CSC429 – Computer Security

LECTURE 7
AUTHENTICATION

Mohammed H. Almeshekah, PhD
meshekah@ksu.edu.sa

User Authentication

- Using a method to validate users who attempt to access a computer system or resources, to ensure they are authorized

- Types of user authentication
  - Something you know
    - User accounts with passwords
  - Something you have
    - Smart cards or other security tokens
  - Something you are
    - Biometrics.
  - Somewhere you are:
    - Location-base access.

Verification vs. Authentication

- Verification: Providing evidence of the rightful claim to a given identity.
  - Here, the user claims an identity (e.g. a username) and provides evidence of that claim (e.g. a password).
- Identification: Providing data to be matched to a role or privilege.
  - Here we may need to compare the user-presented data to a list of data for all registered users and then select the best match.
- Such identification/verification may be used in support of access control (to a border-crossing, a building, a computer, an application, a network,...).

Scenarios Requiring Authentication

- Scenarios
  - Logging into a local computer
  - Logging into a computer remotely
  - Access web sites
- Potential vulnerabilities to consider when client authenticating server
  - channel between the client and the server
  - server compromise
  - client compromise
  - social engineering
  - weak passwords

Threats to Passwords

- Offline dictionary attacks
- Online guessing attempts
- Login spoofing
- Shoulder surfing
- Social engineering
  - e.g., pretexting.

Storing Passwords (UNIX Case Study)

- Old UNIX
  - The file /etc/passwd stores H(password) together with each user's login name, user id, home directory, login shell, etc.
  - file must be world readable
- New UNIX
  - H(password) stored in /etc/shadow, readable only by root
- Brute force attacks possible even if H is one-way
  - how to brute-force when trying to obtain password of any account on a system with many accounts?
  - How to fix it?

Password Salts

- Store [r, H(password,r)] rather than H(password)
  - **r** is randomly chosen for each password
  - **r** is public
  - similar to Initial Vector in CBC & CTR modes
- Benefits
  - Obtaining **any** account becomes difficult.
    - cost of attacking a single account remains the same
  - if two users happen to choose the same password, it doesn't immediately show

Dictionary and Guessing Attacks

- Protect stored passwords (use both cryptography & access control).
- Disable accounts with multiple failed attempts.

Weak Passwords

- Allow long passphrases.
- Randomly generate passwords.
- Check the quality of user-selected passwords
  - use a number of rules
  - run dictionary attack tools
- Give user suggestions/guidelines in choosing passwords
  - e.g., think of a sentence and select letters from it, “It’s 12 noon and I am hungry” => “I’S12&IAH”
  - Using both letter, numbers, and special characters
- Mandate password expiration
- Things to remember: Usability issues

