ZENITHCRYPT OFFENSIVE SECURITY TEAM

CONFIDENTIAL

PENETRATION TEST REPORT

[CLIENT_NAME] Collaboration – Invoice Management Platform

Engagement Parameter	Details
Client Organization	[CLIENT_NAME] Incorporated
Target Application	[CLIENT_NAME] Collaboration - Invoice Management Platform
Testing Period	[Start Date] - [End Date]
Testing Methodology	OWASP Testing Guide v4, OWASP ASVS Level 2
Risk Scoring Standard	CVSS v4.0 Base Metrics (0.1–3.9 Low, 4.0–6.9 Medium, 7.0–8.9 High, 9.0–10.0 Critical)
Test Team	ZenithCrypt Offensive Security Team
Document Classification	**CONFIDENTIAL**

Table of Contents

1. Executive Summary

2. Scope of Assessment

3. Summary of Results

4. Technical Attack Narrative

4.1 CRITICAL: Mass Assignment via Privilege Escalation Parameter (ZC-01)

4.2 HIGH: Unauthenticated SignalR WebSocket Message Injection (ZC-02)

4.3 HIGH: Missing Business Logic Validation on Financial Transactions (ZC-03)

4.4 HIGH: Stored Cross-Site Scripting (XSS) – Multiple Vectors (ZC-04)

4.5 HIGH: Missing Multi-Factor Authentication & CAPTCHA (ZC-05)

4.6 HIGH: Outdated & Vulnerable Backend Components (ZC-06)

4.7 HIGH: Insecure Session Cookie Attributes (ZC-07)

4.8 HIGH: CORS Misconfiguration (ZC-08)

4.9 MEDIUM: Federation Metadata Exposure (ZC-09)

4.10 MEDIUM: Weak SSL/TLS Protocols & Deprecated Cipher Suites (ZC-10)

4.11 MEDIUM: Lack of API & Login Rate Limiting (ZC-11)

4.12 MEDIUM: Web Server & Infrastructure Fingerprinting (ZC-12)

4.13 MEDIUM: Missing HTTP Security Headers (ZC-13)

4.14 MEDIUM: Outdated Frontend Library – Bootstrap (ZC-14)

4.15 LOW: Missing Subresource Integrity – SRI (ZC-15)

5. Appendix A: Vulnerability Detail and Mitigation

1. Executive Summary

A comprehensive penetration test was conducted on the **[CLIENT_NAME] Collaboration Platform** between **[Start Date]** and **[End Date]** to determine its exposure to targeted attacks. All activities simulated a malicious actor engaged in a targeted attack with the primary objectives of identifying whether a remote attacker could penetrate [CLIENT_NAME]'s defenses and determining the impact of a security breach on the confidentiality of private data and internal infrastructure.

Efforts focused on identifying security weaknesses that could allow unauthorized access to organizational data. Attacks were conducted with the level of access a general Internet user would have, following the methodology outlined in the **OWASP Testing Guide v4** and **OWASP ASVS Level 2** under controlled conditions.

> **Overall Assessment Posture: CRITICAL RISK (CVSS v4.0 Calibrated)**
>
> While defense-in-depth and configuration findings fall within the High, Medium, and Low tiers under CVSS v4.0 (1 Critical, 7 High, 6 Medium, 1 Low), an unauthenticated Mass Assignment vulnerability (`IsAdmin=true` , CVSS v4.0: 9.3) enables an external attacker to achieve complete administrative takeover, chain into financial invoice manipulation up to $270M+, and exfiltrate sensitive user directory records.

2. Scope of Assessment

Parameter	Details
Organization	[CLIENT_NAME] Incorporated
Application Name	[CLIENT_NAME] Collaboration - Invoice Management Platform
Audit Type	Application Security - Grey Box Test
Environment	Staging / Sandbox
Target URLs	`https://[TARGET_DOMAIN]`
SSO Provider	`https://[SSO_PROVIDER_DOMAIN]`
Admin Panel	`/APSettings/Index`
Testing Period	[Start Date] - [End Date]
Testing Methodology	OWASP Testing Guide v4, OWASP ASVS Level 2
Test Team	ZenithCrypt
Classification	CONFIDENTIAL

3. Summary of Results

Initial reconnaissance of the [CLIENT_NAME] application resulted in the discovery of multiple critical security weaknesses that could allow a remote attacker to achieve complete system compromise. The assessment began with an examination of the SAML authentication infrastructure, which revealed that the ADFS federation metadata was publicly accessible, exposing signing and encryption certificates, internal endpoints, and claim types.

