ILLUSTRATIVE SAMPLE REPORT - SYNTHETIC TARGET (acmeledger.example)
PROVENANCE NOTICE: This document is a fictional worked example created by briskData to show how manual penetration testing scope, multi-tenant role testing, reproducible HTTP evidence, code-level remediation, and retest verification are structured. AcmeLedger Inc. (acmeledger.example), all tenant IDs, commit hashes, dates, and findings below are entirely synthetic and do not represent a real client system or attestation.

ILLUSTRATIVE SAMPLE DELIVERABLE | DOC ID: BD-SEC-PT-SAMPLE | SYNTHETIC TARGET

Web Application and API Penetration Test Report

ILLUSTRATIVE SAMPLE REPORT - SYNTHETIC TARGET (acmeledger.example)

1. Executive Summary (Illustrative Example)

In this illustrative engagement scenario, briskData security engineers performed an authenticated grey-box manual web application and REST/GraphQL API assessment of the fictional AcmeLedger B2B billing platform (staging-app.acmeledger.example and staging-api.acmeledger.example). Testing focused on cross-tenant data isolation, role-based access enforcement, file/document rendering, payment webhook verification, and session lifecycle controls.

Initial testing identified four illustrative vulnerabilities: one Critical cross-tenant authorization flaw, two High-severity server-side and cryptographic verification flaws, and one Medium session-management defect. In the sample retest sequence below, all four findings were remediated on synthetic build commit 91c4e80b and verified closed.

Severity Tier (CVSS v3.1)	Initial Findings	Verified Fixed at Retest	Open / Accepted	Illustrative Status
Critical (9.0 - 10.0)	1	1	0	Closed in sample retest
High (7.0 - 8.9)	2	2	0	Closed in sample retest
Medium (4.0 - 6.9)	1	1	0	Closed in sample retest
Low / Informational (0.1 - 3.9)	0	0	0	None in sample excerpt
Total Illustrative Findings	4	4	0	All sample items verified

2. Scope, Methodology, and Multi-Tenant Role Matrix

Scope Attribute	Illustrative Engagement Record (Synthetic)
Fictional Target Organization	AcmeLedger Inc. (Synthetic SaaS Example - acmeledger.example)
In-Scope Targets & Build	https://staging-app.acmeledger.example & https://staging-api.acmeledger.example | Release v2.18.4 (commit 4d91b02e)
Illustrative Test & Retest Dates	Initial Window: Oct 12 - Oct 19, 2026 [Illustrative] | Retest: Oct 27, 2026 on commit 91c4e80b [Illustrative]
Assessment Methodology	Manual grey-box testing aligned with OWASP ASVS v4.0 Level 2, OWASP API Security Top 10 (2023), and PTES
Exclusions & Constraints	Denial-of-service flooding, social engineering of staff, and third-party payment processor infrastructure excluded

2.1 Provisioned Multi-Tenant Test Accounts (Synthetic Role Matrix)

Synthetic Tenant	Tenant ID	Provisioned Roles Tested	Isolation Objective Verified
Org_Alpha (Victim Tenant)	org_1042	Owner, Billing Manager, Read-Only Auditor	Confirm Org_Bravo principals cannot read or mutate Org_Alpha records.
Org_Bravo (Attacker Tenant)	org_1043	Owner, Billing Manager, Read-Only Auditor, Revoked API Token	Test horizontal cross-tenant IDOR, vertical privilege escalation, and token revocation.

3. Synthetic Findings Summary Table

Finding ID	Vulnerability Title	Severity & CVSS	CWE / OWASP Category	Sample Retest
BD-2026-01	Cross-Tenant Invoice & ACH Export via Unscoped Object ID (IDOR)	Critical (9.1)	CWE-639 / OWASP API1:2023	Verified Fixed
BD-2026-02	Server-Side Request Forgery (SSRF) via SVG Logo in PDF Renderer	High (8.2)	CWE-918 / OWASP API7:2023	Verified Fixed
BD-2026-03	Payment Webhook Signature Bypass When Header Is Omitted	High (8.1)	CWE-347 / OWASP API8:2023	Verified Fixed
BD-2026-04	Active Session Persistence After Password Reset & Missing Idle Timeout	Medium (5.4)	CWE-613 / OWASP ASVS V3.3	Verified Fixed

4. Detailed Technical Findings (Synthetic Worked Examples)

Finding BD-2026-01: Cross-Tenant Invoice & Bank Detail Export via Unscoped Object ID (IDOR)

Severity / CVSS v3.1	Affected Endpoint	Preconditions	Sample Retest Status
Critical - 9.1 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:N)	GET /api/v2/invoices/{uuid}/export PATCH /api/v2/invoices/{uuid}	Authenticated Read-Only user in Org_Bravo (org_1043)	Verified Fixed (commit 91c4e80b)

Observed Impact (Synthetic): While dashboard list queries filtered invoices by session organization_id, the export and update handlers looked up invoice_uuid by primary key alone after checking only that the caller held billing:read in their own tenant. Because invoice UUIDs appeared in emailed receipt links and webhook logs, a user in Org_Bravo could export or modify ACH remittance details belonging to Org_Alpha.

