GARLIC THREAT

STAY AHEAD OF THE THREAT.

SAMPLE REPORT

VULNERABILITY ASSESSMENT & PENETRATION TEST

Security Assessment Report

What a Garlic Threat engagement delivers: clear risk, proof and a fix for every finding.



7.4 OVERALL RISK

HIGH · 7.4 / 10

TARGET	CLIENT	ASSESSED	REPORT ID
app.acme-demo.io	Acme Demo Co.	02 – 04 NOV 2026	GT-2026-1104-A7F3

CONFIDENTIAL — DO NOT SHARE WITH ANY OUTSIDERS

Fictional target. All findings are illustrative.

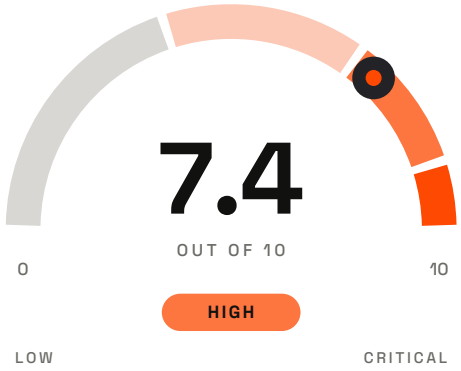
02

Executive summary

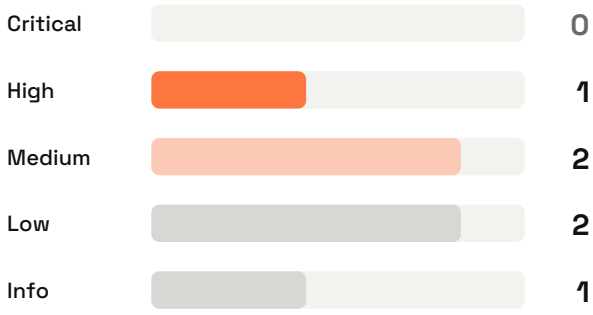
Where you stand, in under a minute.

FOR LEADERSHIP

OVERALL RISK



FINDINGS BY SEVERITY



6 findings · 0 critical · 1 needs action this week

6	1,396	6,923	8h 30m
FINDINGS	ENDPOINTS TESTED	TEST ACTIONS	TESTING TIME

KEY TAKEAWAYS

1. One High-severity access-control flaw (F-01) lets any signed-in user read and edit another organisation's projects. Fix this first.
2. Two Medium issues, login throttling (F-02) and logout CSRF (F-03), weaken account protection but expose no data.
3. No Critical issues. No injection, file-upload or admin-panel exposure. Headers and error handling need tidying.

WORKING WELL

- ✓ TLS and certificates correctly configured
- ✓ Session cookies: Secure, HttpOnly, SameSite
- ✓ No exposed admin or debug interfaces
- ✓ Injection and file-upload tests clean

HOW WE TESTED

22 testing phases, condensed into six stages.

01 Recon & mapping	02 Content discovery	03 Auth & session	04 Access & injection	05 Logic & infrastructure	06 Verify & report
DNS, tech stack, attack surface	Paths, files, endpoints	Login, cookies, CSRF	IDOR, injection, APIs	Business flows, headers, cloud	Proof, CVSS, fixes

03

Findings at a glance

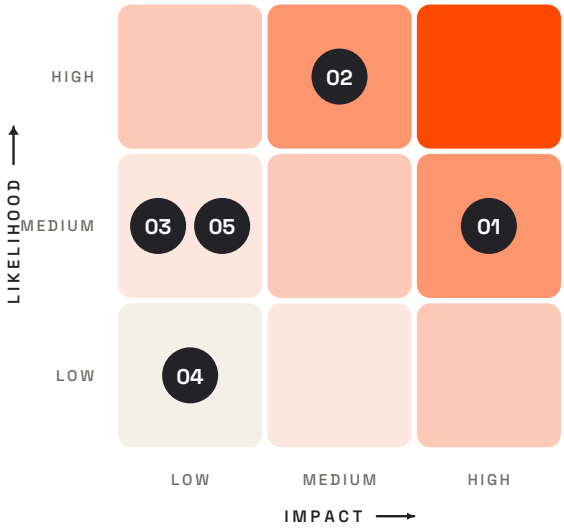
FOR SECURITY LEADS

Six findings, ranked by severity, each mapped to CWE and OWASP.

ID	FINDING	SEVERITY	CVSS	CWE	STATUS
F-01	**Broken object-level authorization** `GET/PATCH /api/v1/projects/{id}`	HIGH	8.1	CWE-639	Verified ✓
F-02	**No rate limiting on magic-link login** `POST /login/magic`	MEDIUM	5.3	CWE-307	Verified ✓
F-03	**Logout CSRF via unguarded POST** `POST /logout`	MEDIUM	4.3	CWE-352	Verified ✓
F-04	**Missing CSP and Permissions-Policy headers** `All HTML responses`	LOW	3.7	CWE-693	Verified ✓
F-05	**Verbose error messages leak stack traces** `GET /api/v1/export?fmt=`	LOW	3.1	CWE-209	Verified ✓
F-06	**Server version banner disclosed** `Response header: Server`	INFO	0.0	CWE-200	Observed

RISK MATRIX

Likelihood × impact



F-06 is informational and not plotted.

