Justin Ciocoi

Feb. 15, 2024

CSCI 400 Textbook Notes

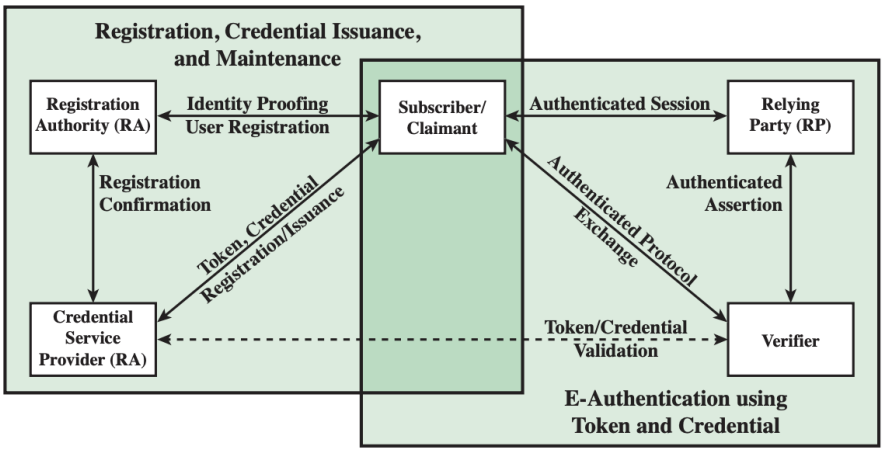
Chapter 3: User Authentication

3.1: Digital User Authentication Principles

- NIST defines digital user authentication as the process of establishing confidence in user identities that are presented electronically to an information system
- The below table illustrates some basic security requirements regarding authentication and their derived requirements

No.	Basic Security Requirement	Derived Security Requirement
1	Identify information system users, processes acting on behalf of users, or devices	Use multi-factor authentication
2	Authenticate or verify the identities as a prerequisite to allowing access to organizational information system	Employ replay resistant authentication mechanisms
3		Prevent reuse of identifiers for a defined period (i.e. login timeouts)
4		Disable identifiers after a defined period of inactivity
5		Enforce a minimum password complexity and change of characters when new passwords are created
6		Prohibit password reuse for a specified number of generations
7		Allow temporary password use for system logons with an immediate change to a permanent password
8		Store and transmit only cryptographically protected passwords (such as hashed passwords using a secure hash function)
9		Obscure feedback of authentication information

- Similarly, a broad model of user authentication can be seen in the below diagram



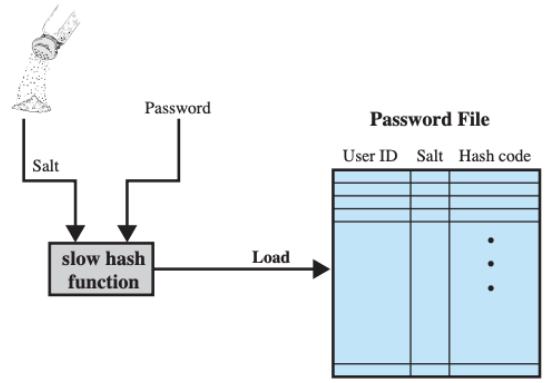
- The following sequence is typical for registration of a digital identity
    - An applicant applies to a **Registration Authority (RA)** to become a subscriber of a **Credential Service Provider (CSP)**
        - The RA is a trusted entity that established and vouches for the identity of the applicant
    - The CSP will then issue some sort of electronic credential to the subscriber
- There are four general means of authenticating a user's identity which can be used either alone or in combination
    - **Something the Individual Knows**, like a password, PIN, or answer to a security question
    - **Something the Individual Possesses**, like an electronic key card or physical key
    - **Something the Individual Is (static biometrics)** such as fingerprint, retina, or face scans
    - **Something the Individual Does (dynamic biometrics)** such as voice recognition or handwriting characteristics
- **Multi-factor Authentication** refers to the implementation of user authentication using two or more of the above factors of ascertaining identity
- An **Assurance Level** describes the level of certainty that an organization has with respect to the validity of user credentials
    - This consists of the degree of confidence in the actual authentication process as well as the confidence in that the individual using the credential isa the one to whom the credential was issued

    - **Level 1** refers to little or no confidence in the asserted identity's validity
    - **Level 2** refers to some confidence in the asserted identity's validity
    - **Level 3** refers to high confidence in the asserted identity's validity
        - This generally refers to the use of at least two factors of security in order to authenticate users
    - **Level 4** refers to very high confidence in the asserted identity's validity
        - Level 4 generally includes multi-factor authentication as well as in-person registration
- There are three defined levels of potential should there be a failure in user authentication
    - **Low** where an error could be expected to have a limited adverse effect on the organization
    - **Moderate** where an error could be expected to have a serious adverse effect on the organization
    - **High** where an error could be expected to have a severe adverse effect on the organization