SYNTHETIC PROOF OF CONCEPT - HTTP REQUEST & RESPONSE

```
GET /api/v2/invoices/7f9c2d10-4b88-41e0-9a12-08d4c1109e33/export HTTP/1.1
Host: staging-api.acmeledger.example
Authorization: Bearer <REDACTED_ORG_BRAVO_JWT_ORG_1043>

HTTP/1.1 200 OK   Content-Type: application/json
{"invoice_id":"7f9c2d10-...","organization_id":1042,"customer_name":"Alpha Example LLC",
 "amount_cents":4850000,"remit_ach_routing":"021000021","remit_ach_last4":"8841"}
```

REMEDIATION GUIDANCE - ENFORCE TENANT PREDICATE AT REPOSITORY LAYER

```
# Scope every object lookup by authenticated organization_id and return 404 on mismatch
stmt = select(Invoice).where(
    Invoice.id == invoice_uuid,
    Invoice.organization_id == current_principal.organization_id,
)
invoice = db.session.execute(stmt).scalar_one_or_none()
if invoice is None:
    raise HTTPException(status_code=404, detail="Invoice not found")
```

Finding BD-2026-02: Server-Side Request Forgery (SSRF) via SVG Branding Upload in PDF Renderer

Severity / CVSS v3.1	Affected Endpoint	Preconditions	Sample Retest Status
High - 8.2 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:N/A:L)	POST /api/v2/settings/branding-logo	Billing Manager role in Org_Bravo (org_1043)	Verified Fixed (commit 91c4e80b)

Observed Impact (Synthetic): Tenant branding settings accepted raw .svg uploads that were subsequently rendered by headless Chromium during PDF invoice generation. Submitting an SVG containing an external <image> reference caused the server renderer to issue internal HTTP requests to link-local and VPC-internal services.

SYNTHETIC REPRODUCTION & REMEDIATION SUMMARY

```
# Uploaded synthetic SVG payload referencing internal test listener:
<svg xmlns="http://www.w3.org/2000/svg" width="200" height="60">
  <image href="http://169.254.169.254/latest/meta-data/" width="200" height="60"/>
</svg>

# Remediation verified at retest:
# 1. Reject SVG uploads; server-side decode and re-encode logos to PNG/WebP only.
# 2. Enforce IMDSv2 (HttpTokens=required, hop-limit=1) on compute nodes.
# 3. Run PDF renderer in an isolated network namespace with RFC1918/link-local egress denied.
```

Finding BD-2026-03: Payment Webhook Signature Verification Bypass When Header Is Omitted

Severity / CVSS v3.1	Affected Endpoint	Preconditions	Sample Retest Status
High - 8.1 (CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:N/I:H/A:L)	POST /api/v1/webhooks/stripe	Unauthenticated external caller	Verified Fixed (commit 91c4e80b)

Observed Impact (Synthetic): The webhook endpoint validated HMAC-SHA256 signatures only when the Stripe-Signature header was present. When the header was omitted, a leftover local-testing branch parsed the raw JSON payload directly, permitting unauthenticated activation of paid enterprise subscription tiers.

SYNTHETIC PROOF OF CONCEPT & REMEDIATION

```
POST /api/v1/webhooks/stripe HTTP/1.1
Host: staging-api.acmeledger.example
Content-Type: application/json   # Stripe-Signature header omitted
{"id":"evt_test_forged_99102","type":"checkout.session.completed",
 "data":{"object":{"client_reference_id":"org_1043","payment_status":"paid"}}}
---
HTTP/1.1 200 OK   {"received":true,"provisioned_tier":"enterprise_annual"}

# Fix: Reject missing Stripe-Signature headers with HTTP 400 unconditionally, verify HMAC
# against unparsed raw request bytes with 300s timestamp tolerance, and store event IDs.
```

Finding BD-2026-04: Active Session Persistence After Password Reset and Missing Idle Timeout

Severity / CVSS v3.1	Affected Endpoint	Preconditions	Sample Retest Status
Medium - 5.4 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:L/A:N)	POST /api/v2/auth/password-reset	Existing session cookie or refresh token	Verified Fixed (commit 91c4e80b)

Observed Impact & Fix (Synthetic): Completing a password reset updated the user's password hash without incrementing token_version or revoking active refresh sessions in Redis, and web sessions lacked an idle timeout (remaining valid for 30 days). Remediation rotated user session keys on password/MFA change and enforced a 30-minute idle / 12-hour absolute session limit.

5. Retest Verification Log and Illustrative Attestation Format

Finding ID	Remediation Change Verified	Retested Build (Synthetic)	Retest Outcome
BD-2026-01	Repository-layer organization_id filter + cross-tenant integration tests	v2.18.5 / 91c4e80b	Remediated & Verified
BD-2026-02	PNG/WebP rasterization only + IMDSv2 hop-limit 1 + egress-blocked renderer	v2.18.5 / 91c4e80b	Remediated & Verified
BD-2026-03	Mandatory raw-body HMAC check + 300s replay guard + event_id unique constraint	v2.18.5 / 91c4e80b	Remediated & Verified
BD-2026-04	Server-side session revocation on credential reset + 30m idle / 12h max timeout	v2.18.5 / 91c4e80b	Remediated & Verified

ILLUSTRATIVE LETTER OF ATTESTATION EXCERPT (SYNTHETIC EXAMPLE - NOT AN ACTUAL ATTESTATION)

In a real engagement where retest verification is included, briskData can provide a standalone customer-shareable Letter of Attestation summarizing authorized scope, testing dates, methodology (OWASP ASVS / API Top 10 / PTES), and retest verification status without exposing internal endpoint paths, source diffs, or infrastructure identifiers. This sample PDF is strictly an educational illustration for fictional acmeledger.example.