# Security Test Report

Scope: `Security Test/Security_Issue_Python_code_unmarked.py`. Deliverables: vulnerability report and corrected code (submitted as `submission/security-fixed-code.zip`).

## Vulnerability Table

| ID | Issue | Severity | Root Cause | CWE/OWASP | Location |
|---|---|---|---|---|---|
| V1 | Hardcoded secrets (API, DB, AWS, SMTP) | High | Secrets embedded in source | CWE-798 / CWE-522 | Top constants, __init__ logging |
| V2 | Secrets logged in plaintext | High | Logging keys/passwords on init/error | CWE-532 | __init__, upload_to_cloud, DB error logs |
| V3 | TLS disabled & warnings suppressed; HTTP webhook | High | `verify=False`, HTTP endpoint | CWE-295 / CWE-319 | Session, call_external_api, process_webhook_data |
| V4 | SQL injection in fetch/delete | High | String formatting user input into SQL | CWE-89 | fetch_user_data, process_webhook_data |
| V5 | Plaintext sensitive data storage | High | Passwords/CC/SSN stored unprotected | CWE-312 / CWE-522 | user_data table schema |
| V6 | Unauthenticated/unsanitized webhook actions | High | No authZ; arbitrary delete; potential SSRF | CWE-306 / CWE-918 | process_webhook_data |
| V7 | Hardcoded S3 creds/bucket; no SSE | Med | Inline keys, no encryption flags | CWE-798 / CWE-311 | upload_to_cloud |
| V8 | Hardcoded SMTP password; logs on failure | Med | Inline credential, logging secret | CWE-798 / CWE-532 | send_notification_email |
| V9 | No timeouts/backoff/rate limit on external calls | Med | Unbounded network calls | CWE-400 / CWE-770 | call_external_api, webhook POST |
| V10 | Global SSL verify disabled & warning suppression | Med | Session verify False, disable_warnings | CWE-295 | __init__ |

## Detailed Findings & Fixes

### V1 Hardcoded secrets

- **Description:** API, DB, AWS, SMTP keys embedded in code.
- **Impact:** Secret leakage, account takeover, data exfiltration.
- **Fix:** Use env vars/secret manager; removed from code/logs.
- **Verification:** Grep shows no secrets; app reads from env.

### V2 Secrets logged

- **Description:** Logging API key, DB password, AWS key on errors.
- **Impact:** Secrets persist in logs/SIEM.
- **Fix:** Removed secret logging; structured logs sans secrets.
- **Verification:** Run flows, inspect logs contain no secrets.

### V3 TLS disabled / HTTP webhook

- **Description:** `verify=False` everywhere; webhook over HTTP.
- **Impact:** MITM, spoofed responses, data leakage.
- **Fix:** Enforce TLS verify, require HTTPS webhook, keep warnings on.
- **Verification:** Requests fail against self-signed; succeed with valid TLS.

### V4 SQL injection

- **Description:** f-string SQL in fetch/delete.
- **Impact:** Data theft or destructive deletes.
- **Fix:** Parameterized queries, type check user_id.
- **Verification:** Pass payload `1 OR 1=1` now rejected; logs warn.

### V5 Plaintext sensitive data

- **Description:** Passwords/credit cards/SSNs stored raw.
- **Impact:** Breach exposes PII and credentials.
- **Fix:** Store password hashes (PBKDF2), last4 only; encrypt at rest via KMS/S3/EBS.
- **Verification:** DB shows hashed/last4; no full values stored.

### V6 Unauthenticated webhook & SSRF risk

- **Description:** No auth/allowlist; arbitrary delete_user; posts to internal HTTP.
- **Impact:** Unauthorized deletion, SSRF into internal network.

- **Fix:** Allowlist actions, type check, require HTTPS endpoint; use auth at gateway.
- **Verification:** Invalid action rejected; endpoint must be HTTPS; deletion only with valid int user_id.

### V7 Hardcoded S3 creds/bucket; no SSE

- **Description:** Inline AWS keys and bucket; no server-side encryption.
- **Impact:** Key leakage; unencrypted objects.
- **Fix:** Use IAM role/default provider; SSE AES256; bucket from env.
- **Verification:** Upload succeeds without inline keys; object shows SSE header.

### V8 SMTP credential handling

- **Description:** Hardcoded SMTP password, logged on failure.
- **Impact:** Credential exposure.
- **Fix:** Env-based secret, no logging of password, TLS enforced.
- **Verification:** Failure logs exclude secrets; auth pulls from env.

### V9 No timeouts/backoff

- **Description:** External calls without timeouts or retries.
- **Impact:** Thread/conn exhaustion, DoS.
- **Fix:** Added timeouts and Retry adapter; rate-limit at gateway.
- **Verification:** Simulate slow API: request times out; retry/backoff observed.

### V10 Global SSL verify disabled

- **Description:** Session verify False + warnings suppressed.
- **Impact:** Accepts forged certs.
- **Fix:** verify=True by default; no suppression.
- **Verification:** Self-signed endpoint now fails.

## Fixed Code

Corrected source is packaged in `submission/security-fixed-code.zip` (file: `security_fixed_code.py`). Key changes:

- Secrets via env vars; no secret logging.
- TLS verification enforced; HTTPS-only webhook; retries + timeouts.
- Parameterized SQL + type validation; hashed passwords and last4 storage.
- IAM-based S3 upload with SSE; SMTP creds from env, no secret logs.

- Allowlisted webhook actions; SSRF/HTTP blocked.

## Verification Plan

- Unit tests: injection payloads rejected; webhook invalid action returns error.
- Integration: upload file -> object has `ServerSideEncryption`; API call fails on bad cert.
- Static scan: no hardcoded secrets; no `verify=False` usages.
- DB inspection: hashed passwords, last4 only.