POLARIS INTELLIGENCE

# Technical Due Diligence Report

express

03 April 2026

Ref: 7d64f479

CONFIDENTIAL

## 1. Executive Summary

**Repository Classification: Library (Confidence: high)**
**This repository is a published library/package. Dependency findings reflect the library's own dependencies, not a consuming application.**

> **No critical findings identified.** The codebase presents a generally low-risk profile based on automated analysis.

*The table below rates risk across 13 dimensions, from Clean (no findings) to Critical (potential deal impact). Together they form the technical risk profile of the target asset.*

| CATEGORY | RATING | SUMMARY |
|---|---|---|
| **Secrets & Credentials** | CLEAN | No credential exposure detected in scanned files |
| **Dependency Vulnerabilities** | LOW RISK | No known vulnerabilities. 16 of 44 dependencies abandoned (health score 5.5/10) |
| **Supply Chain Risk** | CLEAN | No known supply chain incidents |
| **Licence & IP** | CLEAN | Permissive licence, no IP issues detected |
| **Developer Concentration** | CLEAN | Tier 5: Healthy Distribution |
| **Code Quality** | CLEAN | Grade B (79.0/100) |
| **Architecture** | CLEAN | 21 isolated modules (standalone utilities, examples, or platform-specific code) |
| **Malware / Destructive Code** | CLEAN | No suspicious patterns detected in scanned files |
| **Test Coverage** | CLEAN | Strong testing discipline (65% test ratio, CI gates active) |
| **Infrastructure & Deployment** | CLEAN | Limited deployment infrastructure (expected for library) (capped from low — library archetype) |
| **Technical Debt** | CLEAN | Minimal tech debt (2 markers, 0.1/KLOC) |
| **Governance & CI/CD** | LOW RISK | Adequate governance (Grade B, 7.9/10) |
| **Engineering Maturity** | LOW RISK | Adequate maturity (3.7 releases/yr, 80% community health) |

## 2. Transaction Impact Assessment

*This section translates technical findings into their commercial implications for the transaction. Ratings range from Clean (no concern) to Critical (potential deal-breaker), with specific conditions that may need to be met before or after completion.*

| CATEGORY | ASSESSMENT | DETAIL |
|---|---|---|
| **Security Exposure** | LOW RISK | No known vulnerabilities. 16 of 44 dependencies abandoned (health score 5.5/10) |
| **Operational Risk** | CLEAN | Tier 5: Healthy Distribution; Grade B (79.0/100) |
| **IP & Licence Risk** | CLEAN | Permissive licence, no IP issues detected |
| **Integration Complexity** | CLEAN | 21 isolated modules (standalone utilities, examples, or platform-specific code) |
| **Maintenance Burden** | CLEAN | Grade B, 0 quality issues, 2 tech debt markers |

### Remediation Effort Estimate

**No immediate remediation required.**

*Estimates assume a senior developer familiar with the technology stack. Actual effort may vary based on codebase familiarity and organisational context.*

## 3. Scope & Methodology

**Repository:** https://github.com/expressjs/express

**Analysis date:** 03 April 2026

**Codebase size:** 175 files, 26,142 lines

| LANGUAGE | LINES |
|---|---|
| JavaScript | 21,346 |
| Markdown | 4,198 |
| YAML | 395 |
| JSON | 99 |
| HTML | 46 |

### Methodology

This report was produced by Polaris Intelligence automated analysis pipeline. The following scanners were applied:

1. **GitHub Enrichment** — project metadata, release cadence, community health
2. **Secret Scanner** — regex pattern matching + context classification
3. **Dependency Scanner** — manifest parsing + OSV vulnerability cross-reference + exploitability analysis
4. **Supply Chain Intelligence** — cross-reference against known supply chain incidents
5. **Licence Auditor** — declared licence + source header contradiction detection
6. **Bus Factor Analysis** — 5-tier developer concentration taxonomy (24-month window)
7. **Code Quality Scorer** — cyclomatic complexity, duplication, security anti-patterns
8. **Architecture Mapper** — import graph, circular dependencies, module coupling
9. **Engineering Maturity** — release discipline, community governance, project signals
10. **Malware Heuristic** — destructive actions, crypto mining, exfiltration, obfuscation
11. **Governance & CI/CD** — OpenSSF Scorecard, branch protection, dependency management

***Note:*** *File counts may vary between sections because each scanner operates on a different subset of files (e.g. quality analysis covers source code files only, while the scope total includes configuration, documentation, and data files).*

