CETONIX

SECURITY ASSURANCE · OFFENSIVE SECURITY PRACTICE

CONFIDENTIAL · TLP:AMBER+STRICT

PENETRATION TEST REPORT

Web Application · Software-as-a-Service Platform

Black-box, Grey-box and Authenticated Testing

PREPARED FOR

ACME Corporation, Inc.

ACME Workspace — app.acme-corp.example

ENGAGEMENT ID	REPORT VERSION	TEST WINDOW	ISSUED
CTX-PT-2026-0418	1.2 — Final (Post-Retest)	13 - 24 Jul 2026	31 July 2026

REDACTED DISTRIBUTION COPY

Request/response bodies, credentials, tokens, host identifiers and customer data are masked. The client copy contains all evidence in full. See §2 Redaction Key.

Cetonix · EON Freezone, Kharadi, Pune 411014, India · info@cetonix.com

Contents

This report is issued as a redacted distribution copy. Section 2 explains exactly what has been masked and what the client copy contains in its place.

1. Document Control

1.1 Engagement details

Engagement reference	CTX-PT-2026-0418
Client	ACME Corporation, Inc.
Engagement type	Web application penetration test — black-box, grey-box and authenticated (grey/white hybrid)
Target	ACME Workspace — multi-tenant SaaS platform and supporting REST API
Testing window	13 July 2026 – 24 July 2026 (10 working days)
Retest window	28 July 2026 – 29 July 2026
Report issued	31 July 2026
Report version	1.2 — Final (post-retest)
Classification	CONFIDENTIAL · TLP:AMBER+STRICT
Testing standard	OWASP WSTG v4.2 · OWASP ASVS v4.0.3 · PTES · NIST SP 800-115

1.2 Version history

VER.	DATE	AUTHOR	CHANGE
0.1	22 Jul 2026	Lead Consultant	Initial draft issued for internal peer review
0.9	25 Jul 2026	Technical QA Reviewer	Peer review complete; CVSS vectors validated; two findings re-scored
1.0	27 Jul 2026	Practice Manager	Final draft issued to client for remediation
1.1	29 Jul 2026	Lead Consultant	Retest results incorporated; nine findings verified as resolved
1.2	**31 July 2026**	**Practice Manager**	**Final issue — attestation letter appended; redacted distribution copy produced**

1.3 Distribution and authorised recipients

This document is released to the named recipients below. Onward distribution requires written approval from the client's engagement owner. Cetonix retains no copy of client credentials or captured data beyond the retention period stated in section 2.4.

NAME / ROLE	ORGANISATION	COPY ISSUED
[Client Engagement Owner] Head of Security	ACME Corporation, Inc.	**Client copy — unredacted**
[Client Technical Lead] VP Engineering	ACME Corporation, Inc.	**Client copy — unredacted**
[Client Compliance Lead]	ACME Corporation, Inc.	**Client copy — unredacted**

NAME / ROLE	ORGANISATION	COPY ISSUED
Director, GRC		
External auditor / assessor	[Audit firm]	**Redacted distribution copy**
Customer vendor-risk reviewers	Various	**Redacted distribution copy**
Lead Consultant, Test Team	Cetonix	Master copy — encrypted at rest

1.4 Assessment team

ROLE	RESPONSIBILITY	QUALIFICATION PROFILE
Lead Consultant	Test execution, findings, report authorship	Offensive security specialist; web and API application testing; 8 years' practice
Senior Analyst	Authenticated and business-logic testing	Application security background; multi-tenant SaaS specialisation
Technical QA Reviewer	Independent peer review, CVSS validation	Did not participate in testing; reviews all findings prior to issue
Practice Manager	Engagement governance, client communication	Accountable for scope, impartiality screening and final sign-off

Named team, not a rotating pool

The same lead consultant runs the engagement from scoping through retest and is available to your engineering team throughout. Analyst identities are provided to the client in the unredacted copy and withheld here.

2. Confidentiality, Handling and Redaction Key

This copy has been prepared for onward distribution to parties outside the client's security team — auditors, assessors, prospective customers and vendor-risk reviewers. It carries the full narrative, methodology, findings, severity ratings and remediation guidance of the original, with sensitive technical detail masked so that the document cannot itself be used to attack the platform it describes.

What redaction does and does not remove

Nothing about the existence, severity, impact or remediation of a finding is withheld. What is removed is the material that would let a reader reproduce the attack: live identifiers, credentials, tokens, payload strings, internal hostnames and customer data.

2.1 Redaction key

Every masked value is replaced by a solid bar and a bracketed tag naming what was removed. The tags used in this report are listed below.

TAG	WHAT IT MASKS	IN THE CLIENT COPY
[REDACTED — SESSION TOKEN]	Bearer tokens, JWTs, refresh tokens and cookies captured during testing	Full value, truncated to 32 chars
[REDACTED — CREDENTIAL]	Account passwords, API keys, HMAC secrets and MFA codes	Full value
[REDACTED — TENANT UUID]	Workspace and tenant identifiers that map to real customers	Full identifier
[REDACTED — OBJECT ID]	Record identifiers used to demonstrate direct object reference flaws	Full identifier plus the enumeration range tested
[REDACTED — CUSTOMER PII]	Names, email addresses, billing contacts and any personal data observed	Recorded as a data-class count only; raw PII is never retained
[REDACTED — PAYLOAD]	Working exploit strings and injection payloads	Full payload with encoding notes
[REDACTED — HOSTNAME]	Internal, staging and undocumented hostnames discovered during recon	Fully qualified names and resolved addresses
[REDACTED — INTERNAL ENDPOINT]	Internal service URLs and cloud metadata addresses	Full URL and response excerpt
[REDACTED — CREDENTIAL MATERIAL]	Any secret material returned by a vulnerable service	**Masked in every copy — see 2.3**

2.2 Shared copy versus client copy

THIS COPY — REDACTED FOR DISTRIBUTION

```
POST /api/v2/auth/token HTTP/1.1
Host: api.acme-corp.example
{"email":"                    [ACCOUNT]
  "password":"                [CREDENTIAL]
  "otp":"                     [MFA CODE]

HTTP/1.1 200 OK
{"access_token":"             [BEARER TOKEN]
  "refresh_token":"           [REFRESH TOKEN]
  "tenant":"                  [TENANT UUID]
```

Sanitised for auditors, prospects and vendor-risk reviewers.

CLIENT COPY — COMPLETE EVIDENCE

```
POST /api/v2/auth/token HTTP/1.1
Host: api.acme-corp.example
{"email":"test.analyst+al@cetonix...
  "password":"Tr!alAcct-2026-...
  "otp":"4915"

HTTP/1.1 200 OK
{"access_token":"eyJhbGciOiJIUzI1Ni...
  "refresh_token":"rt_7f3a91c4e0b2...
  "tenant":"9c1e4f70-3ab2-4d18-...
```

Issued to named client contacts over encrypted transfer.

Figure 2.1 — The same authentication capture as it appears in this copy (left) and in the client copy (right).

2.3 Material redacted in every copy

Three classes of data are masked even in the client's own copy, because retaining them would create risk that outlasts the engagement:

- Live credential material recovered from a vulnerable service — for example cloud instance credentials reached through CTX-03. Cetonix records that the material was reachable, its type and its scope, and instructs the client to rotate it; the value itself is destroyed at capture.
- Customer personal data observed during cross-tenant testing. Records are counted and classified, never copied. Sampling stops at the minimum needed to prove impact.
- Cardholder data. Testing is designed so that primary account numbers are never retrieved; where a payment reference is returned it is recorded as a token reference only.

2.4 Handling, retention and destruction

Transfer	Encrypted transfer to named recipients only. Reports are not sent as unprotected email attachments.
Storage	Evidence and drafts held encrypted at rest, accessible only to the assigned engagement team and the QA reviewer.
Retention	Evidence retained for 90 days after the retest closes, to support re-issue and auditor queries; the report itself is retained for the period agreed in the master services agreement.
Destruction	All captures, credentials and test accounts are destroyed on client request or at the end of the retention period, with written confirmation.
Third parties	No client data, capture or credential is shared with, or processed by, any third-party service. See section 10 for the position on AI tooling.

3. Executive Summary

PURPOSE

ACME Corporation engaged Cetonix to perform an independent penetration test of the ACME Workspace SaaS platform and its supporting API. The engagement was commissioned to satisfy annual testing obligations across the client's compliance programme and to give the engineering leadership an evidence-based view of the platform's resilience to a motivated attacker holding an ordinary customer account.

OVERALL ASSESSMENT

Overall risk rating: HIGH — reduced to LOW following remediation and retest

Testing identified a chain of authorisation weaknesses that allowed an attacker starting from a self-service trial account to reach records belonging to other tenants. Nine of thirteen findings, including all four rated High or above, were remediated and verified as resolved during the retest window.

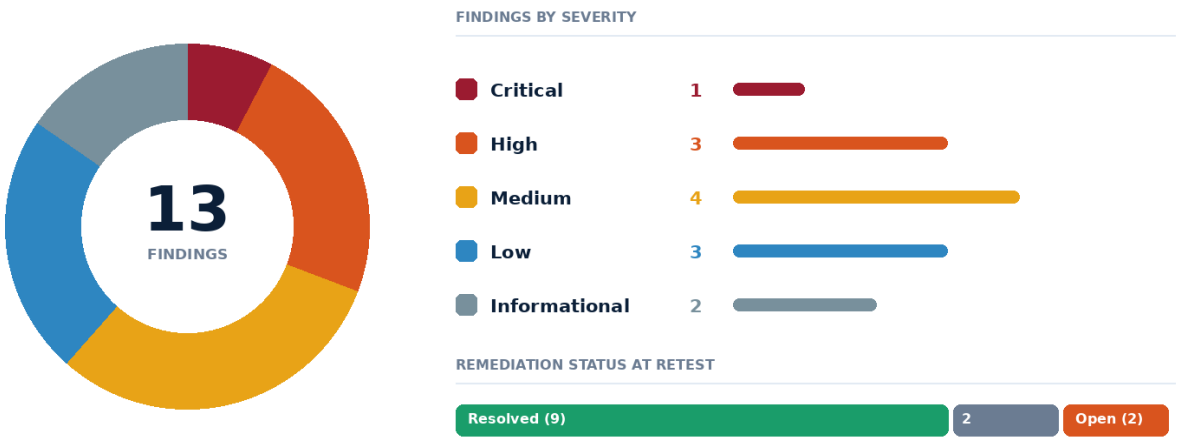


Figure 3.1 — Findings by severity and remediation status at retest.

3.1 What we found

The platform's perimeter is in good order. TLS configuration, patching of internet-facing components and the marketing estate presented no significant issues, and the authentication service resisted direct attack. The weaknesses are not at the edge — they are in the authorisation decisions made once a request is already inside the application.

In short: the application reliably established

In short: the application reliably established ***who*** a caller was, and unreliably established ***what that caller was entitled to reach***. Four findings share that single root cause.

