.
- Some users simply ignore guidelines, or are poor at selecting a ‘strong’ password.
  - Many choose passwords that are too short and very easy to guess.

User authentication mechanisms

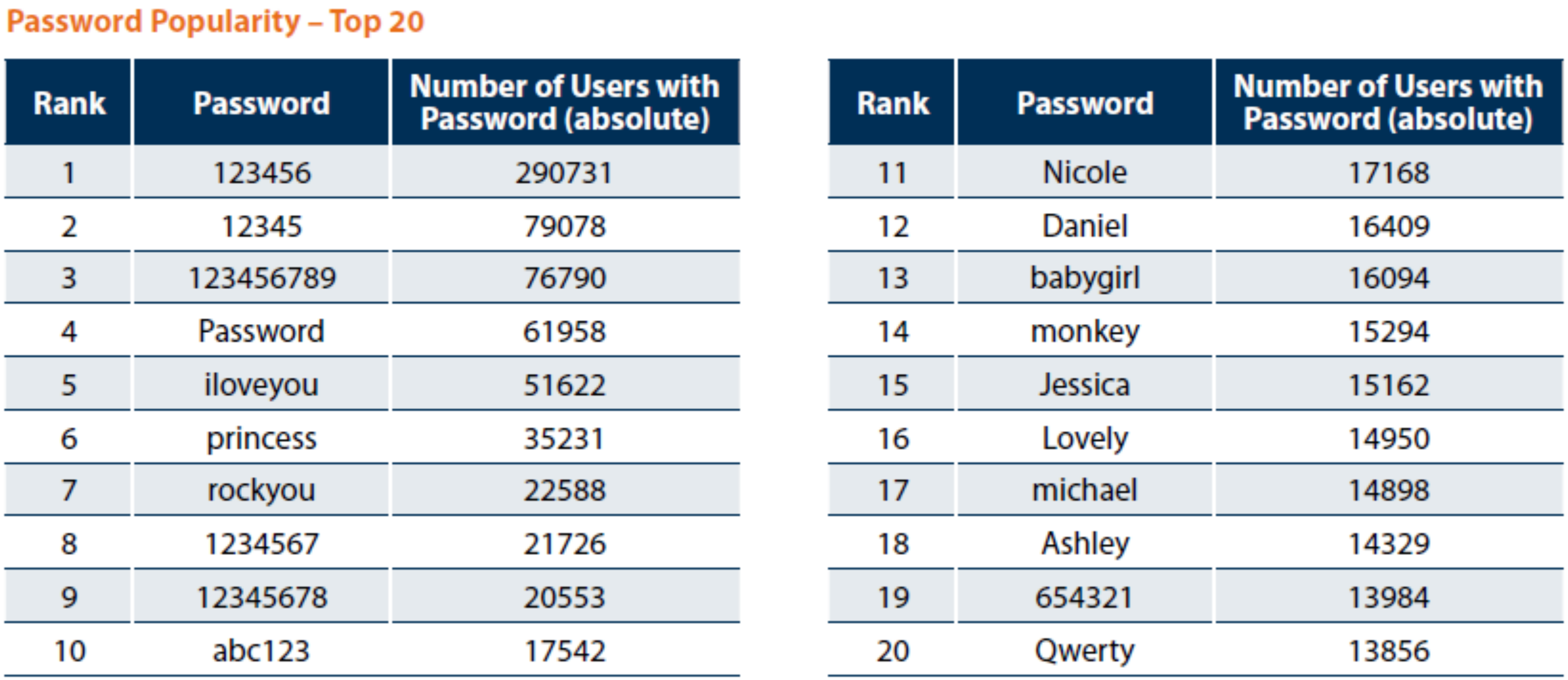
Knowledge based: Passwords

- **Imperva paper: Analysis of user selected passwords**
  - http://www.imperva.com/docs/WP_Consumer_Password_Worst_Practices.pdf
  - December 2009 security breach at Rockyou.com
  - Attacker retrieved and posted 32 million passwords on Internet (Passwords were stored in database as plaintext)
  - Imperva analysis of the password list:
    - About 30% of passwords length less than or equal to six characters.
    - Almost 60% of passwords used characters from limited alpha-numeric set
    - Nearly 50% of users used names, slang words, dictionary words or trivial passwords (consecutive digits, adjacent keyboard keys, and so on).
      - Most common password: “123456”

User authentication mechanisms

Knowledge based: Passwords

- **Imperva paper: Analysis of user selected passwords**
  - **Extract from p 4: Table of 20 most common passwords**



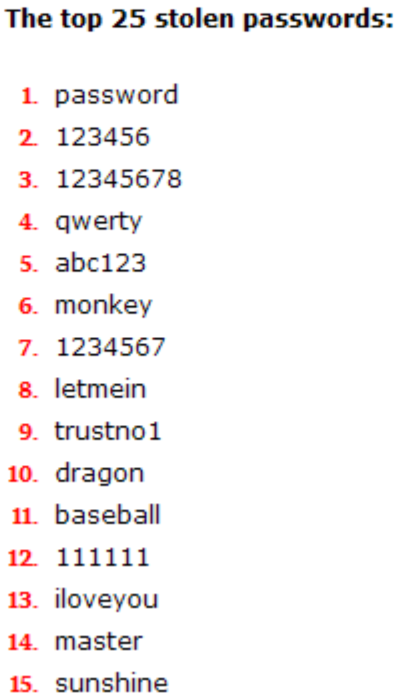
Password Popularity – Top 20

Rank	Password	Number of Users with Password (absolute)
1	123456	290731
2	12345	79078
3	123456789	76790
4	Password	61958
5	iloveyou	51622
6	princess	35231
7	rockyou	22588
8	1234567	21726
9	12345678	20553
10	abc123	17542
11	Nicole	17168
12	Daniel	16409
13	babygirl	16094
14	monkey	15294
15	Jessica	15162
16	Lovely	14950
17	michael	14898
18	Ashley	14329
19	654321	13984
20	Qwerty	13856

