# SECURITY ANALYSIS REPORT

Generated By

Backstage Rookie

**Provider: Backstage Rookie**

**contact - swarajdarekar9@gmail.com**

**Client:** Chandrakant Darekar

darekar138@gmail.com

**Scan ID:** 69b8fd3c-c5d7-46d4-a339-a5514d1ad773

**Date:** 2026-02-10 17:04:20

**Version:** 1.0.0

# TABLE OF CONTENTS

## EXECUTIVE SUMMARY

This security assessment identified critical and high-severity vulnerabilities primarily related to insecure operational practices and external API interactions. A hardcoded JWT token was found, alongside the Flask application running in debug mode, which could lead to arbitrary code execution. Additionally, a lack of timeouts in external API calls introduces reliability risks, and pervasive verbose error messages provide attackers with valuable internal system information. Addressing these issues is crucial for enhancing the application's overall security posture and resilience.

### Identified Vulnerabilities

| ID | Title | CVSS | Page |
|---|---|---|---|
| C1 | Hardcoded JWT Token | 9.8 | |
| H1 | Flask Debug Mode Enabled in Production | 8.8 | |
| M1 | Missing Request Timeout for External API Calls | 6.5 | |
| L1 | Verbose Error Messages Exposing Internal Details | 3.1 | |

# METHODOLOGY

## Introduction

This report details the results of a security assessment conducted on the specified repository. The analysis involved a multi-layered approach, combining automated static analysis tools with advanced, AI-driven verification and enrichment to identify potential security vulnerabilities.

## Objective

The primary objective of this assessment was to identify security weaknesses, assess their potential impact, and provide actionable recommendations for remediation to improve the overall security posture of the application.

## Scope

The assessment was performed on the source code of the repository cloned at the time of the scan. The analysis focused on common web application vulnerabilities, insecure coding practices, and dependency risks.

## Systems in Scope

| Name | IP |
| --- | --- |
| | |
| | |
| | |
| | |
| | |

## User Accounts

As this was a static source code analysis, no user accounts were provisioned or tested.

## FINDINGS

### C1 – Hardcoded JWT Token

**Severity:** Critical

**CVSS Score:** 9.8

**CVSS Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

**Target:** tasks.py

#### Overview

A sensitive JSON Web Token (JWT) is directly embedded within the `tasks.py` file, posing a severe risk of compromise.

#### Details

The `token` variable on line 14 of `tasks.py` contains a full JWT string. While the specific token found might be intended for testing purposes, its presence indicates an insecure practice of hardcoding sensitive information directly into source code. If a production token or a token with elevated privileges were ever hardcoded in a similar manner, it could be inadvertently exposed through various channels such as source code repositories, build artifacts, or even accidentally committed to public version control systems. Attackers could then extract and reuse such tokens to gain unauthorized access, impersonate legitimate users, bypass authentication, or perform other malicious actions, leading to significant data breaches or system compromise.

#### Evidence

- **tasks.py:14:** token = "eyJhbGciOiJIUzI1NiIsImtpZCI6IkRUK3NRQURhd0Uwdm9RbkwiLCJ0eXAiOiJKV1QifQ.eyJpc3MiOiJodHRwczovL2Frd2VtZWZld2ZtenhlZ2lkZXNsLnN1cGFiYXNlLmNvL2F1dGgvdjEi

#### References

- https://owasp.org/www-community/Weak_Authentication_and_Session_Management
- https://cheatsheetseries.owasp.org/cheatsheets/Secrets_Management_Cheat_Sheet.html

#### Recommendation

- Never hardcode sensitive information like API keys, database credentials, or access tokens directly in source code. - Utilize environment variables, a dedicated secrets management service (e.g., AWS Secrets Manager, HashiCorp Vault, Azure Key Vault), or a secure configuration management system to store and retrieve sensitive data at runtime. - Ensure that sensitive data is loaded securely from these external sources and is not logged or exposed in error messages. - Implement strict access controls for repositories and enforce code review policies to prevent accidental exposure of such files.

**Prompt to Solve the Vulnerability:**

In tasks.py, at line 14, a JWT token is hardcoded into the 'token' variable. To secure this, remove the token from the code and store it in a secure environment variable, configuration file, or a dedicated secrets management service. For example, replace 'token = "your_hardcoded_token"' with 'token = os.getenv("YOUR_JWT_TOKEN_ENV_VAR")' and ensure 'YOUR_JWT_TOKEN_ENV_VAR' is set externally without being committed to version control. Also, ensure the environment variable loading (`load_dotenv()`, `os.getenv()`) is correctly configured and used for *all* secrets, including test tokens.

## H1 – Flask Debug Mode Enabled in Production

**Severity:** High

**CVSS Score:** 8.8

**CVSS Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

**Target:** app.py

### Overview

The Flask application is configured to run in debug mode, which enables the Werkzeug debugger, posing a critical security risk in a production environment.

### Details

Line 1872 of `app.py` explicitly calls `app.run(debug=True, port=5000, use_reloader=True)`. Running a Flask application with `debug=True` in a production environment is extremely dangerous. The Werkzeug debugger, when active, allows the execution of arbitrary Python code on the server if an attacker gains access to the debug console (e.g., by triggering an error). This can be exploited to read sensitive files, modify data, execute system commands, and ultimately gain full control over the server. Furthermore, debug mode often exposes detailed tracebacks, environment variables, and other internal application details that can significantly aid an attacker in mapping out the system and identifying further vulnerabilities. The `use_reloader=True` option, while convenient for development, can also expose internal application states and increase the attack surface.

### Evidence

- **app.py:1872:** app.run(debug=True, port=5000, use_reloader=True)

### References

- https://flask.palletsprojects.com/en/2.3.x/deploying/#in-production
- https://owasp.org/www-community/Vulnerabilities/Arbitrary_Code_Execution

### Recommendation

- Disable debug mode (`debug=False`) when deploying the application to production. This is the most critical step. - Use a production-ready WSGI server (e.g., Gunicorn, uWSGI, or mod_wsgi) to serve the Flask application, rather than directly calling `app.run()`. - Configure production environments to never expose detailed error messages or stack traces to end-users. Instead, log these securely server-side. - Ensure that environment-specific configurations properly differentiate between development and production settings, perhaps using configuration files or environment variables to control the debug flag.

**Prompt to Solve the Vulnerability:**

In app.py, at line 1872, the Flask application is run with 'debug=True'. To prevent critical security vulnerabilities in production, change this to 'debug=False' or ensure this line is only executed in a development environment. For production deployment, you should use a production-ready WSGI server like Gunicorn or uWSGI instead of calling 'app.run()' directly. Example production usage with Gunicorn: `gunicorn -w 4 'app:app'`

## M1 – Missing Request Timeout for External API Calls

**Severity:** Medium

**CVSS Score:** 6.5

**CVSS Vector:** CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:N/A:L

**Target:** routes\payments.py

### Overview

Multiple HTTP requests made to external services (Cashfree API) lack a configured timeout, making the application vulnerable to denial-of-service (DoS) or performance degradation.

