CONFIDENTIAL

Balhence

Vulnerability Assessment & Penetration Testing Report

Client: [REDACTED IN SYNTHETIC SAMPLE]
Generated: 2026-08-21
Classification: Confidential

SYNTHETIC SAMPLE - For demonstration purposes only.
It does not describe a real client, engagement, target, or finding.

Contents

1. Executive Summary

Balhence was engaged by [REDACTED IN SYNTHETIC SAMPLE] to perform a comprehensive Vulnerability Assessment and Penetration Testing (VAPT) of their customer-facing web application hosted at [REDACTED IN SYNTHETIC SAMPLE]. In this synthetic scenario, testing occurred during an illustrative five-business-day window. No real organization or target is represented.

The assessment identified **14 unique vulnerabilities** across the application, including **1 Critical**, **4 High**, **4 Medium**, and **5 Low** severity findings. The critical SQL-injection finding demonstrates server-side query control; the high-severity broken-object-authorization finding demonstrates cross-account access in this synthetic scenario. Impact was validated with bounded proofs and controlled records; no bulk extraction or unrelated account access occurred.

14	1	4	4	5
Total Vulnerabilities	Critical	High	Medium	Low

Overall Risk Rating: HIGH

Immediate remediation of the 1 critical and 4 high-severity findings is strongly recommended before the application handles any additional customer data or payment transactions.

2. Scope of Assessment

Parameter	Details
Client	[REDACTED IN SYNTHETIC SAMPLE]
Application	[REDACTED IN SYNTHETIC SAMPLE]
Type	Web Application VAPT (Black Box + Grey Box)
Environment	Production (with rate-limit safeguards)
Testing Period	Illustrative five-business-day window
Assessor	Balhence assessor (synthetic sample)
Methods	OWASP WSTG 4.2, OWASP ASVS 5.0.0, PTES; tailored to scope
Risk Scoring	CVSS v3.1 plus contextual business-impact analysis (sample convention)
Evidence Context	Control cross-references require an exact client-supplied requirement
Report Version	sample-2026-08-21

Accounts used for reproduction

All credentials below are fictional, controlled sample data under the reserved **example.test** domain. They are included to demonstrate the evidence expected in a real report and cannot authenticate to any live system. Live bearer tokens, session cookies, API keys, MFA recovery material, and unrelated secrets are intentionally excluded.

Role	Login / test password	Account identifiers	Controlled object IDs
Standard user - Tenant A	user.a@tenant-a.example.test **SampleOnly!TenantA-2026**	user: usr_test_a account: acct_test_a team: team_test_a tenant: ten_test_a	invoice: inv_test_a document: doc_test_a
Standard user - Tenant B	user.b@tenant-b.example.test **SampleOnly!TenantB-2026**	user: usr_test_b account: acct_test_b team: team_test_b tenant: ten_test_b	invoice: inv_test_b document: doc_test_b
Tenant admin - Tenant A	admin.a@tenant-a.example.test **SampleOnly!AdminA-2026**	user: usr_admin_a account: acct_admin_a team: team_test_a tenant: ten_test_a	invoice: inv_test_a document: doc_test_a

Authorization and Limitations

Testing in this synthetic scenario was limited to the listed application, approved tester IPs, controlled accounts, and agreed hours. Denial of service, destructive changes, persistence, social engineering, third-party systems, and bulk access to production data were excluded. Newly discovered assets required written change approval. These limits materially shape coverage and residual risk.

3. Methodology

Our testing methodology follows industry-standard frameworks including OWASP Testing Guide v4.2, PTES (Penetration Testing Execution Standard), and NIST SP 800-115. The assessment was conducted in the following phases:

Phase 1: Reconnaissance

Passive and active information gathering - DNS enumeration, technology fingerprinting, endpoint discovery, and attack surface mapping.

Phase 2: Tooling-Assisted Coverage

Focused tools supported route, service, and weakness discovery within the agreed rate limits. Automation did not replace manual analysis.

Phase 3: Manual Validation

A tester manually validated every reported finding and assessed authorization, session, tenant, and business-logic paths that scanners cannot reliably decide.

Phase 4: Controlled Impact Analysis

Confirmed weaknesses were demonstrated using client-controlled accounts and the minimum evidence needed. No persistence, destructive action, denial of service, or bulk data extraction was performed.