A closer examination of the SignalR WebSocket implementation revealed that the `/chatHub` endpoint accepts connections without any authentication, allowing arbitrary message injection and data exfiltration. Furthermore, SAML session hijacking was successfully demonstrated by capturing the `MSISAuth` cookie from the ADFS login flow and replaying it to gain full administrative access to the [CLIENT_NAME] application.

A Mass Assignment vulnerability was identified in the admin panel, where the `IsAdmin=true` parameter could be injected to bypass authorization controls. Error messages were found to disclose internal system information including Node names, Activity IDs, and stack traces. In total, the assessment revealed **8 user accounts exposed**, including email addresses and role assignments, demonstrating that existing security controls are inadequate to prevent a cascading failure of the application's security posture.

4. Attack Narrative

4.1 CRITICAL: Federation Metadata Exposure

OWASP Reference: OTG-INFO-005 / A01:2021 – Broken Access Control | **CWE:** CWE-200 | **CVSS:** 9.1

4.1.1 Initial Discovery - Metadata Exposure

During the reconnaissance phase, we identified that the ADFS federation metadata was publicly accessible without any authentication. This exposes the complete SAML configuration including certificates, endpoints, and claim types.

> *[Insert Screenshot: Figure 1 – Federation Metadata Access]* | *[Insert Screenshot: Figure 2 – Exposed Metadata XML]*

The ADFS Federation Metadata XML rendered in a browser exposes the complete SAML configuration. The document tree reveals:

- **Encryption Certificate Exposure (Impact: Critical):** Used to encrypt SAML assertions; exposure allows attackers to analyze weaknesses and potentially decrypt intercepted assertions.
- **Multiple SAML Endpoints Exposed (Impact: Critical):** Exposes the complete attack surface mapping of internal trust services.

4.1 CRITICAL: Mass Assignment via Privilege Escalation Parameter (ZC-01)

OWASP Reference: OTG-INPVAL-020 / A01:2021 – Broken Access Control | CWE-915 | 9.3 (Critical)

CWE:CVSS v4.0:

Vector: `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`

4.1.1 Initial Discovery - Parameter Injection

During the reconnaissance phase, we performed a directory scan to identify the administrative interface. The `/APSettings/` endpoint was discovered, indicating an admin panel.

[Insert Screenshot: Figure 1 – Directory Enumeration Discovering /APSettings/]

4.1.2 Mass Assignment Exploitation - Parameter Injection

The application accepts the `IsAdmin=true` parameter in the URL and user management endpoints (,), which bypasses authorization controls and grants full administrative access.`/APSettings/Index` `/api/InvoiceBillsAPI/AddUser`

[Insert Screenshot: Figure 2 – Mass Assignment Exploitation Request (IsAdmin=true)]

4.2 HIGH: Unauthenticated SignalR WebSocket Message Injection (ZC-02)

OWASP Reference: OTG-CLIENT-010 / A01:2021 – Broken Access Control | **CWE:** CWE-306 | **CVSS v4.0:** 8.7 (High)

Vector: `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:L/VI:H/VA:N/SC:N/SI:L/SA:N`

4.2.1 Initial Discovery - SignalR Hub Identification

During the assessment, we identified that the application uses SignalR for real-time communication. The SignalR hub endpoint `/chatHub` was found to accept WebSocket connections without any authentication or authorization checks, enabling arbitrary message injection and user impersonation.

[Insert Screenshot: Figure 3 – Unauthenticated SignalR WebSocket Connection to /chatHub]

4.3 HIGH: Missing Business Logic Validation on Financial Transactions (ZC-03)

OWASP Reference: A04:2021 – Insecure Design | **CWE:** CWE-840 | **CVSS v4.0:** 8.6 (High) | **Financial Exposure:** Up to $270M+ per invoice

Vector: `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:H/SA:N`

4.3.1 Initial Discovery - Invoice List Display

During the assessment, we identified High-severity Business Logic Validation failures. The application accepts invoice amounts without server-side validation checks, allowing negative amounts, excessive values, and inconsistent line-item distributions (e.g., $270,000,000.99 gross amount vs. $27.99 distribution amount) to be stored and processed, corrupting downstream accounting ledgers.