### Details

The `requests.post` and `requests.get` calls on lines 90, 136, and 663 in `routes\payments.py` do not specify a `timeout` parameter. Without a timeout, if the external Cashfree API becomes unresponsive, excessively slow, or experiences network issues, the application's HTTP client will hang indefinitely while waiting for a response. This can lead to several adverse effects, including: worker processes becoming blocked and exhausted, increased latency for legitimate user requests, and ultimately, a denial-of-service condition for parts or all of the application. This prevents the application from gracefully handling external service failures and managing its resources effectively.

### Evidence

- **routes\payments.py:90:** response = requests.post( f"{CASHFREE_BASE_URL}/orders", headers=get_cashfree_headers(), json=payload )
- **routes\payments.py:136:** response = requests.get( f"{CASHFREE_BASE_URL}/orders/{order_id}", headers=get_cashfree_headers() )
- **routes\payments.py:663:** payout_response = requests.post(payout_api_url, json=cashfree_payout_payload, headers=cashfree_headers)

### References

- https://owasp.org/www-community/attacks/Denial_of_Service
- https://docs.python-requests.org/en/latest/user/advanced/#timeouts

### Recommendation

- Always specify a `timeout` parameter for all external HTTP requests to ensure they do not hang indefinitely. - Choose an appropriate timeout value (e.g., 5-10 seconds) based on the expected response time of the external service and the application's tolerance for delays. - Implement robust error handling around external API calls, including `try-except` blocks to specifically catch `requests.exceptions.Timeout` and `requests.exceptions.ConnectionError`. - Consider implementing retry logic with exponential backoff for transient network errors to improve resilience.

**Prompt to Solve the Vulnerability:**

In routes/payments.py, at lines 90, 136, and 663, HTTP requests are made to external APIs without specifying a timeout. To prevent the application from hanging indefinitely and improve resilience, add a 'timeout' parameter to each 'requests.post' and 'requests.get' call. For example, change 'requests.post(url, ...)' to 'requests.post(url, ..., timeout=5)' (where 5 is an example timeout in seconds), and implement proper exception handling for 'requests.exceptions.Timeout'.

## L1 – Verbose Error Messages Exposing Internal Details

**Severity:** Low

**CVSS Score:** 3.1

**CVSS Vector:** CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N

**Target:** Multiple files (app.py, routes\payments.py)

### Overview

The application frequently returns detailed internal error messages, including stack traces or exception messages (`str(e)`), to the client during API failures.

### Details

Across numerous API endpoints in `app.py` and `routes\payments.py`, broad `Exception` blocks are used to catch errors, which are then returned to the client in JSON responses via `jsonify({'msg': '...', 'error': str(e)})`. While this practice can be useful during development for debugging, exposing the full string representation of an exception (`str(e)`) in a production environment is a security risk. Such verbose error messages can inadvertently reveal sensitive information about the server's internal workings, including file paths, database schema details, names and versions of internal libraries, or specific error conditions. This information can be leveraged by attackers to gather intelligence, map out the system architecture, and identify potential targets for more sophisticated attacks. This violates the principle of least privilege in error reporting.

### Evidence

- **app.py:117:** return jsonify({'msg': 'Registration failed', 'error': str(e)}), 500
- **app.py:187:** return jsonify({'msg': 'Login failed', 'error': str(e)}), 500
- **routes\payments.py:115:** return jsonify({'msg': 'Internal server error', 'error': str(e)}), 500
- **routes\payments.py:186:** return jsonify({'msg': 'Verification failed', 'error': str(e)}), 500

### References

- https://owasp.org/www-community/errors/Detailed_Error_Messages
- https://cwe.mitre.org/data/definitions/209.html

### Recommendation

- Replace verbose error messages with generic, user-friendly messages for end-users. For example, instead of `str(e)`, return a message like 'An unexpected error occurred. Please try again later.' - Log detailed error messages (including stack traces and full exception details) securely on the server-side for debugging purposes, but never expose them directly to clients. - Implement a centralized, custom error handler for the Flask application to ensure a consistent and secure error response format across all endpoints. - Avoid catching generic `Exception` types; instead, catch specific exceptions and handle them appropriately, providing tailored but non-revealing error responses.

**Prompt to Solve the Vulnerability:**

In app.py and routes/payments.py, many error handlers return 'jsonify({'msg': '...', 'error': str(e)})'. To prevent information leakage, replace 'str(e)' with a generic error message for end-users. For example, change 'return jsonify({'msg': 'Login failed', 'error': str(e)}), 500' to 'return jsonify({'msg': 'Login failed', 'error': 'An unexpected error occurred. Please try again later.'}), 500'. Ensure detailed errors are logged server-side but not exposed externally. Implement a global Flask error handler to catch unhandled exceptions and return a standard, non-descriptive error response.

## ENDPOINT SECURITY ANALYSIS

### Endpoint Path /register

| | |
|---|---|
| **Endpoint Path** | /register |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 73 |
| **Authentication Required** | No |
| **Risk Severity** | Critical |
| **CVSS Score** | 9.1 |

#### Authentication Analysis

This endpoint does not enforce authentication, which may expose it to unauthorized access depending on its functionality and the sensitivity of data processed.

#### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

#### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

#### Identified Security Risks

- **AUTH_MISSING** (high): Authentication is missing for the user registration endpoint.

  *Potential Attack Scenario:*

  An attacker could create a large number of fake user accounts, potentially leading to denial-of-service or resource abuse.

#### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | high | Lack of authentication on registration can lead to unauthorized account creation, violating principles of access control and system integrity. |
| **ISO/IEC 27001** | Yes | high | Unauthenticated access to create accounts undermines controls over user registration and access management. |
| **CSA STAR** | Yes | high | Weak controls around user provisioning can impact the integrity and availability of services. |

#### Security Assessment Notes

**CVSS Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:H

##### References

- https://csrc.nist.gov/cve/entry/cve/2023/4094
- https://owasp.org/www-community/vulnerabilities/Cross-Site_Request_Forgery_(CSRF)

### Endpoint Path /login

| | |
|---|---|
| **Endpoint Path** | /login |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 120 |
| **Authentication Required** | No |
| **Risk Severity** | Critical |
| **CVSS Score** | 9.1 |

#### Authentication Analysis

This endpoint does not enforce authentication, which may expose it to unauthorized access depending on its functionality and the sensitivity of data processed.

#### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

#### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

- **AUTH_MISSING** (high): Authentication is missing for the login endpoint. This could allow unauthenticated access or brute-force attacks.