Principal business risks

RISK	HOW IT ARISES	EXPOSURE
Customer data exposure across tenants	Invoice objects are retrieved by identifier without checking that the object belongs to the caller's workspace (CTX-01).	**Records of 3 tenants reached from one trial account**

RISK	HOW IT ARISES	EXPOSURE
Silent privilege escalation	The user-update endpoint binds a role attribute that is absent from the documented schema and invisible in the interface (CTX-02).	**Member → Workspace Admin in a single request**
Internal network reach from a customer account	Webhook destinations are validated as submitted strings rather than resolved hosts, and redirects are followed (CTX-03).	**Cloud metadata service reachable**
Account takeover within a workspace	Stored content is rendered into the dashboard of every workspace member without contextual encoding (CTX-04).	**Admin session context obtained**
Audit and attribution gaps	Privilege changes and webhook modifications are not written to the audit log, so the above would leave little trace.	**Detection unlikely without change**

3.2 What was done well

- Multi-factor authentication is enforced for all administrative roles and could not be bypassed at the login flow.
- Transport security is correctly configured: TLS 1.2 and 1.3 only, HSTS present with a long max-age, no weak cipher suites observed.
- The payment path is properly tokenised — no primary account number was retrievable at any point, and the cardholder data environment remained segmented from the application tier under test.
- Server-side input handling against classic injection classes was sound; no SQL, command or template injection was achievable in 148 tested endpoints.
- The engineering team responded to the two same-day critical notifications within four hours, and had CTX-01 mitigated in production before testing concluded.

3.3 Recommended priorities

1. Introduce a single server-side authorisation layer that resolves the caller's tenant from the session and applies it to every object lookup, rather than trusting a client-supplied workspace header. This one change addresses CTX-01 and hardens the platform against the whole class.
2. Move to explicit allow-lists for request binding so that attributes such as role, entitlement and tenant can never be set by the client.
3. Validate outbound destinations after DNS resolution, block private address ranges, and refuse to follow redirects on customer-supplied URLs.
4. Extend the audit log to cover privilege changes, integration and webhook modification, and administrative impersonation.
5. Adopt continuous authenticated testing between annual assessments so that regressions in these areas are caught at the release that introduces them.

4. Compliance and Regulatory Alignment

The engagement was scoped so that a single test satisfies the technical testing obligations of multiple frameworks at once. Each finding in section 13 carries its own control mapping; the summary below shows the position across the client's programme.

4.1 Framework coverage

FRAMEWORK	OBLIGATION ADDRESSED	FINDINGS AFFECTING	POSITION AT ISSUE
PCI DSS v4.0.1	Req. 11.4.2 / 11.4.3 — internal and external penetration testing at least every 12 months and after significant change, to a documented methodology (11.4.1), with exploitable findings corrected and retested (11.4.4)	CTX-01, CTX-02, CTX-06, CTX-07	**Satisfied — retest complete**
SOC 2 (TSC 2017)	CC4.1 monitoring of controls, CC6.1 logical access, CC6.6 protection against external threats, CC7.1 vulnerability identification	CTX-01, CTX-02, CTX-03, CTX-04, CTX-06	**Evidence provided**
ISO/IEC 27001:2022	Annex A 8.8 technical vulnerability management, A 8.9 configuration management, A 8.28 secure coding, A 8.29 security testing in development and acceptance	All findings	**Evidence provided**
HIPAA Security Rule	§164.308(a)(1)(ii)(A) risk analysis, §164.308(a)(8) periodic technical evaluation, §164.312(a)(1) access control, §164.312(b) audit controls	CTX-01, CTX-02, CTX-05, CTX-06	**Findings affecting ePHI paths remediated**
GDPR	Art. 25 data protection by design, Art. 32 security of processing — including regular testing of technical measures	CTX-01, CTX-04, CTX-08	**Art. 32(1)(d) testing evidence**
NYDFS 23 NYCRR 500	§500.5 penetration testing at least annually from both inside and outside the information system	Applicable if ACME is a covered entity	**External test complete; internal test recommended**

A note on HIPAA and penetration testing

The Security Rule as currently in force does not name penetration testing explicitly; it requires risk analysis and periodic technical evaluation, of which testing is the accepted evidence. The Notice of Proposed Rulemaking published on 27 December 2024 (RIN 0945-AA22) would make vulnerability scanning at least every six months and penetration testing at least every twelve months explicit obligations. That rule remained proposed and not finalised as at the date of this report; organisations handling ePHI should plan on the twelve-month cycle regardless.

4.2 Control mapping by finding

ID	PCI DSS V4.0.1	ISO 27001:2022	SOC 2	HIPAA
CTX-01	7.2.1, 7.2.2	A 5.15, A 8.3	CC6.1, CC6.3	§164.312(a)(1)
CTX-02	7.2.2, 8.2.1	A 5.18, A 8.2	CC6.1, CC6.2	§164.312(a)(1)
CTX-03	1.3.1, 6.2.4	A 8.20, A 8.28	CC6.6, CC6.7	§164.312(e)(1)
CTX-04	6.2.4, 6.4.2	A 8.28	CC6.6, CC7.1	§164.312(c)(1)
CTX-05	8.3.1, 8.3.9	A 8.5	CC6.1	§164.312(d)
CTX-06	8.2.8, 8.6.1	A 8.5, A 5.18	CC6.1	§164.312(a)(2)(iii)
CTX-07	8.3.4	A 8.5	CC6.1, CC7.2	§164.308(a)(5)(ii)(D)
CTX-08	6.5.3, 11.4.3	A 8.31, A 8.9	CC8.1	§164.308(a)(1)(ii)(A)
CTX-09	6.2.4	A 8.28	CC7.1	—
CTX-10	6.4.2	A 8.9	CC6.6	—
CTX-11	8.2.1	A 5.16	CC6.2	—
CTX-12	6.3.3	A 8.8	CC7.1	—
CTX-13	—	A 5.14	CC6.7	—

5. Scope and Rules of Engagement

Scope, authorisation and constraints were agreed in writing before any traffic was sent to client infrastructure. The signed rules of engagement are held with the engagement record and are available to assessors on request.

5.1 In-scope assets

ASSET	TYPE	ENVIRONMENT	TEST DEPTH
app.acme-corp.example	Web application	Production	Black-box + authenticated, all roles
api.acme-corp.example	REST API v2	Production	Authenticated, 148 endpoints from OpenAPI spec
www.acme-corp.example	Marketing site	Production	Black-box, non-intrusive
admin.acme-corp.example	Staff console	Staging mirror	Authenticated, staff role
Single sign-on integration	SAML 2.0	Production	Flow testing; identity provider itself out of scope
Webhook / integration subsystem	Outbound HTTP	Production	Authenticated, configuration abuse

5.2 Explicitly out of scope

EXCLUDED	REASON
Denial-of-service and volumetric testing	Excluded by agreement; production platform serving live customers
Social engineering, phishing and physical access	Not commissioned under this engagement
Third-party identity provider infrastructure	Operated by a third party; no authorisation to test
Payment processor environment	Out of the client's control; tokenised integration tested only at the boundary
Underlying cloud provider infrastructure	Covered by the provider's shared-responsibility model and their own testing programme
Source code review	Not commissioned here; recommended as a follow-on for the authorisation layer

5.3 Test accounts and roles

Two accounts were provisioned at each permission level. The second account at each level is what makes tenant isolation testable — without it, cross-tenant access cannot be demonstrated safely.

ROLE	ACCOUNTS	TENANT	PURPOSE
Anonymous	—	—	Unauthenticated perimeter testing
Trial user (self-service)	2	Attacker-controlled	Entry point for the escalation chain
Member	2	Tenant A and Tenant B	Baseline permissions; horizontal access testing
Workspace Admin	2	Tenant A and Tenant B	Vertical access testing; integration configuration
Staff / support console	1	Cross-tenant by design	Administrative function and impersonation testing

5.4 Rules of engagement

Testing window	09:00–21:00 client local time, Monday to Friday. Out-of-hours testing by prior agreement only.
Rate limiting	Automated requests capped at 5 per second per host; no concurrent scanning of production during business hours.
Data handling	Where a flaw exposed another tenant's data, sampling stopped at the minimum required to evidence impact. No customer data was exported.
Destructive actions	No deletion, modification or corruption of production records. Write operations limited to objects owned by the test accounts.
Critical escalation	Findings of Critical severity reported to the client's named contact within two hours of confirmation, ahead of the report.
Identification	All test traffic carried the header X-Cetonix-Assessment: CTX-PT-2026-0418 and a dedicated user-agent so the client could distinguish it from genuine attacks.
Stop conditions	Testing halts immediately on evidence of prior compromise, on production instability, or on client instruction.

Scope limitations affecting coverage

The staff console was tested against a staging mirror rather than production, at the client's request. Differences between the two environments were reviewed with the client and are not believed to affect the findings, but staff-console results carry that qualification. No other constraint materially limited coverage.

6. Methodology

The engagement followed a documented, repeatable methodology aligned to the OWASP Web Security Testing Guide v4.2 and the Application Security Verification Standard v4.0.3, within the phase structure of PTES and NIST SP 800-115. Automated tooling was used for coverage and discovery only; every finding in this report was manually reproduced before it was written up.

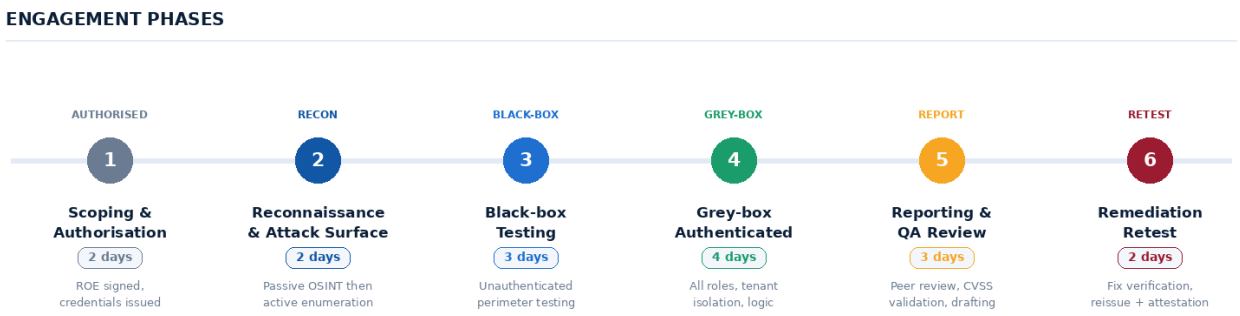


Figure 6.1 — Engagement phases and duration.

6.1 Phase detail

1. Scoping and authorisation	Asset confirmation, role definition, credential issue, rules of engagement signed by both parties. No traffic is sent to client systems before this is complete.
2. Reconnaissance	Passive intelligence gathering first — certificate transparency, passive DNS, public code search, credential exposure corpora — followed by active enumeration within the agreed rate limits. Output is the attack surface in section 7.
3. Black-box testing	Unauthenticated assessment of everything reachable without credentials: the perimeter, authentication flows, error behaviour, exposed interfaces and information disclosure. Simulates an attacker with no prior access.
4. Grey-box authenticated testing	Credentialed testing across every role and both tenants. Access control, tenant isolation, business logic, session management and integration abuse. This is where the material findings were made.
5. Reporting and QA	Independent peer review by a reviewer who did not perform the testing, CVSS vector validation, and drafting. Two findings were re-scored during review.
6. Remediation retest	Verification of the client's fixes, re-issue of the report with each finding marked resolved, accepted or open, and issue of the attestation letter.

