COMP3028 Computer Security

Lecture 9 – Identification and Authentication

This Lecture

- User authentication
- Identification & authentication
- Passwords
  - how to get the password to the user
  - forgotten passwords
  - password guessing
  - password spoofing
  - compromise of the password file
- Biometrics

Introduction

- A secure system might have to track the identities of the users requesting its services.
- Authentication: process of verifying a user's identity.
- Two reasons for authenticating a user:
  - The user identity is a parameter in access control decisions.
  - The user identity is recorded when logging security relevant events in an audit trail.

Identification & Authentication

- When logging on to a computer you enter
  - user name and password
- The first step is called identification
  - You announce who you are.
- The second step is called authentication
  - You prove that you are who you claim to be.
- Authentication by password is widely accepted and not too difficult to implement.

Bootstrapping Authentication

- Passwords should be secrets shared between the user and the system authenticating the user.
- How do you bootstrap a system so that the password ends up in the right places, but nowhere else?
- In an enterprise, users can collect their password personally.
- Otherwise, the password could be sent by mail, email, or phone, or entered by the user on a web page.
- You now have to consider who might intercept the message and who might actually pick it up.

Authenticating a Remote User

- Do not give the password to the caller but call back an authorized phone number
- Send passwords that are valid only for a single log-in request so that the user has to change immediately to a password not known by the sender.
- Send mail by courier with personal delivery.
- Request confirmation on a different channel to activate user account.

Resetting Passwords

- When setting up a new user account some delay in getting the password may be tolerated.
- If you have forgotten your password but are in the middle of an important task you need instant help.
- Procedures for resetting passwords are the same as listed previously, but now reaction should be instant.
  - Organisations must staff a hot desk round the clock,
  - On a web site, auxiliary information may authenticate a user, e.g. mother’s maiden name, phone number, name of pet, etc.
- Password support can become a major cost factor.
- Staff at hot desk needs proper security training.

Guessing Passwords

- **Exhaustive search** (brute force): try all possible combinations of valid symbols up to a certain length.
- **Intelligent search**: search through a restricted name space, e.g. passwords that are somehow associated with a user like name, names of friends and relatives, car brand, car registration number, phone number, etc. or try passwords that are generally popular.
- Typical example for the second approach: **dictionary attack** trying all passwords from an on-line dictionary.
- You cannot prevent an attacker from accidentally guessing a valid password, but you can try to reduce the probability of a password compromise.

Weak Passwords

Rank	2015	2016	2017	2018	2019
1	123456	123456	123456	123456	123456
2	password	password	password	password	123456789
3	12345678	12345	12345678	123456789	qwerty
4	qwerty	12345678	qwerty	12345678	password
5	12345	football	12345	12345	1234567
6	123456789	qwerty	123456789	111111	12345678
7	football	1234567890	letmein	1234567	12345
8	1234	1234567	1234567	sunshine	iloveyou
9	1234567	princess	football	qwerty	111111
10	baseball	1234	iloveyou	iloveyou	123123

Defences

- Set a password: if there is no password for a user account, the attacker does not even have to guess it.
- Change default passwords: often passwords for system accounts have a default value like “manager”.
  - Default passwords help field engineers installing the system; if left unchanged, it is easy for an attacker to break in.
  - Would it then be better to do without default passwords?
- Avoid guessable passwords:
  - Prescribe a minimal password length.
  - Password format: mix upper and lower case, include numerical and other non-alphabetical symbols.
  - Today on-line dictionaries for almost every language exist.

Defences

- Password ageing: set an expiry dates for passwords to force users to change passwords regularly.
- Prevent users from reverting to old passwords, e.g. keep a list of the last ten passwords used.
- Limit login attempts: the system can monitor unsuccessful login attempts and react by locking the user account.
- Inform user: after successful login, display time of last login and the number of failed login attempts since, to warn the user about recently attempted attacks.