User authentication mechanisms

Knowledge based: Passwords

- 2011 SplashData list of top 25 stolen passwords:
- http://www.theglobeandmail.com/news/technology/tech-news/top-25-most-hacked-passwords-revealed/article2244739/



The top 25 stolen passwords:

1. password
2. 123456
3. 12345678
4. qwerty
5. abc123
6. monkey
7. 1234567
8. letmein
9. trustno1
10. dragon
11. baseball
12. 111111
13. iloveyou
14. master
15. sunshine

User authentication mechanisms

Knowledge based: Passwords

- Computer generated passwords avoid the problem of users choosing weak passwords
- But have another security problem:
  - Passwords consisting of random characters are difficult for users to remember, so they write them down.
  - Common locations are:
    - Sticky note on monitor
    - Under mouse pad
    - In top desk drawer
    - In diary
- FIPS PUB 181 http://www.itl.nist.gov/fipspubs/fip181.htm defines an automated password generator
  - Generates ‘word’ by forming pronounceable syllables and concatenating them

User authentication mechanisms

Knowledge based: Passwords

- **Password checking strategies**
- Reactive password checking:
  - System administrator periodically runs a password cracking tool (those available to attackers) on password file looking for those passwords that are easy to recover
    - How often should the administrator do this?
    - Is this activity approved by management?
    - Is there an ‘evil’ side to this?
  - What type of passwords are “easy to recover”?

User authentication mechanisms

Knowledge based: Passwords

- **Password checking strategies**
- Reactive password checking:
  - What type of passwords are “easy to recover”?
    - Short passwords
      - Easy to check all possible combinations of 3 or 4 characters
    - Passwords that are dictionary words …
      - Users more likely to use *apple* than *plape*
    - Or words with an alpha character replaced with digit …
      - Say, appl3
- Reactive checking does not prevent
  - users choosing bad passwords, or
  - using them until detected and asked to change

User authentication mechanisms

Knowledge based: Passwords

- **Password checking strategies**
- Proactive password checking:

1. The user selects a *potential* password
2. This is tested to see if it is strong enough to be used
    - If the password is unsuitable, the user must choose again

– Balance is required:

- If the system rejects too many potential passwords, the users will complain about the process
- If the acceptable password criteria are too ‘simple’, this may aid attackers by allowing a refined search over a smaller password space than might otherwise have been the case

User authentication mechanisms

Knowledge based: Passwords

- Protecting passwords
  - Reusable passwords require confidentiality when in
    - Storage (on authentication server)
    - Transmission (between client and server over network)
    - Use (do not display on screen when being entered!)
  - If ‘clear’ passwords are captured, an attacker may reuse the password and masquerade as the user
    - Http basic authentication effectively transmits ‘clear’ passwords
  - OR an attacker may masquerade as the server, so that the client will disclose the ‘clear’ password to the rogue server

User authentication mechanisms

Knowledge based: Passwords

- Reusable passwords & storage strategies:
  - System password file has entries for each user:
    - Example: Leonie myPassw0rd
  - Alternatives to storing plaintext passwords?

1. Store the hash values of passwords
   - Example: Leonie a422be0e206a05bf03061140e4942382
   - Attacker needs to find a password that produces this value
   - Suppose I see Mark a422be0e206a05bf03061140e4942382 in the password file, and I know what my password hash is?
2. Store salted hash values of passwords
   - Put some random but not secret information in to pwd hash
   - Example: Leonie 86 427b050a20b5eb9301a6957c80a7e9b5
3. Use encryption

User authentication mechanisms

Knowledge based: Passwords

- One-Time Password authentication systems
  - aim to provide security against attacks based on replaying captured reusable passwords
- However:
  - eavesdropping attackers may be able to exploit race conditions if multiple simultaneous authentication sessions are allowed
  - One-Time Password does not provide
    - confidentiality or data integrity services
    - protection against active attacks such as session hijacking
      - known to be present in the current Internet

User authentication mechanisms

Knowledge based: Passwords

- One-Time Password authentication systems
- S/KEY (developed by Bellcore around 1994) provides basis for deriving OneTime Passwords
- OneTime Password described in RFC 2289 (1998) http://www.faqs.org/rfcs/rfc2289.html
  - Uses a secret pass-phrase to generate a sequence of one-time (single use) passwords
  - Password sequence is generated by applying a hash function:
    - To the password P,
    - Then to the hash value H(P)
    - Then to that hash value H(H(P)) …
  - The passwords are used in the reverse order to generation
  - The system only needs to store the password provided in order to verify the next password
  - Security of this One Time Password system is based on the difficulty of inverting a secure cryptographic hash function

Outline

- Access control and user authentication
- Authentication mechanisms:
  - Knowledge based
    - Passwords
  - Object-Based
    - Tokens
  - ID-Based
    - Biometrics
- Summary