  *Potential Attack Scenario:*

  An attacker could attempt to brute-force login credentials, potentially gaining unauthorized access to user accounts.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | high | Lack of authentication or authorization controls on login can lead to unauthorized access, compromising data confidentiality and integrity. |
| **ISO/IEC 27001** | Yes | high | Insecure login mechanisms can lead to unauthorized access, violating access control policies. |
| **CSA STAR** | Yes | high | Weak authentication controls can expose the system to unauthorized access and credential stuffing attacks. |

### Security Assessment Notes

**CVSS Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:H

**References**

- https://owasp.org/www-community/vulnerabilities/Brute_force_attacks

## Endpoint Path /verify-instagram

| | |
|---|---|
| **Endpoint Path** | /verify-instagram |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 192 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 191, 191, 192, 192, 193, 193, 194, 194, 191, 191, 193, 193, 194, 194, 191, 191, 193, 193, 194, 194.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated endpoint for Instagram verification, likely involves secure handling of credentials or tokens. |
| **ISO/IEC 27001** | Yes | low | Authentication is present, contributing to secure access controls. |
| **CSA STAR** | Yes | low | Authenticated endpoint indicates adherence to access control principles. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /verify-instagram/

| | |
|---|---|
| **Endpoint Path** | /verify-instagram/ |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 192 |

| | |
|---|---|
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 191, 191, 192, 192, 193, 193, 194, 194, 191, 191, 193, 193, 194, 194, 191, 191, 193, 193, 194, 194.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated endpoint for Instagram verification, likely involves secure handling of credentials or tokens. |
| **ISO/IEC 27001** | Yes | low | Authentication is present, contributing to secure access controls. |
| **CSA STAR** | Yes | low | Authenticated endpoint indicates adherence to access control principles. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /request-password-reset

| | |
|---|---|
| **Endpoint Path** | /request-password-reset |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 224 |
| **Authentication Required** | No |
| **Risk Severity** | Medium |
| **CVSS Score** | 6.5 |

### Authentication Analysis

This endpoint does not enforce authentication, which may expose it to unauthorized access depending on its functionality and the sensitivity of data processed.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

- **AUTH_MISSING** (medium): Authentication is not explicitly required for password reset requests. This could be vulnerable to abuse if not properly rate-limited or validated.

  *Potential Attack Scenario:*

  An attacker could flood the system with password reset requests for various users, potentially leading to denial-of-service or social engineering attacks if email notifications are not secured.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | medium | Password reset functionality needs robust controls to prevent unauthorized access and denial of service, impacting system integrity and availability. |
| **ISO/IEC 27001** | Yes | medium | Insecure password reset mechanisms can compromise user account security and integrity. |

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **CSA STAR** | Yes | medium | Lack of strong authentication or rate limiting on password resets can be exploited. |

### Security Assessment Notes

**CVSS Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:L/A:H

**References**

- https://owasp.org/www-community/vulnerabilities/Authentication_cheat_sheet
- https://owasp.org/www-project-top-10/2021/A02_2021-Cryptographic_Failure

## Endpoint Path /api/brand/campaigns

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 248 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 247, 247, 248, 248, 249, 249, 253, 253, 247, 247, 249, 249, 253, 253, 247, 247, 249, 249, 253, 253.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated endpoint for campaign creation ensures proper authorization and access control. |
| **ISO/IEC 27001** | Yes | low | Authentication is in place for creating campaigns, aligning with access control requirements. |
| **CSA STAR** | Yes | low | Authenticated POST request indicates controls are in place for creating campaign resources. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 292 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 291, 291, 292, 292, 293, 293, 297, 297, 291, 291, 293, 293, 297, 297, 291, 291, 293, 293, 297, 297.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for listing campaigns ensures data is accessed by authorized users. |
| **ISO/IEC 27001** | Yes | low | Authentication is present for retrieving campaign data, adhering to access control. |
| **CSA STAR** | Yes | low | Authenticated read access to campaign data is a good practice. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/campaigns

| | |
|---|---|
| **Endpoint Path** | /api/campaigns |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 329 |
| **Authentication Required** | No |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint does not enforce authentication, which may expose it to unauthorized access depending on its functionality and the sensitivity of data processed.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

- **AUTH_MISSING** (low): Authentication is not explicitly required for listing all campaigns. Depending on the data sensitivity, this might be acceptable or require review.

  *Potential Attack Scenario:*

  If campaign data contains sensitive information, an unauthenticated attacker could view this data, leading to information disclosure.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | While authentication is missing, this endpoint might be intended for public access. If sensitive data is exposed, a higher risk would apply. |
| **ISO/IEC 27001** | Yes | low | Access controls for retrieving all campaigns should be reviewed based on data sensitivity. |
| **CSA STAR** | Yes | low | Publicly accessible read operations are common, but data sensitivity needs to be considered. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/campaigns/

| | |
|---|---|
| **Endpoint Path** | /api/campaigns/ |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 388 |
| **Authentication Required** | No |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint does not enforce authentication, which may expose it to unauthorized access depending on its functionality and the sensitivity of data processed.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

- **AUTH_MISSING** (low): Authentication is not explicitly required for retrieving a specific campaign by ID. Depending on the data sensitivity, this might be acceptable or require review.

  *Potential Attack Scenario:*

  If campaign details contain sensitive information, an unauthenticated attacker could view this data, leading to information disclosure.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Access controls for retrieving specific campaigns should be reviewed based on data sensitivity. |
| **ISO/IEC 27001** | Yes | low | Authentication should be considered for accessing specific campaign details to maintain data confidentiality. |
| **CSA STAR** | Yes | low | Read access controls should be enforced if campaign details are sensitive. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/creator/your-campaigns

| | |
|---|---|
| **Endpoint Path** | /api/creator/your-campaigns |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 495 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 494, 494, 495, 495, 496, 496, 501, 501, 494, 494, 496, 496, 501, 501, 494, 494, 496, 496, 501, 501.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated endpoint to retrieve creator-specific campaigns ensures data privacy and integrity. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing personal campaign data adheres to access control principles. |
| **CSA STAR** | Yes | low | Authenticated access to creator's campaigns is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/creator/submit-clip

| | |
|---|---|
| **Endpoint Path** | /api/creator/submit-clip |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 583 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 582, 582, 583, 583, 584, 584, 590, 590, 582, 582, 584, 584, 590, 590, 582, 582, 584, 584, 590, 590.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated endpoint for submitting clips ensures only authorized creators can upload content. |
| **ISO/IEC 27001** | Yes | low | Authentication is required for content submission, maintaining integrity. |
| **CSA STAR** | Yes | low | Authenticated access for content submission is a standard security control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/creator/campaign-clips

| | |
|---|---|
| **Endpoint Path** | /api/creator/campaign-clips |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 657 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 656, 656, 657, 657, 658, 658, 662, 662, 656, 656, 658, 658, 662, 662, 656, 656, 658, 658, 662, 662.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated endpoint for retrieving campaign clips ensures creators can only access their own data. |
| **ISO/IEC 27001** | Yes | low | Authentication is required for accessing creator's clips, maintaining data confidentiality. |
| **CSA STAR** | Yes | low | Authenticated read access to creator's campaign clips is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/creator/accepted-clip-details/

| | |
|---|---|
| **Endpoint Path** | /api/creator/accepted-clip-details/ |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 722 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 721, 721, 722, 722, 723, 723, 727, 727, 721, 721, 723, 723, 727, 727, 721, 721, 723, 723, 727, 727.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated endpoint for accessing accepted clip details ensures only authorized creators can view this information. |
| **ISO/IEC 27001** | Yes | low | Authentication is required for accessing specific clip details, protecting data confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access to clip details is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns/

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns/ |
| **HTTP Methods** | DELETE, OPTIONS |
| **Source Location** | ..\Mipoe-Backend\app.py - 761 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

#### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 761, 761, 765, 765, 766, 766, 770, 770, 765, 765, 766, 766, 770, 770, 765, 765, 766, 766, 770, 770.

#### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

#### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

#### Identified Security Risks

No direct security risks were identified for this endpoint.

#### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated DELETE operation for campaigns ensures only authorized entities can remove data, maintaining integrity. |
| **ISO/IEC 27001** | Yes | low | Authentication for deleting campaign data is crucial for data integrity and access control. |
| **CSA STAR** | Yes | low | Authenticated DELETE operations help prevent unauthorized data modification or deletion. |

#### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/creator/clip/

| | |
|---|---|
| **Endpoint Path** | /api/creator/clip/ |
| **HTTP Methods** | DELETE, OPTIONS |
| **Source Location** | ..\Mipoe-Backend\app.py - 833 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

#### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 832, 832, 833, 833, 837, 837, 838, 838, 842, 842, 832, 832, 837, 837, 838, 838, 842, 842, 832, 832, 837, 837, 838, 838, 842, 842.

#### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

#### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

#### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated DELETE operation for clips ensures only authorized creators can remove their content, maintaining data integrity. |
| **ISO/IEC 27001** | Yes | low | Authentication for deleting creator's clips is critical for access control and data integrity. |
| **CSA STAR** | Yes | low | Authenticated DELETE operations help prevent unauthorized data modification or deletion. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/admin/campaigns

| | |
|---|---|
| **Endpoint Path** | /api/admin/campaigns |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 888 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 887, 887, 888, 888, 889, 889, 887, 887, 889, 889, 887, 887, 889, 889.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated access for administrators to view all campaigns is appropriate for management purposes. |
| **ISO/IEC 27001** | Yes | low | Role-based access control for administrative functions like viewing all campaigns is in place. |
| **CSA STAR** | Yes | low | Authenticated administrative access is a standard security measure. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/admin/clip/

| | |
|---|---|
| **Endpoint Path** | /api/admin/clip/ |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 935 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 934, 934, 935, 935, 936, 936, 934, 934, 936, 936, 934, 934, 936, 936.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated administrative PUT operation for clips ensures proper authorization for modifications. |
| **ISO/IEC 27001** | Yes | low | Role-based access control for administrative modification of clip data is implemented. |
| **CSA STAR** | Yes | low | Authenticated administrative access for updating clip data is a sound security practice. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/admin/clip/

| | |
|---|---|
| **Endpoint Path** | /api/admin/clip/ |
| **HTTP Methods** | DELETE, OPTIONS |
| **Source Location** | ..\Mipoe-Backend\app.py - 1026 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1025, 1025, 1026, 1026, 1031, 1031, 1025, 1025, 1031, 1031, 1025, 1025, 1031, 1031.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated administrative DELETE operation for clips ensures only authorized administrators can remove data. |
| **ISO/IEC 27001** | Yes | low | Role-based access control for administrative deletion of clip data is implemented. |
| **CSA STAR** | Yes | low | Authenticated administrative access for deleting clip data is a sound security practice. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/creator/profile

| | |
|---|---|
| **Endpoint Path** | /api/creator/profile |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 1079 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

**Authentication Analysis**

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1078, 1078, 1079, 1079, 1080, 1080, 1084, 1084, 1078, 1078, 1080, 1080, 1084, 1084, 1078, 1078, 1080, 1080, 1084, 1084.

**Request Analysis**

**Content Type:** unknown

No request fields were identified for this endpoint.

**Response Analysis**

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

**Identified Security Risks**

No direct security risks were identified for this endpoint.

**Compliance Impact**

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated endpoint for retrieving creator profile ensures only the creator can access their personal information. |
| **ISO/IEC 27001** | Yes | low | Authentication is required for accessing personal profile data, protecting confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access to creator profiles is appropriate. |

**Security Assessment Notes**

**CVSS Vector:**

**References**

## Endpoint Path /api/creator/profile

| | |
|---|---|
| **Endpoint Path** | /api/creator/profile |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1113 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

**Authentication Analysis**

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1112, 1112, 1113, 1113, 1114, 1114, 1118, 1118, 1112, 1112, 1114, 1114, 1118, 1118, 1112, 1112, 1114, 1114, 1118, 1118.

**Request Analysis**

**Content Type:** unknown

No request fields were identified for this endpoint.

**Response Analysis**

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

**Identified Security Risks**

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated PUT operation for creator profile ensures only the creator can modify their personal information. |
| **ISO/IEC 27001** | Yes | low | Authentication is required for updating personal profile data, protecting integrity and confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access for updating creator profiles is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns//image

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns//image |
| **HTTP Methods** | PUT, OPTIONS |
| **Source Location** | ..\Mipoe-Backend\app.py - 1159 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1158, 1158, 1159, 1159, 1165, 1165, 1166, 1166, 1170, 1170, 1158, 1158, 1165, 1165, 1166, 1166, 1170, 1170, 1158, 1158, 1165, 1165, 1166, 1166, 1170, 1170.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated PUT operation for updating campaign image ensures only authorized users can modify campaign assets. |
| **ISO/IEC 27001** | Yes | low | Authentication for modifying campaign assets aligns with access control and integrity principles. |
| **CSA STAR** | Yes | low | Authenticated access for updating campaign images is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns//budget

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns//budget |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1198 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1197, 1197, 1198, 1198, 1199, 1199, 1204, 1204, 1197, 1197, 1199, 1199, 1204, 1204, 1197, 1197, 1199, 1199, 1204, 1204.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated PUT operation for campaign budget ensures only authorized users can modify financial aspects of campaigns. |
| **ISO/IEC 27001** | Yes | low | Authentication for modifying campaign budgets is crucial for financial integrity and access control. |
| **CSA STAR** | Yes | low | Authenticated access for updating campaign budgets is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns//requirements

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns//requirements |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1230 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1229, 1229, 1230, 1230, 1231, 1231, 1235, 1235, 1229, 1229, 1231, 1231, 1235, 1235, 1229, 1229, 1231, 1231, 1235, 1235.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated PUT operation for campaign requirements ensures only authorized users can modify campaign specifications. |
| **ISO/IEC 27001** | Yes | low | Authentication for modifying campaign requirements aligns with access control and data integrity. |
| **CSA STAR** | Yes | low | Authenticated access for updating campaign requirements is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns//status

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns//status |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1259 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1258, 1258, 1259, 1259, 1260, 1260, 1264, 1264, 1258, 1258, 1260, 1260, 1264, 1264, 1258, 1258, 1260, 1260, 1264, 1264.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated PUT operation for campaign status ensures only authorized users can change campaign lifecycle. |
| **ISO/IEC 27001** | Yes | low | Authentication for modifying campaign status is crucial for data integrity and operational control. |
| **CSA STAR** | Yes | low | Authenticated access for updating campaign status is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns//view_threshold

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns//view_threshold |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1290 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1289, 1289, 1290, 1290, 1291, 1291, 1295, 1295, 1289, 1289, 1291, 1291, 1295, 1295, 1289, 1289, 1291, 1291, 1295, 1295.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated PUT operation for campaign view threshold ensures only authorized users can modify campaign targeting parameters. |
| **ISO/IEC 27001** | Yes | low | Authentication for modifying campaign view thresholds is essential for data integrity and accuracy. |
| **CSA STAR** | Yes | low | Authenticated access for updating campaign view thresholds is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns//deadline