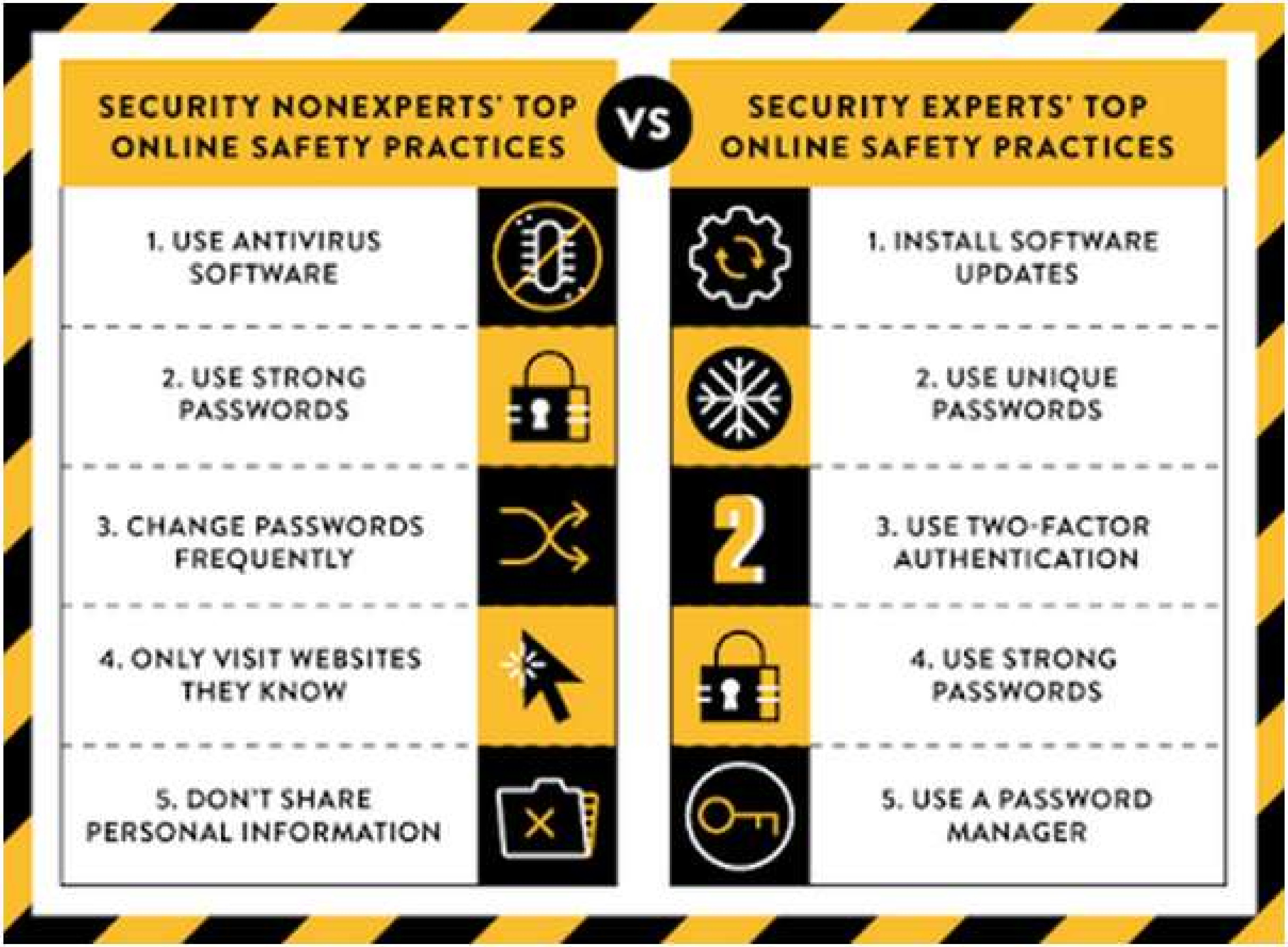
Password Security

- Is security highest if users are forced to use long passwords, mixing upper and lower case characters and numerical symbols, generated for them by the system, and changed repeatedly?
  - Users may have difficulty memorizing complex passwords.
  - Users may have difficulty dealing with frequent password changes.
  - Users may find ways of re-using their favourite password.
- Passwords will be written on a piece of paper kept close to the computer.
  - Security experts routinely look out for passwords on notes posted on computer terminals.
  - Is it always a bad idea to write down your password?

Password Security

- People are best at memorizing passwords they use regularly.
- Passwords work reasonably well in situations where they are entered quite frequently, but not so with systems used only occasionally.
- Good advice:
  - When changing a password, type it immediately several times.
  - Do not change passwords before weekends or holidays.