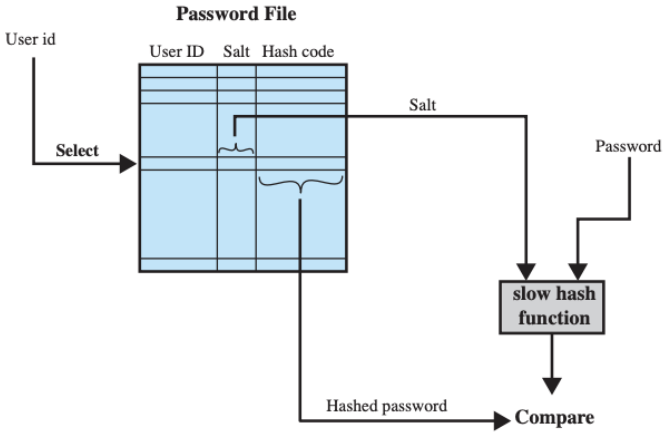
3.2: Password-Based Authentication

- What are the main forms of attack against password-based authentication?
    - **Offline Dictionary Attack**
        - Where a password file is recovered, and hashes are computed in a brute force manner to crack the password
        - The attacker will compare hashes to hashes found on a system in order to break the password
        - Countermeasures include preventing unauthorized access to the password file and intrusion detection mechanisms
    - **Specific Account Attack**
        - This attack guesses password for a single user account until a correct password is found

        - The best countermeasure for this attack is to implement a lockout mechanism locking a user out for a defined period after a specified number of invalid logon attempts
    - **Popular Password Attack**
        - Where a large repository of common passwords is used in a brute-force style attack
        - Countermeasures would including forcing users to adhere to a minimum level of complexity when creating passwords
    - **Workstation Hijacking**
        - Where an attacker approaches a physical workstation that has been left logged in
        - Standard countermeasures for this include implementing an automatic OS lockout after a defined period of inactivity
    - **Exploiting Multiple Password Use**
        - An attacker can exploit password reuse if a password is in use across multiple devices or accounts
        - A common countermeasure is to require unique passwords
- Despite their relative vulnerability, password based authentication is still the most popular form of authentication and thus should be focused on when discussing means of user authentication
- Hashed passwords are often used such that an organization does not store the plaintext password of users and is thus unable to leak these passwords
- Upon every logon attempt, the entered password can be hashed and compared to the hashed value on the organization side and if there is a match, then authentication is granted



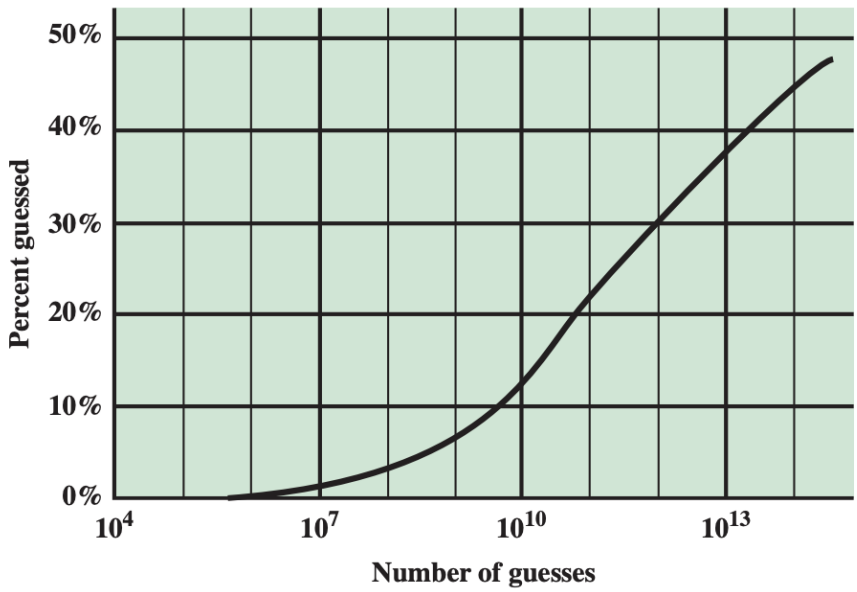
(a) Loading a new password



(b) Verifying a password

- Here, we can see the use of a "salt" in the hashing algorithm, which serves three important purposes
  - This prevents duplicate passwords from being visible in the password file since different salts are used for identical passwords
  - For a salt of length b bits, the number of possible passwords is increased by a factor of 2^b
  - It becomes virtually impossible to tell is a user with passwords on multiple machines has used the same one
- **Password Cracking**
  - **Traditional Approaches**
    - Dictionary attacks where large dictionaries of potential passwords are tried against the contents of the password file using the proper salts and hashing algorithms denoted in the password file

