02230: User Authentication

Robin Sharp

Informatics and Mathematical Modelling
Technical University of Denmark
Phone: (+45) 4525 3749
e-mail: robin@imm.dtu.dk

Basic ideas

User authentication is needed in order to identify users.

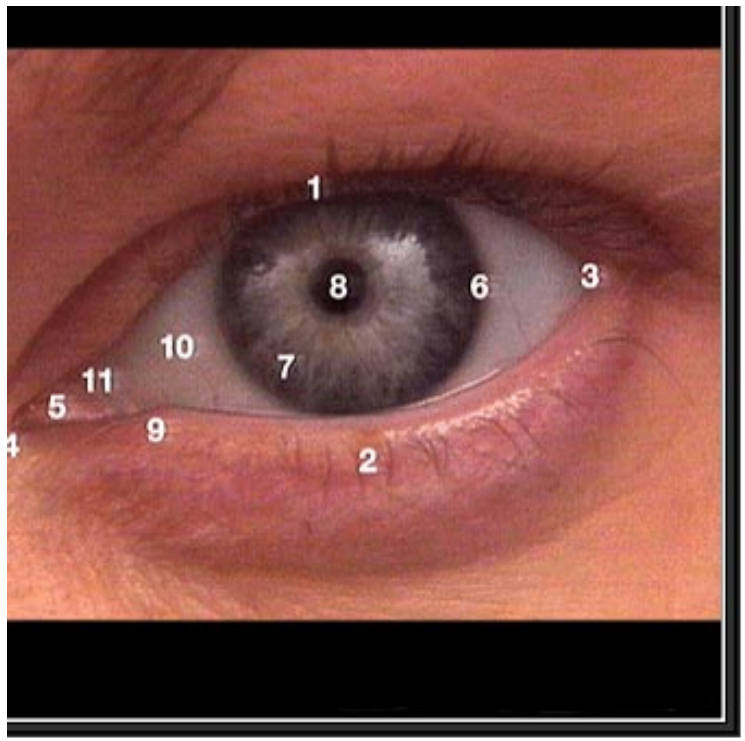
Three possible things to base identification on:

1. Something the user **knows**: Passwords, PIN codes, pass phrases, secret handshakes,...
2. Something the user **has**: ID badge, (physical) key, driving license, uniform,...
3. Something the user **is**: Based on physical characteristics, such as fingerprints, voiceprints, facial features, iris pattern, retina pattern,... Often known as **biometrics**.

Often necessary to use a *combination* in order to achieve real security!

