Initial Post

Identification and authentication securities are critical safeguards to protect sensitive data and systems from unauthorized access. According to Zviran and Erlich (2006), authentication and authorization are vital components in the broader access control framework. While identification determines "who" the user claims to be, authentication verifies that claim by requiring proof, often through knowledge-based, possession-based, or biometric-based methods. Effective authentication practices are essential for minimizing security risks, as weak or compromised methods can lead to unauthorized access, data breaches, and further exploitation of system vulnerabilities. In today's digital world, this could lead to unauthorized access to susceptible data, including financial, biometric, and personal information, resulting in various data compliance violations (Zviran & Erlich, 2006).

Authentication, when relying solely on passwords, faces several challenges. Users often select weak passwords or reuse them across multiple accounts, increasing susceptibility to attacks like credential stuffing and brute force. Integrating multi-factor authentication (MFA), which combines different authentication types, can significantly enhance security, making unauthorized access more challenging (Zviran & Erlich, 2006). Overall, robust authentication methods protect individual accounts and uphold broader information systems' security, especially in high-stakes environments such as online banking or e-commerce platforms (Zviran & Erlich, 2006).

Identification and Authentication Failures are identified as one of the ten primary vulnerabilities in an application by OWASP Top Ten, 2021. These vulnerabilities include weak passwords, insecure session management, and inadequate protections

against brute-force attacks, which can result in unauthorized access to sensitive systems and data.

Below is the list of authentication weaknesses as defined by OWASP, 2021:

1. Weak Username and Password Policies

- **Use of Default, Weak, or Well-Known Passwords**: The application allows passwords like "Password1" or "admin/admin," which are easily guessed.
- **No Enforcement of Strong Passwords**: The application does not enforce complexity requirements, length, or use of unique characters.
- **Reuse of Old Passwords Allowed**: Users can reuse previous passwords, increasing vulnerability to attacks.

2. Insufficient Protection Against Automated Attacks

- **Credential Stuffing Vulnerability**: The application permits automated attempts using pre-collected username/password pairs without rate limiting.
- **Brute Force Attack Vulnerability**: The application allows multiple login attempts without locking accounts or implementing CAPTCHA to prevent automated entry attempts.

3. Insecure Credential Recovery Mechanisms

- **Weak Password Recovery**: The application uses knowledge-based questions (e.g., "What is your mother's name?"), which are often accessible to guess or find.

- **No Multi-Factor Authentication for Recovery**: Recovery or password reset processes lack multi-factor authentication, making it easier for attackers to exploit them.

