Unit 2 Authentication

By
Prof. Pranita Binnar (Research Scholar, Ph.D. Pursuing)
Visiting Faculty, Dept. of Computer Engineering
NMIMS, Navi Mumbai Campus
Email Contact – pranitasadgir@gmail.com
Contact No. - 9699502992

Topics to be covered

- Authentication basics
- Password
- Challenge response
- Biometrics
- SSO

Authentication basics

- **Definition - *Authentication* is the binding of an identity to subject.**
  - Subjects act on behalf of some other, external entity.
  - The identity of that entity controls the actions that its associated subjects may take.
  - Hence, the subjects must bind to the identity of that external entity.

Authentication basics

- The external entity must provide information to enable the system to confirm its identity. This information comes from one (or more) of the following.

1. What the entity knows (such as passwords or secret information such as PIN, Social security number, Mother's maiden name, Date of birth, Name of your pet, etc.)

2. What the entity has (such as a badge or card or tokens)

3. What the entity is (such as fingerprints or retinal characteristics)

4. Where the entity is (such as in front of a particular terminal)

Authentication Process

- The authentication process consists of obtaining the authentication information from an entity, analyzing the data, and determining if it is associated with that entity.
- This means that the computer must store some information about the entity. It also suggests that mechanisms for managing the data are required. We represent these requirements in an *authentication system* consisting of five components.

Authentication System

1. The set *A* of *authentication information* is the set of specific information with which entities prove their identities.

2. The set *C* of *complementary information* is the set of information that the system stores and uses to validate the authentication information.

3. The set *F* of *complementation functions* that generate the complementary information from the authentication information. That is, for $f \in F$, $f: A \rightarrow C$.

4. The set *L* of *authentication functions* that verify identity. That is, for $l \in L$, $l: A \times C \rightarrow \{$ **true**, **false** $\}$.

5. The set *S* of *selection functions* that enable an entity to create or alter the authentication and complementary information.

- The goal of an authentication system is to ensure that entities are correctly identified. If one entity can guess another's password, then the guesser can impersonate the other.

Example

- A user authenticates himself by entering a password, which the system compares with the cleartext passwords stored online. Here, *A* is the set of strings making up acceptable passwords,

  $C = A$, $F = \{ I \}$, and $L = \{$ **eq** $\}$

where *I* is the identity function and **eq** is **true** if its arguments are the same and **false** if they are not.

Passwords

Definition : A *password* is information associated with an entity that confirms the entity's identity.

- Passwords are an example of an authentication mechanism based on what people know: the user supplies a password, and the computer validates it.
- If the password is the one associated with the user, that user's identity is authenticated.
- If not, the password is rejected and the authentication fails.

Password

- The simplest password is some sequence of characters. In this case, the *password space* is the set of all sequences of characters that can be passwords.
- EXAMPLE: One installation requires each user to choose a sequence of 10 digits as a password. Then A has 10^{10} elements (from "0000000000" to "9999999999").

Why Passwords?

- Why is “something you know” more popular than “something you have” and “something you are”?
- **Cost**: passwords are free
- **Convenience**: easier for sysadmin to reset pwd than to issue a new thumb

Good and Bad Passwords

- Bad passwords
  - frank
  - Fido
  - Password
  - incorrect
  - Pikachu
  - 102560
  - AustinStamp
- Good Passwords?
  - jflej,43j-EmmL+y
  - 09864376537263
  - P0kem0N
  - FSa7Yago
  - 0nceuP0nAt1m8
  - PokeGCTall150

Attacks on Passwords

- Attacker could…
  - Target one particular account
  - Target any account on system
  - Target any account on any system
  - Attempt denial of service (DoS) attack
- Common attack path
  - Outsider → normal user → administrator
  - May only require **one** weak password!

Password Attacks

- Phishing. Phishing is when a hacker posing as a trustworthy party sends you a fraudulent email, hoping you will reveal your personal information voluntarily. ...
- Man-in-the-middle attack. Man-in-the middle (MitM) attacks are when a hacker or compromised system sits in between two uncompromised people or systems and deciphers the information they're passing to each other, including passwords.
- Brute force attack. a brute-force attack consists of an attacker submitting many passwords or passphrases with the hope of eventually guessing correctly. The attacker systematically checks all possible passwords and passphrases until the correct one is found.

Password Attacks

- Dictionary attack. A dictionary attack is a systematic method of guessing a password by trying many common words and their simple variations
- Credential stuffing. Credential stuffing is a type of cyberattack in which stolen account credentials, typically consisting of lists of usernames and/or email addresses and the corresponding passwords (often from a data breach), are used to gain unauthorized access to user accounts through large-scale automated login requests directed against a web application
- Keyloggers. a computer program that records every keystroke made by a computer user, especially in order to gain fraudulent access to passwords and other confidential information.

