SpendSmart Model

Owner: oyin
Reviewer:
Contributors:
Date Generated: Sun Aug 18 2024

Executive Summary

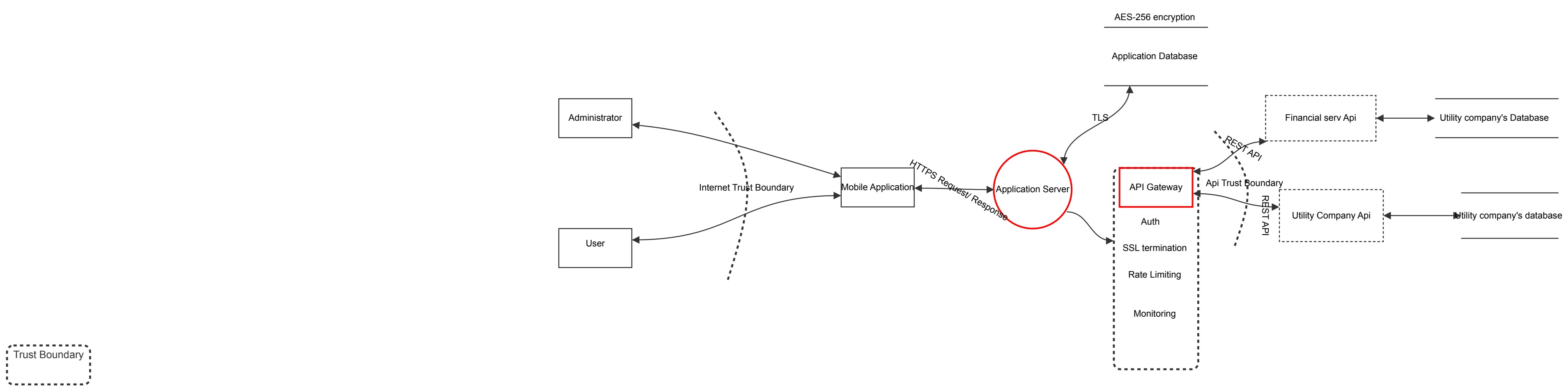
High level system description

data flow information

Summary

Total Threats	21
Total Mitigated	17
Not Mitigated	4
Open / High Priority	1
Open / Medium Priority	3
Open / Low Priority	0
Open / Unknown Priority	0

New STRIDE diagram



New STRIDE diagram

User
(Actor)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Data Flow (Data Flow)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Utility Company Api (Actor) *- Out of Scope*

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Financial serv Api (Actor) *- Out of Scope*

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Utility company's Database (Store)

Number	Title	Type	Priority	Status	Score	Description	Mitigations
38	application database is susceptible to	Tampering	High	Mitigated	8	Unauthorized changes to the data stored in the database.	Implement input validation and sanitation. Use parameterized queries and prepared statements. Regularly audit and log database changes.
39	Application Database is vulnerable to Repudiation attack	Repudiation	High	Mitigated	8	Users denying their actions within the database, such as deleting or modifying records.	Implement detailed logging of all database transactions. Use digital signatures to ensure non-repudiation. Regularly review and audit logs to verify actions and track anomalies.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
40	New STRIDE threat	Information disclosure	High	Mitigated	9	Unauthorized access leading to the exposure of sensitive data. Data intercepted during transmission due to lack of encryption.	Encrypt sensitive data at rest using strong encryption algorithms (e.g., AES-256). Use TLS/SSL for encrypting data in transit. Implement access controls to restrict data access based on user roles.
41	Application Database is vulnerable to DOS attack	Denial of service	High	Mitigated	8	Attacks that exhaust database resources, making the service unavailable.	mplement rate limiting and query throttling. Use database monitoring tools to detect and respond to unusual spikes in activity. Employ database replication and load balancing to distribute the load.

Administrator (Actor)

Number	Title	Type	Priority	Status	Score	Description	Mitigations
44	Administrator is vulnerable to spooofing attack	Spoofing	High	Mitigated	9	An insider or external attacker obtains or guesses admin credentials to access privileged systems or data to manipulation of sensitive information, or disruption of services.	Implement strong authentication mechanisms (e.g., multi-factor authentication), regularly update credentials, and monitor admin access logs for unusual activity.
45	An Administrator is vulnerable to repudiation	Repudiation	Medium	Mitigated	6	An admin makes unauthorized changes but denies responsibility when confronted.	Implement comprehensive logging and auditing mechanisms. Use digital signatures or other cryptographic methods to ensure non-repudiation of actions.