*This is an automated analysis and does not constitute legal, security, or investment advice. Findings should be verified by qualified professionals.*

## 4. Secrets & Credentials CLEAN

*Hardcoded credentials — API keys, database passwords, tokens — are the most common cause of data breaches. Their presence indicates both an immediate security exposure and a gap in the target's engineering practices that transfers with the acquisition.*

> **No hardcoded secrets or credentials detected.**

## 5. Dependency Vulnerabilities LOW RISK

*Modern software relies on hundreds of third-party packages. Known vulnerabilities in these dependencies are publicly catalogued and actively exploited. Unpatched critical CVEs represent a quantifiable security liability that transfers to the acquirer.*

44 dependencies analysed across 1 manifest (28 runtime, 16 dev/test).

> **No known vulnerabilities detected.**

**First-party package checked: npm:express. No known CVEs affecting current version.**

## Historical CVE Record

*These CVEs were filed against prior versions of the target's package. While the current version may not be affected, a history of security incidents indicates recurring risk and may inform the buyer's assessment of the development team's security posture.*

**express**: 5 CVEs on record

| SEVERITY | CVE/ID | SUMMARY |
|---|---|---|
| MEDIUM RISK | CVE-2024-10491 | Express ressource injection |
| MEDIUM RISK | CVE-2014-6393 | No Charset in Content-Type Header in express |
| LOW RISK | CVE-2024-9266 | Express Open Redirect vulnerability |
| LOW RISK | CVE-2024-43796 | express vulnerable to XSS via response.redirect() |
| MEDIUM RISK | CVE-2024-29041 | Express.js Open Redirect in malformed URLs |

## Dependency Health

*"No known CVEs" does not mean healthy dependencies. Stale or abandoned packages receive no security patches and represent latent risk. Health status is derived from package registry release dates.*

| STATUS | | COUNT |
|---|---|---|
| **Active (released within 6 months)** | CLEAN | 17 |
| **Stable (released within 1 year)** | CLEAN | 5 |
| **Stale (1–2 years since last release)** | MEDIUM RISK | 4 |
| **Abandoned (2+ years since last release)** | HIGH RISK | 16 |
| **Unknown (registry lookup failed)** | LOW RISK | 2 |

Dependency health score: **5.5/10**

***Note:*** *2 dependencies returned implausible release dates from package registries and are excluded from the health score calculation.*

### At-Risk Dependencies

| PACKAGE | VERSION | STATUS | LAST RELEASE |
|---|---|---|---|
| escape-html | 1.0.3 | HIGH RISK abandoned | 2015-09-01 (10.6yr ago) |
| vhost | 3.0.2 | HIGH RISK abandoned | 2015-10-13 (10.5yr ago) |
| after | 0.8.2 | HIGH RISK abandoned | 2016-08-16 (9.6yr ago) |
| once | 1.4.0 | HIGH RISK abandoned | 2016-09-06 (9.6yr ago) |
| pbkdf2-password | 1.2.1 | HIGH RISK abandoned | 2016-09-29 (9.5yr ago) |
| etag | 1.8.1 | HIGH RISK abandoned | 2017-09-13 (8.6yr ago) |
| vary | 1.1.2 | HIGH RISK abandoned | 2017-09-24 (8.5yr ago) |
| method-override | 3.0.0 | HIGH RISK abandoned | 2018-07-12 (7.7yr ago) |
| depd | 2.0.0 | HIGH RISK abandoned | 2018-10-26 (7.4yr ago) |
| parseurl | 1.3.3 | HIGH RISK abandoned | 2019-04-16 (7.0yr ago) |
| range-parser | 1.2.1 | HIGH RISK abandoned | 2019-05-11 (6.9yr ago) |
| proxy-addr | 2.0.7 | HIGH RISK abandoned | 2021-06-01 (4.8yr ago) |
| on-finished | 2.4.1 | HIGH RISK abandoned | 2022-02-22 (4.1yr ago) |
| content-type | 1.0.5 | HIGH RISK abandoned | 2023-01-29 (3.2yr ago) |
| merge-descriptors | 2.0.0 | HIGH RISK abandoned | 2023-11-16 (2.4yr ago) |

## 6. Developer Concentration (Bus Factor) CLEAN

*“Bus factor” measures how many developers would need to leave before critical knowledge is lost. High concentration in one developer creates key-person dependency — a material operational risk that can delay integration and increase post-acquisition costs.*

**Tier 5: Healthy Distribution** — Bus factor score: **25.0%**

Developer distribution is healthy across 15 active contributors. Commit concentration is below the 40% threshold, indicating knowledge is reasonably distributed across the team.