6.2 Coverage against OWASP WSTG

The table records which WSTG categories were exercised and the outcome. Categories marked as not applicable had no corresponding functionality in the target.

WSTG CATEGORY	TESTED	OUTCOME
Information gathering (WSTG-INFO)	**Yes**	Three unexpected exposures identified — see 7.3

WSTG CATEGORY	TESTED	OUTCOME
Configuration and deployment (WSTG-CONF)	Yes	CTX-08, CTX-10
Identity management (WSTG-IDNT)	Yes	CTX-11; role provisioning reviewed
Authentication (WSTG-ATHN)	Yes	CTX-05, CTX-07; MFA enforcement confirmed sound
Authorisation (WSTG-ATHZ)	Yes	CTX-01, CTX-02 — principal weakness area
Session management (WSTG-SESS)	Yes	CTX-06
Input validation (WSTG-INPV)	Yes	CTX-03, CTX-04; no injection achievable server-side
Error handling (WSTG-ERRH)	Yes	CTX-09
Cryptography (WSTG-CRYP)	Yes	No issues identified; TLS and at-rest configuration sound
Business logic (WSTG-BUSL)	Yes	Workflow, quota and billing logic exercised; no bypass achieved
Client-side (WSTG-CLNT)	Yes	CTX-04, CTX-12
API testing (WSTG-APIT)	Yes	148 endpoints; OWASP API Top 10 applied in full

7. Reconnaissance and Attack Surface

Reconnaissance establishes what an attacker can see before they have an account. It runs passive-first: everything obtainable from public sources is gathered without touching client infrastructure, and only then is active enumeration performed within the agreed rate limits.

7.1 Passive intelligence

Reconnaissance — passive asset enumeration (read-only)

```
$ cetonix-recon --scope acme-corp.example --passive --rate-limit 2/s

[*] Certificate transparency        → 41 hostnames
[*] Passive DNS                     → 27 hostnames
[*] Public code search              →  3 repositories referencing scope
[*] Credential exposure corpora     →  2 addresses in historic breach sets

[+] app.acme-corp.example           200  nginx      SaaS application
[+] api.acme-corp.example           401  nginx      REST API v2
[+] www.acme-corp.example           200  cloudfront Marketing
[!] [redacted]                      [REDACTED — HOSTNAME]         200  nginx      Legacy API v1 — undocumented
[!] [redacted]                      [REDACTED — HOSTNAME]         200  nginx      Staging — no access control
[!] [redacted]                      [REDACTED — HOSTNAME]         403  nginx      Internal tooling — reachable

[*] Email security posture:  SPF ok · DKIM ok · DMARC p=none  → spoofable
[*] TLS: TLS 1.2/1.3 only · HSTS present · no weak suites observed

[=] 68 hostnames resolved · 6 in scope · 3 unexpected exposures escalated same day
```

Evidence RECON-01 · Passive phase only — no traffic sent to client infrastructure at this stage

Evidence RECON-01 — Passive enumeration output. Hostnames of non-public assets are masked in this copy.

7.2 Discovered attack surface

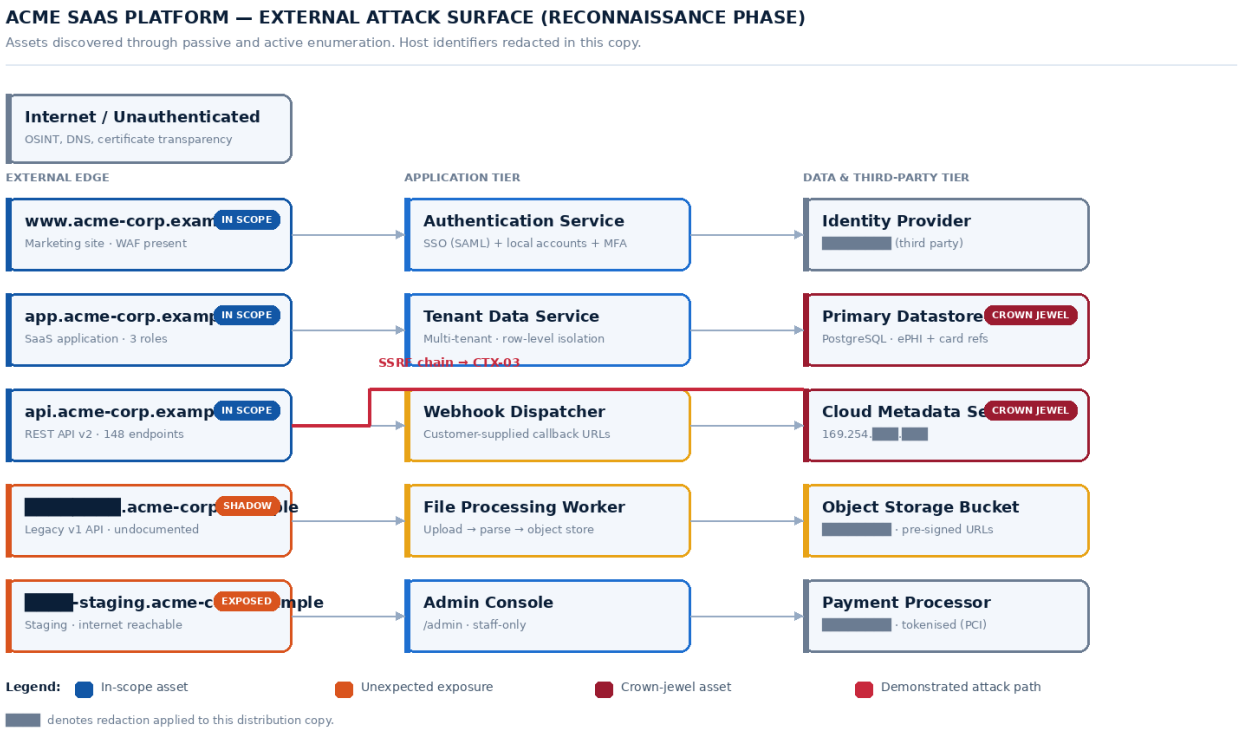


Figure 7.1 — External attack surface with demonstrated attack path overlaid.

7.3 Unexpected exposures

Three assets were reachable from the internet that the client's asset inventory did not record. Each was reported on the day of discovery rather than held for the report.

ASSET	STATUS	ASSESSMENT
Legacy API v1	**Undocumented**	Superseded interface still answering requests, running an older authorisation model. Not in the client's inventory. Raised as CTX-08 and decommissioned during the engagement.
Staging environment	**Internet-reachable**	No network restriction and no access control, populated with a production-like dataset. Contributed to CTX-08.
Internal tooling host	**Reachable, 403**	Responded with an authorisation error rather than being unreachable, confirming its existence. No content obtained; recommended for network-level restriction.

7.4 Perimeter posture

CONTROL	RESULT	DETAIL
TLS configuration	**Pass**	TLS 1.2 and 1.3 only; no weak suites; certificate chain valid; HSTS present with max-age 31536000 and includeSubDomains

CONTROL	RESULT	DETAIL
DNS and email	**Partial**	SPF and DKIM correctly published; DMARC policy set to p=none, so spoofed mail is not rejected — raised as CTX-13
Exposed services	**Pass**	No management interfaces, databases or administrative ports reachable from the internet
Component patching	**Pass**	No internet-facing component with a known, reachable vulnerability was identified
Credential exposure	**Partial**	Two corporate addresses appear in historic third-party breach corpora; neither password was valid against the platform, and MFA was enforced on both accounts
WAF / edge filtering	**Present**	Edge filtering observed on the marketing estate; the application and API are not behind equivalent filtering — noted in remediation

8. Black-box Testing — Unauthenticated

The black-box phase assesses what an attacker with no credentials and no prior knowledge can achieve. It ran for three days against the perimeter, the authentication surface and all publicly reachable functionality.

8.1 Authentication surface

The login flow, password reset, SSO initiation and account provisioning paths were exercised in full.

- **MFA enforcement —** no bypass achieved. Second-factor validation could not be skipped through parameter manipulation, response tampering, flow interruption or direct navigation to post-authentication endpoints.
- **Credential stuffing resistance —** the authentication endpoints applied no rate limiting or lockout, allowing high-volume automated attempts. Raised as CTX-07.
- **Password reset —** tokens were single-use, appropriately random and correctly expired; however the flow did not re-validate the second factor before completing (CTX-05), and produced distinguishable responses for known and unknown accounts (CTX-11).
- **SSO flow —** SAML assertion handling was sound. Signature validation, audience restriction and replay protection all behaved correctly; assertion tampering was rejected.

8.2 Unauthenticated application surface

TEST AREA	RESULT	OBSERVATION
Forced browsing to authenticated routes	Pass	All authenticated routes correctly rejected unauthenticated requests at the server, not only in the client bundle
Injection (SQL, NoSQL, command, template)	Pass	No injection achievable in any unauthenticated parameter
Information disclosure	Fail	Unhandled error states returned framework stack traces including file paths and library versions — CTX-09
Security headers	Fail	No Content-Security-Policy; Referrer-Policy and Permissions-Policy absent — CTX-10
Account enumeration	Fail	Password reset responses distinguished registered from unregistered addresses — CTX-11
Client-side secrets	Pass	JavaScript bundles reviewed; no API keys, tokens or internal endpoints disclosed
HTTP request smuggling	Pass	No desynchronisation achievable between edge and origin
Subdomain takeover	Pass	All CNAME records resolved to live, claimed services

8.3 Outcome of the black-box phase

No unauthenticated path into the application was found. An attacker without credentials is limited to information disclosure and the account-enumeration weakness — useful for targeting, not for access. The material risk begins once an account exists, and an account is available to anyone through self-service sign-up. That is the premise of the authenticated phase.

9. Grey-box Testing — Authenticated

The authenticated phase ran for four days across five role levels and two tenants. Its purpose is to answer three questions: can a user reach data belonging to another user at the same level, can a user acquire a higher level, and can a tenant reach anything belonging to another tenant.

9.1 Access control matrix

Each role was tested against each functional area for both horizontal access (another account at the same level) and vertical access (functionality belonging to a higher level).

FUNCTIONAL AREA	TRIAL	MEMBER	WKSP ADMIN	RESULT
Own workspace records	n/a	**Pass**	**Pass**	Correctly scoped
Other tenant's records	**FAIL**	**FAIL**	**FAIL**	CTX-01 — no tenant check on object retrieval
User and role management	**FAIL**	**FAIL**	**Pass**	CTX-02 — role bound from request body
Billing and subscription	**Pass**	**Pass**	**Pass**	Correctly restricted
Integrations and webhooks	**Pass**	**Pass**	**FAIL**	CTX-03 — destination validation bypass
Audit log access	**Pass**	**Pass**	**Pass**	Read scoping correct; coverage incomplete — see 9.4
Staff console functions	**Pass**	**Pass**	**Pass**	No customer role reached staff functionality

9.2 Tenant isolation

Tenant isolation was tested by performing every read and write operation from Tenant A while substituting identifiers belonging to Tenant B. The platform passes this test for 6 of 7 object types. The exception is the invoice object, where the workspace identifier is read from a client-supplied header and never compared with the object's owner.

Testing stopped at the point of proof. Records belonging to three tenants were confirmed readable; the client was notified the same day and the access was not repeated.