Non-experts vs Experts



SECURITY NONEXPERTS' TOP ONLINE SAFETY PRACTICES	VS	SECURITY EXPERTS' TOP ONLINE SAFETY PRACTICES
1. USE ANTIVIRUS SOFTWARE		1. INSTALL SOFTWARE UPDATES
2. USE STRONG PASSWORDS		2. USE UNIQUE PASSWORDS
3. CHANGE PASSWORDS FREQUENTLY		3. USE TWO-FACTOR AUTHENTICATION
4. ONLY VISIT WEBSITES THEY KNOW		4. USE STRONG PASSWORDS
5. DON'T SHARE PERSONAL INFORMATION		5. USE A PASSWORD MANAGER

Phishing and Spoofing

- Identification and authentication through username and password provide unilateral authentication.
- Computer verifies the user’s identity but the user has no guarantees about the identity of the party that has received the password.
- In phishing and spoofing attacks a party voluntarily sends the password over a channel, but is misled about the end point of the channel.

Spoofing Attacks

- Attacker starts a program that presents a fake login screen and leaves the computer.
- If the next user coming to this machine enters username and password on the fake login screen, these values are captured by the program.
  - Login is then typically aborted with a (fake) error message and the spoofing program terminates.
  - Control returned to operating system, which now prompts the user with a genuine login request.

Countermeasures

- Display number of failed logins: may indicate to the user that an attack has happened.
- Trusted path: guarantee that user communicates with the operating system and not with a spoofing program; e.g., Windows has a secure attention key CTRL+ALT+DEL for invoking the operating system logon screen.
- Mutual authentication: user authenticated to system, system authenticated to user.

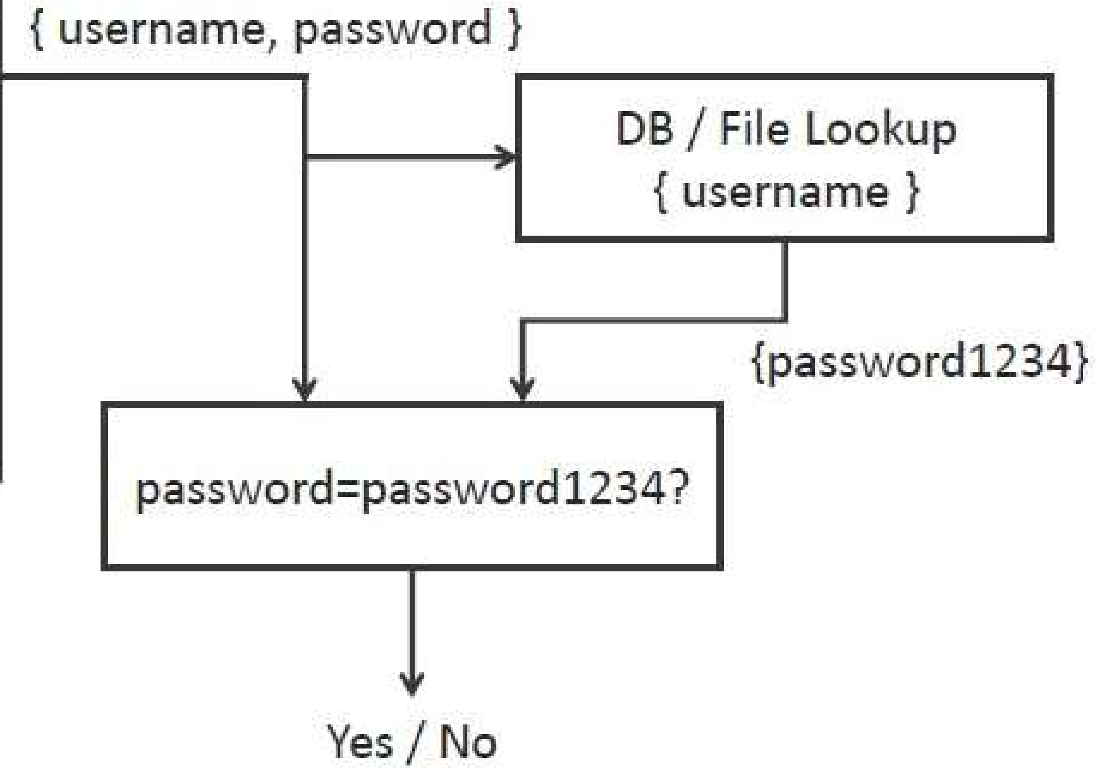
Phishing

- Phishing: attacker impersonates the system to trick a user into releasing the password to the attacker.
  - E.g., a message could claim to come from a service you are using, tell you about an upgrade of the security procedures, and ask you to enter your username and password at the new security site that will offer stronger protection.
- Take care to enter your passwords only at the “right” site (but how do you know?)
- Social engineering: attacker impersonates the user to trick a system operator into releasing the password to the attacker.