### Top Contributors

| DEVELOPER | COMMITS | OWNED FILES | CORE % | STATUS |
|---|---|---|---|---|
| Ulises Gascon | 40 | 1 | 0% | Active |
| Sebastian Beltran | 31 | 4 | 8% | Active |
| Wes Todd | 25 | 5 | 4% | Departed |
| Phillip Barta | 17 | 10 | 4% | Active |
| Jon Church | 15 | 8 | 0% | Active |
| Shivam Sharma | 8 | 5 | 4% | Active |
| Blake Embrey | 8 | 3 | 8% | Departed |
| Mert Can Altin | 6 | 0 | 0% | Departed |
| S M Mahmudul Hasan | 3 | 0 | 0% | Departed |
| Carlos Serrano | 3 | 0 | 0% | Departed |

### Departed Developer Risk

| DEVELOPER | MONTHS INACTIVE | CORE FILES OWNED | RISK |
|---|---|---|---|
| Szymon Łągiewka | 15 | 14 | HIGH RISK |
| Blake Embrey | 16 | 2 | MEDIUM RISK |
| Wes Todd | 12 | 1 | MEDIUM RISK |
| prajesh | 14 | 1 | MEDIUM RISK |
| Shahan Arshad | 16 | 1 | MEDIUM RISK |
| Mert Can Altin | 12 | 0 | LOW RISK |

## 7. Licence & Intellectual Property CLEAN

*Copyleft licences (GPL, AGPL) require derivative works to be released under the same open-source terms. If copyleft code is embedded in a proprietary product, the acquirer may face an obligation to open-source their own code — or costly remediation to replace the affected components.*

### Declared Licences

| SPDX ID | RISK | SOURCE | FILE |
|---|---|---|---|
| MIT | CLEAN | licence_file | `LICENSE` |
| MIT | CLEAN | package_metadata | `package.json` |

**No licence contamination detected.**

## 8. Code Quality & Technical Debt CLEAN

*Code quality directly predicts the cost and speed of post-acquisition development. A low grade signals elevated technical debt — higher bug rates, slower feature delivery, and more expensive onboarding for new developers joining after the transaction.*

Quality grade: **B (79.0/100)** — Acceptable

### Grade Scale

| GRADE | MEANING |
|---|---|
| **A** | Excellent maintainability |
| **B** | Good engineering quality |
| **C** | Moderate technical debt |
| **D** | High technical debt |
| **F** | Severe structural risk |

| METRIC | VALUE |
|---|---|
| **Total files** | 50 |
| **Code lines** | 2,365 |
| **Functions** | 32 |
| **Classes** | 0 |
| **Avg function length** | 12.7 lines |
| **Avg complexity** | 2.7 |

### Complexity Hotspots

| SEVERITY | FILE | FUNCTION | COMPLEXITY |
|---|---|---|---|
| MEDIUM RISK | `lib/response.js` | sendfile | 14 |

### Large Files (>500 lines)

| FILE | TOTAL LINES | CODE LINES |
|---|---|---|
| `lib/response.js` | 1048 | 475 |
| `lib/application.js` | 632 | 262 |
| `lib/request.js` | 528 | 154 |

### Security Anti-Patterns

*These patterns represent common security vulnerabilities (OWASP Top 10, CWE). Each warrants developer review to confirm whether it represents an actual risk in context.*

| SEVERITY | PATTERN | LOCATION | DESCRIPTION |
|---|---|---|---|
| HIGH RISK | eval() | `test/res.redirect.js:115` | Dynamic code execution via eval() — potential code injection vector |
| HIGH RISK | .innerHTML = | `test/res.redirect.js:115` | Direct innerHTML assignment — potential XSS vector |