Phase 5: Reporting and QA

Findings were reviewed for reproducibility, duplicates, severity rationale, sensitive-data minimization, root-cause remediation, and explicit coverage limitations.

Tools Used:

Representative tooling: intercepting proxy, browser developer tools, API client, port/service enumeration, focused templates, and small review scripts. Actual tools vary by authorized scope; tool output alone is not reported as a verified finding.

4. Findings Summary

#	Vulnerability	Severity	CVSS	Status
1	SQL Injection - Login API	Critical	9.8	Open
2	Broken Access Control - IDOR on /api/users/{id}	High	8.1	Open
3	Stored XSS in User Profile Bio	High	7.6	Open
4	IDOR - [REDACTED IN SYNTHETIC SAMPLE]	High	7.5	Open
5	JWT Token Not Expiring on Logout	High	7.4	Open
6	Missing Rate Limiting on Login Endpoint	Medium	6.5	Open
7	Sensitive Data in Error Responses	Medium	5.3	Open
8	CORS Misconfiguration on [REDACTED IN SYNTHETIC SAMPLE]	Medium	5.0	Open
9	Session Fixation via URL Parameter	Medium	4.8	Open
10	Missing Security Headers (CSP, X-Frame-Options)	Low	3.7	Open
11	Verbose Server Banner Disclosure	Low	3.1	Open
12	[REDACTED IN SYNTHETIC SAMPLE]	Low	2.6	Open
13	Autocomplete Enabled on Sensitive Fields	Low	2.1	Open
14	Cookie Without Secure/HttpOnly Flags	Low	2.0	Open

5. Detailed Findings

5.1 SQL Injection in Login API

Critical

CVSS: 9.8 (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H)

Category: A03:2021 - Injection **Affected:** [REDACTED IN SYNTHETIC SAMPLE]

Description

The redacted login endpoint is vulnerable to SQL Injection via the **email** parameter. The application directly concatenates user input into a SQL query without parameterized statements. Controlled boolean and time-based responses confirmed that input altered query behavior. Broader confidentiality, integrity, and availability impact is plausible but was not exercised in this sample scenario.

Impact

An unauthenticated attacker could potentially read or modify data reachable by the application database account. The assessment confirmed query-control impact without extracting customer records. Potential blast radius depends on database privileges, segmentation, and the affected query path and should be validated during remediation.

Steps to Reproduce

1. Navigate to [REDACTED IN SYNTHETIC SAMPLE]
2. Intercept the login POST request using Burp Suite
3. Send paired control and boolean-condition values in the email parameter using the approved test account
4. Observe a repeatable response difference attributable to the injected condition
5. Confirm with a bounded time-delay expression; do not enumerate schema or extract records

[Screenshot / Evidence Redacted in Sample Report]

Remediation

Priority: Immediate

1. Use parameterized queries / prepared statements for all database interactions
2. Implement input validation with strict type checking on the email field
3. Deploy a Web Application Firewall (WAF) as an additional defense layer
4. Restrict the application database account to least privilege
5. Review all query-construction paths and add regression tests for injection

5.2 Broken Access Control - IDOR

High

CVSS: 8.1 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N)

Category: A01:2021 - Broken Access Control **Affected:** [REDACTED IN SYNTHETIC SAMPLE]

Description

The redacted user profile API endpoint does not enforce proper authorization checks. Changing the user ID allowed one client-controlled low-privilege account to read and update a second client-controlled account. Sequential identifiers make discovery easier but are not the root cause; missing server-side object authorization is.

Impact

A low-privilege user may access or alter another user's records. The controlled two-account test confirmed cross-account confidentiality and integrity impact. Platform-wide blast radius is plausible if the same missing check applies consistently, but bulk enumeration was not performed.