Protecting the Password File

- Operating system maintains a file with user names and passwords
- Attacker could try to compromise the confidentiality or integrity of this password file.
- Options for protecting the password file:
  - cryptographic protection,
  - access control enforced by the operating system,
  - combination of cryptographic protection and access control, possibly with further measures to slow down dictionary attacks.

Password Authentication - Bad



{ username, password }

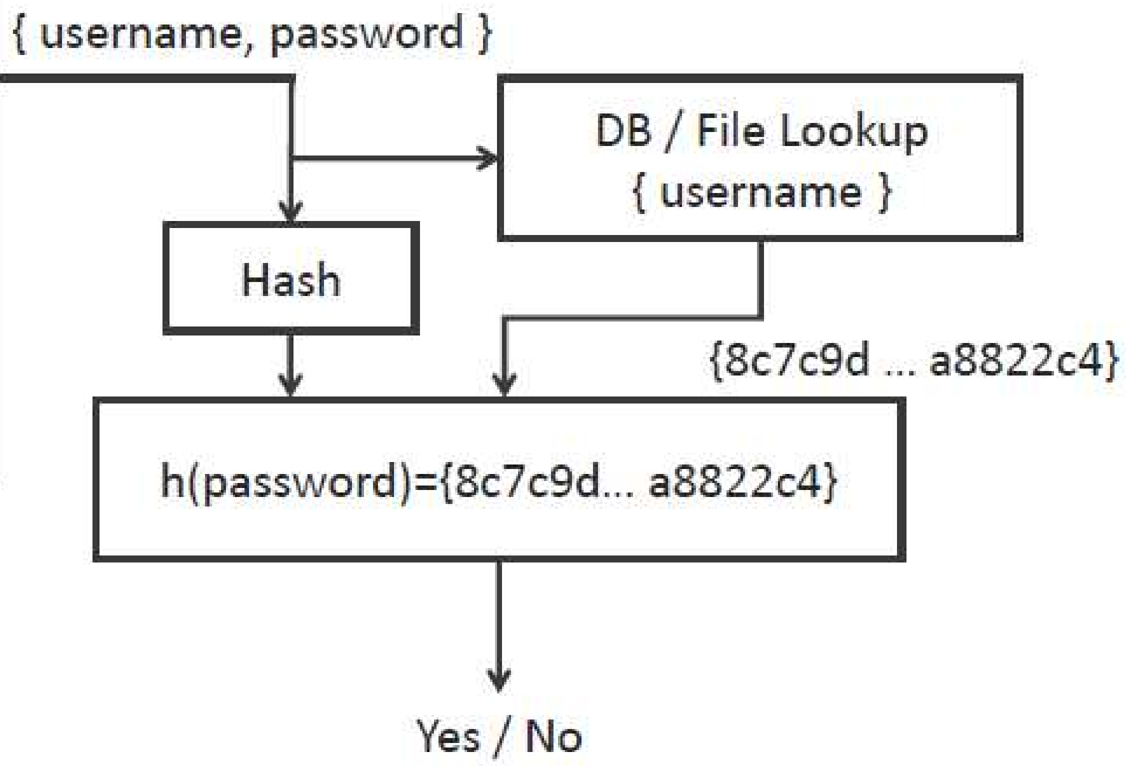
DB / File Lookup
{ username }

{password1234}

password=password1234?