Mobile Application (Actor)

Number	Title	Type	Priority	Status	Score	Description	Mitigations
27	#1. Mobile Application is vulnerable to spoofing	Spoofing	Medium	Mitigated	6	Phishing attacks via SMS or email to trick users into revealing login credentials.	Implement strong authentication and session management: Use multi-factor authentication (MFA) and securely manage session tokens to prevent identity spoofing.
28	#2.	Repudiation	High	Mitigated	8	Insufficient verification of user actions, making it difficult to prove the authenticity of transactions.	Maintain comprehensive logs of user activities and implement mechanisms to detect and prevent repudiation.
29	3. application is vulnerable to Tampering	Spoofing	High	Mitigated	8	Modifying data sent between the mobile app and backend servers to alter transaction details.	Implement encryption and message authentication to protect data from tampering during transmission.
30	4. Application is vulnerable to Denial Of Service	Spoofing	High	Mitigated	8	IFlooding the mobile app's server with excessive requests, causing it to crash or become unresponsive.	Use rate limiting, CAPTCHA, and other techniques to mitigate DoS attacks.

API Gateway (Actor)

Number	Title	Type	Priority	Status	Score	Description	Mitigations
42	API Gateway	Spoofing	High	Mitigated	8	Unauthorized entities masquerade as legitimate users or systems to gain access to the API Gateway.	Implement strong authentication mechanisms such as OAuth, API keys, or client certificates to verify the identity of clients accessing the API Gateway. Use secure protocols like HTTPS to prevent interception and spoofing of credentials.

Number	Title	Type	Priority	Status	Score	Description	Mitigations
43	API Gateway is vulnerable to Repudiation Attack	Repudiation	Medium	Mitigated	6	Users or clients deny performing actions or accessing resources through the API Gateway.	Implement logging and auditing mechanisms that record API requests and responses, including client identities and actions performed. Use digital signatures or timestamps to ensure non-repudiation of transactions.
71	The Gateway is vulnerable to Tampering	Spoofing	High	Open	8	Attackers intercept and potentially alter communication between clients and the API gateway.	Use Transport Layer Security (TLS) to encrypt data in transit, ensuring the integrity and confidentiality of communication.
72	The API is Vulneable to DDOS Attac	Spoofing	Low	Mitigated	4	Attackers flood the API gateway with a high volume of requests, aiming to overwhelm the system and cause service outages.	Enforce strong authentication practices, such as multi-factor authentication (MFA), and use OAuth2.0 or other secure token-based authentication methods.

Utility company's database (Store)

Number	Title	Type	Priority	Status	Score	Description	Mitigations

Application Server (Process)

Number	Title	Type	Priority	Status	Score	Description	Mitigations
57	Application is vulnerable to Tampering	Tampering	Medium	Open		Malicious actors modify data in transit or on the server to manipulate application behavior or compromise data integrity.	Use encryption (both in transit and at rest) to protect sensitive data from unauthorized modification. Employ data validation and integrity checks at multiple layers (input validation, server-side validation) to detect and prevent tampering attempts.
58	Application Server is vulnerable to Information Disclosure	Information disclosure	Medium	Open		Unauthorized parties gain access to sensitive information stored on or transmitted by the application server.	Implement encryption for data both at rest (in databases) and in transit (over networks) to protect against eavesdropping and unauthorized access.
60	Application Server is vulnerable to DOS	Denial of service	Medium	Open		Provide a description for this threat	Provide remediation for this threat or a reason if status is N/A
64	Application Server is vulnrable to Spoofing	Spoofing	Medium	Mitigated	6	Attackers attempt to impersonate legitimate users or systems to gain unauthorized access to the application server.	Implement strong authentication mechanisms, such as multi-factor authentication (MFA), to verify user identities. Use secure protocols (e.g., HTTPS, TLS) for communication to prevent interception and tampering.

Application Database (Store)

Number	Title	Type	Priority	Status	Score	Description	Mitigations
49	Application Database is vulnerable to Tampering	Tampering	High	Mitigated	8	Malicious actors modify data in transit or on the server to manipulate application behavior or compromise data integrity.	Data Validation: Implement input validation to ensure data integrity and prevent injection attacks (e.g., SQL injection). Access Controls: Use database permissions and auditing to track and restrict data modifications based on user roles. Backup and Recovery: Regularly backup database contents and implement secure backup storage to recover data in case of tampering. Repudiation

Number	Title	Type	Priority	Status	Score	Description	Mitigations
50	Application Database is vulnerable to Information Disclosure	Information disclosure	High	Mitigated	9	Unauthorized parties gain access to sensitive information stored on or transmitted by the application server.	Implement encryption for data both at rest (in databases) and in transit (over networks) to protect against eavesdropping and unauthorized access. Apply access controls and role-based permissions to limit who can access sensitive data.
54	Application Server is vulnerable to DOS	Denial of service	High	Mitigated	8	Implement rate limiting and throttling mechanisms to restrict the number of requests a user or IP address can make within a specific timeframe. Use load balancers and scalable architecture to distribute traffic and absorb DoS attacks. Monitor server performance metrics and implement anomaly detection to identify and mitigate DoS attacks promptly.	Provide remediation for this threat or a reason if status is N/A