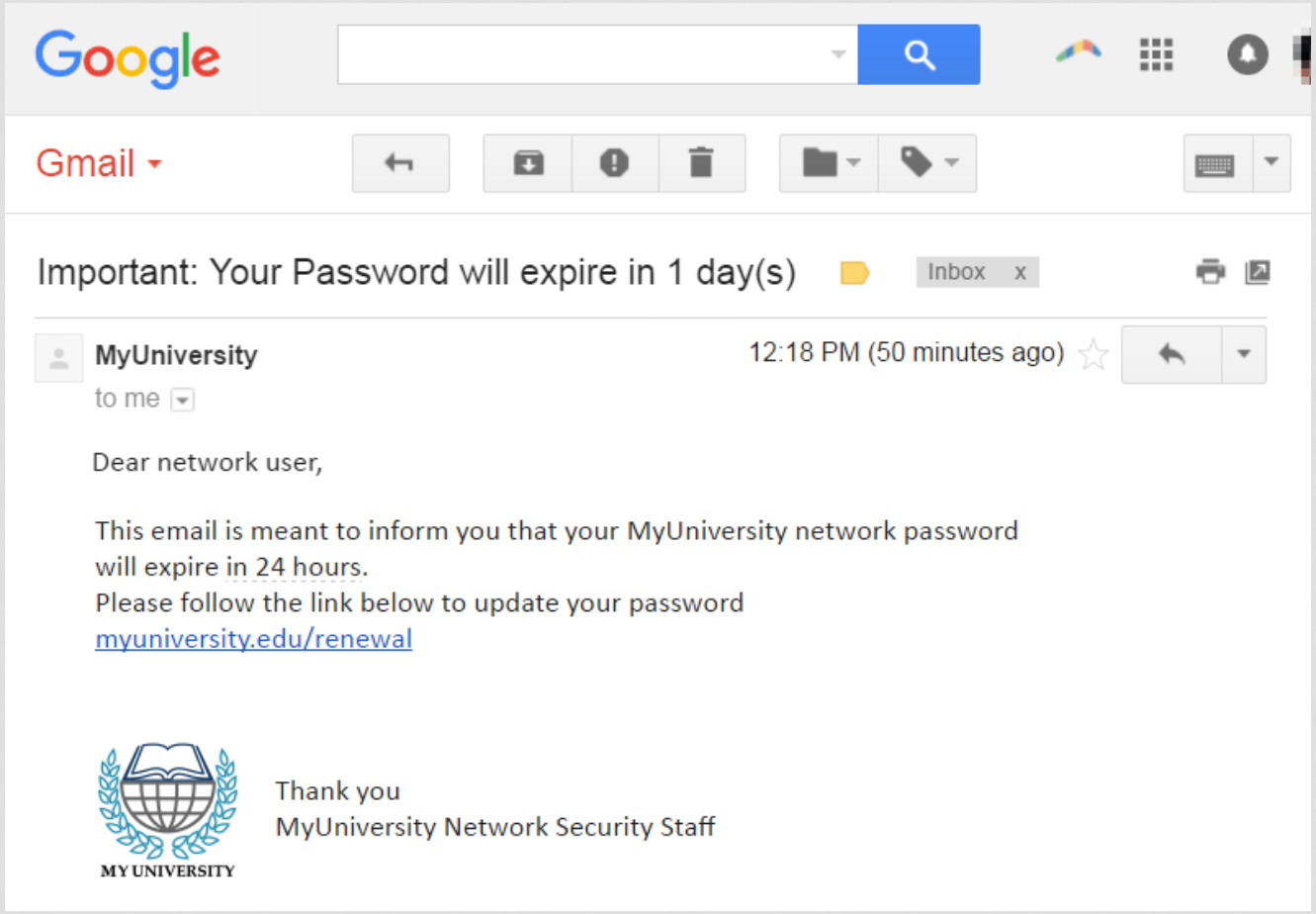
Trusted Path

- Attacks:
  - write a program showing a login window on screen and record the passwords.
- Defense: Trusted Path
  - Mechanism that provides confidence that the user is communicating with what the real server (typically Trusted Computing Base in OSes)
    - attackers can't intercept or modify whatever information is being communicated.
    - defends attacks such as fake login programs
  - Example: Ctrl+Alt+Del for log in on Windows
- How to apply this to web?!

Phishing Threats

- The Cost of a phishing attach for a mid-size company is $1.6 million
- According to SANS, 95% of all enterprise attacks are a result of a successful phishing.
- Different kind of phishing:
  - Phishing
  - Spear phishing
  - Whaling

Phishing Example



Other Threats

- Use ideas from recent research:
  - graphical passwords
  - Keystrokes
- Go beyond passwords
  - security tokens
  - biometrics
  - 2-factor authentication

Using Passwords Over Insecure Channels

- One-time passwords
  - Each password is used only once
  - Defend against passive adversaries who eavesdrop and later attempt to impersonate
- Challenge response
  - Send a response related to both the password and a challenge
- Zero knowledge proof of knowledge

