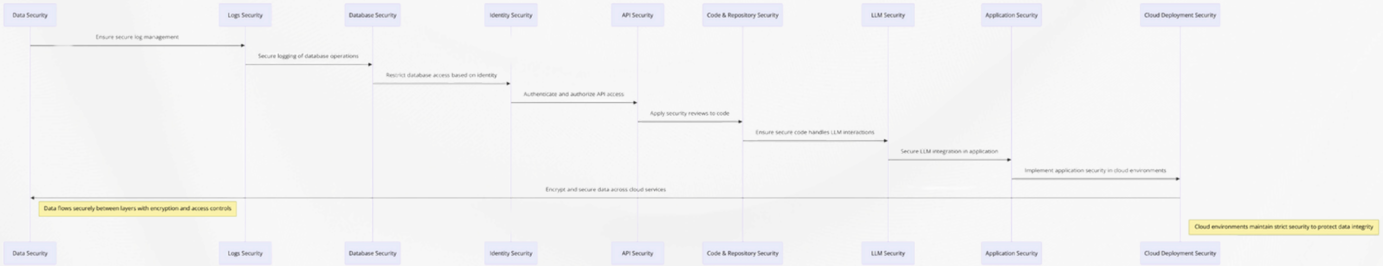
DATA SECURITY DOCUMENTATION

21st May 2024


Data Security
Logs Security
Database Security
Identity Security
API Security
Code & Repository Security
LLM Security
Application Security
Cloud Deployment Security
Ensure secure log management
Secure logging of database operations
Restrict database access based on identity
Authenticate and authorize API access
Apply security reviews to code
Ensure secure code handles LLM interactions
Secure LLM integration in application
Implement application security in cloud environments
Encrypt and secure data across cloud services
Data flows securely between layers with encryption and access controls
Cloud environments maintain strict security to protect data integrity

1. Data Security

Data security is paramount in handling financial data, particularly when interacting with generative AI models. Ensuring that sensitive data is not exposed to unintended parties, including the LLM, is crucial.

Data Ingestion: Data is ingested through secure channels, ensuring end-to-end encryption using protocols like TLS.
Data Anonymization: Before any data is sent to the LLM, it is anonymized to strip out personally identifiable information (PII).
Data Storage: Data at rest is encrypted using strong encryption standards like AES-256.
Data Transmission: Data is transmitted over secure channels using encryption protocols like TLS or HTTPS.
Secure API Gateway: All interactions with the LLM are routed through a secure API gateway, ensuring that data is encrypted and authenticated.
Data Minimization: Only essential data is passed to the LLM, minimizing the risk of data exposure.

2. Logs Security

Logs are essential for auditing and troubleshooting but can also contain sensitive information.

Encryption: Logs are encrypted both in transit and at rest.
Access Controls: Strict access controls ensure that only authorized personnel can access logs.
Log Redaction: Sensitive information in logs is redacted to prevent data leakage.
Log Rotation: Regular log rotation and archival policies ensure that old logs are properly managed and deleted.

3. Database Security

Databases storing financial data must be secured to prevent unauthorized access and data breaches.

Encryption: Data is encrypted at rest and in transit.
Access Controls: Role-based access controls (RBAC) are implemented to restrict access to sensitive data.
Database Auditing: Regular audits are conducted to monitor access and changes to the database.
Intrusion Detection: Systems are in place to detect and respond to unauthorized access attempts.

DATA SECURITY DOCUMENTATION

21st May 2024

4. Identity Security (User Login)

User identity and access management ensure that only authorized users can access the system.

Single Sign-On (SSO): Integration with client SSOs allows for centralized authentication and authorization.
Multi-Factor Authentication (MFA): MFA adds an additional layer of security by requiring multiple forms of verification.
Access Controls: Fine-grained access controls ensure users can only access data and functions necessary for their roles.
Identity Federation: Federated identity management allows for secure cross-domain authentication.

5. API Security

APIs are the core of interaction between the application components and external systems.

Authentication and Authorization: APIs are secured using OAuth 2.0 and JWT tokens.
Rate Limiting: To prevent abuse, rate limiting is implemented.
Input Validation: All inputs are validated to prevent injection attacks.
Encryption: All API communications are encrypted using HTTPS.

6. Code & Repository Security Standards

Secure coding practices and repository management are crucial to maintaining application security.

Environment Variables: Sensitive information is stored in environment variables and not hardcoded into the codebase.
Code Reviews: Regular code reviews and security audits are conducted.
Dependency Management: Dependencies are regularly updated to patch vulnerabilities.
Repository Access: Access to code repositories is restricted to authorized personnel and is audited.

7. LLM Security

Ensuring that data shared with the LLM is secure and not exposed.

Data Encryption: Data passed to the LLM is encrypted using strong encryption standards.
Secure Containers: LLMs run in secure containers isolated from other processes.
Data Masking: Sensitive data is masked before being processed by the LLM.
Access Controls: Strict access controls ensure only authorized services can interact with the LLM.

21st May 2024

8. Application Security

Application security involves protecting the application from various threats and vulnerabilities.

Secure Development Lifecycle: Security is integrated into every stage of the development lifecycle.
Penetration Testing: Regular penetration tests are conducted to identify and fix vulnerabilities.
WAF (Web Application Firewall): A WAF protects the application from common web exploits.
Security Patches: Regular updates and patches are applied to fix security vulnerabilities.

9. Cloud Deployment Security

Deploying applications in the cloud requires a robust security strategy to protect data and services.

Encryption: Data in transit and at rest in the cloud is encrypted.
Network Security: Virtual Private Clouds (VPCs), security groups, and firewalls are used to secure network traffic.
IAM: AWS Identity and Access Management (IAM) is used to manage access to cloud resources.
Monitoring and Logging: Continuous monitoring and logging are implemented to detect and respond to security incidents.