SSD_project_lost_and_found

Current Risk Summary report

Sun Apr 20 2025 20:22:50 GMT+0000 (Coordinated Universal Time)

Project description: *No description*

Filtered by: Critical risk | High risk | Medium risk | Low risk | Very Low risk | Unmitigated threats | No mitigation planned

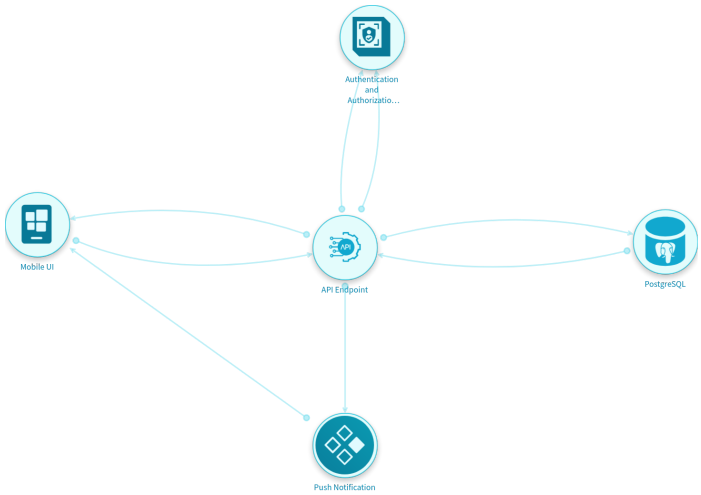
This report only includes threats that match the filters above.

Unique ID: ssd_project_lost_and_found-1744131289774

Owner: Hassaan Ali Bukhari, Muhammad Abdullah

Workflow state: Draft

Tags: *No tags*



Content menu

Current Risk summary

Inherent risk description: The Inherent Risk before countermeasures were applied.
- **Risk Rating:** 63% High

The Current Risk description (the risk we are at now): The Current Risk is based on the current implementation status of the countermeasures and test results.
- **Risk Rating:** 63% High

Projected Risk description: The Projected Risk is the level of risk that would be reached should the required countermeasures be implemented.
- **Risk Rating:** 63% High

Components

- API Endpoint

  Model questionnaire information:
  - **Credit Card Data: How is it handled by this component?** Sent from component
  - **Credit Card Data: How is it handled by this component?** Received by component
  - **Customer Data: How is it handled by this component?** Sent from component
  - **Customer Data: How is it handled by this component?** Received by component
  - **Do you have proper authorization checks implemented?** Yes, it is implemented
  - **Do you have rate limiting and robust load balancing in place for your API endpoint?** Yes, it is implemented
  - **Do you implement logging and monitoring in your API endpoints?** Yes, it is implemented
  - **Do you regularly conduct security audits and review configurations in your API endpoint environment?** Yes, it is implemented
  - **Do you use Role-Based Access Control (RBAC) to manage permissions based on user roles (assigning access levels to users based on their roles)?** Yes, it is implemented
  - **Do you use data encryption for both data at rest (stored data) and data in transit (data being sent or received)?** Yes, it is implemented
  - **Do you use strict security headers (like Content Security Policy, X-Content-Type-Options, X-Frame-Options, and Strict-Transport-Security) for your API endpoints?** Yes, it is implemented
  - **Does this component handle personally identifiable information from citizens of the European Union?** No
  - **Does this component handle personally identifiable information from citizens of the European Union?** Yes
  - **Does this component have to be CCPA-compliant?** No
  - **Personally Identifiable Information: How is it handled by this component?** Sent from component
  - **Personally Identifiable Information: How is it handled by this component?** Received by component
  - **Protected Health Information: How is it handled by this component?** Sent from component
  - **Protected Health Information: How is it handled by this component?** Received by component