One-Time Passwords

- Time-synchronized OTP
  - Based on a seed.
- Shared lists of one-time passwords
  - What happens when you exhaust the list?!
- Using a hash chain (Lamport)

Lamport Hash Chains

- One-time setup:
  - A selects a value w, a hash function H(), and an integer t, computes $w_0 = H^t(w)$ and sends w_0 to B
  - B stores w_0
- Protocol: to authenticate to B for the i^{th} time, $1 \leq i \leq t$
  - A sends to B: A, i, $w_i = H^{t-i}(w)$
  - B checks $i = i_A$, $H(w_i) = w_{i-1}$
  - if both holds, $i_A = i_A + 1$
- Example:
  - $h(s), h(h(s), h(h(h(s))), \ldots, h^{1000}(s)$

Challenge-Response Protocols

- Goal: one entity authenticates to other entity proving the knowledge of a secret, 'challenge'

- Approach:
  - Use time-variant parameters to prevent replay, interleaving attacks, provide uniqueness and timeliness
  - e.g., nonce (used only once), timestamps

Challenge-Response Examples

- Unilateral authentication, timestamp-based
  - A → B: $MAC_K(t_A, B)$
- Unilateral authentication, nonce-based
  - B → A: r_B
  - A → B: $MAC_K(r_B, B)$
- Mutual authentication, nonce-based
  - B → A: r_B
  - A → B: r_A, $MAC_K(r_A, r_B, B)$
  - B → A: $MAC_K(r_B, r_A)$

Issues to Consider in Password Systems

- Which types of attacks to defend against?
  - targeted attack on one account
  - attempt to penetrate any account on a system
  - attempt to penetrate any account on any system
  - denial of service attack
- Whether to protect users against each other?
- Can users be trained? Will they follow the suggestions?
- Will the passwords be used in other systems?
- Whether the passwords will be used in a controlled environment?

Biometrics

- Authentication occurs through factors arising from human physical or behavioural characteristics.
- Biometrics (bios = life, metron = measure) measures features of certain body parts.
- Biometrics can only ever provide a probabilistic measure of authentication.
  - As we shall see, biometrics systems are intrinsically subject to errors that limit performance.

Key Requirements for Biometrics

1. Persistence (Stability): Low rate of change over required period of time.
2. Distinctness (Uniqueness): High variance between individuals (preferably even among identical twins).
3. Universality: Should be present in as many individuals as possible.
4. Detectability (Collectability): Features must be efficiently detectable and collectable.
5. Fraud Resistance: Features should be difficult to reproduce and must provide means for testing *liveness*.
6. Acceptability: Method must be acceptable to users.
7. Performance: Method have reasonable accuracy, and speed and processing requirements.