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns//deadline |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1321 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1320, 1320, 1321, 1321, 1322, 1322, 1326, 1326, 1320, 1320, 1322, 1322, 1326, 1326, 1320, 1320, 1322, 1322, 1326, 1326.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated PUT operation for campaign deadline ensures only authorized users can modify campaign timing. |
| **ISO/IEC 27001** | Yes | low | Authentication for modifying campaign deadlines is crucial for operational control and data integrity. |
| **CSA STAR** | Yes | low | Authenticated access for updating campaign deadlines is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/campaigns//pending-payouts

| | |
|---|---|
| **Endpoint Path** | /api/brand/campaigns//pending-payouts |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 1358 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1357, 1357, 1358, 1358, 1365, 1365, 1366, 1366, 1371, 1371, 1357, 1357, 1365, 1365, 1366, 1366, 1371, 1371, 1357, 1357, 1365, 1365, 1366, 1366, 1371, 1371.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for pending payouts ensures access to financial data is restricted to authorized personnel. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing financial payout information is critical for data confidentiality and integrity. |
| **CSA STAR** | Yes | low | Authenticated access to financial data is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/profile

| | |
|---|---|
| **Endpoint Path** | /api/brand/profile |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 1455 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1454, 1454, 1455, 1455, 1456, 1456, 1460, 1460, 1454, 1454, 1456, 1456, 1460, 1460, 1454, 1454, 1456, 1456, 1460, 1460.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for brand profile ensures only authorized users can view brand information. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing brand profile data maintains confidentiality and integrity. |
| **CSA STAR** | Yes | low | Authenticated access to brand profiles is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/brand/profile

| | |
|---|---|
| **Endpoint Path** | /api/brand/profile |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1482 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

**Authentication Analysis**

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1481, 1481, 1482, 1482, 1483, 1483, 1487, 1487, 1481, 1481, 1483, 1483, 1487, 1487, 1481, 1481, 1483, 1483, 1487, 1487.

**Request Analysis**

**Content Type:** unknown

No request fields were identified for this endpoint.

**Response Analysis**

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

**Identified Security Risks**

No direct security risks were identified for this endpoint.

**Compliance Impact**

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated PUT operation for brand profile ensures only authorized users can modify brand information. |
| **ISO/IEC 27001** | Yes | low | Authentication for updating brand profile data is crucial for integrity and confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access for updating brand profiles is appropriate. |

**Security Assessment Notes**

**CVSS Vector:**

**References**

## Endpoint Path /api/admin/clip//view-count

| | |
|---|---|
| **Endpoint Path** | /api/admin/clip//view-count |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1514 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

**Authentication Analysis**

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1513, 1513, 1514, 1514, 1528, 1528, 1529, 1529, 1513, 1513, 1528, 1528, 1529, 1529, 1513, 1513, 1528, 1528, 1529, 1529.

**Request Analysis**

**Content Type:** unknown

No request fields were identified for this endpoint.

**Response Analysis**

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

**Identified Security Risks**

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated administrative PUT operation for clip view count ensures only authorized personnel can modify metrics. |
| **ISO/IEC 27001** | Yes | low | Authentication for modifying clip view counts is important for data integrity and preventing manipulation. |
| **CSA STAR** | Yes | low | Authenticated administrative access for updating clip view counts is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/admin/campaign//update-views

| | |
|---|---|
| **Endpoint Path** | /api/admin/campaign//update-views |
| **HTTP Methods** | PUT |
| **Source Location** | ..\Mipoe-Backend\app.py - 1580 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1579, 1579, 1580, 1580, 1591, 1591, 1592, 1592, 1579, 1579, 1591, 1591, 1592, 1592, 1579, 1579, 1591, 1591, 1592, 1592.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated administrative PUT operation for campaign views ensures only authorized personnel can modify campaign metrics. |
| **ISO/IEC 27001** | Yes | low | Authentication for modifying campaign view counts is important for data integrity and preventing manipulation. |
| **CSA STAR** | Yes | low | Authenticated administrative access for updating campaign view counts is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/admin/analytics/campaign-performance/

| | |
|---|---|
| **Endpoint Path** | /api/admin/analytics/campaign-performance/ |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 1635 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1634, 1634, 1635, 1635, 1640, 1640, 1641, 1641, 1634, 1634, 1640, 1640, 1641, 1641, 1634, 1634, 1640, 1640, 1641, 1641.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for campaign performance analytics ensures administrative access to sensitive performance data. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing analytics data ensures confidentiality and integrity. |
| **CSA STAR** | Yes | low | Authenticated administrative access to performance analytics is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/auth/google-sync

| | |
|---|---|
| **Endpoint Path** | /api/auth/google-sync |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 1738 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1737, 1737, 1738, 1738, 1738, 1739, 1744, 1744, 1745, 1745, 1748, 1748, 1761, 1761, 1761, 1737, 1737, 1739, 1744, 1744, 1745, 1745, 1748, 1748, 1772, 1772, 1778, 1737, 1737, 1744, 1744, 1745, 1745, 1772, 1772, 1778, 1772, 1772, 1782.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for Google sync implies secure handling of authentication tokens and user data. |
| **ISO/IEC 27001** | Yes | low | Authentication for integrating with external services like Google is crucial for data security and privacy. |
| **CSA STAR** | Yes | low | Authenticated integration with third-party services is a standard security practice. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /refresh

| | |
|---|---|
| **Endpoint Path** | /refresh |
| **HTTP Methods** | POST |
| **Source Location** | ..\Mipoe-Backend\app.py - 1816 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1815, 1815, 1816, 1816, 1821, 1821, 1822, 1822, 1815, 1815, 1821, 1821, 1822, 1822, 1815, 1815, 1821, 1821, 1822, 1822.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for refreshing tokens is standard practice for maintaining secure sessions. |
| **ISO/IEC 27001** | Yes | low | Secure token refresh mechanisms contribute to session management and access control. |
| **CSA STAR** | Yes | low | Authenticated token refresh is a common security control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /logout

| | |
|---|---|
| **Endpoint Path** | /logout |
| **HTTP Methods** | DELETE |
| **Source Location** | ..\Mipoe-Backend\app.py - 1832 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1831, 1831, 1832, 1832, 1836, 1836, 1831, 1831, 1836, 1836, 1831, 1831, 1836, 1836.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated DELETE request for logout ensures proper session termination. |
| **ISO/IEC 27001** | Yes | low | Secure logout procedures are part of access control and session management. |
| **CSA STAR** | Yes | low | Authenticated logout is a standard security control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /api/health

| | |
|---|---|
| **Endpoint Path** | /api/health |
| **HTTP Methods** | GET |
| **Source Location** | ..\Mipoe-Backend\app.py - 1843 |
| **Authentication Required** | No |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint does not enforce authentication, which may expose it to unauthorized access depending on its functionality and the sensitivity of data processed.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Health check endpoints are typically public and do not require authentication. However, ensuring they don't expose sensitive system information is important. |
| **ISO/IEC 27001** | Yes | low | Health check endpoints are common for monitoring and typically do not require authentication. Information disclosed should be limited. |
| **CSA STAR** | Yes | low | Publicly accessible health checks are standard. Ensure no sensitive system details are revealed. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /create-deposit-order