Attacking a Password System

1. A dictionary attack is the guessing of a password
2. Countering Password Guessing
3. Random Selection of Passwords
4. Pronounceable and Other Computer-Generated Passwords
5. User Selection of Passwords
6. Reusable Passwords and Dictionary Attacks
7. Password Aging

Attacking a Password System

- The simplest attack against a password-based system is to guess passwords.
- **Definition :** A *dictionary attack* is the guessing of a password by repeated trial and error.
- The name of this attack comes from the list of words (a “dictionary”) used for guesses.
- The dictionary may be a set of strings in random order or (more usually) a set of strings in decreasing order of probability of selection.

Countering Password Guessing

- Password guessing requires either the set of complementation functions (F) and complementary information (C) or access to the authentication functions (L).
- In both approaches, the goal of the defenders is to maximize the time needed to guess the password.
- Let *P* be the probability that an attacker guesses a password in a specified period of time T. Let *G* be the number of guesses that can be tested in one time unit. Let *T* be the number of time units during which guessing occurs. Let *N* be the number of possible passwords. Then,

$$P \geq \frac{TG}{N}$$

Random Selection of Passwords

- **Random Selection of Passwords**

- **Pronounceable and Other Computer-Generated Passwords**
- Pronounceable passwords are based on the unit of sound called a *phoneme*. In English, phonemes for constructing passwords are represented by the character sequences *cv*, *vc*, *cvc*, or *vcv*, where *v* is a vowel and *c* a consonant.
  - EXAMPLE: The passwords “helgoret” and “juttelon” are pronounceable passwords; “przbqxdf” and “zxrptglfn” are not.

User Selection of Passwords

- Rather than selecting passwords for users, one can constrain what passwords users are allowed to select. This technique, called *proactive password selection*, enables users to propose passwords they can remember, but rejects any that are deemed too easy to guess.

1. Passwords based on account names
    - a. Account name followed by a number
    - b. Account name surrounded by delimiters
2. Passwords based on user names
    - a. Initials repeated 0 or more times
    - b. All letters lower- or uppercase
    - c. Name reversed
    - d. First initial followed by last name reversed
3. Passwords based on computer names
4. Dictionary words
5. Reversed dictionary words
6. Dictionary words with some or all letters capitalized

User Selection of Passwords

- Conjugations or declensions of dictionary words
- Patterns from the keyboard
- Passwords shorter than six characters
- Passwords containing only digits
- Passwords containing only uppercase or lowercase letters, or letters and numbers, or letters and punctuation
- Passwords that look like license plate numbers
- Concatenations of dictionary words
- Dictionary words preceded or followed by digits, punctuation marks, or spaces
- Dictionary words with all vowels deleted
- Dictionary words with white spaces deleted

Reusable Passwords and Dictionary Attacks

- Reusable passwords are quite susceptible to dictionary attacks. The goal of random passwords, pronounceable passwords, and proactive password checking is to maximize the time needed to guess passwords.
- **Password Aging**
- Guessing of passwords requires that access to the complement, the complementation functions, and the authentication functions be obtained. If none of these have changed by the time the password is guessed, then the attacker can use the password to access the system.

Password Cracking Tools

- Popular password cracking tools
  - Password Crackers
  - Password Portal
  - L0phtCrack and LC4 (Windows)
  - John the Ripper (Unix)
- Admins should use these tools to test for weak passwords since attackers will
- Good articles on password cracking
  - Passwords - Conerstone of Computer Security
  - Passwords revealed by sweet deal

Challenge-Response

- Passwords have the fundamental problem that they are *reusable*. If an attacker sees a password, she can later *replay* the password.
- The system cannot distinguish between the attacker and the legitimate user, and allows access.
- An alternative is to authenticate in such a way that the transmitted password changes each time. Then, if an attacker replays a previously used password, the system will reject it.

Definition

1. Let user *U* desire to authenticate himself to system *S*.
2. Let *U* and *S* have an agreed-on secret function *f*.
3. A *challenge-response* authentication system is one in which *S* sends a random message *m* (the challenge) to *U*, and *U* replies with the transformation $r = f(m)$ (the response).
4. *S* validates *r* by computing it separately.

Pass Algorithms

- **Definition:** Let there be a challenge-response authentication system in which the function *f* is the secret. Then *f* is called a *pass algorithm*.
- Under this definition, no cryptographic keys or other secret information may be input to *f*.
- The algorithm computing *f* is itself the secret.

One-Time Passwords

- **Definition :** A *one-time* password is a password that is invalidated as soon as it is used.
- The ultimate form of password aging occurs when a password is valid for exactly one use. In some sense, challenge-response mechanisms use one-time passwords.
- Think of the response as the password. As the challenges for successive authentications differ, the responses differ.
- Hence, the acceptability of each response (password) is invalidated after each use.
- The challenge is the number of the authentication attempt; the response

is the one-time password.

Hardware-Supported Challenge-Response Procedures

- One-time passwords are considerably simpler with hardware support because the passwords need not be printed on paper or some other medium.
- Hardware support comes in two forms: Both perform the same functions.