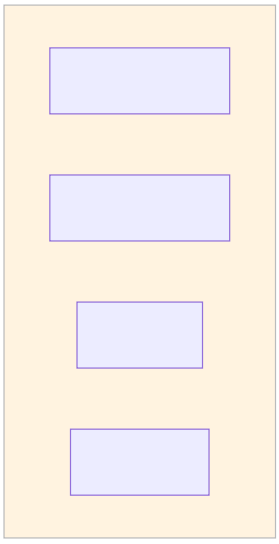
## 9. Architectural Assessment CLEAN

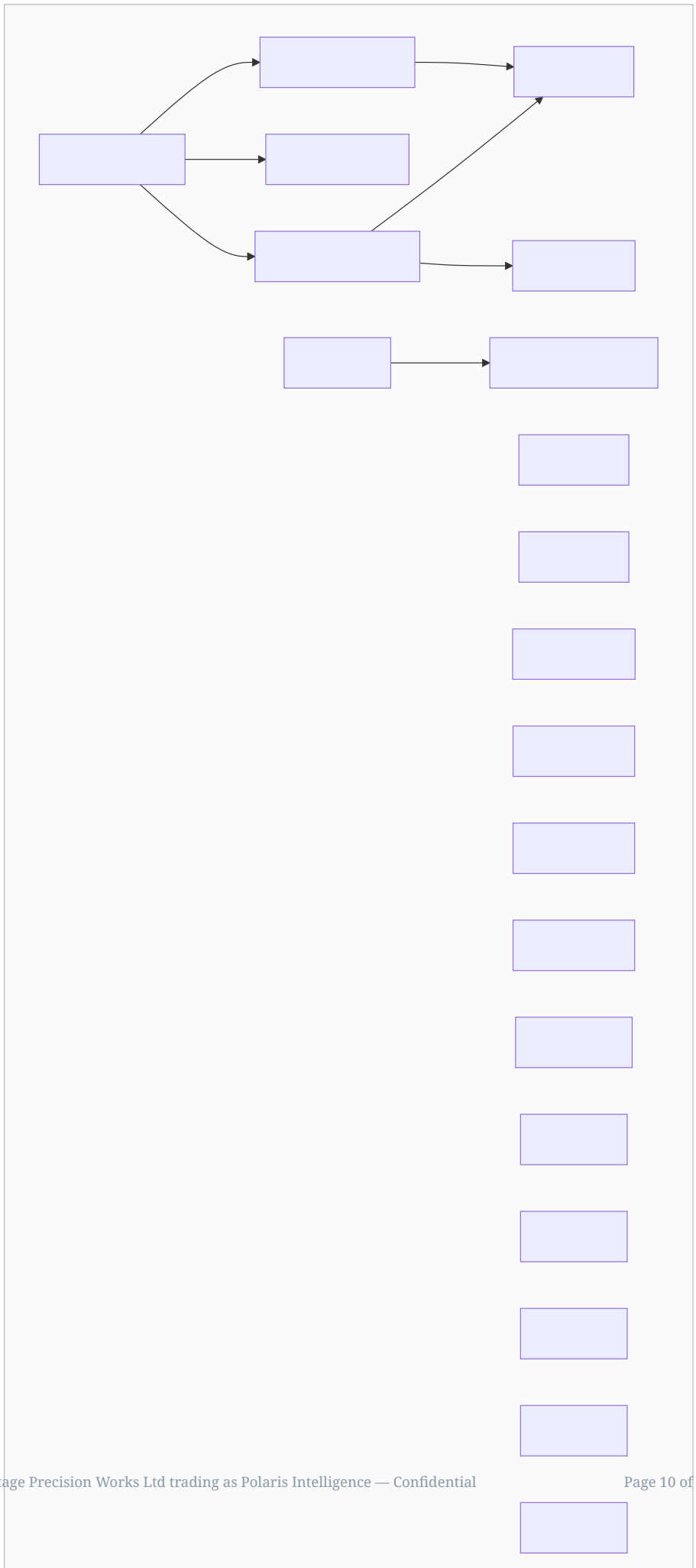
50 production modules, 22 internal dependencies, 52 external imports.

21 isolated modules detected (no internal imports or dependents). These are typically standalone utilities, examples, platform-specific implementations, or generated code.

### Module Dependency Map

*Modules are grouped by architectural layer and coloured accordingly. Arrows show import dependencies. Numbers in parentheses indicate how many other modules depend on each file. Thick red borders highlight high fan-in modules (change risk).*





## Layer Distribution

*Modules are grouped by their role in the application: business logic (core functionality), presentation (user-facing), data (storage), and test. A well-structured codebase separates these concerns clearly.*

| LAYER | MODULES |
|---|---|
| **other** | 34 |
| **presentation** | 8 |
| **infrastructure** | 8 |