User authentication mechanisms

Object based: Tokens

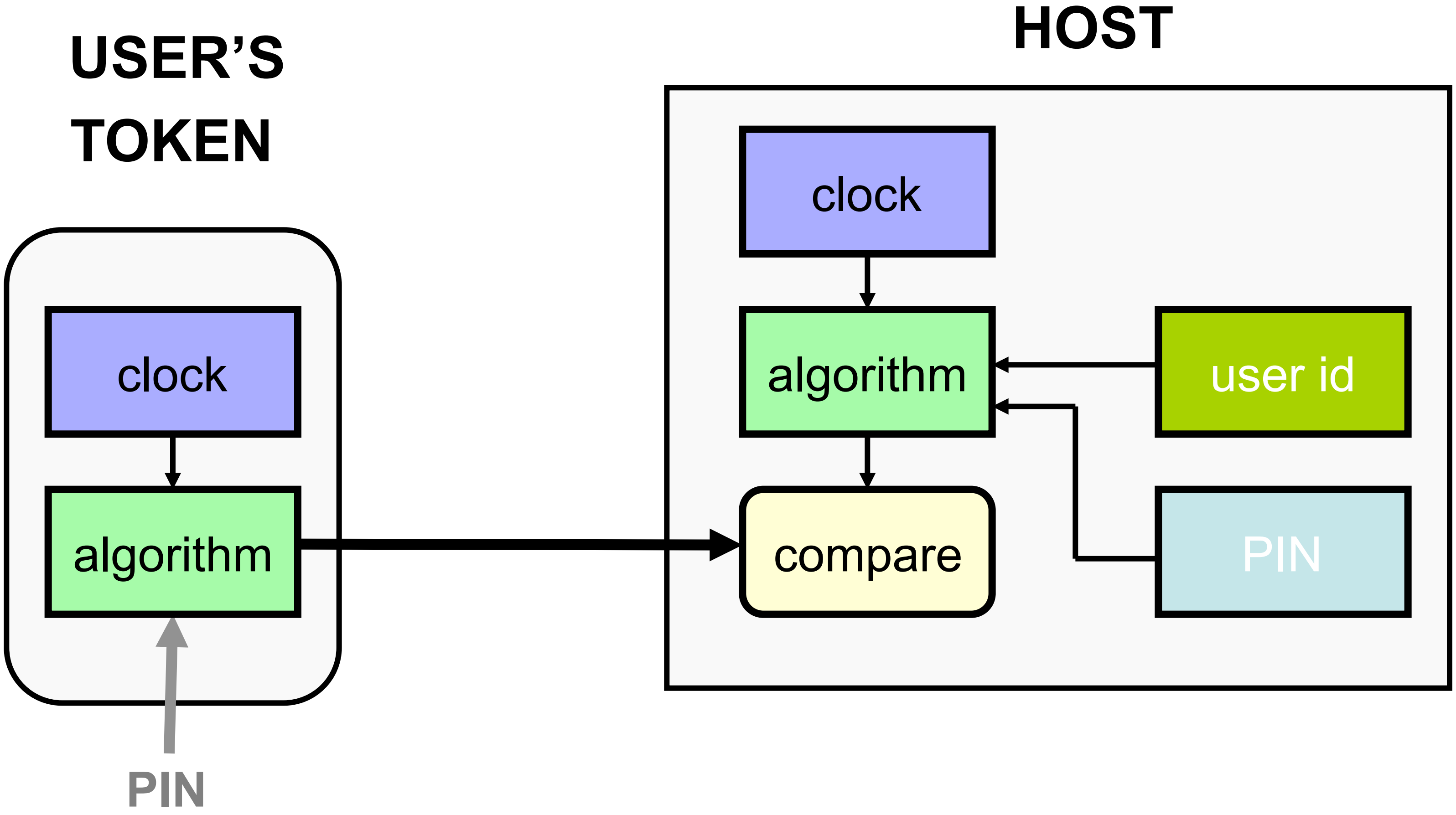
- A token could be
  - A physical key,
  - A swipe card,
  - An ID badge
- Can also have token that generates a sequence of one-time passwords
  - Need to have password generators in the token and at the host system that are synchronized to produce the same sequence of random passwords
- Two general methods:
  - Clock-based tokens
  - Counter-based tokens

User authentication mechanisms

Object based: Clock-based tokens

- Clock-based tokens: Operation
- Token display shows a constantly changing value
  - Clock time is used as input to algorithm calculating display value
  - User provides ID and then types in current value as authenticator when requesting access (log in to system)
- Possession of token is necessary to know the correct value for the current time
- NOTE: Clocks in token and at host system must be synchronised
- Some tokens have extra security feature: require a PIN to access display value
  - Something you have, or something you know?

User authentication mechanisms
Object based: Clock-based tokens



User authentication mechanisms

Object based: Clock-based tokens

- Example: RSA SecurID
- Each RSA SecurID authenticator has a unique symmetric key (uses cryptography)
- The key is used with a proprietary algorithm (SecurID Hash) to generate a new code every 30/60 seconds
- The code is unpredictable and dynamic
  - Difficult to guess the correct code at any given time.

RSA SecurID SID700

RSA SecurID SD520

User authentication mechanisms

Object based: Clock-based tokens

- Clock-based tokens: Issues
- Requires synchronisation:
  - The system fails if clocks in token and at host are not synchronised
- For network usage, need to provide an acceptable window of time to allow for network delays
  - Introduces the possibility of attack
    - An intermediate node could capture a password and then log in as user