1. A program for a general-purpose computer.
2. Special-purpose hardware support.

A program for a general-purpose computer.

- The first type of hardware device, informally called a *token*, provides mechanisms for hashing or enciphering information.
- With this type of device, the system sends a challenge.
- The user enters it into the device.
- The device returns the appropriate response.
- Some devices require the user to enter a personal identification number or password, which is used as a cryptographic key or is combined with the challenge to produce the response.

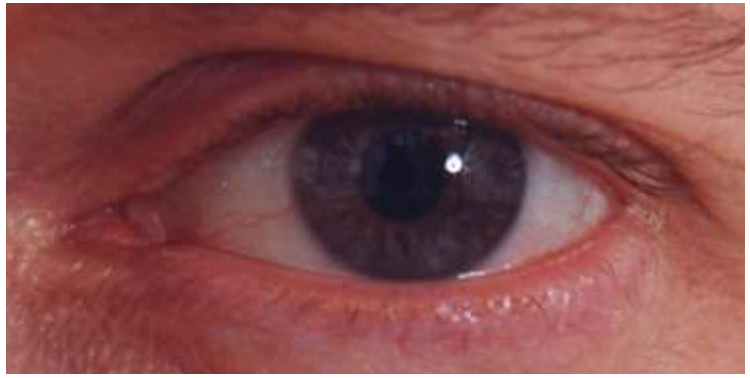
Special-purpose hardware support.

- The second type of hardware device is temporally based. Every 60 seconds, it displays a different number. The numbers range from 0 to $10n - 1$, inclusive.
- A similar device is attached to the computer. It knows what number the device for each registered user should display.
- To authenticate, the user provides his login name. The system requests a password.
- The user then enters the number shown on the hardware device, followed by a fixed (reusable) password.
- The system validates that the number is the one expected for the user at that time and that the reusable portion of the password is correct.

Identification vs Authentication

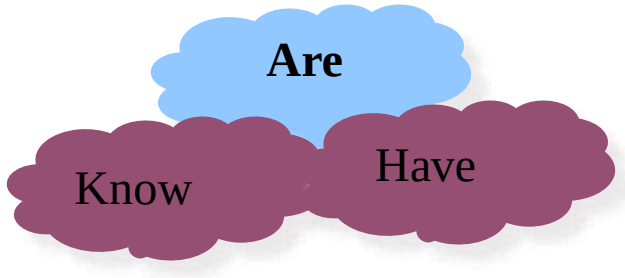
- **Identification** — Who goes there?
  - Compare **one-to-many**
  - Example: FBI fingerprint database
- **Authentication** — Are you who you say you are?
  - Compare **one-to-one**
  - Example: Thumbprint mouse
- Identification problem is more difficult
  - More “random” matches since more comparisons
- We are (mostly) interested in authentication

Biometrics



Something You Are

- Biometric
  - **"You are your key"** — Schneier
- Examples
  - Fingerprint
  - Handwritten signature
  - Facial recognition
  - Speech recognition
  - Gait (walking) recognition
  - "Digital doggie" (odor recognition)
  - Many more!



Biometrics Authentication

- May be better than passwords
- But, cheap and reliable biometrics needed
  - Today, an active area of research
- Biometric authentication involves using some part of your physical makeup to authenticate you.
- Biometrics **are** used in security today
  - Thumbprint mouse
  - Palm print for secure entry
  - Fingerprint to unlock car door, etc.

Biometrics Authentication

- A single characteristic or multiple characteristics could be used.
- It all depends on the infrastructure and the level of security desired.
- With biometric authentication, the physical characteristic being examined is usually mapped to a username. This username is used to make decisions after the person has been authenticated.
- But biometrics not really that popular

Single Sign on (SSO)

- Authenticating to multiple systems is unpopular with users.
- Left on their own, users will reuse the same password to avoid having to remember many different passwords.
- For example, users become frustrated at having to authenticate to a computer, a network, a mail system, an accounting system, and numerous web sites.
- The panacea for this frustration is called **single sign-on**.
- A user authenticates once per session, and the system forwards that authenticated identity to all other processes that would require authentication.

Security Assertion Markup Language (SAML)

- SAML is a standard for logging users into applications based on their sessions in another context. This single sign-on (SSO) login standard has significant advantages over logging in using a username/password:
  - No need to type in credentials
  - No need to remember and renew passwords
  - No weak passwords
- Most organizations already know the identity of users because they are logged in to their Active Directory domain or intranet.
- It makes sense to use this information to log users in to other applications, such as web-based applications, and one of the more elegant ways of doing this is by using SAML.

SSO Protocol

- Kerberos --- example single sign-on protocol
- Kerberos is a computer-network authentication protocol that works on the basis of tickets to allow nodes communicating over a non-secure network to prove their identity to one another in a secure manner.
- Its designers aimed it primarily at a client–server model, and it provides mutual authentication—both the user and the server verify each other's identity.

Thank you