9.3 Business logic

- Subscription quota enforcement was tested for bypass through concurrent requests, direct API invocation and state manipulation — no bypass achieved.
- Billing arithmetic was tested for price, quantity and currency manipulation across the checkout and invoice adjustment flows — server-side recalculation was correctly applied in every case.
- Race conditions were tested against invitation acceptance, seat allocation and one-time coupon redemption using concurrent request bursts — all handled atomically.

- Workflow sequencing (approve before submit, activate before pay) could not be bypassed by replaying or reordering requests.

9.4 Logging and detectability

A secondary objective was to establish whether the client's own monitoring would have detected the test. It would have detected the noisy parts and missed the dangerous ones.

ACTIVITY	LOGGED	COMMENT
Failed authentication attempts	**Yes**	High-volume attempts were recorded, though no alert threshold fired — see CTX-07
Successful authentication	**Yes**	Recorded with source address and user agent
Privilege change (role escalation)	**No**	The escalation in CTX-02 produced no audit record of any kind
Webhook creation and modification	**No**	Integration changes are not audited, so CTX-03 would leave no trace
Cross-tenant object access	**No**	Object reads are not logged with the caller's tenant, so CTX-01 is not reconstructable after the fact
Staff impersonation of a customer	**Partial**	Session start is recorded; actions taken while impersonating are attributed to the customer, not the staff member

The detectability gap matters as much as the findings

Three of the four most serious findings would have produced no audit record. Even after the vulnerabilities are fixed, the client cannot currently answer the question an incident responder or a regulator would ask first: did anyone else do this before you found it?

10. Use of Artificial Intelligence in this Engagement

Cetonix discloses the role of AI tooling in every engagement, because a client buying assurance is entitled to know what was assessed by a person and what was not. The position is simple: AI accelerates the analyst, and never substitutes for them.

Enterprise models only — your data is never used for training

All AI assistance runs exclusively on enterprise-tier model services procured under commercial agreements that contractually prohibit the use of customer or engagement data for model training, with zero data retention enabled where the provider offers it. Consumer and free-tier AI services are prohibited by Cetonix policy and are blocked on engagement systems. Nothing you disclose to us — code, credentials, captures, findings or this report — enters a training corpus.

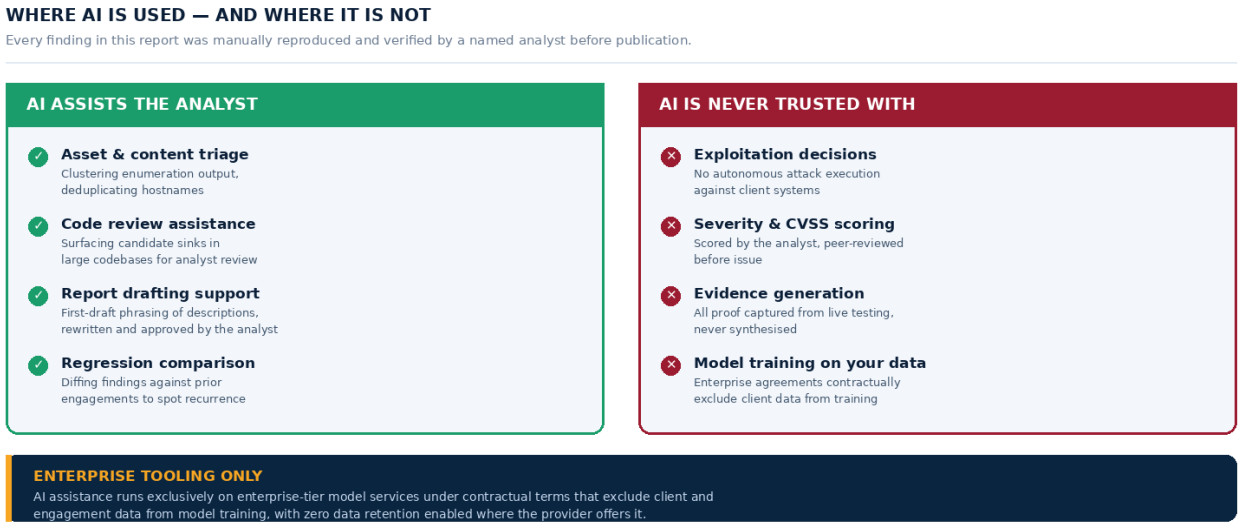


Figure 10.1 — Division between AI-assisted and analyst-only activity.

10.1 Where AI was used in this engagement

ACTIVITY	HOW AI WAS USED	HUMAN CONTROL
Enumeration triage	Clustering and deduplicating 68 discovered hostnames into candidate groups for review	Analyst confirmed every in-scope asset manually against the client inventory before testing
Endpoint prioritisation	Ranking 148 API endpoints by likely authorisation sensitivity to sequence the testing effort	All 148 endpoints were tested regardless of rank; prioritisation affected order only
Response differential review	Highlighting statistically unusual responses across large request sets for analyst attention	Every highlighted response was manually inspected; CTX-01 was confirmed by hand
Report drafting	First-draft phrasing of finding descriptions and remediation text from analyst notes	Rewritten and approved by the lead consultant; technical content originates from the analyst

ACTIVITY	HOW AI WAS USED	HUMAN CONTROL
Regression comparison	Comparing this engagement's findings against the client's prior assessment to identify recurrence	Analyst validated each match; two recurrences confirmed and noted in section 15

10.2 Where AI was not used

Exploitation	No autonomous agent was permitted to interact with client systems. Every request that reached the client's systems was issued by, or under the direct control of, a named analyst.
Severity and CVSS scoring	All ratings were assigned by the analyst and independently validated by the QA reviewer. No score in this report was generated by a model.
Evidence	All evidence is captured from live testing. Nothing in section 13 is reconstructed, illustrative or synthesised.
Finding determination	No finding was published on the basis of a model's assertion. Reproduction by a human analyst is a precondition of inclusion.

10.3 Model provenance and data handling

The controls below are contractual and technical, not aspirational. They are available for review during vendor due diligence, and they apply to every engagement Cetonix performs.

Service tier	Enterprise or business-tier model services only, procured under a signed commercial agreement. Consumer and free-tier AI services are prohibited by policy and blocked on engagement systems.
Training exclusion	Every agreement contractually excludes customer content — prompts, files, outputs and engagement material — from model training, fine-tuning and human review for product improvement.
Retention	Zero data retention is enabled where the provider offers it. Where it is not, retention is limited to the provider's minimum abuse-monitoring window and no client identifiers are submitted.
Data minimisation	Client credentials, captured traffic, customer records and personal data are never submitted to an AI service. Assistance operates on analyst notes and Cetonix-side metadata.
Residency and access	Requests are issued from Cetonix-controlled accounts with named-user access, audit logging and no third-party plugin or connector access to engagement material.
Evidence on request	The relevant provider terms and the internal AI usage policy can be supplied to the client's procurement or security function on request.

Our commitment on AI

If a finding appears in a Cetonix report, a named analyst reproduced it, scored it and stands behind it. AI changes how quickly we get to the interesting requests; it does not change who decides what is true, and it never learns from your data.

11. Findings Summary

ID	FINDING	SEVERITY	CVSS	CWE	STATUS
CTX-01	Cross-tenant object access in the invoice API	Critical	9.1	CWE-639	Resolved
CTX-02	Privilege escalation via mass assignment of the role attribute	High	8.1	CWE-915	Resolved
CTX-03	Server-side request forgery via webhook destination	High	8.6	CWE-918	Resolved
CTX-04	Stored cross-site scripting in the workspace name field	High	7.6	CWE-79	Resolved
CTX-05	Second factor not revalidated during password reset	Medium	6.5	CWE-308	Resolved
CTX-06	Sessions survive password change and role downgrade	Medium	6.3	CWE-613	Resolved
CTX-07	No rate limiting on authentication endpoints	Medium	5.9	CWE-307	Resolved
CTX-08	Staging and legacy interfaces reachable from the internet	Medium	5.3	CWE-1188	Resolved
CTX-09	Framework stack traces returned on unhandled errors	Low	3.7	CWE-209	Resolved
CTX-10	Content-Security-Policy and hardening headers absent	Low	3.1	CWE-693	Open
CTX-11	Account enumeration through password reset responses	Low	3.7	CWE-204	Accepted
CTX-12	Outdated client-side library with no reachable sink	Info	—	CWE-1104	Open
CTX-13	DMARC policy set to p=none	Info	—	CWE-16	Accepted

11.1 Risk positioning

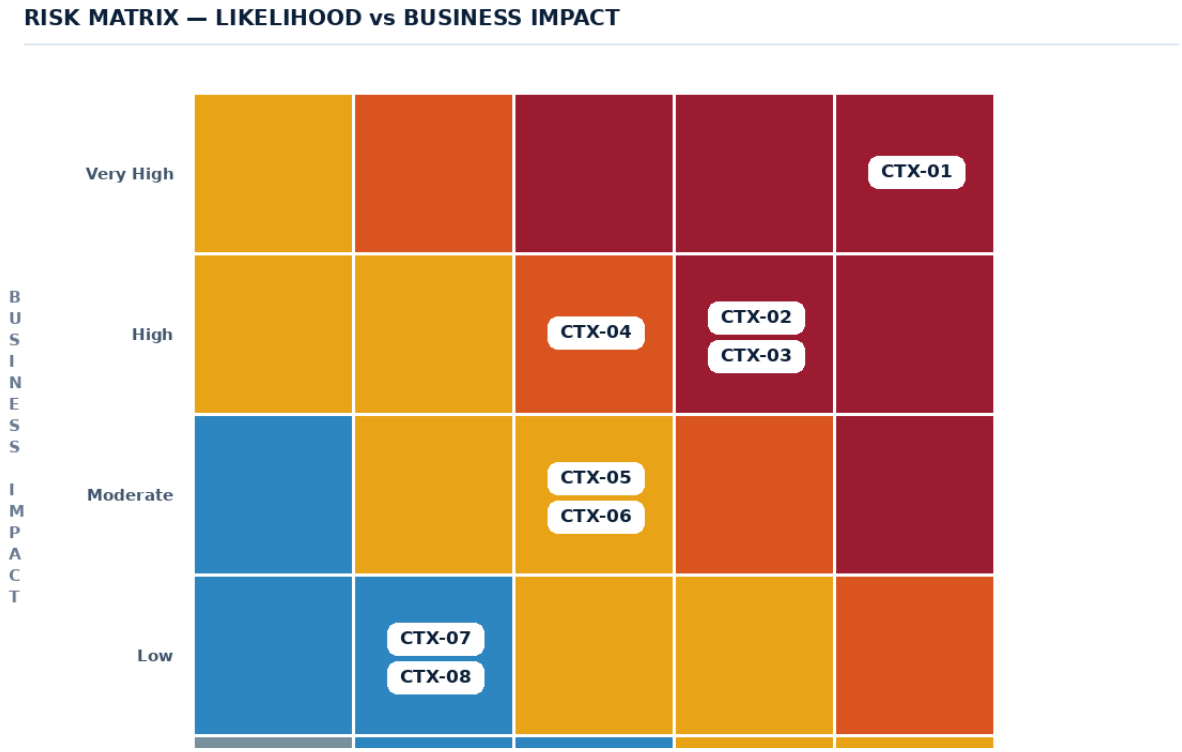


Figure 11.1 — Findings plotted by likelihood of exploitation against business impact.