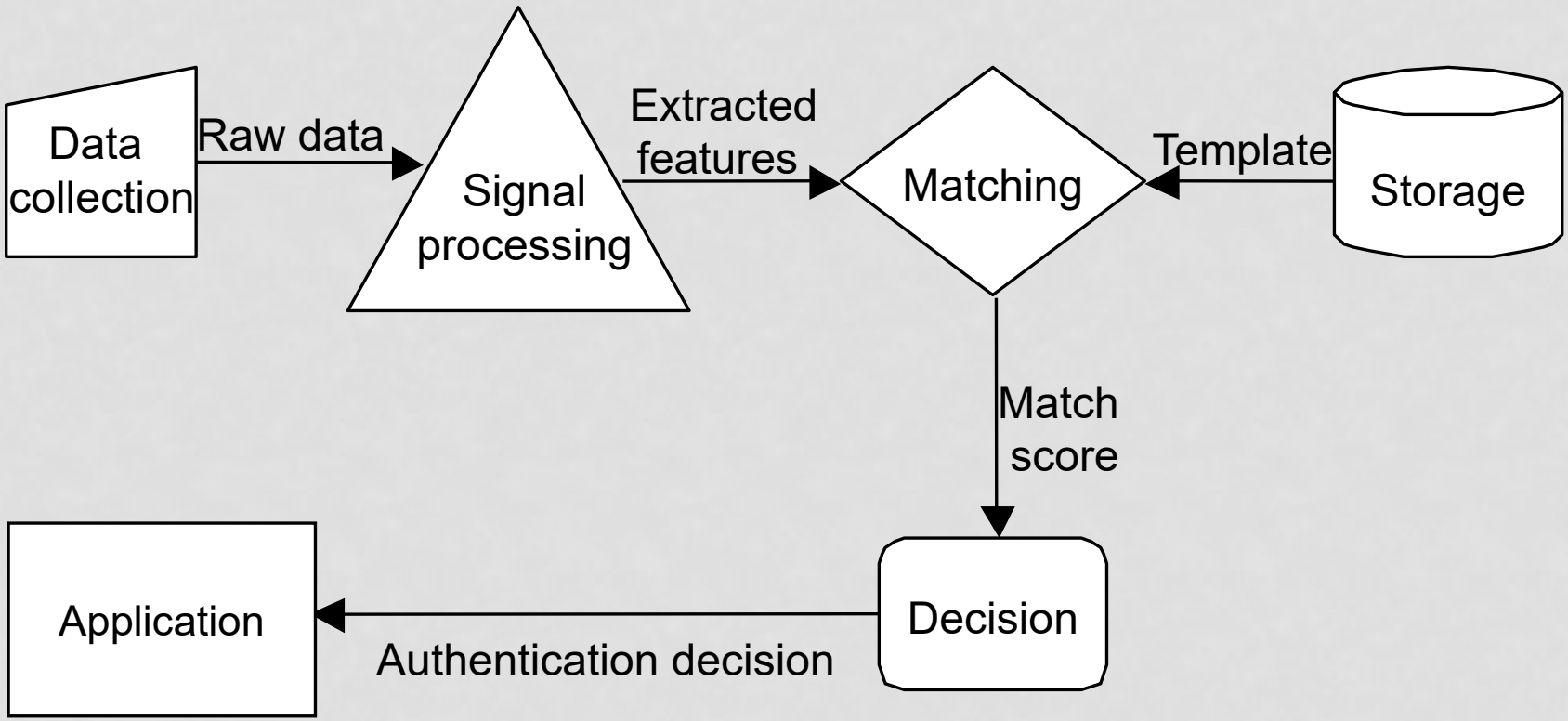
Examples of Biometric Techniques

- Fingerprint biometrics:
  - Fingerprint recognition.
- Eye biometrics:
  - Iris and retinal scanning.
- Face biometrics:
  - Face recognition using visible or infrared light (called facial thermography) in 2D or 3D.
- Hand geometry biometrics.
- Signature biometrics:
  - Signature recognition, including signature dynamics.
- Voice biometrics:
  - Speaker recognition.

"Exotic" Biometric Techniques

- Additional examples of biometric techniques (mostly research-oriented):
  - Vein pattern recognition (e.g. hand).
  - Palm-print recognition.
  - Gait recognition.
  - Body odour measurements.
  - Ear shape (an implicit part of face recognition).
  - Keystroke dynamics.
  - DNA/RNA.