Yes / No

Password Authentication - Better


{ username, password }

DB / File Lookup { username }

Hash

{8c7c9d ... a8822c4}

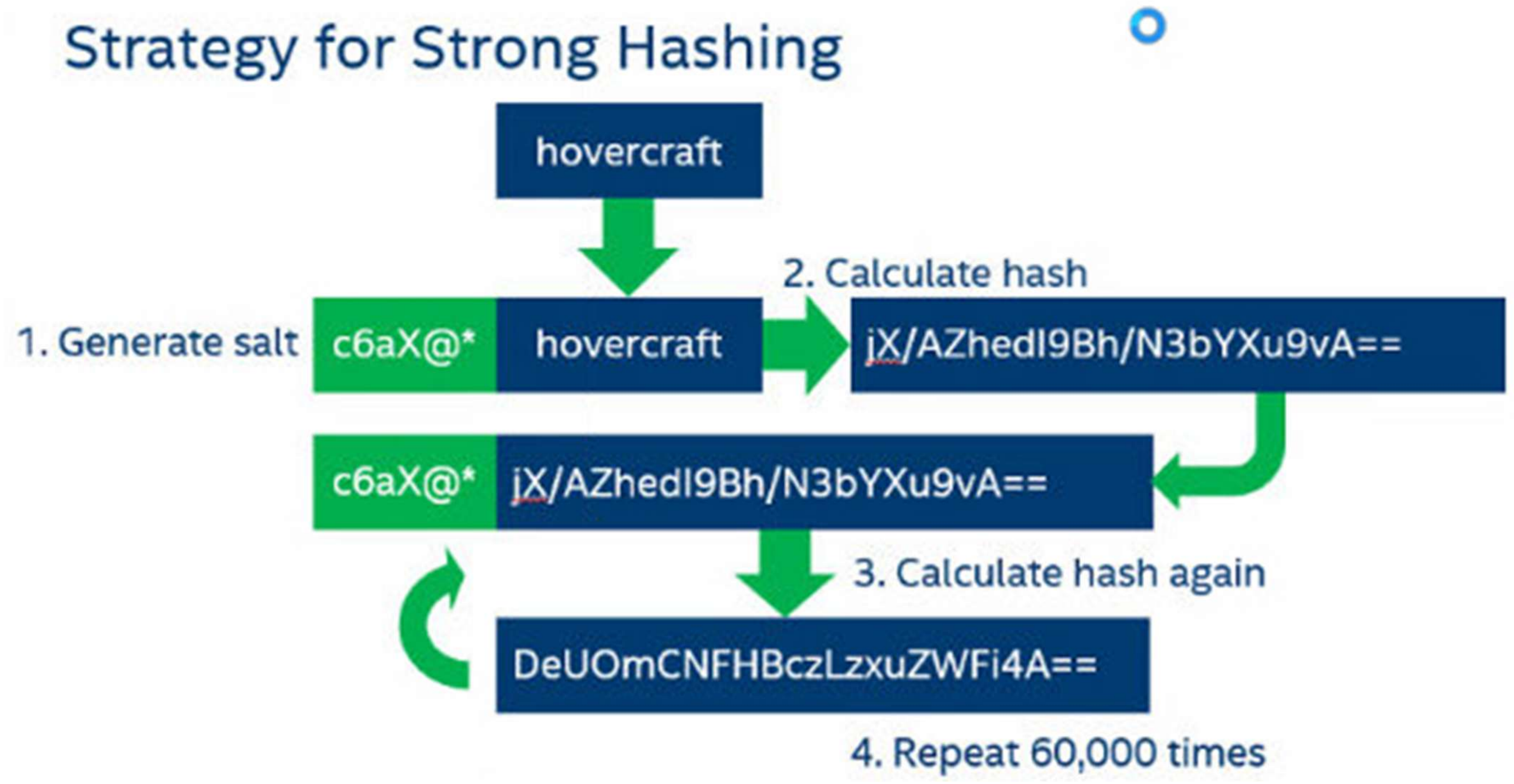
h(password)={8c7c9d... a8822c4}

Yes / No

Password Salting

- Following common practice we refer to f(x) as the encrypted password; to be precise, it is the hash of x.
- To slow down dictionary attacks, a salt is appended to the password before encryption and stored with the encrypted password.
  - If two users have the same password, they will now have different entries in the file of encrypted passwords.
  - Example: Unix uses a 12 bit salt.

Password Salting



Strategy for Strong Hashing

hovercraft

1. Generate salt

c6aX@* hovercraft

2. Calculate hash

jX/AZhedI9Bh/N3bYXu9vA==

c6aX@* jX/AZhedI9Bh/N3bYXu9vA==

3. Calculate hash again

DeUOmCNFHBczLzxuZWFi4A==

4. Repeat 60,000 times

Access Control Settings

- Only privileged users must have write access to the password file.
  - Otherwise, an attacker could get access to the data of other users simply by changing their password, even if it is protected by cryptographic means.
- If read access is restricted to privileged users, then passwords in theory could be stored unencrypted.
- If password file contains data required by unprivileged users, passwords must be “encrypted”; such a file can still be used in dictionary attacks.
  - Typical example is /etc/passwd in Unix; many versions of Unix thus store encrypted passwords in a shadow password file that is not publicly accessible.