| | |
|---|---|
| **Endpoint Path** | /create-deposit-order |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 49 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 48, 48, 49, 49, 50, 50, 55, 55, 48, 48, 50, 50, 55, 55, 48, 48, 50, 50, 55, 55.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for creating deposit orders ensures financial transactions are authorized. |
| **ISO/IEC 27001** | Yes | low | Authentication for financial transaction initiation is crucial for integrity and preventing fraud. |
| **CSA STAR** | Yes | low | Authenticated access for financial transactions is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /verify-deposit

| | |
|---|---|
| **Endpoint Path** | /verify-deposit |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 121 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 120, 120, 121, 121, 122, 122, 127, 127, 120, 120, 122, 122, 127, 127, 120, 120, 122, 122, 127, 127.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for deposit verification ensures financial transaction integrity. |
| **ISO/IEC 27001** | Yes | low | Authentication for financial transaction verification is crucial for preventing unauthorized modifications. |
| **CSA STAR** | Yes | low | Authenticated access for financial transaction verification is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /virtual-account

| | |
|---|---|
| **Endpoint Path** | /virtual-account |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 192 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 191, 191, 192, 192, 193, 193, 198, 198, 191, 191, 193, 193, 198, 198, 191, 191, 193, 193, 198, 198.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for virtual account details ensures access to financial information is restricted. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing financial account details is crucial for confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access to virtual account details is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /wallet-balance

| | |
|---|---|
| **Endpoint Path** | /wallet-balance |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 243 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 242, 242, 243, 243, 244, 244, 247, 247, 242, 242, 244, 244, 247, 247, 242, 242, 244, 244, 247, 247.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for wallet balance ensures access to financial information is restricted. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing financial balance information is crucial for confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access to wallet balance is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /allocate-budget

| | |
|---|---|
| **Endpoint Path** | /allocate-budget |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 269 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 268, 268, 269, 269, 270, 270, 275, 275, 268, 268, 270, 270, 275, 275, 268, 268, 270, 270, 275, 275.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for budget allocation ensures financial operations are authorized. |
| **ISO/IEC 27001** | Yes | low | Authentication for financial budget management is crucial for integrity and preventing unauthorized use of funds. |
| **CSA STAR** | Yes | low | Authenticated access for budget allocation is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /reclaim-budget

| | |
|---|---|
| **Endpoint Path** | /reclaim-budget |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 349 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 348, 348, 349, 349, 350, 350, 355, 355, 348, 348, 350, 350, 355, 355, 348, 348, 350, 350, 355, 355.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for budget reclamation ensures financial operations are authorized. |
| **ISO/IEC 27001** | Yes | low | Authentication for financial budget reclamation is crucial for integrity and preventing unauthorized fund movement. |
| **CSA STAR** | Yes | low | Authenticated access for budget reclamation is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /distribute-to-creator

| | |
|---|---|
| **Endpoint Path** | /distribute-to-creator |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 436 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 435, 435, 436, 436, 455, 455, 460, 460, 435, 435, 455, 455, 460, 460, 435, 435, 455, 455, 460, 460.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for distributing funds to creators ensures financial transactions are authorized and logged. |
| **ISO/IEC 27001** | Yes | low | Authentication for fund distribution is crucial for preventing fraud and maintaining financial integrity. |
| **CSA STAR** | Yes | low | Authenticated access for fund distribution is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /creator-withdraw

| | |
|---|---|
| **Endpoint Path** | /creator-withdraw |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 560 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 559, 559, 560, 560, 578, 578, 583, 583, 559, 559, 578, 578, 583, 583, 559, 559, 578, 578, 583, 583.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for creator withdrawals ensures financial transactions are authorized and recorded. |
| **ISO/IEC 27001** | Yes | low | Authentication for creator withdrawals is crucial for preventing fraud and maintaining financial integrity. |
| **CSA STAR** | Yes | low | Authenticated access for creator withdrawals is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /creator/payout-details

| | |
|---|---|
| **Endpoint Path** | /creator/payout-details |
| **HTTP Methods** | POST, PUT |
| **Source Location** | ..\routes\payments.py - 735 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 734, 734, 735, 735, 750, 750, 755, 755, 734, 734, 750, 750, 755, 755, 734, 734, 750, 750, 755, 755.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST/PUT operations for payout details ensure creators can manage their financial information securely. |
| **ISO/IEC 27001** | Yes | low | Authentication for managing financial payout details is crucial for data integrity and preventing unauthorized access. |
| **CSA STAR** | Yes | low | Authenticated access for managing payout details is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /creator/payout-details

| | |
|---|---|
| **Endpoint Path** | /creator/payout-details |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 823 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 822, 822, 823, 823, 828, 828, 833, 833, 822, 822, 828, 828, 833, 833, 822, 822, 828, 828, 833, 833.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for payout details ensures creators can only access their own financial information. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing financial payout details is crucial for confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access to payout details is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /creator/verify-payout-details

| | |
|---|---|
| **Endpoint Path** | /creator/verify-payout-details |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 873 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 872, 872, 873, 873, 878, 878, 883, 883, 872, 872, 878, 878, 883, 883, 872, 872, 878, 878, 883, 883.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for verifying payout details ensures the integrity of financial information. |
| **ISO/IEC 27001** | Yes | low | Authentication for verifying financial payout details is crucial for integrity and preventing unauthorized changes. |
| **CSA STAR** | Yes | low | Authenticated access for verifying payout details is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /creator/withdrawals

| | |
|---|---|
| **Endpoint Path** | /creator/withdrawals |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 933 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 932, 932, 933, 933, 940, 940, 945, 945, 932, 932, 940, 940, 945, 945, 932, 932, 940, 940, 945, 945.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for withdrawal history ensures creators can only access their own transaction records. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing withdrawal history is crucial for confidentiality and integrity. |
| **CSA STAR** | Yes | low | Authenticated access to withdrawal history is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /creator/notifications/

| | |
|---|---|
| **Endpoint Path** | /creator/notifications/ |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 995 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

**Authentication Analysis**

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 994, 994, 995, 995, 999, 999, 1000, 1000, 1002, 1002, 994, 994, 999, 999, 1000, 1000, 1002, 1002, 994, 994, 999, 999, 1000, 1000, 1002, 1002, 1000, 1000, 1002, 1002, 1002, 1002, 1002, 1002.

**Request Analysis**

**Content Type:** unknown

No request fields were identified for this endpoint.

**Response Analysis**

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

**Identified Security Risks**

No direct security risks were identified for this endpoint.

**Compliance Impact**

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for creator notifications ensures creators can only access their own notifications. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing personal notifications is crucial for confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access to creator notifications is appropriate. |

**Security Assessment Notes**

**CVSS Vector:**

**References**

## Endpoint Path /transactions//

| | |
|---|---|
| **Endpoint Path** | /transactions// |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 1030 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

**Authentication Analysis**

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1029, 1029, 1030, 1030, 1030, 1031, 1031, 1035, 1035, 1035, 1038, 1042, 1029, 1029, 1031, 1031, 1035, 1035, 1035, 1038, 1042, 1029, 1029, 1031, 1031, 1035, 1035, 1038, 1042, 1038, 1042, 1038, 1042.

**Request Analysis**

**Content Type:** unknown

No request fields were identified for this endpoint.