- Authentication and Authorization Module

  Model questionnaire information:
  - **Credit Card Data: How is it handled by this component?** Processed
  - **Credit Card Data: How is it handled by this component?** Sent from component
  - **Credit Card Data: How is it handled by this component?** Received by component
  - **Customer Data: How is it handled by this component?** Processed
  - **Customer Data: How is it handled by this component?** Sent from component
  - **Customer Data: How is it handled by this component?** Received by component
  - **Do you have a policy and workflow for comprehensive logging and monitoring (tracking user activities and system events to detect suspicious behavior) in place?** Not sure
  - **Do you have a policy and workflow for comprehensive logging and monitoring (tracking user activities and system events to detect suspicious behavior) in place?** Yes, it is implemented
  - **Do you have a secure access control mechanism (a way to ensure only authorized users can access certain parts of your application) in your workflow?** Yes, it is implemented
  - **Do you have rate limiting (a technique to control the number of requests a user can make) and proper resource management (efficient use of server resources) active in your module?** Yes, it is implemented
  - **Do you have secure configuration (ensuring settings are properly set to protect the application) and encryption standards (rules for converting data into a secure format) enforced in your project?** Yes, it is implemented
  - **Do you thoroughly validate and sanitize user input (checking and cleaning data provided by users to prevent harmful code or data from causing issues)?** Yes, it is implemented
  - **Does this component handle personally identifiable information from citizens of the European Union?** No
  - **Does this component handle personally identifiable information from citizens of the European Union?** Yes
  - **Does this component have to be CCPA-compliant?** No
  - **Personally Identifiable Information: How is it handled by this component?** Processed
  - **Personally Identifiable Information: How is it handled by this component?** Sent from component
  - **Personally Identifiable Information: How is it handled by this component?** Received by component
  - **Protected Health Information: How is it handled by this component?** Processed
  - **Protected Health Information: How is it handled by this component?** Sent from component
  - **Protected Health Information: How is it handled by this component?** Received by component

- Mobile UI

  Model questionnaire information:
  - **Credit Card Data: How is it handled by this component?** Processed
  - **Customer Data: How is it handled by this component?** Processed
  - **Do you encrypt communications in your application?** Yes, it is implemented
  - **Do you encrypt communications in your application?** Not sure
  - **Do you properly utilize the app sandbox feature to contain the app and limit its access?** Not sure
  - **Do you use Two-Factor Authentication (2FA) in your mobile application?** Not sure
  - **Do you use a secure method to store data in your mobile application?** Not sure
  - **Do you validate user input to protect against injection attacks?** Not sure
  - **Does this component handle personally identifiable information from citizens of the European Union?** No
  - **Does this component have to be CCPA-compliant?** No
  - **Personally Identifiable Information: How is it handled by this component?** Processed
  - **Protected Health Information: How is it handled by this component?** Processed

- PostgreSQL

  Model questionnaire information:
  - **Credit Card Data: How is it handled by this component?** Stored

- **Customer Data: How is it handled by this component?** Stored
- **Does this component handle personally identifiable information from citizens of the European Union?** No
- **Does this component have to be CCPA-compliant?** No
- **Is PostgreSQL database file permissions setting enforced and updated regularly in your current system?** Yes, it is implemented
- **Is PostgreSQL database file permissions setting enforced and updated regularly in your current system?** No, and this is not applicable
- **Is PostgreSQL routinely updated to its latest secure version in the current system?** No, but it is required
- **Is rate limiting and resource throttling active and regularly updated in PostgreSQL?** Yes, it is implemented
- **Is the PostgreSQL configuration hardened and network access restricted currently?** Yes, it is implemented
- **Is the TLS encryption enforced for all connections to PostgreSQL?** Yes, it is implemented
- **Is the robust authentication and role-based access control currently in effect for PostgreSQL?** Not sure
- **Is the secure backup procedure with encryption and access controls for PostgreSQL currently implemented?** Yes, it is implemented
- **Is the use of parameterized queries and validation of inputs in place for PostgreSQL to prevent SQL injection attacks?** Yes, it is implemented
- **Personally Identifiable Information: How is it handled by this component?** Stored
- **Protected Health Information: How is it handled by this component?** Stored

- Push Notification

Model questionnaire information:

- **Credit Card Data: How is it handled by this component?** Processed
- **Customer Data: How is it handled by this component?** Processed
- **Do you have rate limiting (a technique to control the number of requests a user can make to a server in a given timeframe) in place?** Yes, it is implemented
- **Do you use content validation and filtering (checks to ensure input data is safe and appropriate) in your application?** Not sure
- **Do you use strong authentication (verifying the identity of users) and authorization checks (controlling user access) for notifications and their triggering mechanisms?** Not sure
- **Does this component handle personally identifiable information from citizens of the European Union?** No
- **Does this component have to be CCPA-compliant?** No
- **Personally Identifiable Information: How is it handled by this component?** Processed
- **Protected Health Information: How is it handled by this component?** Processed

Accepted Risks

No data

Current Risks

Component: API Endpoint

Use case: Information Disclosure

CRT1. Threat name: Attackers abuse of Missing or Insecure Security Headers
- **Inherent risk:** Medium
- **Current risk:** Medium
- **Projected risk:** Medium
- **State:** Expose
- **CR1. Countermeasure name:** Implement Strict Security Headers
  - **Status:** RECOMMENDED

CRT2. Threat name: Attackers exploit Misconfiguration
- **Inherent risk:** Critical
- **Current risk:** Critical
- **Projected risk:** Critical
- **State:** Expose
- **CR2. Countermeasure name:** Regular Security Audits and Configuration Reviews
  - **Status:** RECOMMENDED

CRT3. Threat name: Attackers take advantage of Sensitive Data Exposure
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR3. Countermeasure name:** Data Encryption at Rest and in Transit
  - **Status:** RECOMMENDED

Use case: Denial of Service

CRT4. Threat name: Attackers carry out denial of service by API Abuse/Flooding
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR4. Countermeasure name:** Implement rate limiting and robust load balancing
  - **Status:** RECOMMENDED

Use case: Elevation of Privilege

CRT5. Threat name: Attackers gain elevated privilege from Broken Function Level Authorization
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR5. Countermeasure name:** Role-Based Access Control (RBAC)
  - **Status:** RECOMMENDED

Use case: Tampering

CRT6. Threat name: Attackers perform Injection Attacks and Scripting
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR6. Countermeasure name:** Implement Proper Authorization Checks
  - **Status:** RECOMMENDED

Use case: Repudiation

CRT7. Threat name: Lack of evidences of actions due to insufficient Auditing and Logging
- **Inherent risk:** Medium
- **Current risk:** Medium
- **Projected risk:** Medium
- **State:** Expose
- **CR7. Countermeasure name:** Implement Logging and Monitoring
  - **Status:** RECOMMENDED

Component: Authentication and Authorization Module

Use case: Elevation of Privilege

CRT8. Threat name: Attackers gain unauthorized access or elevated privileges, e.g., via stolen credentials, cookies, or tokens

- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR8. Countermeasure name:** Use secure access control mechanisms
  - **Status:** RECOMMENDED

Use case: Tampering

CRT9. Threat name: Attackers inject malicious content, e.g., SQL queries, to manipulate or access data

- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR9. Countermeasure name:** Input validation and sanitization
  - **Status:** RECOMMENDED

Use case: Information Disclosure

CRT10. Threat name: Attackers intercept or eavesdrop on sensitive information during transmission

- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR10. Countermeasure name:** Enforce secure configuration and encryption
  - **Status:** RECOMMENDED

Use case: Denial of Service

CRT11. Threat name: Attackers use enumeration to discover valid user identifiers, potentially creating a Denial of Service (DoS) condition

- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR11. Countermeasure name:** Rate limiting and proper resource management
  - **Status:** RECOMMENDED

Use case: Repudiation

CRT12. Threat name: Lack of evidences of misuse due to insufficient logging

- **Inherent risk:** Medium
- **Current risk:** Medium
- **Projected risk:** Medium
- **State:** Expose
- **CR12. Countermeasure name:** Create a policy and workflow for comprehensive logging and monitoring
  - **Status:** RECOMMENDED