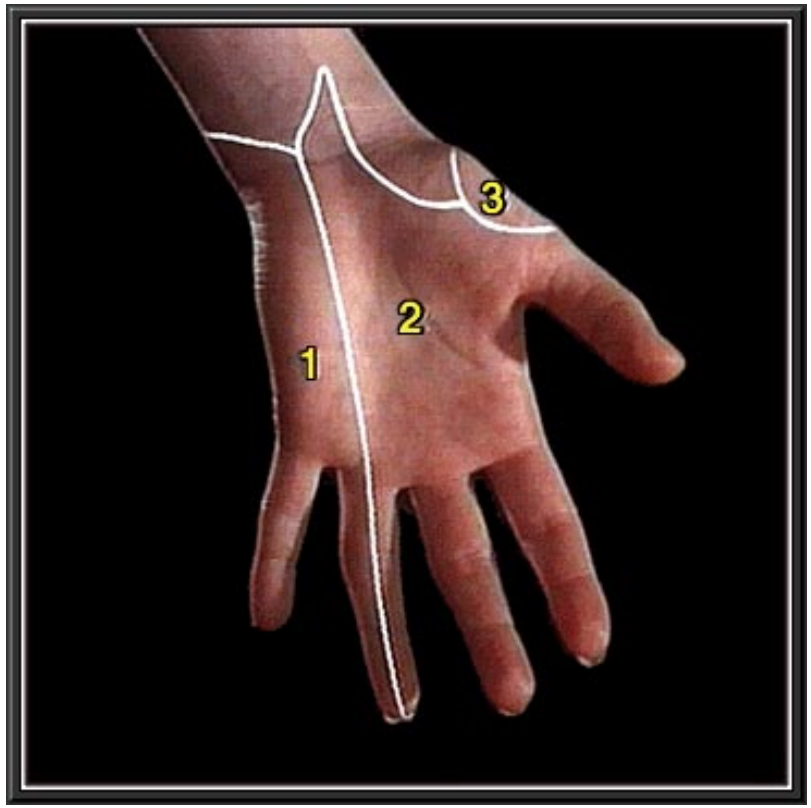
Iris recognition

- Attempts to find characteristic patterns in the iris of the eye:



Hand and finger patterns

- Attempts to find characteristic measurement data by inspection of the fingers of the hand:



Passwords

- “Everybody knows” that passwords should be difficult to guess. So:
  - Make the password long enough
  - Don’t use common words or phrases
  - Include characters from a large character set
- **Attacks** on passwords include:
  - Try all possible passwords (combinatorial)
  - Try many probable passwords (dictionary attack)
  - Try passwords likely for the user
  - Exploit access to the system list of passwords
  - Ask the user (social engineering)

Loose-lipped systems

- Some systems help the attacker to break in more quickly!

```
Welcome to XYZ databar at ABC
Enter user name: alan
Invalid user name, unknown user
Enter user name:
```

- Better systems give no information until the end of the dialogue:

```
Welcome to the XYZ databar at DEF
Enter user name: alan
Enter password: ********
Invalid access
Enter user name:
```

Password storage

- Passwords are nowadays stored in encrypted form, so the plaintext password cannot be found directly.
- Risks:
  - If encrypted password file is accessible, and you have a dictionary, it is easy to try encrypting all words in the dictionary to see if one matches...
  - If several users use same password, their entries in the file will be identical.
- Typical solution (Unix): encryption of user's password is parameterised by a **salt**: 12-bit number derived from system time and process ID. (Salt is stored in plaintext form in file together with password.)

Choosing and using passwords

Password must be difficult to guess, but easy to remember!

To ensure good passwords:

- Give users good guidelines for password design.
- Use computer-generated passwords.
- Reactive checking: Use password cracker at regular intervals.
- Proactive checking: Check password when it is selected by user; reject if not good enough.

Checking criteria include:

- Minimum length.
- Not in (large) dictionary.
- Markov model for predicting guessable passwords.

The good weep and the evil laugh...

- Many investigations show that, if there are no special rules, people choose **weak passwords**:

  E.g. Morris & Thompson, 1979:

0.5%	Single ASCII character
2%	Two ASCII characters
14%	Three ASCII characters
14%	Four letters
21%	Five letters, all same case
18%	Six lower case letters
15%	Words in dictionaries or lists of names
14%	*"Better" choices*

One-time passwords

- A password that changes each time it is used.
- Often known as **challenge-response systems**.
- The user has to *manipulate* the challenge in an agreed manner in order to be accepted. E.g.:
  - f(x)=x+1. System prompts with a value, user replies with the modified value.
  - f(a1,a2,a3,a4,a5,a6)=(a3,a1,a1,a2). System supplies a string, user transforms it in an agreed manner.
  - f(E(x))= E(D(E(x))+1). User must decrypt an encrypted value, modify it and re-encrypt it.
- Some methods are suitable for *people*, others for *computers* to identify themselves.

Trends in user authentication

- Modern trend is to combine several authentication methods: **Multi-factor authentication**.
  - Something you **have** + something you **know**.

    E.g. *Credit card* + *PIN code*
  - Something you **know** + something you **are**.

    E.g. Password + *fingerprint*
  - Something you **have** + something you **are**.

    E.g. *Smart card* + *iris recognition*
- Requires the true user to be present and to make use of something he/she knows or has.
- Probably good enough for many years to come?