12. Attack Narrative

Individually, several of these findings would read as routine. The reason the engagement is rated High is that they compose. The sequence below was executed end to end from a free trial account created through the public sign-up page, using no credential belonging to ACME or its customers.

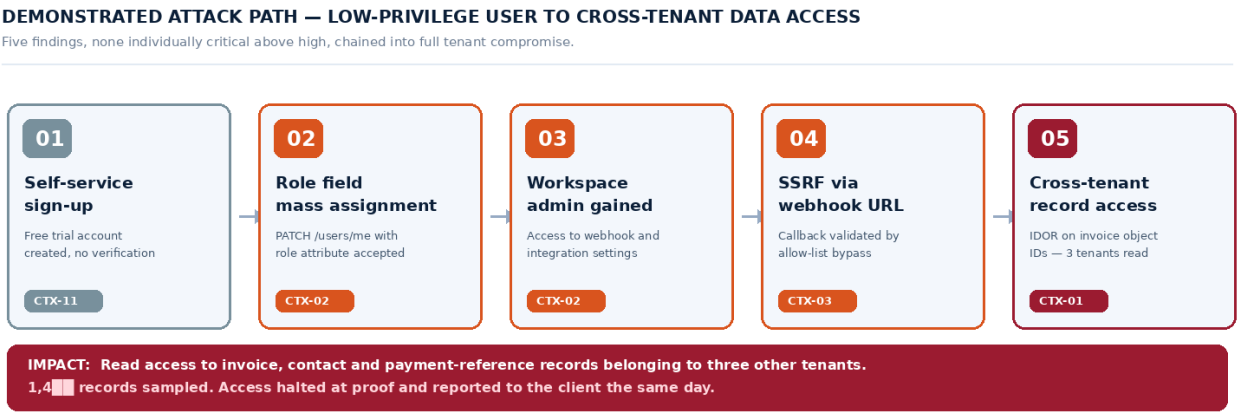


Figure 12.1 — Demonstrated attack path from anonymous sign-up to cross-tenant data access.

12.1 Narrative

1. **Entry.** A trial account was created through public self-service sign-up. No email verification, domain restriction or approval step gated the creation of a working session (CTX-11).
2. **Escalation.** The account's own profile-update request was extended with an attribute that does not appear in the published API schema and is not exposed anywhere in the interface. The server bound it and returned an elevated role, with no re-authentication, no approval and no audit record (CTX-02).
3. **Expansion.** Workspace Admin permissions unlocked the integrations area, including webhook configuration — functionality that instructs the platform to make outbound HTTP requests on the tenant's behalf.
4. **Internal reach.** A webhook destination was accepted after passing the allow-list check as a submitted string, then resolving elsewhere and following a redirect. The platform issued the request from inside its own network and returned the response body to the tenant through the delivery log (CTX-03).
5. **Objective.** Separately, invoice objects were retrieved by identifier while authenticated as a different tenant. The server read the workspace identifier from a client-supplied header and never compared it to the object's owner. Records belonging to three tenants were confirmed readable before testing was halted and the client notified (CTX-01).

> **Elapsed time from account creation to cross-tenant data access: 3 hours 41 minutes**
> None of the steps required specialist tooling, insider knowledge or a zero-day. Each was a routine authorisation mistake; the composition is what makes it a breach.

12.2 What would have broken the chain

- A server-side tenant resolution that ignores client-supplied workspace headers would have stopped step 5, independently of every other fix.
- An explicit binding allow-list on the user-update endpoint would have stopped step 2, and with it steps 3 and 4.
- Post-resolution destination validation would have stopped step 4 alone.
- Audit coverage of privilege and integration changes would not have prevented the chain, but would have made it visible — which is the difference between an incident and an undetected breach.

13. Detailed Findings

Each finding records what was found, how it was proved, what it means for the business, and what to do about it. Evidence is reproduced as captured, with sensitive values masked according to the key in section 2.

CTX-01 Cross-tenant object access in the invoice API — CRITICAL CVSS 9.1

CVSS v3.1	AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:N/A:N	**CWE**	CWE-639 · CWE-284
Affected asset	api.acme-corp.example — /api/v2/invoices/{id}	**OWASP**	API1:2023 BOLA · A01:2021
Status	**Resolved — verified 29 Jul 2026**	**Compliance**	PCI 7.2.1 · ISO A 8.3 · SOC 2 CC6.1 · HIPAA §164.312(a)(1)
References	OWASP API Security Top 10 — API1:2023 Broken Object Level Authorization · OWASP WSTG-ATHZ-04 · OWASP ASVS v4.0.3 § 4.2.1 · CWE-639 Authorization Bypass Through User-Controlled Key		

Description

The invoice retrieval endpoint resolves the requested object by its identifier and returns it without verifying that the object belongs to the workspace of the authenticated caller. Tenant scoping is derived from the X-Workspace-Id request header — a value supplied by the client and never reconciled against the session or the object's owner.

The consequence is that any authenticated user of the platform, at any permission level including a free trial account, can retrieve invoice records belonging to any other customer by presenting that customer's object identifier. Identifiers are sequential within a predictable range, so they do not need to be known in advance.

This is the single most serious finding of the engagement, and it is the objective the attack chain in section 12 reaches.

Business impact

Invoice records contain billing contact details, line-item descriptions that frequently disclose the customer's own business activity, amounts, and payment references. For ACME's regulated customers, these records constitute personal data under GDPR and, in the healthcare segment, may constitute ePHI.

A single trial account is sufficient to harvest this data across the customer base at scale. Because object reads are not logged with the caller's tenant (see 9.4), a prior exploitation of this flaw by a third party could not be ruled in or out from existing logs.

Cetonix confirmed access to records belonging to three tenants and stopped immediately. A malicious actor would not have stopped.

Evidence

Evidence CTX-01-A — A Tenant A session retrieving a Tenant B object. Identifiers, tokens and customer data masked.

Reproduction

1. Authenticate as any user in Tenant A and capture a valid session token.
2. Retrieve an invoice owned by the caller and note the identifier format: GET /api/v2/invoices/[redacted] **[REDACTED — OBJECT ID]**
3. Obtain an identifier belonging to another tenant. Identifiers are sequential within a narrow range, so this requires no prior knowledge.
4. Reissue the request with the Tenant A session and the Tenant B identifier, leaving the X-Workspace-Id header set to the caller's own workspace.
5. Observe HTTP 200 with the full object body of the other tenant's invoice, including billing contact and payment reference.

Remediation

- Resolve the caller's tenant server-side from the authenticated session. Never accept a workspace or tenant identifier from a client-supplied header, parameter or body field for authorisation purposes.
- Apply the tenant predicate in the data access layer rather than in each controller, so that a missing check in a new endpoint cannot reintroduce the flaw. Row-level security in the database is the strongest form of this.
- Return HTTP 404 rather than 403 for objects outside the caller's tenant, so the response does not confirm the existence of another tenant's records.
- Replace sequential identifiers with unguessable ones (UUIDv4 or equivalent). This is defence in depth, not a fix — it raises the cost of enumeration but does not restore authorisation.
- Add an automated test asserting that every object-returning endpoint rejects a cross-tenant identifier, and run it in the pipeline on every release.

> **Fix the class, not the endpoint**
> CTX-01 and CTX-02 are the same mistake in two places: trusting the client to describe the caller's authority. A centralised authorisation layer resolves both and prevents the next instance, which a per-endpoint patch will not.

CTX-02 Privilege escalation via mass assignment of the role attribute

HIGH
CVSS 8.1

CVSS v3.1	AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N	**CWE**	CWE-915 · CWE-269
Affected asset	api.acme-corp.example — PATCH /api/v2/users/me	**OWASP**	API3:2023 BOPLA · A01:2021
Status	**Resolved — verified 29 Jul 2026**	**Compliance**	PCI 7.2.2 · ISO A 5.18 · SOC 2 CC6.2 · HIPAA §164.312(a)(1)
References	OWASP API Security Top 10 — API3:2023 Broken Object Property Level Authorization · OWASP WSTG-BUSL-01 · OWASP ASVS v4.0.3 § 5.1.2 · CWE-915 Improperly Controlled Modification of Dynamically-Determined Object Attributes		

Description

The self-service profile update endpoint deserialises the request body directly onto the user model. The binding is inclusive rather than explicit: any attribute present on the model can be set by the caller, including attributes that are absent from the published API schema and not exposed anywhere in the user interface.

Among those attributes is the account's role. A standard Member account was able to set its own role to Workspace Admin in a single request. No re-authentication was required, no approval workflow was invoked, and no audit record was produced.

Business impact

Any user of a workspace — including one added as a read-only collaborator, and including a self-service trial account — can grant themselves administrative control of that workspace. That control includes member management, billing configuration, integration settings and data export.

In the attack chain this is the pivot: it converts an ordinary account into the Workspace Admin required to reach the webhook functionality exploited in CTX-03.

Because privilege changes are not audited, an organisation would have no record that the escalation occurred and no means of attributing subsequent administrative actions to it.

Evidence

Evidence CTX-02-A — A Member-role session setting its own role. Role values masked to prevent direct replay.

Reproduction

1. Authenticate as a standard Member of any workspace.
2. Issue a profile update containing an additional attribute not present in the published schema: "role": [redacted] **[REDACTED — ROLE VALUE]**
3. Observe HTTP 200 and the elevated role reflected in the response body.
4. Reload the application. Administrative functions — member management, billing, integrations — are now available to the account.
5. Review the workspace audit log. No entry corresponding to the privilege change is present.

Remediation

- Bind request bodies through an explicit allow-list of client-settable fields (a data transfer object), rather than deserialising onto the persistence model. Attributes governing authority must never appear on that list.
- Move role assignment to a dedicated, separately authorised endpoint that requires an existing administrator and re-authentication.
- Emit an audit event for every privilege change, recording actor, subject, previous value, new value and source address.
- Add a regression test that asserts a non-administrative session cannot alter its own role, entitlements or tenant, and run it on every release.
- Review all other endpoints that deserialise onto persistence models; the pattern is rarely present in only one place.

CTX-03 Server-side request forgery via webhook destination

HIGH
CVSS 8.6

CVSS v3.1	AV:N/AC:L/PR:H/UI:N/S:C/C:H/I:L/A:N	**CWE**	CWE-918
Affected asset	api.acme-corp.example — POST /api/v2/webhooks	**OWASP**	API7:2023 SSRF · A10:2021
Status	**Resolved — verified 29 Jul 2026**	**Compliance**	PCI 1.3.1, 6.2.4 · ISO A 8.20 · SOC 2 CC6.6 · HIPAA §164.312(e)(1)
References	OWASP API Security Top 10 — API7:2023 Server Side Request Forgery · OWASP WSTG-INPV-19 · OWASP SSRF Prevention Cheat Sheet · CWE-918 Server-Side Request Forgery		

Description

Workspace administrators may register webhook destinations to receive platform events. The destination is validated against an allow-list, but the validation is applied to the submitted string before DNS resolution, and the dispatcher follows HTTP redirects without re-applying the check.

Two consequences follow. A hostname that passes validation may resolve to an internal address, and a destination that passes validation may redirect the dispatcher to one that would not have. In both cases the platform issues the request from inside its own network boundary.

The response body of the outbound request is written to the delivery log, which is readable by the tenant. This converts a blind request-forgery primitive into a full read primitive against internal services.