**Response Analysis**

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

**Identified Security Risks**

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for transactions, with user_id parameter, ensures users can only access their own transaction history. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing transaction data is crucial for confidentiality and integrity. |
| **CSA STAR** | Yes | low | Authenticated access to transaction history is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /refund-campaign

| | |
|---|---|
| **Endpoint Path** | /refund-campaign |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 1110 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1109, 1109, 1110, 1110, 1126, 1126, 1131, 1131, 1109, 1109, 1126, 1126, 1131, 1131, 1109, 1109, 1126, 1126, 1131, 1131.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for refunding campaigns ensures only authorized actions can be taken on campaign funds. |
| **ISO/IEC 27001** | Yes | low | Authentication for refunding campaigns is crucial for financial integrity and preventing unauthorized refunds. |
| **CSA STAR** | Yes | low | Authenticated access for refunding campaigns is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /campaign-summary/

| | |
|---|---|
| **Endpoint Path** | /campaign-summary/ |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 1203 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1202, 1202, 1203, 1203, 1215, 1215, 1217, 1217, 1202, 1202, 1215, 1215, 1217, 1217, 1202, 1202, 1215, 1215, 1217, 1217.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for campaign summary ensures access to campaign performance data is restricted. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing campaign summary data is crucial for confidentiality and integrity. |
| **CSA STAR** | Yes | low | Authenticated access to campaign summaries is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /calculate-earnings//

| | |
|---|---|
| **Endpoint Path** | /calculate-earnings// |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 1270 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1269, 1269, 1270, 1270, 1283, 1283, 1285, 1285, 1269, 1269, 1283, 1283, 1285, 1285, 1269, 1269, 1283, 1283, 1285, 1285.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for calculating earnings, with creator_id, ensures creators can only view their own earnings. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing earnings calculation is crucial for confidentiality and integrity. |
| **CSA STAR** | Yes | low | Authenticated access to earnings calculation is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /bulk-distribute

| | |
|---|---|
| **Endpoint Path** | /bulk-distribute |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 1364 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1363, 1363, 1364, 1364, 1384, 1384, 1389, 1389, 1363, 1363, 1384, 1384, 1389, 1389, 1363, 1363, 1384, 1384, 1389, 1389.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for bulk distribution ensures only authorized users can initiate large-scale financial transactions. |
| **ISO/IEC 27001** | Yes | low | Authentication for bulk fund distribution is crucial for preventing fraud and maintaining financial integrity. |
| **CSA STAR** | Yes | low | Authenticated access for bulk distribution is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /request-refund

| | |
|---|---|
| **Endpoint Path** | /request-refund |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 1535 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1534, 1534, 1535, 1535, 1548, 1548, 1553, 1553, 1534, 1534, 1548, 1548, 1553, 1553, 1534, 1534, 1548, 1548, 1553, 1553.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for refund requests ensures only authorized users can initiate refund processes. |
| **ISO/IEC 27001** | Yes | low | Authentication for refund requests is crucial for financial integrity and preventing fraudulent claims. |
| **CSA STAR** | Yes | low | Authenticated access for refund requests is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /refund-requests

| | |
|---|---|
| **Endpoint Path** | /refund-requests |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 1629 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1628, 1628, 1629, 1629, 1636, 1636, 1641, 1641, 1628, 1628, 1636, 1636, 1641, 1641, 1628, 1628, 1636, 1636, 1641, 1641.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for refund requests ensures users can only view their own refund statuses. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing refund request status is crucial for confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access to refund request statuses is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /admin/approve-refund

| | |
|---|---|
| **Endpoint Path** | /admin/approve-refund |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 1696 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1695, 1695, 1696, 1696, 1709, 1709, 1714, 1714, 1695, 1695, 1709, 1709, 1714, 1714, 1695, 1695, 1709, 1709, 1714, 1714.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for approving refunds ensures only authorized administrators can process refunds. |
| **ISO/IEC 27001** | Yes | low | Authentication for administrative refund approval is crucial for financial integrity and auditability. |
| **CSA STAR** | Yes | low | Authenticated administrative access for approving refunds is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /admin/reject-refund

| | |
|---|---|
| **Endpoint Path** | /admin/reject-refund |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 1810 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1809, 1809, 1810, 1810, 1821, 1821, 1826, 1826, 1809, 1809, 1821, 1821, 1826, 1826, 1809, 1809, 1821, 1821, 1826, 1826.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for rejecting refunds ensures only authorized administrators can manage refund decisions. |
| **ISO/IEC 27001** | Yes | low | Authentication for administrative refund rejection is crucial for financial integrity and auditability. |
| **CSA STAR** | Yes | low | Authenticated administrative access for rejecting refunds is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /refund-status/

| | |
|---|---|
| **Endpoint Path** | /refund-status/ |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 1871 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1870, 1870, 1871, 1871, 1878, 1878, 1880, 1880, 1870, 1870, 1878, 1878, 1880, 1880, 1870, 1870, 1878, 1878, 1880, 1880.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for refund status ensures users can only view the status of their own refund requests. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing refund status is crucial for confidentiality. |
| **CSA STAR** | Yes | low | Authenticated access to refund status is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /admin/refund-audit-trail

| | |
|---|---|
| **Endpoint Path** | /admin/refund-audit-trail |
| **HTTP Methods** | GET |
| **Source Location** | ..\routes\payments.py - 1940 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 1939, 1939, 1940, 1940, 1947, 1947, 1939, 1939, 1947, 1947, 1939, 1939, 1947, 1947.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated GET request for refund audit trail ensures only authorized administrators can access audit logs. |
| **ISO/IEC 27001** | Yes | low | Authentication for accessing audit trails is crucial for integrity and accountability. |
| **CSA STAR** | Yes | low | Authenticated administrative access to audit trails is appropriate. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## Endpoint Path /creator/revert-withdrawal

| | |
|---|---|
| **Endpoint Path** | /creator/revert-withdrawal |
| **HTTP Methods** | POST |
| **Source Location** | ..\routes\payments.py - 2019 |
| **Authentication Required** | Yes (token) |
| **Risk Severity** | Low |
| **CVSS Score** | N/A |

### Authentication Analysis

This endpoint enforces authentication using a token-based mechanism. Authentication checks were detected at the following code locations: 2018, 2018, 2019, 2019, 2024, 2024, 2029, 2029, 2018, 2018, 2024, 2024, 2029, 2029, 2018, 2018, 2024, 2024, 2029, 2029.

### Request Analysis

**Content Type:** unknown

No request fields were identified for this endpoint.

### Response Analysis

**Content Type:** unknown

**Status Codes:**

**Contains Sensitive Data:** No

### Identified Security Risks

No direct security risks were identified for this endpoint.

### Compliance Impact

| Regulation | Applicable | Risk Level | Reason |
|---|---|---|---|
| **SOC 2** | Yes | low | Authenticated POST request for reverting withdrawals ensures only authorized actions can be taken on financial transactions. |
| **ISO/IEC 27001** | Yes | low | Authentication for reverting withdrawals is crucial for financial integrity and auditability. |
| **CSA STAR** | Yes | low | Authenticated access for reverting withdrawals is a necessary control. |

### Security Assessment Notes

**CVSS Vector:**

**References**

## METRICS SUMMARY

**Total Findings:** 4

| Severity | Count |
|---|---|
| CRITICAL | 1 |
| HIGH | 1 |
| MEDIUM | 1 |
| LOW | 1 |

# BUSINESS RISK ADVICE

Based on a comprehensive security analysis, here are the prioritized recommendations to enhance the security posture and mitigate identified risks.