User authentication mechanisms

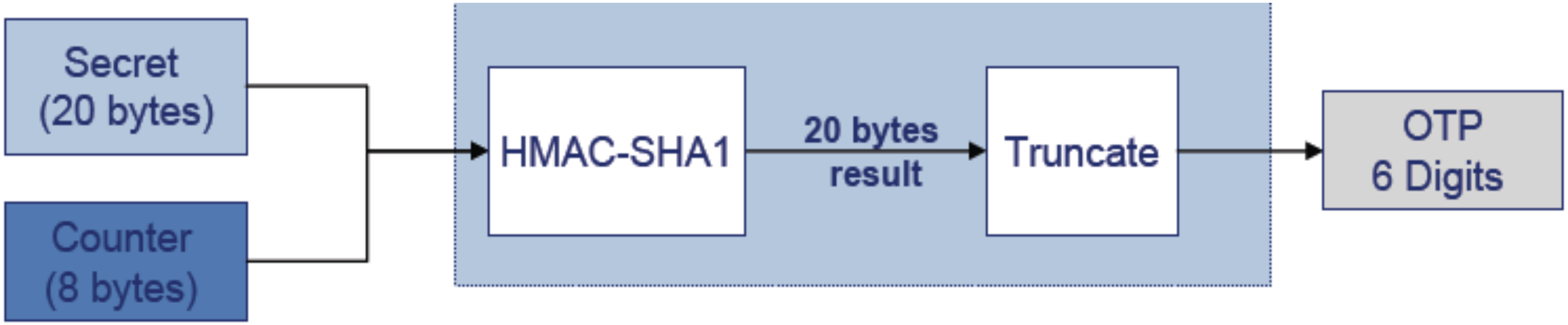
Object based: Counter-based tokens

- Counter-based tokens: Operation
- Instead of a clock value, the token device generates a ‘password’ result value as a function of an internal counter and other internal data, without external inputs.
- Example: HOTP
  - HMAC-Based One-Time Password Algorithm
  - described in RFC 4226 (Dec 2005)
    - http://www.rfc-archive.org/getrfc.php?rfc=4226
  - Uses a shared secret value (stored on token) as shared key
  - Value displayed on the token is HMAC value of the current counter value