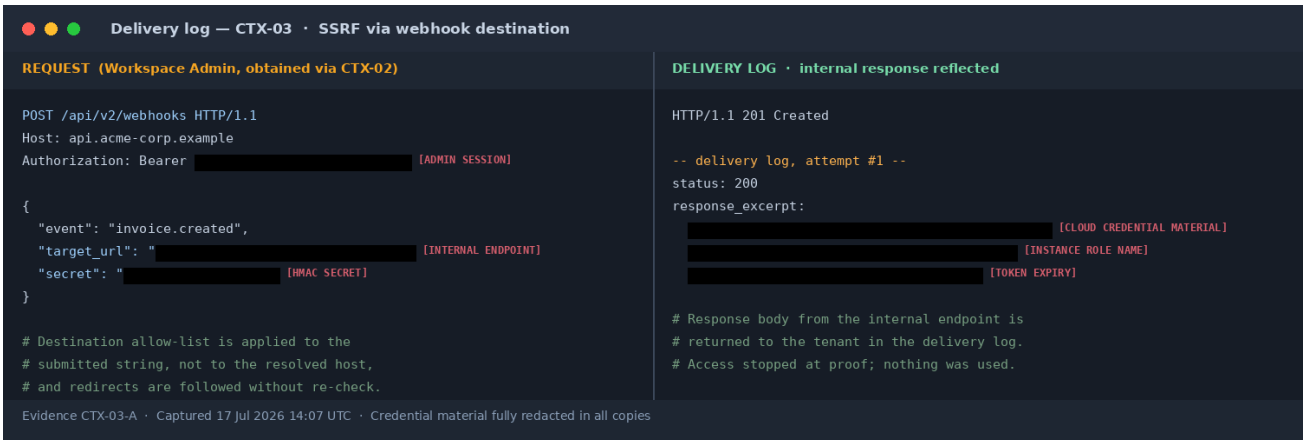
Business impact

The cloud metadata service was reachable through this path, and its response was returned to the tenant through the delivery log. Metadata services commonly expose instance credentials; where they do, a tenant of the SaaS platform obtains the platform's own cloud identity.

Any internal service reachable from the application tier — administrative interfaces, service discovery, internal APIs — becomes readable through the same mechanism, subject only to what the dispatcher will follow.

Because webhook configuration is not audited, the change that establishes this access leaves no record.

Evidence



Evidence CTX-03-A — Delivery log reflecting an internal response. Credential material is masked in all copies, including the client's.

Reproduction

1. Authenticate as a Workspace Admin. In the attack chain, this role was obtained through CTX-02 rather than granted.
2. Register a webhook whose destination passes the allow-list check as submitted but resolves to, or redirects toward, an internal address: "target_url": [redacted] **[REDACTED — INTERNAL ENDPOINT]**
3. Trigger the subscribed event so the dispatcher makes the outbound request.
4. Open the webhook delivery log for the attempt.
5. Observe the response body from the internal endpoint rendered in the log. Cetonix confirmed reachability and halted; no credential material was used or retained.

Remediation

- Validate destinations after DNS resolution, not before. Resolve the hostname, then reject any address in a private, loopback, link-local or cloud-metadata range, and re-check on every attempt to defeat DNS rebinding.
- Do not follow redirects on customer-supplied destinations. If redirects must be supported, re-apply the full validation to each hop.
- Send webhook traffic through a dedicated egress proxy with no route to internal services, so that a validation failure is not sufficient to reach anything sensitive.
- Do not return upstream response bodies to the tenant. A delivery log needs the status code and timing; it does not need the body.
- Require IMDSv2 (or the platform equivalent) so that metadata access requires a session token that a forged request cannot obtain.
- Audit webhook creation and modification as a security-relevant configuration change.

> **Rotate before you patch**
> Where an SSRF has reached a metadata service, the credentials it could have exposed should be treated as compromised and rotated before the vulnerability is closed. ACME rotated the affected instance role on the day of notification.

CTX-04 Stored cross-site scripting in the workspace name field

HIGH
CVSS 7.6

CVSS v3.1	AV:N/AC:L/PR:L/UI:R/S:C/C:H/I:L/A:N	**CWE**	CWE-79
Affected asset	app.acme-corp.example — /settings/general	**OWASP**	A03:2021 Injection
Status	**Resolved — verified 29 Jul 2026**	**Compliance**	PCI 6.2.4, 6.4.2 · ISO A 8.28 · SOC 2 CC6.6 · HIPAA §164.312(c)(1)
References	OWASP WSTG-INPV-02 · OWASP Cross Site Scripting Prevention Cheat Sheet · OWASP ASVS v4.0.3 § 5.3.3 · CWE-79 Improper Neutralization of Input During Web Page Generation		

Description

The workspace name is stored without restriction and rendered into the dashboard header of every member of that workspace. The rendering path inserts the value as markup rather than as text, so script content supplied by any user able to edit workspace settings executes in the browser of every other member.

No interaction beyond loading the dashboard is required. The absence of a Content-Security-Policy (CTX-10) removes the control that would have limited the impact.

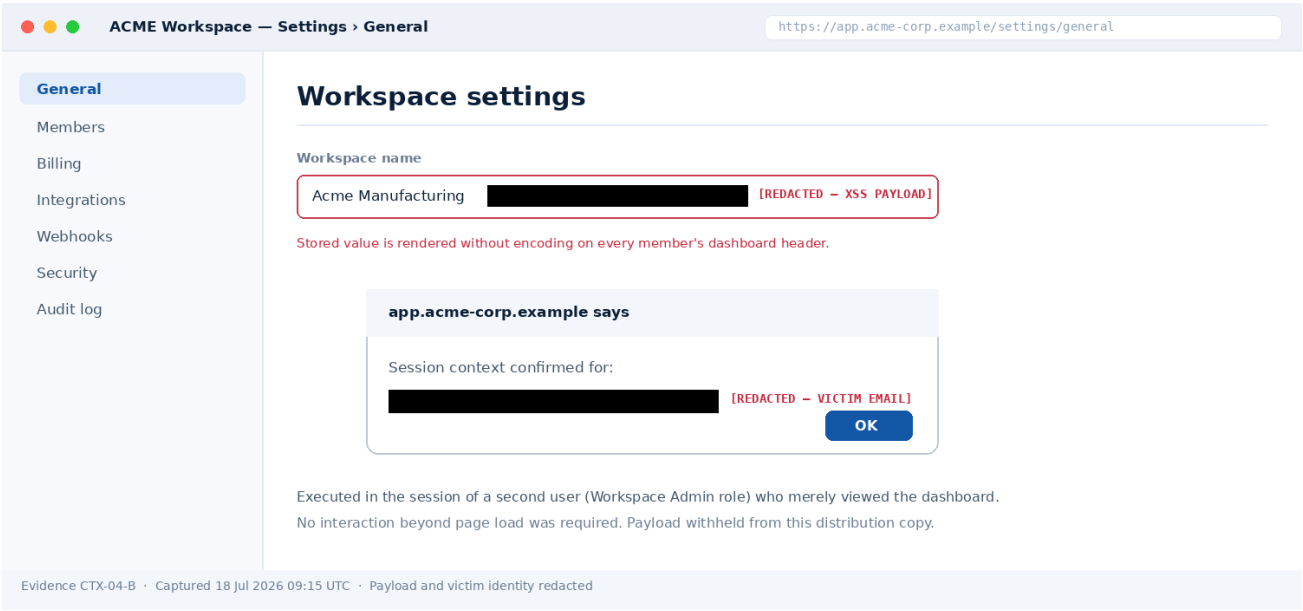
Business impact

A member with settings access can execute script in the session of every colleague who opens the dashboard, including Workspace Admins. Cetonix demonstrated capture of an administrator's session context.

Combined with CTX-02, which allows any member to obtain settings access, the pre-condition for this finding is simply holding an account in the workspace.

In the multi-tenant context the blast radius is the workspace rather than the platform, but for ACME's larger customers a workspace is the whole organisation.

Evidence



Evidence CTX-04-B — Execution in a second user's session. Payload and victim identity redacted.

Reproduction

1. Authenticate as a user with workspace settings access.
2. Set the workspace name to a value containing script content: [redacted] **[REDACTED — PAYLOAD]**
3. Save the setting and sign out.
4. Authenticate as a different member of the same workspace and open the dashboard.
5. Observe execution in the second user's session context, with no interaction beyond page load.

Remediation

- Encode on output, contextually. HTML-encode the value where it is rendered into markup; the correct encoding depends on the context, so apply it at the point of rendering rather than at input.
- Use the framework's text-binding primitives rather than raw HTML insertion for any value that originates from a user.
- Validate the field on input for length and permitted character classes — as defence in depth, never as the primary control.
- Deploy a Content-Security-Policy that disallows inline script execution (CTX-10). This converts a successful injection from account compromise into a blocked request.
- Review every other location where stored values are rendered into markup; workspace name is unlikely to be the only one.

CTX-05 Second factor not revalidated during password reset

MEDIUM
CVSS 6.5

CVSS v3.1	AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N	**CWE**	CWE-308 · CWE-640
Affected asset	app.acme-corp.example — /auth/reset	**Status**	**Resolved — verified 29 Jul 2026**
Compliance	PCI 8.3.1, 8.3.9 · ISO A 8.5 · SOC 2 CC6.1 · HIPAA §164.312(d)		
References	OWASP WSTG-ATHN-09 · OWASP Forgot Password Cheat Sheet · OWASP ASVS v4.0.3 § 2.5.6 · NIST SP 800-63B § 6.1.2		

Description

Multi-factor authentication is correctly enforced at login and could not be bypassed there. The password reset flow, however, completes on possession of the emailed reset token alone: the account's registered second factor is not challenged before the new password takes effect, and an authenticated session is issued immediately on completion.

The reset token itself was well implemented — single-use, sufficiently random and correctly expiring. The weakness is that it is treated as sufficient rather than as one factor.

Business impact

An attacker who compromises a user's email account — through a reused password, a mailbox forwarding rule or a session theft — obtains the platform account outright, despite MFA being enrolled. This reduces the platform's MFA guarantee to the security of the weakest email provider among its users, which for a multi-tenant SaaS platform is not a guarantee at all.

Evidence

```
POST /auth/reset/complete HTTP/1.1
Host: app.acme-corp.example

{ "token": [REDACTED — RESET TOKEN]
  "new_password": [REDACTED — CREDENTIAL]
  # no otp / second-factor field is accepted or required
}

HTTP/1.1 200 OK
Set-Cookie: session= [REDACTED — SESSION TOKEN]
# authenticated session issued without a second-factor challenge
```

Remediation

- Challenge the registered second factor before a password reset completes, for every account with MFA enrolled.
- Where the second factor is unavailable, route the user to an account-recovery process with identity verification rather than allowing the reset to proceed.
- Do not issue an authenticated session on reset completion; require a fresh login with the new credential and the second factor.

- Notify the account holder through a channel other than the reset email when a password is changed.