## Entry Points (25)

| LOCATION | PATTERN |
|---|---|
| `lib/request.js:231` | `app.get(` |
| `lib/application.js:353` | `app.get(` |
| `lib/response.js:161` | `app.get(` |
| `examples/error/index.js:10` | `app.get(` |
| `examples/error-pages/index.js:30` | `app.get(` |
| `examples/online/index.js:50` | `app.get(` |
| `examples/params/index.js:47` | `app.get(` |
| `examples/search/index.js:52` | `app.get(` |
| `examples/multi-router/index.js:10` | `app.get(` |
| `examples/route-middleware/index.js:70` | `app.get(` |

## Isolated Modules (21)

*These files neither use nor are used by any other file in the codebase. They may be unused code, standalone utilities, or components loaded indirectly. Each should be reviewed to confirm it is genuinely needed.*

- `examples/auth/index.js`
- `examples/cookie-sessions/index.js`
- `examples/cookies/index.js`
- `examples/downloads/index.js`
- `examples/ejs/index.js`
- `examples/error-pages/index.js`

- `examples/error/index.js`
- `examples/hello-world/index.js`
- `examples/markdown/index.js`
- `examples/mvc/controllers/main/index.js`
- `examples/online/index.js`
- `examples/params/index.js`
- `examples/resource/index.js`
- `examples/search/index.js`
- `examples/search/public/client.js`

## 9b. Architecture Topology

*This section applies graph-theoretic analysis to the module dependency network identified in the Architectural Assessment. Community detection reveals natural subsystem boundaries; centrality analysis identifies critical bridge modules whose failure or refactoring would disproportionately affect the codebase.*

| METRIC | VALUE |
|---|---|
| **Modules analysed** | 50 |
| **Internal dependencies** | 22 |
| **Subsystems detected** | 8 |
| **Modularity score** | 0.773 (strong modular structure) |
| **Unclustered modules** | 21 |

### Subsystem Overview

*Communities are groups of modules that are more tightly connected to each other than to the rest of the codebase. Each represents a natural subsystem boundary.*

| SUBSYSTEM | MODULES |
|---|---|
| **examples/view-constructor/ (other, 2 modules)** | 2 |
| **examples/mvc/ (other, 2 modules)** | 2 |
| **lib/ (infrastructure, 9 modules)** | 9 |
| **examples/mvc/ (presentation, 4 modules)** | 4 |
| **examples/content-negotiation/ (other, 3 modules)** | 3 |

| SUBSYSTEM | MODULES |
|---|---|
| **examples/route-separation/ (other, 4 modules)** | 4 |
| **examples/view-locals/ (other, 2 modules)** | 2 |
| **examples/multi-router/ (presentation, 3 modules)** | 3 |

### Structural Gaps

*Subsystem pairs with very few connecting dependencies that may indicate missing integration.*

| SUBSYSTEM A | SUBSYSTEM B | CONNECTING EDGES |
|---|---|---|
| **examples/view-constructor/ (other, 2 modules)** | examples/mvc/ (other, 2 modules) | 0 |
| **examples/view-constructor/ (other, 2 modules)** | lib/ (infrastructure, 9 modules) | 0 |
| **examples/view-constructor/ (other, 2 modules)** | examples/mvc/ (presentation, 4 modules) | 0 |
| **examples/view-constructor/ (other, 2 modules)** | examples/content-negotiation/ (other, 3 modules) | 0 |
| **examples/view-constructor/ (other, 2 modules)** | examples/route-separation/ (other, 4 modules) | 0 |
| **examples/view-constructor/ (other, 2 modules)** | examples/view-locals/ (other, 2 modules) | 0 |
| **examples/view-constructor/ (other, 2 modules)** | examples/multi-router/ (presentation, 3 modules) | 0 |
| **examples/mvc/ (other, 2 modules)** | lib/ (infrastructure, 9 modules) | 0 |
| **examples/mvc/ (other, 2 modules)** | examples/mvc/ (presentation, 4 modules) | 0 |
| **examples/mvc/ (other, 2 modules)** | examples/content-negotiation/ (other, 3 modules) | 0 |

## 10. Test Coverage CLEAN

*Automated tests act as safety nets for the codebase. When developers make changes, tests verify nothing else has broken. Strong test coverage means the acquirer's team can modify and extend the code with confidence; weak or absent testing means changes carry a higher risk of introducing undetected problems.*

**Strong testing discipline — automated tests cover a significant portion of the codebase and run automatically before changes are released.**

| METRIC | VALUE |
|---|---|
| **Test files** | 91 |
| **Source files** | 141 |
| **Test-to-source ratio** | 64.5% |
| **Test functions / cases** | 1684 |

This is a healthy ratio, indicating testing is a routine part of development.