[Insert Screenshot: Figure 4 – Invoice List Displaying Excessive Amount]

API Field	Observed Value	Validation Issue
`totalAmount`	`-270000000.99`	Negative amount accepted without server-side rejection
`grossAmount`	`-270000000.99`	No upper/lower range validation on invoice amount
`tax`	`0.00`	Inconsistent – Tax calculation bypassed (expected ~$21.6M)

4.4 HIGH: Stored Cross-Site Scripting (XSS) – Multiple Vectors (ZC-04)

OWASP Reference: OTG-INPVAL-002 / A03:2021 – Injection | **CWE:** CWE-79 | **CVSS v4.0:** 8.5 (High)

Vector: `CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:N/SC:L/SI:L/SA:N`

4.4.1 Initial Discovery - Stored XSS in Invoice Payment Notes

During the assessment, we identified Stored XSS vulnerabilities across multiple components. Payloads embedded into invoice fields are stored in the database and passively executed when rendered in the application's UI by finance staff or administrators.

[Insert Screenshot: Figure 5 – Payment Notes XSS Injection Request] | *[Insert Screenshot: Figure 6 – Payment Notes XSS Rendered in UI]*

Key Finding: The application stored and rendered the XSS payload (`<script>alert('XSS')</script>`) without any sanitization, HTML encoding, or input validation.

4.5 – 4.8 Additional High-Severity Findings (CVSS v4.0: 7.5 – 8.2)

- **4.5 HIGH – Missing Multi-Factor Authentication & CAPTCHA (ZC-05 | CVSS v4.0: 8.2):** User accounts and authentication endpoints rely solely on single-factor credentials without MFA or CAPTCHA protections ().

- `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`
- **4.6 HIGH – Outdated & Vulnerable Backend Components (ZC-06 | CVSS v4.0: 8.2):** Underlying server frameworks and application libraries run unpatched versions with publicly disclosed vulnerabilities (`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`).
- **4.7 HIGH – Insecure Session Cookie Attributes (ZC-07 | CVSS v4.0: 7.7):** Session cookies (including SAML `MSISAuth` tokens) lack hardened `HttpOnly`, `Secure`, and `SameSite` attributes, enabling token theft when chained with Stored XSS (`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`).
- **4.8 HIGH – CORS Misconfiguration (ZC-08 | CVSS v4.0: 7.5):** Dynamic reflection of arbitrary `Origin` headers with credentials enabled allows unauthorized external sites to read authenticated API responses (`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:L/VA:N/SC:N/SI:N/SA:N`).

4.9 MEDIUM: Federation Metadata Exposure (ZC-09)

OWASP Reference: OTG-INFO-005 / A01:2021 – Broken Access Control | **CWE:** CWE-200 | **CVSS v4.0:** 6.9 (Medium)

Vector: `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`

4.9.1 Initial Discovery - Metadata Exposure

During reconnaissance, we identified that the ADFS federation metadata was publicly accessible without authentication, exposing public X.509 certificates, internal SAML endpoints, and claim schemas. While private keys are not exposed, this provides complete identity provider attack-surface mapping.

[Insert Screenshot: Figure 7 – Federation Metadata Access] | [Insert Screenshot: Figure 8 – Exposed Metadata XML]

```
<KeyDescriptor use="encryption">
  <X509Certificate>[REDACTED_CERTIFICATE_STRING]</X509Certificate>
</KeyDescriptor>
<fed:TargetScopes>
  <Address>https://[SSO_PROVIDER_DOMAIN]/adfs/services/trust/.../issuedtoken...
</Address>
</fed:TargetScopes>
```

4.10 MEDIUM: Weak SSL/TLS Protocols & Cipher Suites (ZC-10)

OWASP Reference: OTG-CRYPST-001 / A02:2021 – Cryptographic Failures | **CVSS v4.0:** 6.8 (Medium)

Vector: `CVSS:4.0/AV:N/AC:H/AT:P/PR:N/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`

4.10.1 Initial Discovery - Weak SSL/TLS Protocols

[Insert Screenshot: Figure 9 – SSL/TLS Protocol & Cipher Suite Configuration]

Key Finding: The server supports weak cipher suites including CBC mode ciphers, static RSA key exchange (lacking Perfect Forward Secrecy), and the deprecated 3DES cipher (112-bit effective security), requiring active network interception conditions to exploit.