OWASP TOP 10 COVERAGE

Where findings landed

A01 Access control — 2	A02 Crypto failures ✓
A03 Injection ✓	A04 Insecure design ✓
A05 Misconfiguration — 3	A06 Vulnerable components ✓
A07 Auth failures — 1	A08 Integrity failures ✓
A09 Logging & monitoring ✓	A10 SSRF ✓

Dark tiles = findings. Outlined tiles = tested, clear.

CVSS V3.1 SEVERITY SCALE

INFO 0.0 · LOW 0.1 – 3.9 · MEDIUM 4.0 – 6.9 · HIGH 7.0 – 8.9 · CRITICAL 9.0 – 10

Each finding is verified with a working proof of concept where it is safe to do so. Status shows whether the issue was exploited (Verified) or only observed (Observed).

F-01

HIGH

Broken object-level authorization on the project API

Any signed-in user can read and edit another organisation's projects.

CVSS 3.1	CWE	OWASP	STATUS
8.1 High	639	A01 · API1:2023	Verified

AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N

ENDPOINT `/api/v1/projects/{id}`

WHAT WE FOUND

The project endpoint verifies that a caller is logged in, but never checks that the requested project belongs to the caller's organisation. Changing the project ID in the URL returns, and allows editing of, projects owned by other customers.

BUSINESS IMPACT

Cross-tenant disclosure of project names, members and settings, plus unauthorised modification. This is a direct breach of customer data isolation and may trigger contractual and regulatory notification duties.

ATTACK PATH

1. **Attacker signs in** — as a normal user of Org A
2. **Swaps the project ID** — in the request URL
3. **API checks login only** — ownership is never verified
4. **Org B data exposed** — read and write access

PROOF OF CONCEPT (SANITISED)

REQUEST

```
GET /api/v1/projects/prj_b7e204 HTTP/1.1
Host: app.acme-demo.io
Authorization: Bearer eyJ...  (user A, org_A)
Accept: application/json
```

RESPONSE

```
HTTP/1.1 200 OK
Content-Type: application/json
{ "id": "prj_b7e204",
  "org": "org_B",
  "name": "Billing", "members": 14 }
```

User A belongs to org_A. The API still returns org_B's project because it never compares the owner to the caller.

RECOMMENDED FIX

- **Scope every lookup** — Filter object queries by the caller's organisation, e.g. WHERE id = ? AND org_id = caller.org.
- **Return 404, not 403** — Treat objects outside the caller's scope as non-existent, and cover each route with an authorization test.
- **Harden identifiers** — Use unguessable IDs (UUIDv4) as defence in depth. Not a substitute for the ownership check.

FIX EFFORT

Low

~1 developer-day

RETEST

After patch

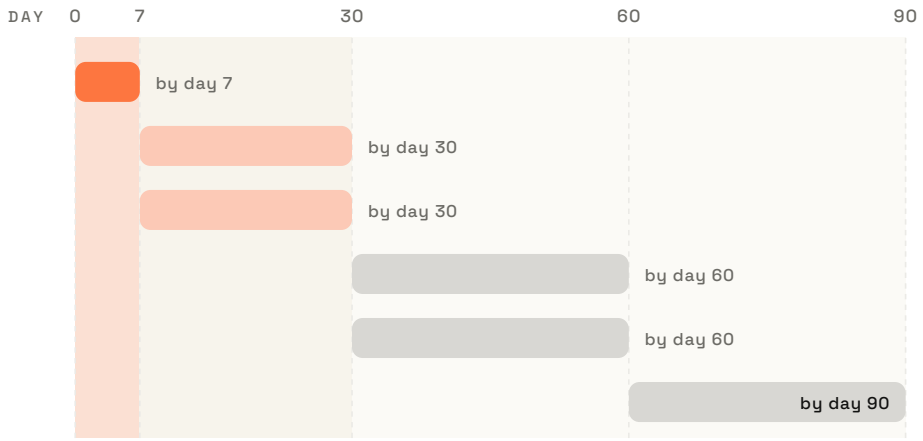
05

Remediation roadmap

What to fix, and in which order.

FOR ENGINEERING

FIX SCHEDULE



Orange = fix first. Dates are targets from the day the report is delivered.

WHAT HAPPENS NEXT

01 Remediate

Your team ships the fixes in the order above. We are on hand for questions.

02 Retest

We re-run the original proof of concept for every finding against your patched build.

03 Close out

You receive a short close-out note confirming which findings are resolved.

SCOPE AND LIMITATIONS

Scope: the hosts listed on the cover. Systems outside that list were not tested. Testing combines automated tooling with manual verification and was performed with written authorisation from the asset owner.

Limitations: no assessment finds every vulnerability. Treat this report as a point-in-time view and pair it with code review, regular retesting and secure development practices.

Stay ahead of the threat.

Want a report like this for your own product? We will scope it with you.

darshil@garlicthreat.tech

GARLIC THREAT
STAY AHEAD OF THE THREAT.

CONFIDENTIAL — DO NOT SHARE WITH ANY OUTSIDERS