> **Tests run automatically before code changes are accepted. This means new code is verified before it reaches the main codebase, reducing the chance of regressions.**

CI configuration found in: .github/workflows/scorecard.yml, .github/workflows/codeql.yml, .github/workflows/legacy.yml, .github/workflows/ci.yml

## 11. Infrastructure & Deployment CLEAN

*This section assesses whether the deployment process — how the software is built, packaged, and released — is documented in code or relies on undocumented manual steps. Codified deployment means a new team can operate the software independently; undocumented deployment creates dependency on the original developers and increases transition risk.*

**Limited build automation detected. Package publication may rely on manual steps.**

### Infrastructure Found

| CATEGORY | TOOLS / CONFIGURATION |
|---|---|
| **CI/CD Automation** | github-actions (4) |

> **As a library distributed via package registry, containerisation, orchestration, and deployment infrastructure are not expected. CI/CD for build and publish is the primary infrastructure concern.**

## 12. Technical Debt CLEAN

*Technical debt represents shortcuts, deferred work, and known problems that the development team has acknowledged but not yet fixed. Every codebase carries some debt; what matters is the volume and severity. High technical debt increases the cost of post-acquisition development and the risk that changes introduce new problems.*

**Minimal technical debt — the codebase is well-maintained with few acknowledged shortcuts or deferred work items.**

Developers have left **2** notes in the code flagging work that needs to be done (TODO items, known bugs, temporary workarounds). That is **0.1 markers per 1,000 lines of code**. This density is typical for a well-maintained project.

### Debt Markers by Type

| TYPE | COUNT |
|---|---|
| **TODO** | 2 |

## 13. Governance & CI/CD Security LOW RISK

*Governance measures the security and maturity of development processes — CI pipeline hardening, release signing, code review enforcement, and dependency management. Weak governance increases post-acquisition remediation costs and ongoing operational risk.*

Overall governance: B (7.9/10) [OpenSSF Scorecard + local analysis]

OpenSSF Scorecard aggregate: 8.2/10

*Individual check scores below reflect Scorecard's automated assessment. Low scores on checks such as branch protection, signed releases, or code review are common for open-source projects and do not necessarily indicate a governance risk — the overall rating accounts for the project's classification and context.*

### CI Pipeline Security — Grade A

| CHECK | SCORE | SOURCE | DETAIL |
|---|---|---|---|
| **Token-Permissions** | 10/10 | scorecard | GitHub workflow tokens follow principle of least privilege |
| **Pinned-Dependencies** | 6/10 | scorecard | dependency not pinned by hash detected -- score normalized to 6 |
| **Dangerous-Workflow** | 10/10 | scorecard | no dangerous workflow patterns detected |

| CHECK | SCORE | SOURCE | DETAIL |
|---|---|---|---|
| **CI-Tests** | 10/10 | scorecard | 30 out of 30 merged PRs checked by a CI test -- score normalized to 10 |
| **SAST** | 10/10 | scorecard | SAST tool is run on all commits |

CI pipelines demonstrate strong security practices.

### Release Engineering — Grade A

| CHECK | SCORE | SOURCE | DETAIL |
|---|---|---|---|
| **Signed-Releases** | 3/10 | scorecard | Releases exist (3.7/yr via GitHub tags) but are not cryptographically signed. Score reflects absence of signing, not absence of releases. |
| **Packaging** | N/A | scorecard | packaging workflow not detected |
| **Maintained** | 10/10 | scorecard | 30 commits and 9 issue activity found in the last 90 days -- score normalized to 10 |
| **release_frequency** | 3/10 | enrichment | 3.7 releases/year, 100% semver compliant |

Release engineering practices are mature.

### Governance Posture — Grade A

| CHECK | SCORE | SOURCE | DETAIL |
|---|---|---|---|
| **Security-Policy** | 10/10 | scorecard | security policy file detected |
| **Contributors** | 10/10 | scorecard | project has 116 contributing companies or organizations |
| **CII-Best-Practices** | 0/10 | scorecard | no effort to earn an OpenSSF best practices badge detected |
| **License** | 10/10 | scorecard | license file detected |

Project demonstrates strong governance practices.