## Eliminate Hardcoded Secrets and Disable Debug Mode (Priority: Immediate)

**Description:** Remove the hardcoded JWT token from `tasks.py` and ensure Flask debug mode (`debug=True`) is never enabled in production environments. These vulnerabilities provide critical attack vectors.

**Why it Matters:** Direct exposure of sensitive credentials allows unauthorized access, bypassing authentication and potentially leading to full system compromise (CVSS 9.8). Running Flask in debug mode in production enables arbitrary code execution via the Werkzeug debugger (CVSS 8.8). Both violate fundamental security principles of confidentiality, integrity, and availability, impacting SOC 2, ISO/IEC 27001, and CSA STAR compliance.

**Recommended Actions:**

- Replace the hardcoded `token` variable in `tasks.py` (line 14) with a reference to a secure environment variable or a dedicated secrets management service (e.g., AWS Secrets Manager, HashiCorp Vault).
- Modify `app.py` (line 1872) to ensure `app.run(debug=True, ...)` is strictly conditional to a development environment, or better, use a production-ready WSGI server (e.g., Gunicorn, uWSGI) to serve the Flask application in production.
- Perform a comprehensive codebase scan to identify and remove any other hardcoded sensitive information.
- Implement strict access controls and code review policies to prevent future occurrences of hardcoded secrets or misconfigured debug settings.

**Expected Outcome:** Elimination of critical attack vectors for unauthorized access and arbitrary code execution, significantly improving the application's security posture and resilience against severe breaches.

## Implement Robust Authentication and Rate Limiting for User Management (Priority: Immediate)

**Description:** Address the missing authentication controls on the `/register` and `/login` endpoints, and harden the `/request-password-reset` endpoint against abuse.

**Why it Matters:** The absence of robust authentication for user registration and login (CVSS 9.1 for both) allows for trivial brute-force attacks, account enumeration, and denial-of-service, directly compromising user accounts and system integrity. Inadequate controls on password reset (CVSS 6.5) can be exploited for account takeover. These issues directly violate access control and system integrity requirements for SOC 2, ISO/IEC 27001, and CSA STAR.

**Recommended Actions:**

- **For `/register` (app.py:73)**: Implement CAPTCHA or reCAPTCHA, enforce strong password policies, and introduce email verification workflows. Apply aggressive rate limiting to prevent automated account creation.
- **For `/login` (app.py:120)**: Implement strong, adaptive rate limiting, multi-factor authentication (MFA), and account lockout policies after a configurable number of failed login attempts. Ensure secure session management post-login.
- **For `/request-password-reset` (app.py:224)**: Implement server-side rate limiting on password reset requests. Ensure reset tokens are single-use, cryptographically strong, and have a very short expiration time. Validate all input carefully and provide generic error messages.
- Ensure all user management API responses provide only generic messages to prevent user enumeration (e.g., do not reveal if a username exists).

**Expected Outcome:** Prevention of automated user account abuse, brute-force attacks, and account takeover attempts, enhancing user account security and overall system resilience.

## Enforce Timeouts for All External API Calls (Priority: Short-Term)

**Description:** Configure explicit timeouts for all HTTP requests made to external services, such as the Cashfree API.

**Why it Matters:** Requests to external APIs without a defined timeout can cause the application to hang indefinitely if the external service becomes unresponsive or slow. This leads to worker process exhaustion, increased latency, and ultimately, denial-of-service for the application (CVSS 6.5). This impacts application availability and reliability.

**Recommended Actions:**

- Add a `timeout` parameter to all `requests.post` and `requests.get` calls (e.g., `requests.post(url, ..., timeout=5)` for 5 seconds). This applies to `routes\payments.py` (lines 90, 136, 663) and any other external calls.
- Implement `try-except` blocks to specifically catch `requests.exceptions.Timeout` and `requests.exceptions.ConnectionError` around all external API interactions to handle network and service unresponsiveness gracefully.
- Define and standardize appropriate timeout values across the application based on expected service response times and business requirements.
- Consider implementing retry logic with exponential backoff for transient network errors to improve robustness without indefinite waits.

**Expected Outcome:** Enhanced application resilience against external service failures, prevention of resource starvation, and improved overall stability and responsiveness.

### Standardize Secure Error Handling and Logging (Priority: Mid-Term)

**Description:** Replace verbose internal error messages that include stack traces or specific exception details with generic, user-friendly messages for client-facing responses.

**Why it Matters:** Exposing detailed error messages, such as `str(e)`, directly to clients (e.g., `app.py:117`, `app.py:187`, `routes\payments.py:115`, `routes\payments.py:186`) provides attackers with valuable internal system information. This information (e.g., file paths, database schema, library versions) aids in reconnaissance and can facilitate more sophisticated attacks, violating the principle of least privilege in information disclosure (CVSS 3.1).

**Recommended Actions:**

- Implement a centralized error handling mechanism (e.g., using Flask's `@app.errorhandler` decorator) to intercept and process all unhandled exceptions.
- For all client-facing API responses, replace specific exception details (`str(e)`) with generic, non-revealing error messages (e.g., 'An unexpected error occurred. Please try again later.' or 'Invalid input provided.').
- Ensure that all detailed error information, including full stack traces and exception data, is securely logged server-side to an internal log management system or SIEM, accessible only to authorized personnel.
- Refactor existing broad `try-except Exception as e:` blocks to catch more specific exception types where possible, allowing for more precise handling without compromising security.

**Expected Outcome:** Prevention of sensitive information leakage through error messages, significantly reducing the reconnaissance surface for attackers while retaining necessary diagnostic information for developers and operations teams.

### Integrate Security into the Software Development Lifecycle (SSDLC) (Priority: Long-Term)

**Description:** Establish and enforce a comprehensive Secure Software Development Lifecycle (SSDLC) to proactively identify, mitigate, and prevent security vulnerabilities throughout the development process.

**Why it Matters:** Reactive security measures are less effective and more costly than proactive integration of security throughout the SDLC. A robust SSDLC ensures continuous compliance with standards like SOC 2, ISO/IEC 27001, and CSA STAR, reduces security debt, and fosters a security-first culture, preventing recurring vulnerabilities.

**Recommended Actions:**

- Provide mandatory and recurring security training for all developers covering common vulnerabilities (OWASP Top 10), secure coding practices, and framework-specific security guidelines.
- Integrate automated security testing tools, such as Static Application Security Testing (SAST) and Dependency Scanning, into the CI/CD pipeline to detect vulnerabilities early in the development cycle.
- Implement regular manual code reviews and threat modeling sessions for critical components and new features to identify design-level security flaws.
- Establish a formal vulnerability management program including regular security assessments, penetration testing, and a defined process for tracking and remediating findings.
- Formalize secure configuration management practices for all environments (development, staging, production) to prevent misconfigurations that lead to vulnerabilities.

**Expected Outcome:** A proactive and mature security posture, significant reduction in security vulnerabilities reaching production, lower costs associated with vulnerability remediation, and continuous compliance with relevant security standards and regulations.