CTX-06 Sessions survive password change and role downgrade

MEDIUM
CVSS 6.3

CVSS v3.1	AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:L/A:N	**CWE**	CWE-613
Affected asset	Platform-wide session management	**Status**	**Resolved — verified 29 Jul 2026**
Compliance	PCI 8.2.8, 8.6.1 · ISO A 8.5 · SOC 2 CC6.1 · HIPAA §164.312(a)(2)(iii)		
References	OWASP WSTG-SESS-07 · OWASP Session Management Cheat Sheet · OWASP ASVS v4.0.3 § 3.3.1 · CWE-613 Insufficient Session Expiration		

Description

Existing sessions are not invalidated when the underlying authority changes. A session established before a password change continues to authenticate afterwards, and a session established at an elevated role retains that role's permissions after the account is downgraded or removed from the workspace.

Permission is evaluated once at session establishment and cached for the session's lifetime, which was observed to be 30 days with sliding renewal.

Business impact

The two actions a user takes in response to a suspected compromise — changing their password, and an administrator revoking access — do not achieve what the user believes they achieve. An attacker holding a stolen session retains access for up to 30 days after the credential is changed. In an offboarding context, a departing employee's session outlives their account.

Remediation

- Invalidate all other sessions on password change, and give the user the option to keep the current one.
- Bind sessions to a credential or authority version that increments on password change, role change or workspace removal; reject sessions presenting a stale version.
- Re-evaluate authorisation per request from server-side state rather than from claims cached at session establishment.
- Reduce the session lifetime for administrative roles and require re-authentication for sensitive actions.
- Provide users with a visible list of active sessions and a means to revoke them.

CTX-07 No rate limiting on authentication endpoints — MEDIUM CVSS 5.9

CVSS v3.1	AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:N/A:N	**CWE**	CWE-307
Affected asset	api.acme-corp.example — /api/v2/auth/*	**Status**	**Resolved — verified 29 Jul 2026**
Compliance	PCI 8.3.4 · ISO A 8.5 · SOC 2 CC6.1, CC7.2 · HIPAA §164.308(a)(5)(ii)(D)		
References	OWASP WSTG-ATHN-03 · OWASP Credential Stuffing Prevention Cheat Sheet · OWASP ASVS v4.0.3 § 2.2.1 · CWE-307 Improper Restriction of Excessive Authentication Attempts		

Description

The authentication, password reset and MFA verification endpoints applied no rate limiting, account lockout, progressive delay or challenge. Sustained automated attempts were accepted at the tested ceiling of 5 requests per second without throttling, block or alert.

Attempts were recorded in the application log, but no alert threshold fired during the engagement despite several thousand failures from a single source.

Business impact

Credential stuffing against the platform is unconstrained, which matters because two corporate addresses in scope already appear in historic breach corpora. The MFA verification endpoint being unthrottled is the more serious half: a six-digit code with unlimited attempts is not a second factor. Testing was capped by the rules of engagement, not by any control in the platform.

Remediation

- Apply rate limiting per account, per source address and globally, with progressive delay on repeated failure.
- Apply a strict attempt cap to MFA verification specifically — a small number of attempts per code, after which the code is invalidated and a new challenge required.
- Introduce a challenge (CAPTCHA or equivalent) after a failure threshold rather than blocking outright, to avoid a denial-of-service against legitimate users.
- Alert on authentication failure volume and on credential-stuffing patterns such as many accounts from one source or one password across many accounts.
- Consider a breached-password check at registration and at password change.

CTX-08 Staging and legacy interfaces reachable from the internet — MEDIUM CVSS 5.3

CVSS v3.1	AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N	**CWE**	CWE-1188 · CWE-1059
Affected asset	Two undocumented hosts — identifiers redacted	**Status**	**Resolved — decommissioned 22 Jul 2026**
Compliance	PCI 6.5.3, 11.4.3 · ISO A 8.31, A 8.9 · SOC 2 CC8.1 · HIPAA §164.308(a)(1)(ii)(A)		
References	OWASP WSTG-CONF-05 · OWASP API Security Top 10 — API9:2023 Improper Inventory Management · CWE-1188 Insecure Default Initialization of Resource		

Description

Reconnaissance identified two hosts reachable from the internet that did not appear in the client's asset inventory: a superseded v1 API still answering requests under an older authorisation model, and a staging environment with no network restriction and no access control.

The staging environment was populated with a production-like dataset. Whether that data was synthetic or a copy of production could not be determined from the outside and was escalated to the client on the day of discovery.

Business impact

Unmanaged assets sit outside the controls the managed estate receives — they are not patched on the same cycle, not monitored, and not included in the scope of assessments unless discovered. The legacy API's older authorisation model is of particular concern, since it may not carry the fixes applied to v2. Under PCI DSS the presence of undocumented internet-facing systems also undermines the scope definition on which an assessment depends.

Remediation

- Decommission the legacy interface, or place it behind authentication and network restriction if a client still depends on it, with a published sunset date.
- Restrict staging to an allow-list of source addresses or an internal network, and require authentication regardless.
- Confirm whether staging holds production-derived data; if so, treat the exposure as a potential data incident and apply masking going forward.
- Reconcile the asset inventory against certificate transparency and passive DNS on a recurring schedule, so that unmanaged assets are discovered internally rather than by an assessor.

CTX-09 — Framework stack traces returned on unhandled errors

LOW CVSS 3.7

CVSS v3.1	AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:N/A:N	**CWE**	CWE-209
Affected asset	app.acme-corp.example · api.acme-corp.example	**Status**	**Resolved — verified 29 Jul 2026**
Compliance	PCI 6.2.4 · ISO A 8.28 · SOC 2 CC7.1		
References	OWASP WSTG-ERRH-01 · OWASP ASVS v4.0.3 § 7.4.1 · CWE-209 Generation of Error Message Containing Sensitive Information		

Description

Malformed input to several endpoints produced unhandled exceptions, and the resulting responses contained framework stack traces including absolute file paths, internal module names, library versions and fragments of the query structure.

The disclosure does not by itself grant access. Its value to an attacker is in target selection: knowing the exact framework and library versions in use converts a broad search for weaknesses into a narrow one.

Business impact

Reconnaissance value to an attacker, and an information-disclosure finding that auditors and customer security reviewers will raise. Internal path disclosure also assists in constructing traversal and deserialisation attempts against other endpoints.

Evidence

```
HTTP/1.1 500 Internal Server Error
Content-Type: text/html

Traceback (most recent call last):
  File "[redacted]" [REDACTED — INTERNAL PATH]
  File "[redacted]" [REDACTED — INTERNAL PATH]
    [redacted] [REDACTED — LIBRARY + VERSION]
# framework debug handler reachable in production
```

Remediation

- Disable debug error handlers in production and return a generic error page carrying only a correlation identifier.
- Log the full trace server-side against that identifier so support can still diagnose the failure.
- Add a catch-all exception handler so that an unanticipated failure cannot fall through to the framework's own handler.
- Test error paths explicitly — malformed types, oversized values, unexpected content types — as part of the release process.

CTX-10	Content-Security-Policy and hardening headers absent	LOW CVSS 3.1

CVSS v3.1	AV:N/AC:H/PR:L/UI:R/S:U/C:L/I:L/A:N	**CWE**	CWE-693
Affected asset	app.acme-corp.example — all responses	**Status**	**Open — scheduled for Q4 2026**
Compliance	PCI 6.4.2 · ISO A 8.9 · SOC 2 CC6.6		
References	OWASP WSTG-CONF-12 · OWASP Secure Headers Project · OWASP ASVS v4.0.3 § 14.4.3 · CWE-693 Protection Mechanism Failure		

Description

No Content-Security-Policy header is issued on any application response. Referrer-Policy and Permissions-Policy are also absent. HSTS and X-Content-Type-Options are correctly present.

CSP is the control that limits what a successful injection can do. Its absence is why CTX-04 resulted in session compromise rather than a blocked inline script.

Business impact

Defence in depth is missing at the point where it matters most. Any future injection flaw — in the application or in a third-party script it loads — executes without restriction. The client has scheduled this for Q4 2026; Cetonix recommends bringing it forward given CTX-04.

Remediation

- Deploy a Content-Security-Policy in report-only mode first, review the reports, then enforce. Target a policy without unsafe-inline for scripts.
- Adopt nonce-based or hash-based script allowances so inline script can be eliminated rather than permitted.
- Add Referrer-Policy (strict-origin-when-cross-origin) and a Permissions-Policy denying unused browser features.
- Monitor CSP violation reports as a detection signal — they frequently reveal injection attempts before anything else does.

CTX-11 Account enumeration through password reset responses

LOW
CVSS 3.7

CVSS v3.1	AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N	**CWE**	CWE-204
Affected asset	app.acme-corp.example — /auth/reset, /signup	**Status**	**Risk accepted by client**
Compliance	PCI 8.2.1 · ISO A 5.16 · SOC 2 CC6.2		
References	OWASP WSTG-IDNT-04 · OWASP Authentication Cheat Sheet · OWASP ASVS v4.0.3 § 2.2.3 · CWE-204 Observable Response Discrepancy		

Description

The password reset flow returns distinguishable responses for registered and unregistered addresses, and the sign-up flow rejects addresses already in use with a distinct message. Response timing also differed measurably between the two cases.

Self-service sign-up additionally requires no email verification before a working session is issued, which is the entry point used in the attack chain in section 12.

Business impact

An attacker can confirm which addresses hold accounts before attempting credential stuffing or phishing, raising the yield of both. For a platform serving named enterprise customers, enumeration also discloses which organisations are customers — commercially sensitive independent of the security impact.

Remediation

- Return an identical response and an equivalent response time whether or not the address is registered, and communicate the outcome by email instead.
- Apply the same principle to sign-up: accept the submission and inform the user by email that an account already exists.
- Require email verification before a session is issued on self-service sign-up.
- Rate-limit both endpoints per source address to make bulk enumeration impractical.

CTX-12 Outdated client-side library with no reachable sink

INFORMATIONAL
CVSS —

A third-party JavaScript library bundled by the application is several minor versions behind current and carries a published advisory. The vulnerable function is not called by application code and the sink is not reachable through any tested path, so no exploitation was achievable and no severity is assigned.

It is recorded because dependency currency is a control in its own right — a future change to application code could make the sink reachable without anyone noticing that a known-vulnerable version is present, and customer security questionnaires routinely ask about component inventory.

Remediation

- Update the library to a current version as part of routine maintenance.
- Introduce software composition analysis in the pipeline, with a policy distinguishing reachable from unreachable vulnerable code so the signal stays useful.
- Maintain a software bill of materials for the client-side bundle; enterprise customers increasingly request one.

References: OWASP A06:2021 Vulnerable and Outdated Components · CWE-1104 Use of Unmaintained Third Party Components

CTX-13 DMARC policy set to p=none

INFORMATIONAL
CVSS —

SPF and DKIM are correctly published for the primary domain, but the DMARC record specifies p=none. Receiving mail servers are therefore told to take no action on messages that fail authentication, so the two controls that are correctly configured have no enforcement behind them.

This is outside the application boundary and does not affect the platform's security posture directly. It is recorded because it materially affects the plausibility of phishing against the client's own staff and against its customers — the initial access vector in the majority of real incidents.

Remediation

- Monitor DMARC aggregate reports until legitimate sending sources are fully accounted for.
- Move to p=quarantine, then to p=reject once reporting shows no legitimate mail is affected.
- Publish DMARC records for parked and non-sending domains with p=reject, as those are commonly abused.