### Branch Protection & Code Review — Grade C

| CHECK | SCORE | SOURCE | DETAIL |
|---|---|---|---|
| **Branch-Protection** | 1/10 | scorecard | branch protection is not maximal on development and all release branches |

| CHECK | SCORE | SOURCE | DETAIL |
|---|---|---|---|
| **Code-Review** | 9/10 | scorecard | Found 18/19 approved changesets -- score normalized to 9 |

Branch protection has gaps — review requirements may be bypassable.

### Dependency Management — Grade B

| CHECK | SCORE | SOURCE | DETAIL |
|---|---|---|---|
| **Dependency-Update-Tool** | 10/10 | scorecard | update tool detected |
| **Vulnerabilities** | 10/10 | scorecard | 0 existing vulnerabilities detected |
| **Fuzzing** | 0/10 | scorecard | project is not fuzzed |

Dependency management practices are adequate.

## 14. Engineering Maturity LOW RISK

*Engineering maturity measures the project's operational health beyond source code quality — release discipline, community governance, and project signals that indicate long-term viability.*

Overall maturity: **B** — minor maturity gaps only (risk rating: LOW)

### Release Cadence

| METRIC | VALUE |
|---|---|
| **Releases per year** | 3.7 |
| **Days since last release** | 122 |
| **Semver compliance** | 100.0% |
| **Grade** | B |

### Community Health

| DOCUMENT | STATUS |
|---|---|
| **SECURITY.md** | Present |

| DOCUMENT | STATUS |
|---|---|
| **CONTRIBUTING.md** | Present |
| **CODE_OF_CONDUCT** | Present |
| **Issue template** | Missing |
| **PR template** | Present |
| **Health score** | 80% |
| **Grade** | B |

### Project Signals

| METRIC | VALUE |
|---|---|
| **Stars** | 68,891 |
| **Forks** | 23,022 |
| **Open issues** | 208 |
| **Contributors** | 100 |
| **Repository created on GitHub** | 2009-06-26 |
| **Grade** | A |

***Note:*** *GitHub API reports 100 contributors while git history analysis (Bus Factor section) identified 30. This discrepancy arises because GitHub counts all commit authors across the full history, while git analysis may use a limited clone depth or different author-deduplication rules.*

## 15. Malware & Destructive Action Scan CLEAN

*This scan searches for code patterns commonly associated with protestware, supply-chain attacks, and sabotage — filesystem wipers, obfuscated payloads, unauthorised network calls, and install-hook abuse. Findings are heuristic and warrant manual review rather than automatic condemnation.*

Files scanned: 142

**No suspicious patterns detected.**

## 16. Risk Summary & Recommendations

*Recommendations are prioritised by their potential impact on the transaction. Immediate and Urgent items should be addressed as conditions precedent; Medium items can be scheduled into the post-acquisition integration roadmap.*

| PRIORITY | CATEGORY | RECOMMENDED ACTION |
|---|---|---|
| MEDIUM RISK | Dependency Health | 16 abandoned or deprecated dependencies. Evaluate alternatives to reduce latent supply chain risk. |

## Appendix A: Raw Data

### Dependency Inventory (44 packages)

| ECOSYSTEM | PACKAGE | VERSION | VULNS | HEALTH | LAST RELEASE |
|---|---|---|---|---|---|
| npm | accepts | 2.0.0 | 0 | unknown | 2022-02-02 |
| npm | body-parser | 2.2.1 | 0 | active | 2026-01-07 |
| npm | content-disposition | 1.0.0 | 0 | active | 2025-11-18 |
| npm | content-type | 1.0.5 | 0 | abandoned | 2023-01-29 |
| npm | cookie | 0.7.1 | 0 | active | 2025-11-26 |
| npm | cookie-signature | 1.2.1 | 0 | stale | 2024-10-29 |
| npm | debug | 4.4.0 | 0 | stable | 2025-09-13 |
| npm | depd | 2.0.0 | 0 | abandoned | 2018-10-26 |
| npm | encodeurl | 2.0.0 | 0 | abandoned | 2024-03-29 |
| npm | escape-html | 1.0.3 | 0 | abandoned | 2015-09-01 |
| npm | etag | 1.8.1 | 0 | abandoned | 2017-09-13 |
| npm | finalhandler | 2.1.0 | 0 | active | 2025-12-01 |
| npm | fresh | 2.0.0 | 0 | unknown | 2017-09-14 |
| npm | http-errors | 2.0.0 | 0 | active | 2025-11-20 |
| npm | merge-descriptors | 2.0.0 | 0 | abandoned | 2023-11-16 |
| npm | mime-types | 3.0.0 | 0 | active | 2025-11-20 |
| npm | on-finished | 2.4.1 | 0 | abandoned | 2022-02-22 |
| npm | once | 1.4.0 | 0 | abandoned | 2016-09-06 |
| npm | parseurl | 1.3.3 | 0 | abandoned | 2019-04-16 |
| npm | proxy-addr | 2.0.7 | 0 | abandoned | 2021-06-01 |
| npm | qs | 6.14.2 | 0 | active | 2026-02-15 |
| npm | range-parser | 1.2.1 | 0 | abandoned | 2019-05-11 |