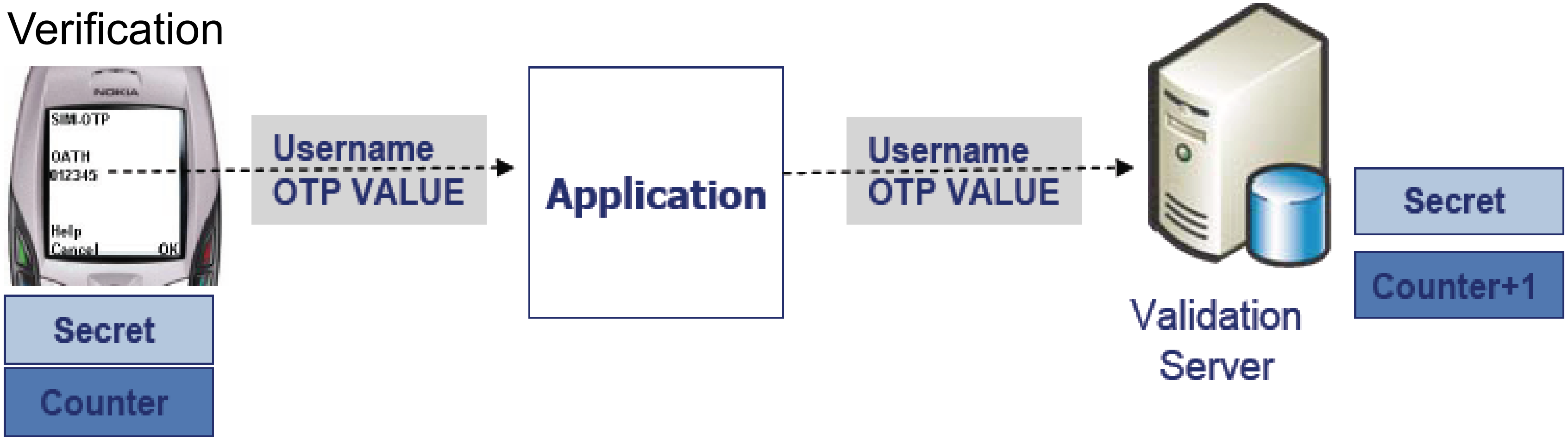
User authentication mechanisms

Object based: Counter-based tokens - HOTP

OTP Generation



Verification



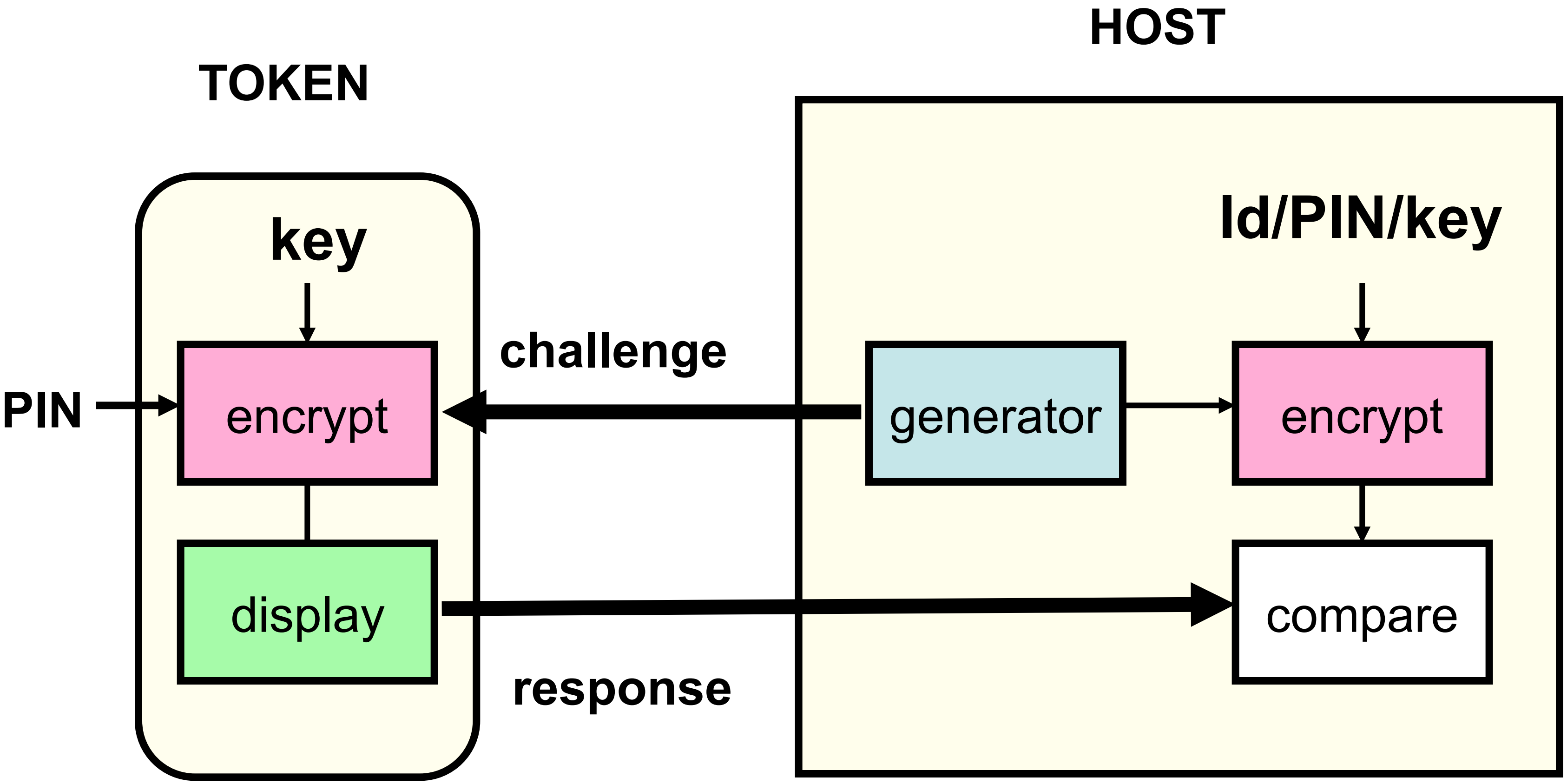
User authentication mechanisms

Object based: C-R token systems

- Challenge response systems: Operation
- User makes access request
- System sends a challenge (generally a number) to user
- User types challenge into device
  - Device computes and displays response
    - Response is a cryptographic one-way function of challenge and other info such as key and PIN.
- User reads response off device display and sends response to host
  - If response is as expected by host, then access is granted
- Different challenges produce different responses
  - Prevents replay attacks

User authentication mechanisms

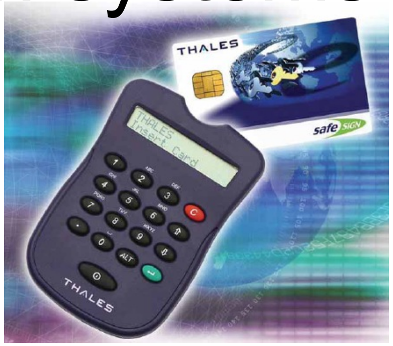
Object based: C-R token systems



Symmetric Case

User authentication mechanisms

Object based: Chall.-resp. token systems



Example: **SafeSign Personal Security Module** uses a smart card (containing cryptographic key material) and reader.

Operation:

- User logs into host system and enters userID,
- Host generates random seven-digit number (the challenge), and displays it on the terminal
- User enters PIN into the token keypad, then the seven-digit challenge, token displays response for user to enter
- Host system performs a similar computation using the PIN and key values stored with the userID
- If results of host computation check out against response entered by the user, then access is granted

•SafeSign mobile authentication uses mobile phone rather than dedicated hardware device, MCode CR is challenge response

Outline

- Access control and user authentication
- Authentication mechanisms:
  - – Knowledge based
    - Passwords
  - – Object-Based
    - Tokens
  - – ID-Based
    - Biometrics
- Summary

User authentication mechanisms

ID based: Biometrics

- What is a biometric system?
  - Automated methods of verifying or recognising a person based upon a physiological or behavioural characteristic.
- Biometric examples:
  - fingerprint
  - facial recognition
  - eye retina/iris scanning
  - hand geometry
  - written signature
  - voice print
  - keystroke dynamics

User authentication mechanisms

ID based: Biometrics

- Requirements for biometric characteristics:
  - **Universality**:
    - each person should have the characteristic
  - **Distinctiveness**:
    - any two persons should be sufficiently different in terms of the characteristic
  - **Permanence**:
    - the characteristic should be sufficiently invariant (with respect to the matching criterion) over a period of time
  - **Collectability**:
    - the characteristic can be measured quantitatively