Steps to Reproduce

1. Authenticate as controlled Account A and record its own profile request
2. Replace the object identifier with controlled Account B's identifier
3. Observe Account B's seeded test profile data in the response
4. Update Account B's harmless test-only display label and observe success
5. Revert the display label and verify that no other record was accessed or changed

[Screenshot / Evidence Redacted in Sample Report]

Remediation

Priority: Immediate

1. Implement server-side authorization checks - verify the authenticated user owns the requested resource
2. Deny by default and centralize object/function authorization where practical
3. Add negative authorization tests across roles, tenants, and object states
4. Treat non-sequential IDs only as defense-in-depth, never as an authorization control
5. Log and alert on anomalous cross-object access patterns

5.3 Stored XSS in User Profile

High

CVSS: 7.6 (CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:C/C:H/I:L/A:N)

Category: A03:2021 - Injection (XSS) **Affected:** [REDACTED IN SYNTHETIC SAMPLE]

Description

The user profile bio field does not sanitize or encode user input before rendering it on other users' screens. An attacker can inject arbitrary JavaScript that executes in the context of another user who views the attacker's profile. The assessment confirmed controlled JavaScript execution; follow-on session or action impact was not exercised.

Impact

Arbitrary script execution in the application origin can potentially read non-HttpOnly data, perform same-origin actions as the viewing user, or alter trusted content. Realistic impact depends on cookie flags, CSP, exposed actions, and which users can be induced to view the content.

Steps to Reproduce

1. Navigate to Profile > Edit Bio
2. Enter a benign execution marker: `[REDACTED IN SYNTHETIC SAMPLE]`
3. Save the profile
4. View the profile using controlled Account B
5. Observe the marker in the browser console; no token or data is transmitted

[Screenshot / Evidence Redacted in Sample Report]

Remediation

Priority: Within 48 hours

1. Implement output encoding (HTML entity encoding) on all user-generated content
2. Deploy Content Security Policy (CSP) headers to prevent inline script execution
3. Use a proven sanitization library (e.g., DOMPurify) for any rich-text fields
4. Set HttpOnly and Secure flags on all session cookies

Detailed entries 5.4 through 5.14 follow the same format. They have been omitted from this sample report for brevity. They are summarized above but omitted from this synthetic sample.

6. Risk Rating Matrix

All findings are rated using CVSS v3.1 (Common Vulnerability Scoring System) and mapped to the following severity classification:

Severity	CVSS Range	Count	Description
Critical	9.0 - 10.0	**1**	Severe modeled technical impact; prioritize immediate containment and remediation.
High	7.0 - 8.9	**4**	Significant modeled technical impact; prioritize based on exposure and business context.
Medium	4.0 - 6.9	**4**	Material weakness with more limited impact or additional exploitation conditions.
Low	0.1 - 3.9	**5**	Limited direct impact; address in a risk-based remediation plan.

7. Remediation Priority

Immediate (0-48 hours): SQL Injection (#1), Broken Access Control (#2)

Within 1 week: Stored XSS (#3), IDOR on Invoices (#4), JWT Expiry (#5)

Within 30 days: Rate Limiting (#6), Error Disclosure (#7), CORS (#8), Session Fixation (#9)

Next release cycle: Security Headers (#10), Server Banner (#11), Directory Listing (#12), Autocomplete (#13), Cookie Flags (#14)

8. Re-test Terms

In this synthetic example, the proposal includes **one re-test round requested within 30 days** for the finding IDs listed in the signed scope. The re-test verifies those findings against the named environment/build; it is not a new full assessment. Results are recorded as **Fixed**, **Partially Fixed**, **Not Fixed**, **Cannot Verify**, or **Risk Accepted**.

Re-test Coverage	Finding IDs 01-05 in this illustrative SOW
Request Window	Within 30 days of initial report delivery
Environment	Named build and target supplied as remediation-ready
Rounds	One; redesigns and new scope require a change
Deliverable	Updated finding statuses with residual limitations

9. Appendix

A. References

- OWASP Top 10 (2021) - https://owasp.org/Top10/
- CVSS v3.1 Calculator - https://www.first.org/cvss/calculator/3-1
- NIST SP 800-115 - Technical Guide to Information Security Testing
- PTES - Penetration Testing Execution Standard
- OWASP ASVS - https://owasp.org/www-project-application-security-verification-standard/
- CWE - https://cwe.mitre.org/

B. Disclaimer

This synthetic report demonstrates structure only and does not describe a real engagement. A real assessment is a point-in-time, scope-limited review and cannot guarantee discovery of every weakness, future security, or compliance. Conclusions must be read with the exact target inventory, access, assumptions, exclusions, and testing constraints stated in the signed engagement.