| ECOSYSTEM | PACKAGE | VERSION | VULNS | HEALTH | LAST RELEASE |
|---|---|---|---|---|---|
| npm | router | 2.2.0 | 0 | stale | 2025-03-27 |
| npm | send | 1.1.0 | 0 | active | 2025-12-15 |
| npm | serve-static | 2.2.0 | 0 | active | 2025-12-15 |
| npm | statuses | 2.0.1 | 0 | stable | 2025-06-06 |
| npm | type-is | 2.0.1 | 0 | stale | 2025-03-27 |
| npm | vary | 1.1.2 | 0 | abandoned | 2017-09-24 |
| npm | after | 0.8.2 | 0 | abandoned | 2016-08-16 |
| npm | connect-redis | 8.0.1 | 0 | stable | 2025-06-14 |
| npm | cookie-parser | 1.4.7 | 0 | stale | 2024-10-08 |
| npm | cookie-session | 2.1.1 | 0 | stable | 2025-07-17 |
| npm | ejs | 3.1.10 | 0 | active | 2026-03-05 |
| npm | eslint | 8.47.0 | 0 | active | 2026-03-20 |
| npm | express-session | 1.18.1 | 0 | active | 2026-01-22 |
| npm | hbs | 4.2.0 | 0 | active | 2026-04-01 |
| npm | marked | 15.0.3 | 0 | active | 2026-03-20 |
| npm | method-override | 3.0.0 | 0 | abandoned | 2018-07-12 |
| npm | mocha | 10.7.3 | 0 | active | 2025-11-05 |
| npm | morgan | 1.10.1 | 0 | stable | 2025-07-17 |
| npm | nyc | 17.1.0 | 0 | active | 2026-02-22 |
| npm | pbkdf2-password | 1.2.1 | 0 | abandoned | 2016-09-29 |
| npm | supertest | 6.3.0 | 0 | active | 2026-01-06 |
| npm | vhost | 3.0.2 | 0 | abandoned | 2015-10-13 |

## Module Dependencies (top 30)

| MODULE | FAN-IN | LAYER |
|---|---|---|
| `lib/express.js` | 3 | infrastructure |
| `examples/mvc/db.js` | 3 | other |
| `examples/content-negotiation/db.js` | 2 | other |
| `lib/utils.js` | 2 | infrastructure |
| `lib/request.js` | 1 | infrastructure |
| `lib/view.js` | 1 | infrastructure |
| `examples/mvc/lib/boot.js` | 1 | infrastructure |
| `lib/application.js` | 1 | infrastructure |
| `lib/response.js` | 1 | infrastructure |

| MODULE | FAN-IN | LAYER |
|---|---|---|
| `examples/view-locals/user.js` | 1 | other |
| `examples/multi-router/controllers/api_v2.js` | 1 | presentation |
| `examples/route-separation/user.js` | 1 | other |
| `examples/route-separation/post.js` | 1 | other |
| `examples/view-constructor/github-view.js` | 1 | other |
| `examples/multi-router/controllers/api_v1.js` | 1 | presentation |
| `examples/route-separation/site.js` | 1 | other |
| `examples/search/index.js` | 0 | other |
| `examples/static-files/public/js/app.js` | 0 | presentation |
| `examples/static-files/index.js` | 0 | other |
| `examples/view-constructor/index.js` | 0 | other |
| `examples/params/index.js` | 0 | other |
| `examples/search/public/client.js` | 0 | presentation |
| `examples/cookies/index.js` | 0 | other |
| `examples/hello-world/index.js` | 0 | other |
| `examples/session/index.js` | 0 | other |
| `examples/route-map/index.js` | 0 | other |
| `examples/downloads/index.js` | 0 | other |
| `index.js` | 0 | other |
| `examples/vhost/index.js` | 0 | other |
| `examples/web-service/index.js` | 0 | other |