Generic Biometric System Architecture



Errors in Boimetrics

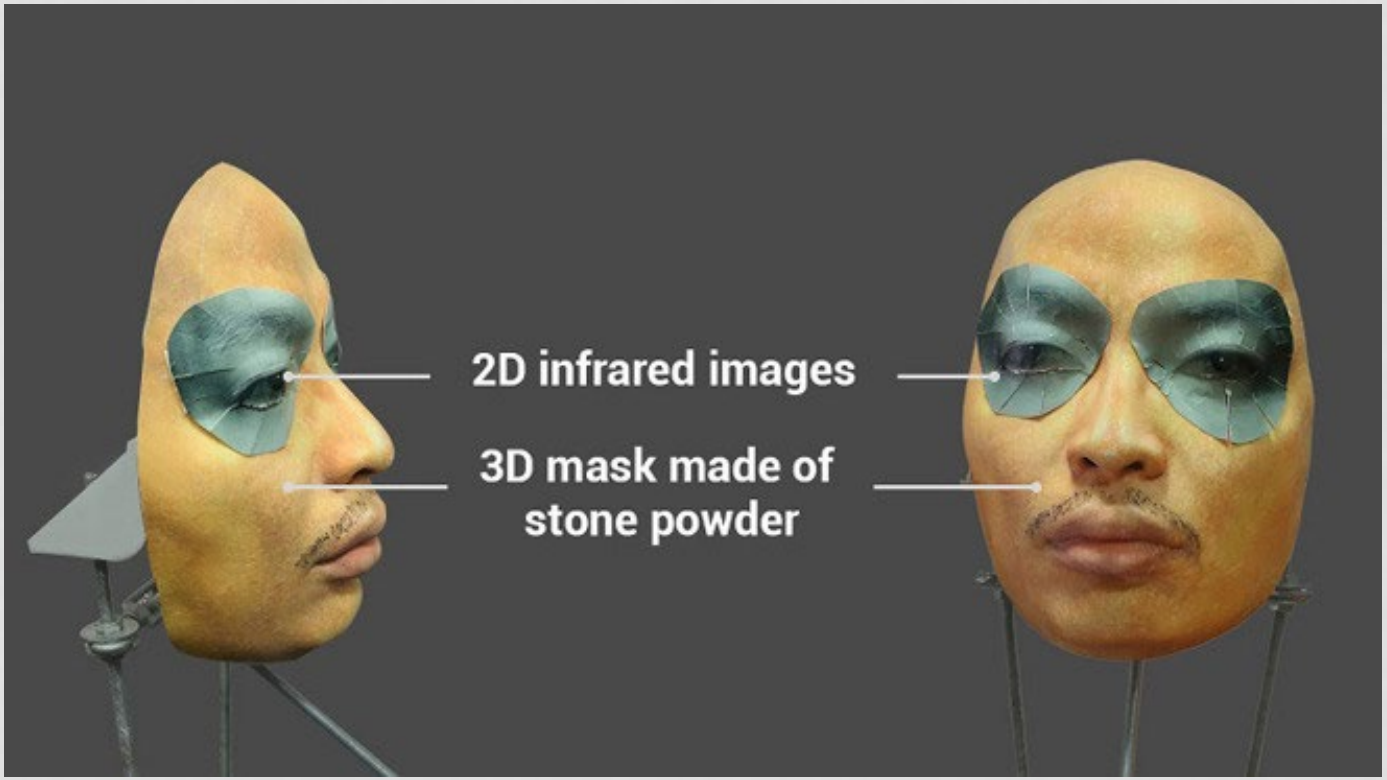
- Rejection of genuine claim:
  - Referred to as Type I error or false negative.
- Acceptance of impostor claim:
  - Known as Type II error or false positive.
- These errors are not independent!

Attacks on Boimetrics

- Thin latex glue imprinted with fingerprint taken from another individual (e.g. a latent print from a surface) can be hard to detect.
- Famous "gummy finger" attack by Matsumoto *et al.*

Attacks on Face Recognition

https://www.youtube.com/watch?v=rhiSBc061JU



Next Lecture

- Access Control.

- Readings for next lecture:
  - UNIX File and Directory Permissions and Modes
    - http://www.hccfl.edu/pollock/AUnix1/FilePermissions.htm
  - Unix file permissions
    - http://www.unix.com/tips-tutorials/19060-unix-file-permissions.html

  - Anderson's book – sections 4.1, 4.2.1, 4.2.2, 4.2.3 and 4.2.6,