Broken Authentication

Broken authentication is a flaw in a system's authentication process. It is the failure of a system to properly verify a user's identity before granting access to resources. This can happen for a variety of reasons, including weak passwords, a lack of proper password policies, and insufficient security measures for verifying identity.

Broken authentication is a common problem that can have serious consequences, such as unauthorized access to sensitive information, data breaches, and financial loss. To protect against this type of vulnerability, strong authentication practices must be implemented.

Some best practices for mitigating broken authentication vulnerabilities include:

1) Implementing strong password policies: This includes requiring strong passwords, requiring regular password changes, and prohibiting the reuse of old passwords.

2) Enabling two-factor authentication: This adds an extra layer of security by requiring a second form of authentication, such as a code sent to a user's phone, in addition to a username and password.

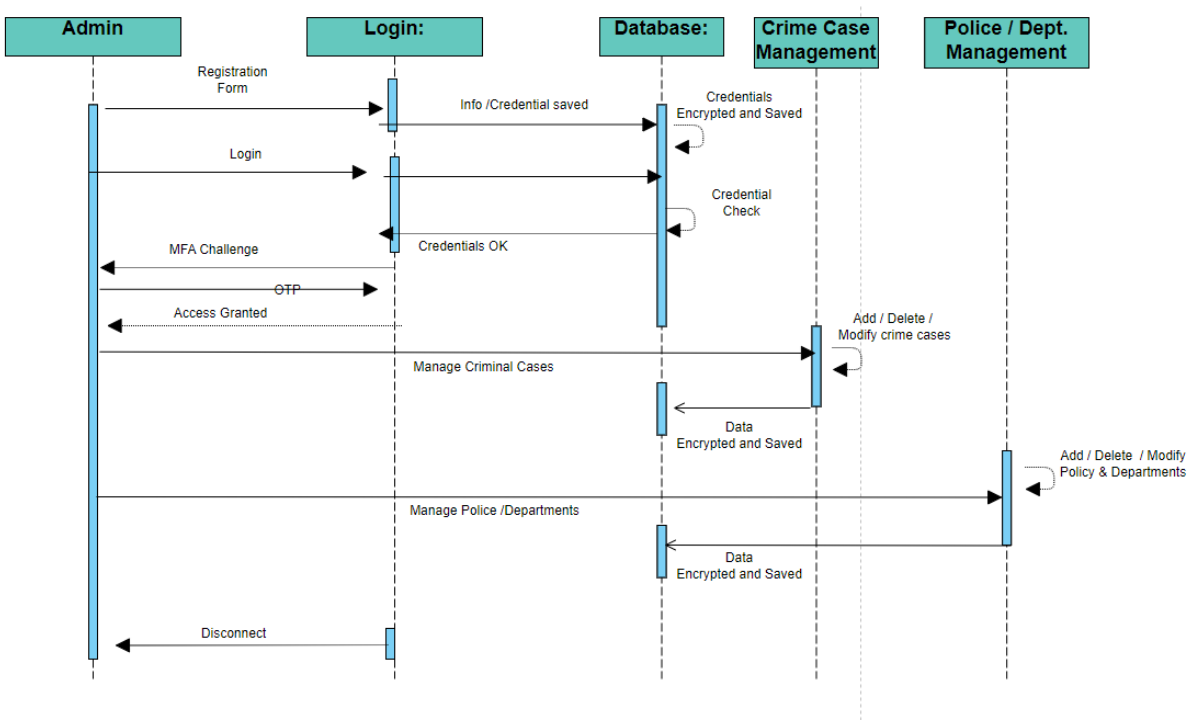
3) Implementing proper session management: This includes using secure session tokens, setting appropriate session timeouts, and invalidating inactive sessions.

4) Monitoring and auditing authentication logs on a regular basis can help identify and address potential vulnerabilities and malicious activity.

5) Providing secure methods for recovering lost or forgotten passwords: This can help prevent unauthorized account access.

6) By implementing these best practices, organizations can significantly reduce the risk of broken authentication vulnerabilities and protect against unauthorized access to sensitive information.

We can use class and sequence diagrams to present the proposed software's design because they provide a clear visual representation of the system and how it works. Class diagrams can depict the relationships between various objects, as well as their attributes and methods, whereas sequence diagrams can depict the flow of control and the interactions between objects over time. These diagrams, when combined, can provide a comprehensive view of the system and aid in the identification of potential vulnerabilities and design flaws.



References:

Unit 3 Exercise Submission