SAMM Assessment spreadsheet

Software Development Plan for SecureSync

1. Governance

- **Strategy & Metrics**:
  - **Create & Promote**: Establish a security roadmap for SecureSync, outlining the security objectives and goals. Define KPIs for measuring security effectiveness.
  - **Measure & Improve**: Continuously evaluate security metrics and implement strategies for improvement.
- **Policy & Compliance**:
  - **Policy & Standards**: Create security policies adhering to relevant regulations. Define standards for encryption, access control, and data storage.
  - **Compliance Management**: Monitor compliance regularly to ensure SecureSync follows international security standards .
- **Education & Guidance**:
  - **Training & Awareness**: Regular cybersecurity training for all employees and developers involved with SecureSync.
  - **Organization & Culture**: Foster a security-first culture within the development team by embedding security into development workflows.

2. Design

- **Threat Assessment**:
  - **Application Risk Profile**: Identify potential threats for SecureSync by conducting threat modeling exercises to assess application vulnerabilities.
  - **Threat Modeling**: Use data flow diagrams to map out potential attack vectors and define mitigations.
- **Security Requirements**:
  - **Software Requirements**: Define security requirements for data encryption, access controls, and secure API development.
  - **Supplier Security**: Ensure third-party services and dependencies integrated into SecureSync follow robust security standards.
- **Secure Architecture**:
  - **Architecture Design**: Define a secure architecture for SecureSync, using a layered defense model to mitigate risks across different components.
  - **Technology Management**: Select technologies that support encryption, access control, and scalability.

3. Implementation

- **Secure Build**:
  - **Build Process**: Integrate security testing, automating static analysis, and vulnerability scanning.
  - **Software Dependencies**: Ensure that third-party libraries used by SecureSync are vetted and updated regularly to avoid vulnerabilities.
- **Secure Deployment**:
  - **Deployment Process**: Use secure deployment processes including role-based access control, environment hardening, and containerization for scalability and security.

    Secret Management: Implement secrets management to protect sensitive keys and credentials.

- **Defect Management**:
    - **Defect Tracking**: Establish defect tracking for all security vulnerabilities found during development.
    - **Metrics & Feedback**: Analyze metrics from defects to identify security trends and weak spots in the software development lifecycle.

4. Verification

- **Architecture Assessment**:

    Architecture Validation: Periodic reviews of the architecture to ensure it aligns with security policies and standards.

    Architecture Compliance: Ensure the architecture complies with regulations.

- **Requirements-driven Testing**:

    Control Verification: Use automated tests to verify security controls, including authentication mechanisms, encryption, and access control lists.

    Misuse/Abuse Testing: Conduct penetration testing and simulate misuse scenarios to identify potential security gaps.

- **Security Testing**:

    Scalable Baseline: Implement scalable security testing as part of the QA process for SecureSync.

    Deep Understanding: Focus on deep vulnerability assessments for critical components such as database encryption and access control mechanisms.

5.Operations

- **Incident Management**:

    Incident Detection: Set up real-time monitoring and alerting systems to detect unusual activity or breaches.

    Incident Response: Develop an incident response plan for SecureSync that includes procedures for containment, eradication, and recovery.

- **Environment Management**:

    Configuration Hardening: Regularly harden configurations for servers, databases, and cloud services to reduce attack surfaces.

    Patch & Update: Ensure timely patching of all systems, libraries, and third-party components.

- **Operational Management**:

    Data Protection: Regularly audit data protection mechanisms, ensuring encryption is applied to all sensitive data, both at rest and in transit.

    Legacy Management: Develop a plan for managing legacy systems, including deprecating insecure technologies or protocols.