        - Rainbow Table Attacks pre-compute tables of hash values for all salts, but can be countered through the use of a sufficiently large salt
        - Password crackers are software packages which allow attackers to exploit the fact that people often use short, easily guessable passwords
            - *John the Ripper* is an open-source password cracker first developed in 1996 and it uses a combination of brute force and dictionary techniques
    - **Modern Approaches**
        - Password policies have largely improved through computing's history
            - Minimum complexity requirements
            - Regular password changing
        - However, password cracking approaches have also greatly improved
            - Processing capacity of password cracking hardware has increased
            - Complex algorithms used for password generation based on passwords in use have also been developed
    - In a pure brute force attack, the following graph illustrates the relationship between number of guesses and percentage of possible passwords guessed



- **Password File Access Control**

    - Hashed password information is often stored in a file separately from user ID usually referred to as a **shadow password file**
    - Unanticipated break-ins could lead to unintended unauthorized access to a machines password file
    - An error could occur rendering the password file readable and exposing the data of all accounts
    - A lack of physical security could also allow an attacker access to a physical machine that will allow them access either directly or indirectly to the password file
    - Capturing network traffic is another approach that functions in a similar way to password file exfiltration on the side of the attacker
- Because of these issues, there must be policies in place with regard to user password selection
- There are four general techniques currently in use meant to achieve this goal
    - User Education
    - Computer-Generated Passwords
    - Reactive Password Checking
    - Minimum Password Complexity Requirements
- Password rules such as minimum complexity should be enforced by system admins, and passwords should also be checked for weakness at the time of creation by comparison to a large database of passwords that are considered weak

3.3: Token Based Authentication

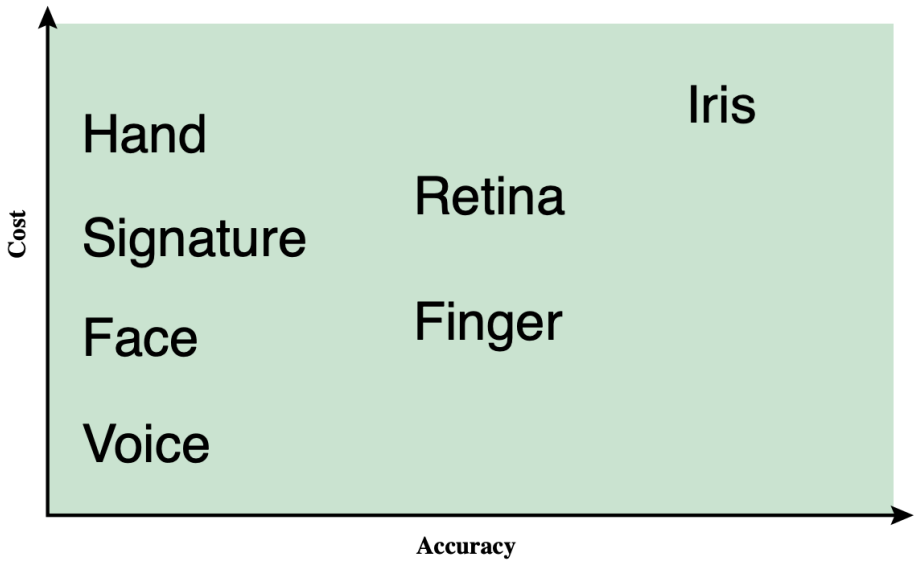
- **Memory Cards**
    - Memory cards are capable of storing, but not processing, data
    - The most common of these is the magnetic stripe card
    - Might include internal electronic memory
    - These can be used alone for physical access, such as in the case of a hotel room key
    - Memory cards provide significantly greater security when combined with a PIN, such as with an ATM card

    - There are several drawbacks of memory cards however
        - They require a special reader
        - The user is capable of losing the token, locking them out and potentially providing an unauthorized user access
        - Users might be dissatisfied
- **Smart Tokens**
    - Smart Tokens include an embedded microprocessor, and can look like just about anything
    - Manual interfaces include a keypad and display for human/token interaction
    - There is also an electronic interface allowing for proper communication with compatible readers and writers in both a contact and contactless capacity
    - Smart tokens include an authentication protocol that can be classifies into one of three categories
        - Static
        - Dynamic Password Generator
        - Challenge-Response
- **Smart Cards**
    - These are the most important category of smart token
    - They have the appearance of a credit card, an electronic interface, and may use any of the possible smart token protocols
    - They contain
        - Processor
        - Memory
        - I/O Ports
    - They typically include three types of memory
        - ROM (Read-Only Memory)