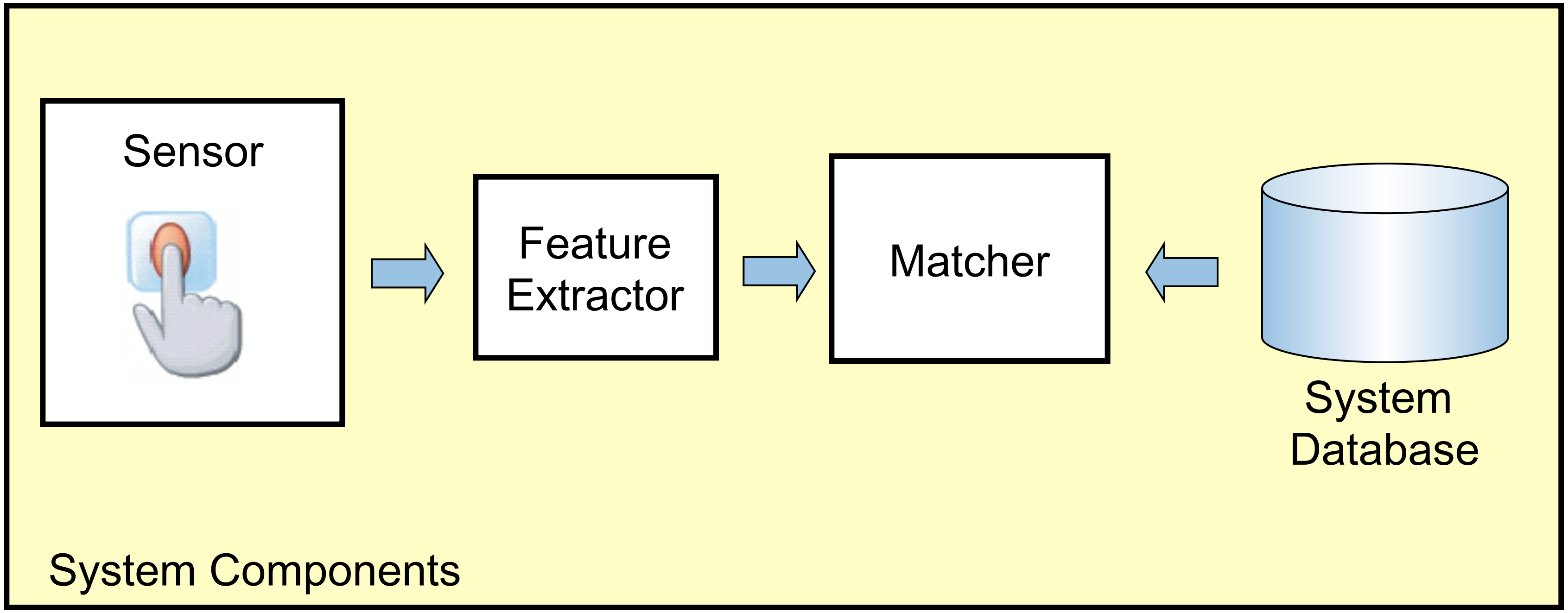
User authentication mechanisms

ID based: Biometrics

- Practical considerations:
  - Performance:
    - What is the achievable recognition accuracy and speed?
    - What resources are required to achieve desired recognition accuracy and speed?
    - Are there operational and environmental factors affecting the accuracy and speed?
  - Acceptability:
    - The extent to which people are willing to accept the use of a particular biometric identifier (characteristic)
  - Circumvention:
    - How easy is it to fool the system using fraudulent methods?

User authentication mechanisms

ID based: Biometrics



User authentication mechanisms

ID based: Biometrics

- System components:
  - Sensor module: captures the biometric signal of an individual.
    - Example: fingerprint sensor that images the ridge and valley structure of a user's finger.
  - Feature extraction module: processes the acquired biometric signal to extract a set of discriminatory features.
    - Example: feature extraction module of a fingerprint-based biometric system extracts the position and orientation of minutiae points (local ridge and valley singularities) in a fingerprint image.

User authentication mechanisms

ID based: Biometrics

- System components:
  - System database module: used to store the biometric templates of enrolled users
  - Users are included in biometric system database in enrolment phase:
    - Biometric characteristic of individuals scanned by a biometric reader to produce a digital representation (feature values) of the characteristic.
    - Quality check performed to ensure acquired sample can be reliably processed by successive stages.
    - Input digital representation is further processed by a feature extractor to generate a compact representation called a *template*.
    - May store multiple templates to account for variations

User authentication mechanisms

ID based: Biometrics

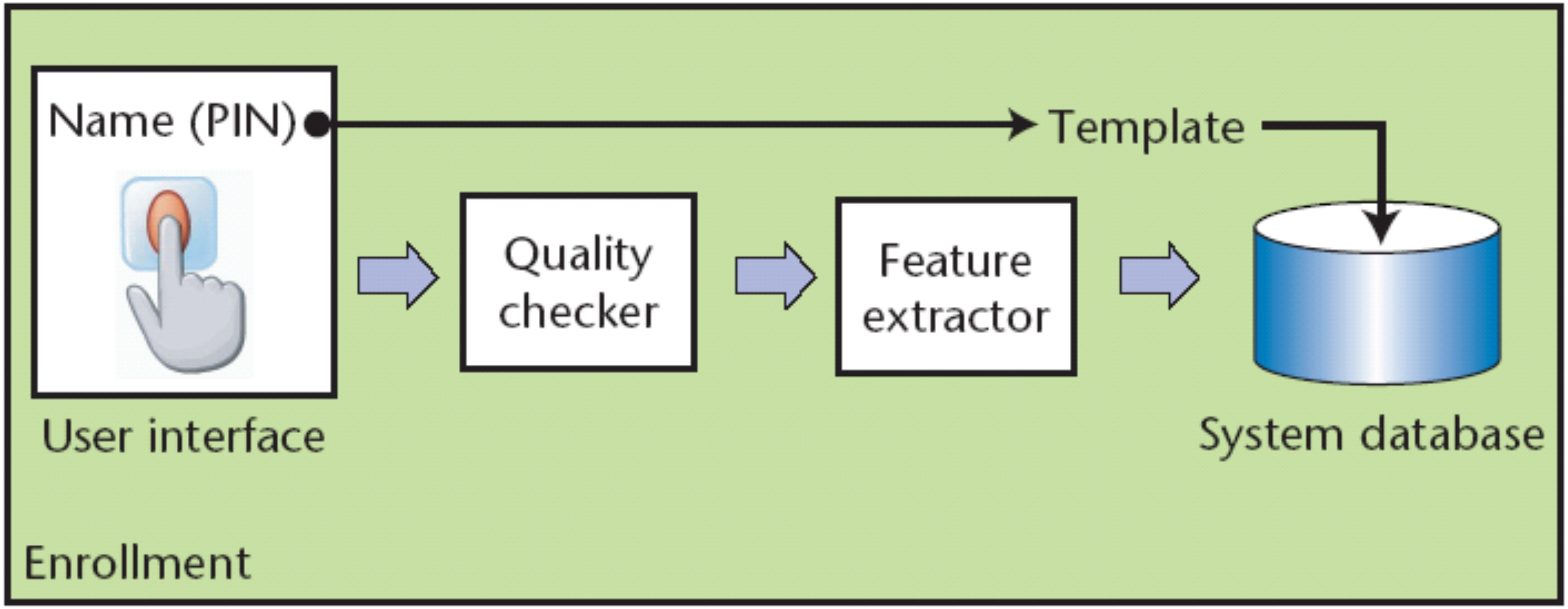
- System components:
  - Matcher module: features captured during recognition are compared against the stored templates to generate matching scores.
    - Example: matcher module of a fingerprint-based biometric system determines the number of matching minutiae between the input and the template fingerprint images and reports a matching score.
  - Matcher module also includes a decision making module which uses matching score to
    - confirm a user's claimed identity (verification) or
    - establish a user's identity (identification)