4.11 – 4.12 MEDIUM: Rate Limiting & Infrastructure Fingerprinting (ZC-11, ZC-12)

- **4.11 MEDIUM – Lack of API & Login Rate Limiting (ZC-11 | CVSS v4.0: 6.3):** Authentication and user lookup endpoints lack throttling, enabling automated account enumeration across 8 active accounts (`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:L/VI:N/VA:L/SC:N/SI:N/SA:N`).

- **4.12 MEDIUM – Web Server & Infrastructure Fingerprinting (ZC-12 | CVSS v4.0: 5.3):** HTTP response headers (`Server: awselb/2.0`) and unhandled error messages disclose internal node names, Activity IDs, ASP.NET/IIS versions, and full stack traces (`CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`).

4.13 MEDIUM: Missing HTTP Security Headers (ZC-13)

OWASP Reference: OTG-CONFIG-014 / A05:2021 – Security Misconfiguration | **CVSS v4.0:** 5.1 (Medium)

Vector: `CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:A/VC:L/VI:L/VA:N/SC:N/SI:N/SA:N`

HTTP responses from the application and SSO provider omit key defense-in-depth browser headers, increasing susceptibility to clickjacking, MIME-sniffing, and Stored XSS impact.

Header	Present	Security Status
`Strict-Transport-Security`	Yes	Secure
`Content-Security-Policy`	No	**Medium - Missing**
`X-Frame-Options`	No	**Medium - Missing**
`X-Content-Type-Options`	No	**Low - Missing**

4.14 – 4.15 Frontend Library & Subresource Integrity Findings (ZC-14, ZC-15)

- **4.14 MEDIUM – Outdated Frontend Library: Bootstrap (ZC-14 | CVSS v4.0: 5.1):** The frontend loads an end-of-life Bootstrap version affected by known DOM-based XSS gadgets when combined with attacker-controlled HTML attributes (`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:A/VC:L/VI:L/VA:N/SC:N/SI:N/SA:N`).

- **4.15 LOW – Missing Subresource Integrity: SRI (ZC-15 | CVSS v4.0: 2.1):** External third-party scripts and stylesheets lack cryptographic `integrity` hashes, requiring upstream CDN compromise prior to exploitation (`CVSS:4.0/AV:N/AC:H/AT:P/PR:N/UI:P/VC:L/VI:L/VA:N/SC:N/SI:N/SA:N`).

5. Appendix A: Vulnerability Detail and Mitigation

A.1 CRITICAL: Mass Assignment (IsAdmin=true) [ZC-01]

Attribute	Details
OWASP 2021 / CWE	A01:2021 Broken Access Control | CWE-915
CVSS v4.0 Score	**9.3 (Critical)** – `CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`
Affected Component	`/APSettings/Index`, `/api/InvoiceBillsAPI/AddUser`

Description: The application accepts the `IsAdmin=true` parameter in the URL, bypassing authorization controls and granting administrative access.

Remediation: Remove mass assignment of security-sensitive fields; use explicit Data Transfer Object (DTO) mapping for user roles; implement strict server-side authorization checks.

A.2 HIGH: Unauthenticated SignalR WebSocket Message Injection [ZC-02]

CVSS v4.0: **8.7 (High)** | **Description:** The SignalR hub `/chatHub` accepts WebSocket connections without authentication, allowing attackers to impersonate users and broadcast spoofed messages. **Remediation:** Decorate the SignalR hub with `[Authorize]` and validate user identity claims on every hub method.

A.3 HIGH: Missing Business Logic Validation [ZC-03]

CVSS v4.0: **8.6 (High)** | **Description:** Invoice endpoints accept negative or extreme amounts ($270,000,000.99 gross vs. $27.99 distribution) without server-side validation. **Remediation:** Enforce positive amount bounds, reconcile line-item sums and tax calculations server-side, and enforce multi-tier approval workflows.