References: RFC 7489 · ISO/IEC 27001:2022 Annex A 5.14 · CWE-16 Configuration

14. Remediation Roadmap

Findings are sequenced by risk reduction per unit of effort rather than by severity alone. The client completed the immediate and short-term actions during the retest window.

14.1 Prioritised actions

ID	ACTION	PRIORITY	EFFORT	OWNER
CTX-01	Server-side tenant resolution and data-layer scoping	**Immediate**	Medium	Platform engineering
CTX-02	Explicit binding allow-list; separate role-change endpoint	**Immediate**	Low	Platform engineering
CTX-03	Post-resolution destination validation; egress proxy	**Immediate**	Medium	Platform engineering
CTX-04	Contextual output encoding on stored values	**Short term**	Low	Application team
—	Audit coverage for privilege and integration changes	**Short term**	Medium	Platform engineering
CTX-05	Second-factor challenge in password reset	**Short term**	Low	Identity team
CTX-06	Session invalidation on authority change	**Short term**	Medium	Identity team
CTX-07	Rate limiting and MFA attempt caps	**Short term**	Low	Platform engineering
CTX-08	Decommission legacy API; restrict staging	**Short term**	Low	Infrastructure
CTX-09	Disable debug handlers in production	**Medium term**	Low	Application team
CTX-10	Content-Security-Policy, report-only then enforce	**Medium term**	Medium	Application team
CTX-11	Uniform responses on reset and sign-up	**Medium term**	Low	Identity team
CTX-12	Dependency update and SCA in pipeline	**Routine**	Low	Application team
CTX-13	DMARC progression to p=reject	**Routine**	Low	IT operations

14.2 Structural recommendations

Four of the thirteen findings share one root cause: authorisation decided per endpoint, from data the client supplied. Patching each endpoint resolves this report; it does not resolve the next one.

- **Centralise authorisation.** A single layer that resolves tenant and role from the session and applies them at the data-access boundary would have prevented CTX-01, CTX-02 and the cross-tenant half of CTX-03.

- **Make authorisation testable.** Add pipeline tests that assert cross-tenant and cross-role access is refused for every object-returning endpoint. Authorisation regressions are invisible in code review and obvious in a test.
- **Close the audit gap.** Privilege changes, integration changes and administrative impersonation must produce records. Without them the client cannot evidence to a regulator that a flaw was not exploited before discovery.
- **Test continuously, not annually.** Authenticated dynamic testing on each release, with static analysis in pull requests, would have caught CTX-02 and CTX-09 at the commit that introduced them.
- **Follow with a targeted code review.** A review of the authorisation layer and request-binding code would establish whether the patterns behind CTX-01 and CTX-02 exist elsewhere in paths this engagement did not reach.

15. Retest Results

The client remediated during the engagement and a formal retest was performed on 28–29 July 2026. Each finding was re-tested using the original reproduction steps, and the fix was probed for bypass rather than merely confirmed present.

ID	FINDING	RESULT	VERIFICATION
CTX-01	Cross-tenant object access	**Resolved**	Tenant now resolved from the session; client-supplied header ignored. Re-tested across all seven object types and both tenants, including header manipulation and parameter pollution attempts.
CTX-02	Mass assignment privilege escalation	**Resolved**	Binding allow-list implemented; role moved to a separate endpoint requiring an existing administrator. Twelve undocumented attributes retested; all rejected.
CTX-03	SSRF via webhook destination	**Resolved**	Validation now post-resolution with private ranges blocked and redirects refused. Response bodies no longer returned in the delivery log. Instance credentials rotated on 17 Jul.
CTX-04	Stored XSS in workspace name	**Resolved**	Contextual output encoding applied. Retested with encoding-variant and mutation payloads; no execution achieved.
CTX-05	MFA not revalidated on reset	**Resolved**	Second-factor challenge added; no session issued on completion.
CTX-06	Session survival on authority change	**Resolved**	Sessions bound to an authority version. Verified across password change, role downgrade and workspace removal.
CTX-07	No authentication rate limiting	**Resolved**	Per-account and per-source limits with progressive delay; MFA codes invalidated after five attempts.
CTX-08	Staging and legacy exposure	**Resolved**	Legacy API decommissioned 22 Jul; staging restricted to an allow-list and placed behind authentication.
CTX-09	Stack traces in production	**Resolved**	Debug handlers disabled; generic error page with correlation identifier confirmed.
CTX-10	Security headers absent	**Open**	Scheduled for Q4 2026. Cetonix recommends bringing this forward given the demonstrated injection path.
CTX-11	Account enumeration	**Accepted**	Client accepts the risk, citing sign-up conversion. Compensating rate limiting was applied.
CTX-12	Outdated client-side library	**Open**	Scheduled in the routine maintenance cycle.

ID	FINDING	RESULT	VERIFICATION
CTX-13	DMARC p=none	**Accepted**	Monitoring in progress ahead of a policy change; owned outside the platform team.

Post-retest position

All Critical and High findings are resolved. Residual risk is limited to two Low and two Informational items, of which two are accepted by the client with compensating controls in place. On the evidence of the retest, the platform's authorisation model is materially stronger than at the start of the engagement.

15.1 Recurrence against the prior assessment

Two findings recur from the client's previous assessment: the absence of a Content-Security-Policy (CTX-10) and account enumeration on the reset flow (CTX-11). Both were raised previously and remain open or accepted. Recurrence is not in itself a failure — accepted risk is a legitimate decision — but it should be reconfirmed by the risk owner each cycle rather than carried forward by default.

Appendix A — Severity and Scoring Methodology

Severity is assigned from CVSS v3.1 base scores, adjusted for the client's environment and reviewed independently before issue. Where the environmental context materially changes the risk, the adjustment and its reasoning are stated in the finding.

SEVERITY	CVSS	DEFINITION AND EXPECTED RESPONSE
Critical	9.0 – 10.0	Direct compromise of sensitive data or platform integrity, exploitable with little effort and no special access. Notified within two hours of confirmation; remediation expected immediately.
High	7.0 – 8.9	Significant compromise, or a reliable step toward one. Notified within one business day; remediation expected within 30 days.
Medium	4.0 – 6.9	Meaningful weakening of the security posture, typically requiring a precondition or chaining. Remediation expected within 90 days.
Low	0.1 – 3.9	Limited direct impact; contributes to attacker capability or defence in depth. Address in routine maintenance.
Informational	—	No direct security impact at present; recorded for completeness, hygiene or future relevance.

Business impact is assessed separately from technical severity. A technically modest flaw affecting regulated data may be escalated, and a technically severe flaw in an isolated component may be moderated. Every adjustment is recorded in the finding and validated by the QA reviewer.

Appendix B — Tooling

Tools support coverage and discovery. No finding in this report rests on tool output alone; each was reproduced manually before inclusion.

CATEGORY	PURPOSE	NOTES
Intercepting proxy	Request interception and manipulation	Primary manual testing platform for all phases
Application scanner (authenticated)	Breadth coverage across 148 endpoints	Output triaged by an analyst; no unverified result reported
API testing harness	Specification-driven endpoint exercise	Driven from the client's OpenAPI specification
Subdomain and DNS enumeration	Attack surface discovery	Passive sources first; active enumeration within agreed limits
TLS and header analysis	Transport and configuration review	Non-intrusive
Credential exposure search	Breach corpus review for in-scope domains	Read-only; no credential was used against the platform
Custom tooling	Tenant isolation and object enumeration harness	Written for this engagement; rate-limited per the rules of engagement

Appendix C — Scope Asset Inventory

The inventory as tested, reconciled against the client's own asset register. Host identifiers for non-public assets are redacted in this copy.

ASSET	TYPE	IN INVENTORY	DISPOSITION
app.acme-corp.example	Application	**Yes**	Tested in full — all roles
api.acme-corp.example	REST API	**Yes**	148 endpoints tested
www.acme-corp.example	Marketing	**Yes**	Non-intrusive testing
admin.acme-corp.example	Staff console	**Yes**	Staging mirror tested
[redacted].acme-corp.example	Legacy API v1	**No**	CTX-08 — decommissioned 22 Jul 2026
[redacted]-staging.acme-corp.example	Staging	**No**	CTX-08 — restricted 23 Jul 2026
[redacted].acme-corp.example	Internal tooling	**No**	Reported; network restriction recommended

Appendix D — Glossary

BOLA	Broken Object Level Authorization — retrieving another user's or tenant's object by presenting its identifier. The flaw in CTX-01.
Mass assignment	Binding request data directly onto an internal object, allowing a caller to set fields that were never intended to be client-controlled. The flaw in CTX-02.
SSRF	Server-Side Request Forgery — causing the server to issue requests of the attacker's choosing, typically to internal systems. The flaw in CTX-03.
Tenant isolation	The guarantee that one customer of a shared platform cannot reach another customer's data.
Black-box testing	Testing with no credentials and no prior knowledge, simulating an external attacker.
Grey-box testing	Testing with credentials at each role level, simulating an attacker who has obtained an account.
CVSS	Common Vulnerability Scoring System — an industry-standard method of expressing technical severity as a score and a vector string.
CWE	Common Weakness Enumeration — a catalogue of software weakness types, used to classify the underlying defect.
Rules of engagement	The written agreement defining what may be tested, when, how aggressively, and what is prohibited.

Appendix E — Attestation Letter

The letter below may be shared with auditors, assessors, customers and prospects. It confirms that testing was performed and that findings were remediated, without disclosing technical detail about the platform.

CETONIX

SECURITY ASSURANCE · OFFENSIVE SECURITY PRACTICE

31 July 2026

LETTER OF ATTESTATION — PENETRATION TEST

To whom it may concern,

Cetonix was engaged by ACME Corporation, Inc. to perform an independent penetration test of the ACME Workspace software-as-a-service platform and its supporting application programming interface. Testing was conducted between 13 and 24 July 2026 under engagement reference CTX-PT-2026-0418.

The assessment comprised unauthenticated (black-box) testing and authenticated (grey-box) testing across all defined user roles and two customer tenants, following a documented methodology aligned to the OWASP Web Security Testing Guide v4.2, the OWASP Application Security Verification Standard v4.0.3, the Penetration Testing Execution Standard and NIST Special Publication 800-115. All findings were manually verified by a named analyst and independently peer-reviewed prior to issue.

Thirteen findings were identified, comprising one of Critical severity, three of High severity, four of Medium severity, three of Low severity and two of an informational nature. A remediation retest was performed on 28 and 29 July 2026. All findings rated Critical and High have been remediated and independently verified as resolved. Remaining items are of Low or informational severity and are either scheduled for remediation or formally accepted by the client with compensating controls in place.

This engagement satisfies the annual penetration testing obligation under PCI DSS v4.0.1 Requirements 11.4.2 and 11.4.3, and provides technical testing evidence supporting ISO/IEC 27001:2022 Annex A 8.8 and A 8.29, SOC 2 Trust Services Criteria CC4.1 and CC7.1, and the evaluation requirement at 45 CFR §164.308(a)(8) of the HIPAA Security Rule.

This letter reflects the security posture of the assessed environment at the conclusion of the retest window. It does not constitute a guarantee against future compromise, and it does not extend to changes made to the environment after that date.

Yours faithfully,

[Name], Practice Manager — Offensive Security
Cetonix · info@cetonix.com · Verification of this letter may be requested at the address above, quoting CTX-PT-2026-0418.

END OF REPORT