Caching Passwords

- Our description of login has been quite abstract: password travels directly from user to the password checking routine.
- In reality, it will be held temporarily in intermediate storage locations like buffers, caches, or a web page.
- The management of these storage locations is normally beyond the control of the user; a password may be kept longer than the user has bargained for.

Single Sign-on

- Having to remember many passwords for different services is a nuisance; with a single sign-on service, you have to enter your password only once.
- A simplistic single-sign on service could store your password and do the job for you whenever you have to authenticate yourself.
  - Such a service adds to your convenience but it also raises new security concerns.
- System designers have to balance convenience and security; ease-of-use is an important factor in making IT systems really useful, but many practices which are convenient also introduce new vulnerabilities.

More on Authentication

- If you are dissatisfied with the level of security provided by passwords, what else can you do?
- In general, the following options are open.
- You can be authenticated on the basis of
  - something you know,
  - something you hold,
  - who you are,
  - what you do,
  - where you are.

Something You Know

- The user has to know some secret to be authenticated.
- Examples: password, personal identification number (PIN), personal information like home address, date of birth, name of spouse etc.
- Anybody who obtains your secret “is you”.
- You leave no trace if you pass your secret to somebody else.
- There is a case of computer misuse where somebody has logged in using your username and password.
  - Can you prove your innocence?
  - Can you prove that you have not divulged your password?

Something You Hold

- User presents a physical token to be authenticated.
- Examples: keys, cards or identity tags, smart cards.
- Physical tokens can be lost or stolen.
- Anybody who is in possession of the token has the same rights as the legitimate owner.
- To increase security, physical tokens are often used in combination with something you know, e.g. bank cards come with a PIN

Who You Are

- Biometric schemes use unique physical characteristics of a person such as face, finger prints, iris patterns, hand geometry, maybe even DNA.
- Biometrics may seem to offer the most secure solution for authenticating a person.
- We will use the example of fingerprints to sketch how biometric authentication works.

Fingerprint

- Enrolment: reference sample of the user’s fingerprint is acquired at a fingerprint reader.
- Features are derived from the sample.
  - Fingerprint minutiae: end points of ridges, bifurcation points, core, delta, loops, whorls, etc.
- For higher accuracy, record features for more than one finger.
- Feature vectors are stored in a secure database.
- When the user logs on, a new reading of the fingerprint is taken; features are compared against the reference features.

Biometrics

- Biometrics are used for two purposes:
  - Identification: 1:*n* comparison tries to identify the user from a database of *n* persons.
  - Verification: 1:1 comparison checks whether there is a match for a given user.
- Authentication by password: clear reject or accept at each authentication attempt.
- Biometrics: the stored reference template will hardly ever match precisely template derived from the current measurements.

Forged Fingers

- Fingerprints, and biometric traits in general, may be unique but they are no secrets.
- You are leaving your fingerprints in many places.
- Rubber fingers that defeat most commercial fingerprint recognition systems can be fabricated quite easily.
  - If authentication takes place in the presence of security personnel this would be a minor issue.
  - When authenticating remote users additional precautions have to be taken to counteract this type of fraud.

What You Do

- People perform mechanical tasks in a way that is both repeatable and specific to the individual.
- Experts look at the dynamics of handwriting to detect forgeries.
- Users could sign on a special pad that measures attributes like writing speed and writing pressure.
- On a keyboard, typing speed and key strokes intervals can be used to authenticate individual users.

Where You Are

- Some operating systems grant access only if you log on from a certain terminal.
  - A system manager may only log on from an operator console but not from an arbitrary user terminal.
  - Users may be only allowed to log on from a workstation in their office.
- Decisions of this kind will be even more frequent in mobile and distributed computing.
- Global Positioning System (GPS) might be used to established the precise geographical location of a user during authentication.

TOCTTOU

- Time of check to time of use.
- Example: an attacker uses an unattended machine where another user is logged on.
- The operating system checks a user's identity at the start of a session but uses the identity to make access control decisions later on during the session.
- Solution: demand authentication also at intervals within the session: lock screen automatically when a user is idle for too long.

Multi-Factor Authentication

- Combines something you know and something you have.
- Common examples:
  - Text codes to mobiles
  - One-time passwords, Google authenticator, Microsoft authenticator, etc.
  - USB devices, e.g. Yubico

Readings

- Anderson Chapter 2
- Gollman Chapter 4