A.4 HIGH: Stored Cross-Site Scripting (XSS) [ZC-04]

CVSS v4.0: **8.5 (High)** | **Description:** Payloads embedded in invoice payment notes execute persistently in victim browsers. **Remediation:** Apply context-aware HTML output encoding, sanitize rich-text inputs, and deploy a strict `Content-Security-Policy`.

A.5 HIGH: Missing MFA, Cookie Hardening & CORS Controls [ZC-05, ZC-07, ZC-08]

CVSS v4.0: **7.5 – 8.2 (High)** | **Remediation:** Enforce TOTP/FIDO2 MFA across all user roles; configure `HttpOnly; Secure; SameSite=Strict` on `MSISAuth` and session cookies; replace wildcard CORS reflection with an explicit domain allowlist.

A.6 MEDIUM & LOW: Federation Metadata, TLS, Security Headers & SRI [ZC-09 – ZC-15]

CVSS v4.0: **5.1 – 6.9 (Medium)** & **2.1 (Low)** | **Remediation:** Restrict `FederationMetadata.xml` access if external discovery is unneeded; enforce TLS 1.2/1.3 with PFS ciphers on AWS ALB; suppress verbose stack traces; inject `Content-Security-Policy`, `X-Frame-Options: DENY`, and `X-Content-Type-Options: nosniff` headers; upgrade Bootstrap and add SHA-384 SRI hashes.

6. Vulnerability Risk Summary (CVSS v4.0)

Finding	Severity	CVSS v4.0	CVSS v4.0 Base Vector
ZC-01: Mass Assignment (`IsAdmin=true`)	**CRITICAL**	9.3	`CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`
ZC-02: SignalR Message Injection	**HIGH**	8.7	`CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:L/VI:H/VA:N/SC:N/SI:L/SA:N`
ZC-03: Business Logic Validation Missing	**HIGH**	8.6	`CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:N/VC:N/VI:H/VA:N/SC:N/SI:H/SA:N`
ZC-04: Stored XSS (Payment Notes)	**HIGH**	8.5	`CVSS:4.0/AV:N/AC:L/AT:N/PR:L/UI:P/VC:H/VI:H/VA:N/SC:L/SI:L/SA:N`
ZC-05: Missing MFA & CAPTCHA	**HIGH**	8.2	`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`
ZC-06: Outdated Backend Components	**HIGH**	8.2	`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N`

Finding	Severity	CVSS v4.0	CVSS v4.0 Base Vector
ZC-07: Insecure Cookie Attributes	HIGH	7.7	`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N`
ZC-08: CORS Misconfiguration	HIGH	7.5	`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:P/VC:H/VI:L/VA:N/SC:N/SI:N/SA:N`
ZC-09: Federation Metadata Exposure	MEDIUM	6.9	`CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`
ZC-10: Weak SSL/TLS Protocols & Ciphers	MEDIUM	6.8	`CVSS:4.0/AV:N/AC:H/AT:P/PR:N/UI:N/VC:H/VI:N/VA:N/SC:N/SI:N/SA:N`
ZC-11: Lack of API & Login Rate Limiting	MEDIUM	6.3	`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:N/VC:L/VI:N/VA:L/SC:N/SI:N/SA:N`
ZC-12: Server Fingerprinting & Stack Traces	MEDIUM	5.3	`CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:L/VI:N/VA:N/SC:N/SI:N/SA:N`
ZC-13: Missing HTTP Security Headers	MEDIUM	5.1	`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:A/VC:L/VI:L/VA:N/SC:N/SI:N/SA:N`
ZC-14: Outdated Bootstrap Version	MEDIUM	5.1	`CVSS:4.0/AV:N/AC:L/AT:P/PR:N/UI:A/VC:L/VI:L/VA:N/SC:N/SI:N/SA:N`
ZC-15: Missing Subresource Integrity (SRI)	LOW	2.1	`CVSS:4.0/AV:N/AC:H/AT:P/PR:N/UI:P/VC:L/VI:L/VA:N/SC:N/SI:N/SA:N`

Prepared by: ZenithCrypt Offensive Security Team | **Date:** [Date] | **Classification:** CONFIDENTIAL