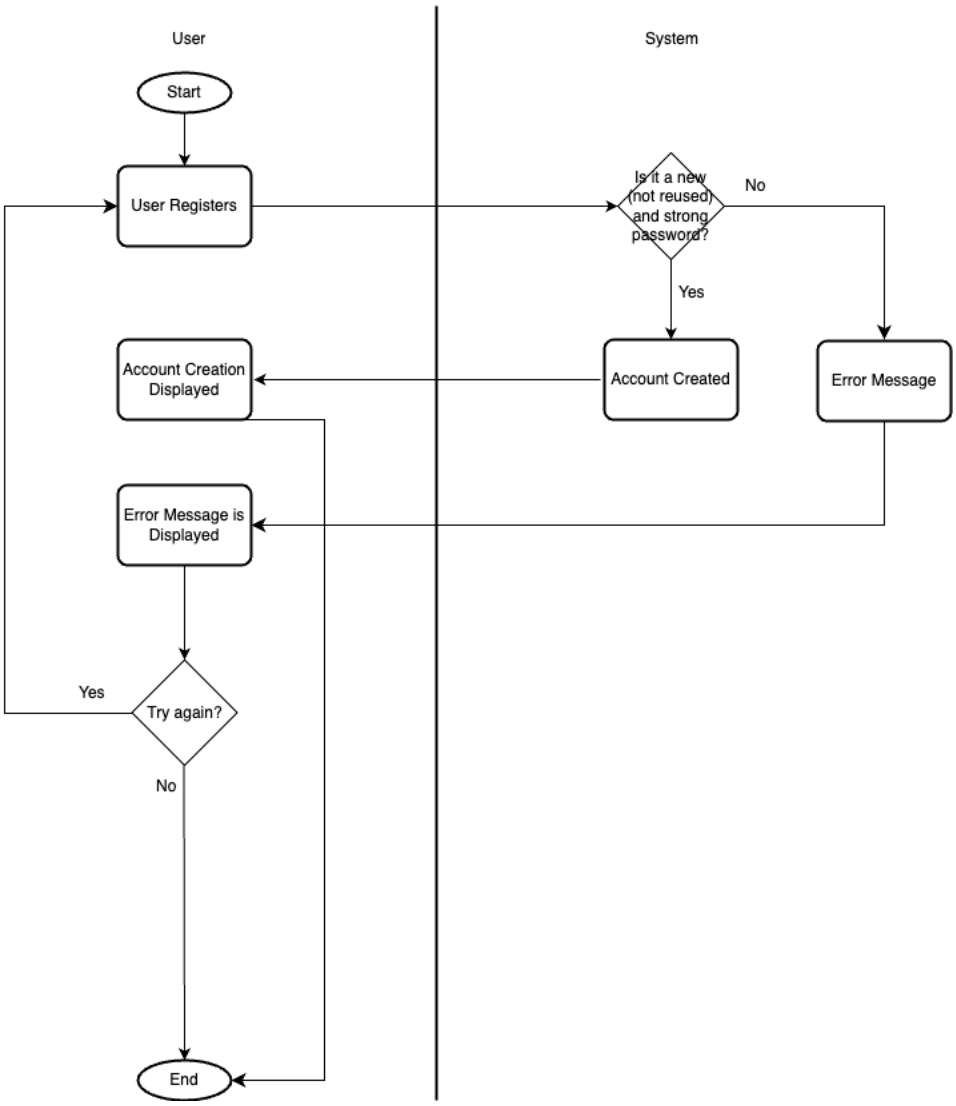
4. Inadequate Session Management and Security

- **Exposing Session Identifiers in URLs**: The application includes session IDs in URLs, making it easy for attackers to intercept them (e.g., through browser history or referrer headers).
- **Reused Session Identifiers Post-Login**: The application reuses session IDs upon re-authentication, making sessions vulnerable to hijacking.
- **Failure to Invalidate Sessions Properly**: The application fails to invalidate session tokens during logout or after a period of inactivity, allowing attackers to use the session token even after the user logs out.