Component: Mobile UI

Use case: Spoofing

CRT13. Threat name: Attackers attempt to access sensitive information

- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR13. Countermeasure name:** Use secure storage mechanisms
  - **Status:** RECOMMENDED
- **CR14. Countermeasure name:** Implement encryption of communications
  - **Status:** RECOMMENDED
- **CR15. Countermeasure name:** Implement Two-Factor Authentication to provide an extra layer of security
  - **Status:** RECOMMENDED

Use case: Elevation of Privilege

CRT14. Threat name: Attackers exploit vulnerable dependencies to execute malicious activities

- **Inherent risk:** Medium
- **Current risk:** Medium
- **Projected risk:** Medium

- **State:** Expose
- **CR16. Countermeasure name:** Use the app sandbox feature provided by the Operating System to contain the app and limit its access
  - **Status:** RECOMMENDED

CRT15. Threat name: Attackers gain unauthorized access to the App's functions or data
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR17. Countermeasure name:** Validate all user inputs to protect against injection attacks
  - **Status:** RECOMMENDED

Component: PostgreSQL

Use case: Denial of Service

CRT16. Threat name: Attackers cause denial of service through resource exhaustion
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR18. Countermeasure name:** Implement rate limiting and resource throttling
  - **Status:** RECOMMENDED

Use case: Information Disclosure

CRT17. Threat name: Attackers exfiltrate data due to insecure backup procedures
- **Inherent risk:** Critical
- **Current risk:** Critical
- **Projected risk:** Critical
- **State:** Expose
- **CR19. Countermeasure name:** Implement secure backup procedures with encryption and access controls
  - **Status:** RECOMMENDED

CRT18. Threat name: Attackers exploit misconfigurations in postgresql settings
- **Inherent risk:** Critical
- **Current risk:** Critical
- **Projected risk:** Critical
- **State:** Expose
- **CR20. Countermeasure name:** Harden postgresql configuration and restrict network access
  - **Status:** RECOMMENDED

CRT19. Threat name: Attackers exploit sql injection vulnerabilities
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR21. Countermeasure name:** Use parameterized queries and validate inputs
  - **Status:** RECOMMENDED

CRT20. Threat name: Attackers intercept data due to unencrypted communications
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR22. Countermeasure name:** Enforce TLS encryption for all connections
  - **Status:** RECOMMENDED

Use case: Tampering

CRT21. Threat name: Attackers exploit outdated postgresql vulnerabilities
- **Inherent risk:** Critical
- **Current risk:** Critical
- **Projected risk:** Critical
- **State:** Expose
- **CR23. Countermeasure name:** Regularly update postgresql to the latest secure version
  - **Status:** RECOMMENDED

CRT22. Threat name: Attackers tamper with data due to insecure file permissions
- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR24. Countermeasure name:** Enforce secure file permissions on PostgreSQL database files
  - **Status:** RECOMMENDED

Use case: Spoofing

CRT23. Threat name: Attackers gain unauthorized access due to weak authentication

- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR25. Countermeasure name:** Implement robust authentication and role-based access control
  - **Status:** RECOMMENDED

Component: Push Notification

Use case: Tampering

CRT24. Threat name: Attackers intercept, manipulate, or take control of notifications

- **Inherent risk:** Critical
- **Current risk:** Critical
- **Projected risk:** Critical
- **State:** Expose
- **CR26. Countermeasure name:** Implement strong authentication and authorization checks
  - **Status:** RECOMMENDED

Use case: Denial of Service

CRT25. Threat name: Attackers overload users by sending too many notifications

- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR27. Countermeasure name:** Implement rate limiting
  - **Status:** RECOMMENDED

Use case: Spoofing

CRT26. Threat name: Attackers send notifications to impersonate legitimate services

- **Inherent risk:** High
- **Current risk:** High
- **Projected risk:** High
- **State:** Expose
- **CR28. Countermeasure name:** Implement content validation and filtering mechanisms
  - **Status:** RECOMMENDED

End of Current Risk Report