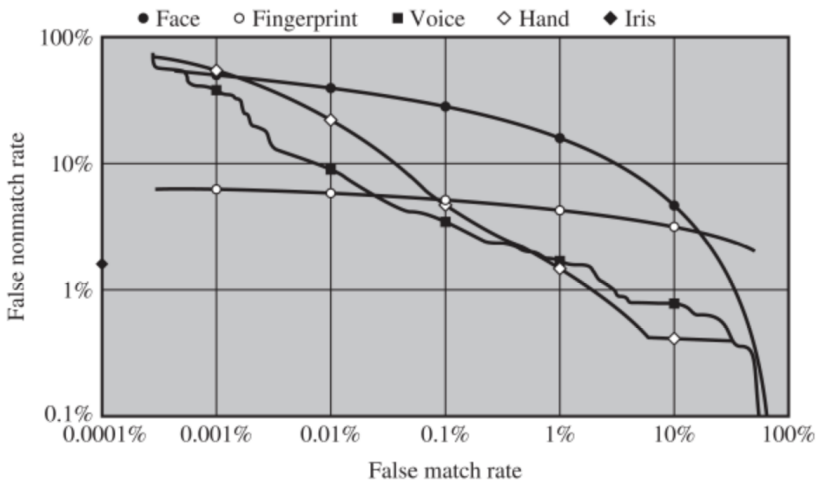
        - EEPROM (Electrically Erasable Programmable Read-Only Memory)
        - RAM (Random Access Memory)
- **Electronic Identity Cards**
    - These function as identifying documents for citizens of a particular country
    - The most advanced deployment of such a system comes in the form of Germany's *neuer Personalausweis* which includes some of the following human-readable data printed on its surface
        - **Personal Data**, such as name, DOB, address, etc
        - **Document Number** or identifier for the card itself
        - **Card Access Number**
        - **Machine Readable Zone**

3.4: Biometric Authentication

- Biometric authentication aims to authenticate users based on unique physical characteristics that are difficult to impersonate
- A number of different physical characteristics can be used in user authentication, the most common of which are as follows
    - **Facial Characteristics**
    - **Fingerprints**
    - **Hand Geometry**
    - **Retinal Pattern**
    - **Iris**
    - **Signature**
    - **Voice**
- The below graph illustrates the relationship between cost and accuracy of the above biometric factors used in authentication



- Biometrics can be used alone for identification, or in tandem with another factor of authentication for verification
- The following shows how often the different biometrics will deliver false matches and false non-matches



3.5: Remote User Authentication

- **Password Protocol**
  - Real password protocols, like Kerberos, are slightly more complicated and will be discussed at length in future chapters
  - A random number r, called a **nonce**, or number used once, is used in conjunction with a hash function in a challenge-response method

- The **Token Protocol** functions similarly to the password protocol but using a token rather than password
- **Biometric Protocols** also follow the same basic idea

3.6: Security Issues for User Authentication

Attack Type	Authenticator	Example(s)	Typical Defenses
Client	Password	Guessing, Exhaustive Search	Large password entropy, limited password attempts
Client	Token	Exhaustive Search	Large token entropy, limited token attempts, theft of object requires presence
Client	Biometric	False Match	Large biometric entropy, limited authentication attempts
Host	Password	Plaintext theft, Exhaustive Search	Hashing and protection of password database
Host	Token	Passcode Theft	Same as password, 1-time passcode
Host	Biometric	Template theft	Capture device authentication, challenge-response
Eavesdropping, Theft, Copying	Password	Shoulder Surfing	Security diligence, multi-factor authentication
Eavesdropping, Theft, Copying	Token	Theft, Counterfeiting	Multi-Factor Authentication, tamper resistant token
Eavesdropping, Theft, Copying	Biometric	Spoofing Biometrics	Capture device authentication
Replay	Password	Replay stolen password response	Challenge Response Protocol
Replay	Token	Replay stolen passcode response	1-Time Passcode
Replay	Biometric	Replay stolen biometric template response	Capture device Authentication

Attack Type	Authenticator	Example(s)	Typical Defenses
Trojan Horse	All	Installation of rogue client or capture device	Capture device within a trusted security perimeter
Denial of Service	All	Lockout by multiple failed authentications	Multi-Factor with Token