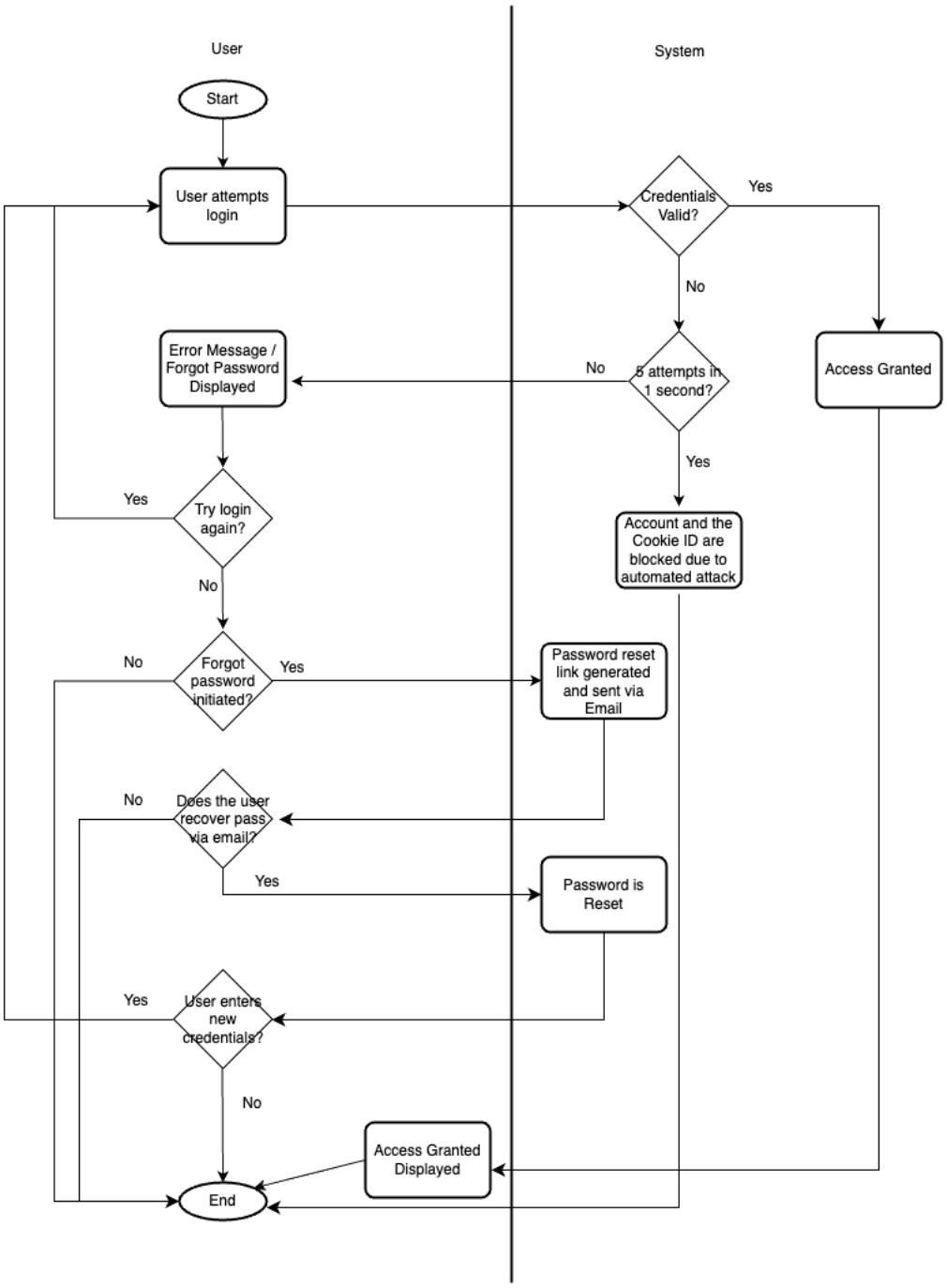
5. Lack of Multi-Factor Authentication (MFA)

- **No MFA for Sensitive Actions**: The application does not require multi-factor authentication for critical actions, such as logging in or changing credentials.
- **Weak or Ineffective MFA Implementation**: The application's MFA implementation is bypassable or ineffective (e.g., easily guessed codes or bypassable security questions).

The first flowchart demonstrates the registration process, which enforces the use of strong passwords and blocks default, weak, well-known, or reused passwords:



The second flowchart shows the log-in authentication process, which protects against automated attacks and has a secure credential recovery mechanism (e.g. email-based password recovery process instead of a knowledge-based one):


User
System
Start
User attempts login
Credentials Valid?
Yes
No
Access Granted
Error Message / Forgot Password Displayed
No
5 attempts in 1 second?
Yes
Account and the Cookie ID are blocked due to automated attack
Yes
Try login again?
No
No
Forgot password initiated?
Yes
Password reset link generated and sent via Email
No
Does the user recover pass via email?
Yes
Password is Reset
Yes
User enters new credentials?
No
Access Granted Displayed
End

References:

OWASP Top 10. (2021) A07:2021 – Identification and Authentication Failures. Available from: **https://owasp.org/Top10/A07_2021-Identification_and_Authentication_Failures/** [Accessed 26 October 2024]

OWASP (2021) OWASP Top 10. *OWASP*. Available from: **https://owasp.org/www-project-top-ten/** [Accessed 25 October 2024]

Zviran, M., & Erlich, Z. (2006) Identification and Authentication: Technology and Implementation Issues. *Communications of the Association for Information Systems* 17. DOI: **https://doi.org/10.17705/1CAIS.01704**