Course: CY341 - Secure Software Development Lifecycle

Team Members: Asad Ali (2022903), Ammaid Saleem (2022344), M.Mursaleen (2022401), Hamayun Bajwa (2022189)

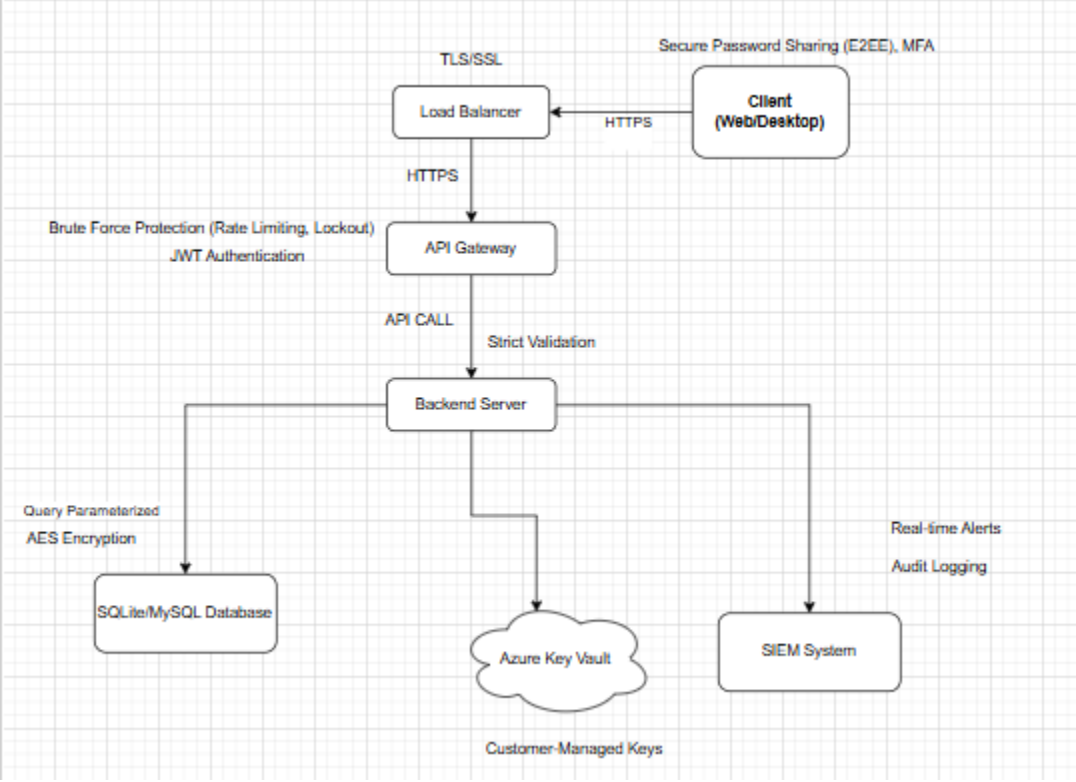
Faculty: CYS

Submitted to: Dr. Zubair Ahmad

Introduction

The Secure Password Manager is a web and desktop application designed to store and share passwords securely. The architecture mitigates critical threats (e.g., SQL injection, token theft, session hijacking) using robust controls like JWT authentication, AES encryption, and Multi-Factor Authentication (MFA).

System Architecture:



Components

- **Client (Web/Desktop)**: The frontend interface for users to manage passwords, supporting MFA and End-to-End Encryption (E2EE) for secure password sharing.
- **Load Balancer**: Distributes HTTPS traffic with TLS/SSL for secure communication.
- **API Gateway**: Authenticates requests using JWT and enforces brute force protection (rate limiting, account lockout).
- **Backend Server**: Processes application logic, implements Role-Based Access Control (RBAC), and uses strict input validation.
- **SQLite/MySQL Database**: Stores passwords encrypted with AES.

- **Azure Key Vault**: Manages customer-managed encryption keys.
- **SIEM System**: Provides real-time alerts and audit logging for monitoring.

Data Flow

- Users send HTTPS requests from the Client through the Load Balancer to the API Gateway.
- The API Gateway verifies JWTs and forwards API calls to the Backend Server.
- The Backend Server uses parameterized queries to access the Database, retrieves keys from Azure Key Vault, and sends logs/alerts to the SIEM System.

Security Controls

 JWT Authentication (API Gateway):

- Uses short-lived tokens (15 minutes) and revocation lists to prevent token theft
- **Multi-Factor Authentication (MFA)** (Client):
  - Requires a second factor (e.g., OTP) to counter session hijacking and MFA bypass.
- **AES Encryption** (Database):
  - Encrypts passwords with AES-256 to prevent data leaks from backups or misconfigured vaults.
- **End-to-End Encryption (E2EE)** (Client):
  - Secures password sharing, ensuring only intended recipients can decrypt.
- **Role-Based Access Control (RBAC)** (Backend Server):
  - Restricts actions based on user roles (e.g., admin vs. user), mitigating privilege escalation.
- **Parameterized Queries** (Backend Server):
  - Sanitizes database inputs to prevent SQL injection.
  - Example (SQL):

```
SELECT * FROM users WHERE username = ? AND password = ?;
```

- **Brute Force Protection** (API Gateway):
  - Implements rate limiting and account lockout to mitigate DoS attacks.
- **TLS/SSL** (Load Balancer):
  - Ensures secure communication between Client, Load Balancer, and API Gateway.
- **Customer-Managed Keys** (Azure Key Vault):
  - Secures encryption keys, preventing misconfiguration risks.
- **SIEM Logging and Alerts** (SIEM System):
  - Provides audit trails and real-time alerts to counter log forgery

Security Design Measures

The architecture follows secure design principles to mitigate Week 2 threats:

1. **Least Privilege**:
    - RBAC ensures users only access authorized resources, reducing elevation of privilege risks.
2. **Defense-in-Depth**:
    - Multiple layers (MFA, JWT, AES, TLS/SSL) protect against spoofing, tampering, and data leaks.
3. **Input Validation**:
    - Strict validation and parameterized queries prevent tampering (e.g., SQL injection, JWT manipulation).
4. **Secure Defaults**:
    - Short-lived JWTs, encrypted data, and customer-managed keys minimize vulnerabilities.
5. **Auditability**:
    - SIEM logging ensures all actions are traceable, addressing repudiation risks.

Threat Mitigation Summary

- **Spoofing**: MFA and JWT authentication secure user identity.
- **Tampering**: Parameterized queries and strict validation block SQL injection and JWT manipulation.
- **Information Disclosure**: AES encryption, E2EE, and Azure Key Vault protect data.
- **Denial of Service**: Rate limiting and load balancing mitigate brute force attacks.
- **Elevation of Privilege**: RBAC restricts unauthorized access.
- **Repudiation**: SIEM provides verifiable audit logs.

Conclusion

The Secure Password Manager's architecture integrates robust security controls to address Week 2 threats, ensuring a secure and compliant system. The design aligns with OWASP Top 10, NIST SP 800-53, and GDPR/ISO 27001 standards.