User authentication mechanisms
ID based: Biometrics

- Modes of operation:
- Enrolment:
  - Analog capture of the user's biometric attribute.
  - Processing captured data to develop a template of user's attribute
- Verification of claimed identity (1-to-1):
  - Capture of a new biometric sample.
  - Comparison of the new sample with that of the user's stored template.
  - Decision on access acceptance or rejection.
- Identification (1-to-many):
  - Capture of a new biometric sample.
  - Search database of stored templates for a match based solely on the biometric.

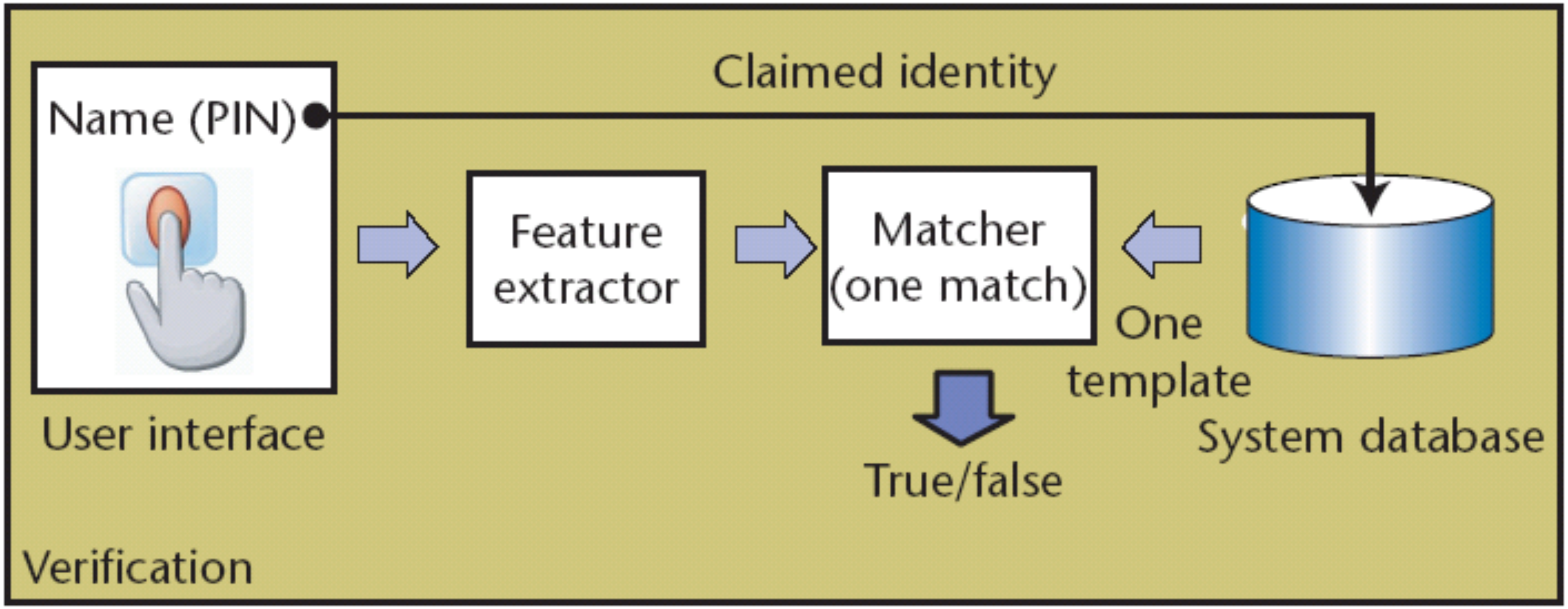
User authentication mechanisms

ID based: Biometrics - Enrolment



Source: Biometric Recognition: Security and Privacy Concerns

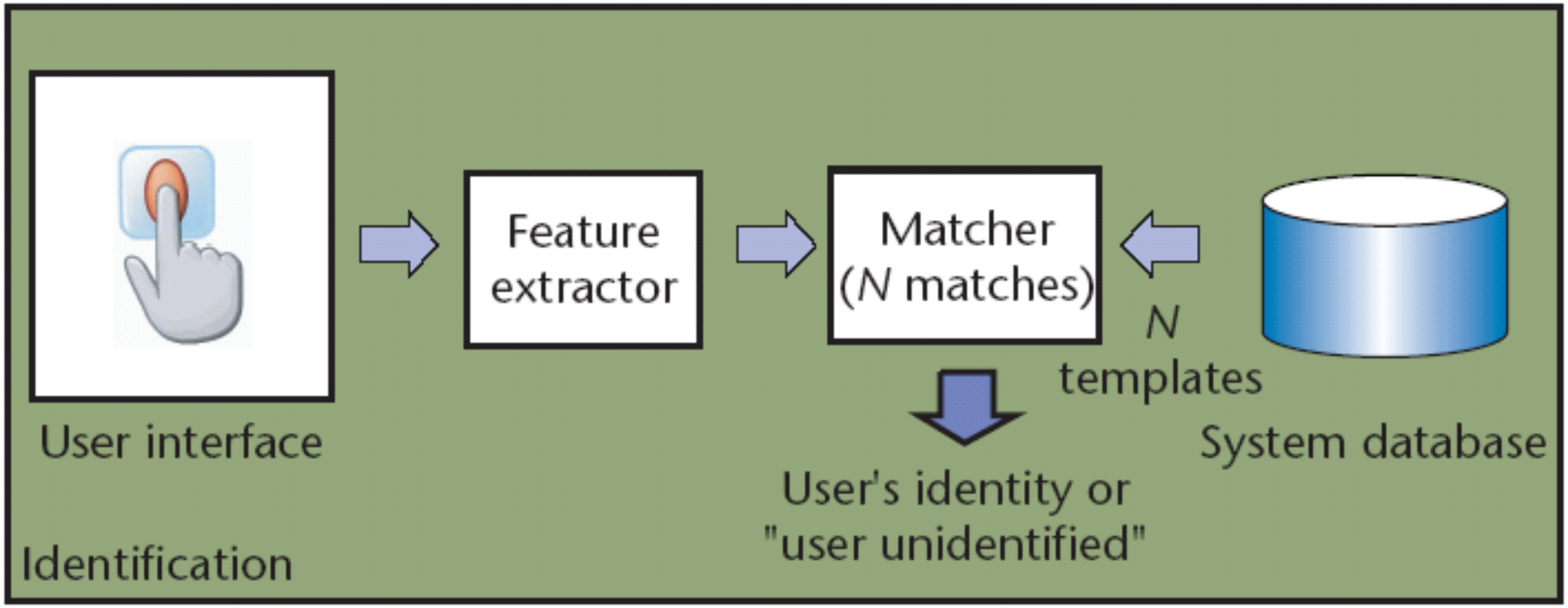
User authentication mechanisms
ID based: Biometrics - Verification



Source: Biometric Recognition: Security and Privacy Concerns

User authentication mechanisms

ID based: Biometrics - Identification



Source: Biometric Recognition: Security and Privacy Concerns

User authentication mechanisms

ID based: Biometrics - Issues

- Possible errors:
  - Two samples of the same biometric characteristic from the same person (for example, two impressions of a user's right index finger) are not exactly the same due to:
    - imperfect imaging conditions (e.g. sensor noise and dry fingers),
    - changes in the user's physiological or behavioral characteristics (e.g. cuts and bruises on the finger),
    - ambient conditions (e.g. temperature and humidity) and
    - user's interaction with the sensor (e.g. finger placement).
  - Captured features are compared against stored template to generate matching scores used for decision making, so variations can result in error

User authentication mechanisms

ID based: Biometrics - Issues

- Two possible errors:
  - Reject legitimate user
    - If expect sample to be an exact match with stored template, then any variation leads to rejection of legitimate user
    - To reduce this possibility (rejecting a legitimate user) system may be adjusted to tolerate small variation between captured features and stored template
