Group №5

LuminaX

Software development plan

Use Case №2

1. **Introduction and Scope**

This Software Development Plan describes the approach for building a secure and reliable software solution for LuminaX’s LumiMind product. LumiMind is designed to help users with content creation, analytics, and automation, making their work easier and more efficient. Because LuminaX has a very low tolerance for cybersecurity risks, this plan focuses heavily on security throughout the software development process. The aim is to protect against issues like unauthorized access, data breaches, and service disruptions. By keeping security at the forefront of development, LuminaX intends to deliver a product that meets high standards for both reliability and compliance with data protection regulations.

2. **Objective**:

To deliver a secure, AI-powered product for content generation, analytics, and automation, providing a seamless and secure experience.

3. **Roles and Responsibilities**

**IT Security Team**: Ensures secure authentication, access controls, and ongoing threat monitoring.

**HR Manager**: Conducts cybersecurity training to increase employee awareness and prevent phishing and data mishandling.

**Database Administrator**: Implements encryption, access controls, and database monitoring.

**Software Development Team**: Integrates security controls (RBAC, MFA) into the backend and frontend and performs code reviews.

**Cloud Security Team**: Manages secure cloud configurations and reviews for potential misconfigurations.

**Network Security Team**: Manages network infrastructure and its security, redundancy, and failover mechanisms.

**Legal & Compliance Team**: Ensures the software adheres to relevant data protection regulations (e.g., GDPR, CCPA).

4. **Requirements and Specifications**

Functional Requirements:

* + Multi-factor authentication (MFA) and OAuth 2.0 with JWT for secure login.
  + Role-based access control (RBAC) across all user roles and data.
  + Automated content generation, data processing, and analytics capabilities.

Non-Functional Requirements:

* + AES-256 encryption for data at rest and HTTPS/TLS for data in transit.
  + Compliance with data protection regulations (GDPR, CCPA).

Technical Requirements:

* + Microservices architecture with Docker and Kubernetes for scalability.
  + Secure APIs with input validation, rate limiting, and Web Application Firewall (WAF).
* Real-time monitoring and logging with Prometheus, Grafana, and ELK Stack.

5. **Development Methodology**

* Agile Methodology: Adopted to allow iterative development, enabling the team to integrate security improvements based on testing feedback continuously.
* Sprints: Security reviews at the end of each sprint to ensure the integration of security practices in each component before moving forward.

6. **Quality Assurance (QA) and Testing**

Security Testing:

* + Regular penetration testing on APIs, data flows, and storage to identify potential vulnerabilities.
  + Use automated tools for continuous scanning of security and compliance.

API Security Testing:

* + Validate input and set up rate limiting, ensuring all external interfaces are secure.

Performance Testing:

* + Simulate high request volumes to ensure resilience against DoS attacks, especially on the API Gateway and AI Engine.

Code Reviews and Audits:

* + Regular code reviews focusing on secure coding practices.
  + Audit logs for unusual activity or security breaches.

7. **Resource and Staffing Plan**

**Security Tools**: Tools for encryption (AES-256), token management (OAuth 2.0, JWT), and secure API testing.

**Database and Logging Tools**: RDBMS with encrypted storage, audit log capabilities, and the ELK Stack for log analysis.

**Training Resources**: Materials for regular cybersecurity training, phishing simulations, and secure data handling.

8. **Maintenance and Support Plan**

**Continuous Monitoring**: Use Prometheus, Grafana, and ELK Stack to monitor the system for potential threats.

**Scheduled Audits and Reviews**: Conduct quarterly security audits and compliance reviews.

**Employee Training**: Regular cybersecurity training, focusing on updates in phishing methods, secure data handling, and password policies.

**Compliance Tracking**: Ensure continuous alignment with GDPR, CCPA, and any emerging data protection regulations.

9. **Tools for securing software development**

* **SonarQube** for code quality and security analysis.
* **OWASP ZAP** and **Burp Suite** for web application security testing.
* **Snyk** and **Checkmarx** for dependency and code vulnerability scanning.
* **HashiCorp Vault** for secrets management.
* **Aqua Security** and **Docker Bench for Security** for container security.
* **Prometheus** and **Grafana** for monitoring and logging.
* **ELK Stack** for log management and analysis.