      - This introduces another possible error:
  - Accept unauthorised user
    - An unauthorised user may be able to provide a sample with captured features similar enough to those of the stored template for a legitimate user that the unauthorised user gains the access rights of legitimate user

User authentication mechanisms

ID based: Biometrics - Issues

- So biometric verification systems can make two types of errors in decision making:
  - False match:
    - mistaking biometric measurements from two different persons to be from the same person, (results in allowing unauthorised person access) and
  - False non-match:
    - mistaking two biometric measurements from the same person to be from two different persons (so rejecting legitimate user)
  - There is a trade-off between false match rate (FMR) and false non-match rate (FNMR) in every biometric system

User authentication mechanisms

ID based: Biometrics - Issues

- The system decision is tuned by a threshold *t*:
  - Pairs of biometric samples generating matching scores higher than or equal to *t* are inferred as mate pairs (same person);
  - Pairs of biometric samples generating scores lower than *t* are inferred as non-mate pairs (different persons).
- Both FMR and FNMR are functions of the system threshold *t*:
  - If *t* is decreased to make the system more tolerant to input variations and noise, then FMR increases
  - If *t* is raised to make the system more secure, then FNMR increases accordingly

Summary

- Access control is important in information security
  - Access to resources may be restricted – decision to allow access depends on identity of requester
- User authentication is fundamental - want to
  - prevent unauthorised access, and
  - permit authorised access to information assets
- Common access control mechanisms are
  - Knowledge based - Passwords
  - Object-Based - Tokens
  - ID-Based - Biometrics
- Some mechanisms require cryptography
- Advantages and disadvantages for each type
  - Cost, level of security provided, user acceptability, etc