

disposable-email-domains

github.com/amieiro/disposable-email-domains

Scanned 24 September 2026 at 10:18 · Report generated 29 October 2026 · PHP · 90 stars

OVERALL ASSESSMENT

Medium

Combined score 25 · 10 findings after filtering

Summary

This repository has been flagged as having low overall vulnerability risk, but it shows signs of potential malicious activity that should be treated with caution. The most serious issue is the use of non-constant commands in code, which could allow attackers to inject harmful instructions and execute arbitrary system commands. Additionally, there are patterns that match known cryptocurrency mining tools, suggesting the project might be connected to or attempting to integrate with crypto-mining pools. While these findings indicate a higher level of malicious intent than typical insecure coding, they do not necessarily mean the code is actively dangerous right now. The developer should review the specific files mentioned in the reports to understand exactly how these patterns are being used before deciding whether to proceed with using this codebase.

Written by a language model running locally on the scanning machine, from the findings in this report. The risk levels above and below were decided by the scoring model, not by this summary.

Risk, by type

MalDet scores two different questions separately, because they call for different responses: code can be badly written without being hostile, and hostile code is not always badly written.

VULNERABILITY

Low (18)

Insecure coding an attacker could exploit. Usually mistakes, and usually fixable in place.

MALICIOUS PATTERN

Low (7)

Signs the code may be trying to do harm: hidden payloads, credential access, persistence, exfiltration.

What each detector found

DETECTOR	LOOKS FOR	HIGH	MEDIUM	LOW	TOTAL
semgrep	Insecure coding patterns across many languages	2	6	0	8

DETECTOR	LOOKS FOR	HIGH	MEDIUM	LOW	TOTAL
yara	Signatures of malicious behaviour in source files	2	0	0	2

6 of 10 findings were rated more likely genuine than noise by the classifier.

Evidence

Identical findings are grouped, so a single rule firing many times takes one row. Ordered worst first. “Seen” is how many times that finding occurred.

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
HIGH	semgrep	Executing non-constant commands. This can lead to command injection.	2	/creator/app/Console/Commands/CreateDisposableEmailDomainsFilesCommand.php:565
HIGH	yara	YARA rule matched: detect_crypto_miner — Cryptocurrency mining pool connection patterns	2	/denyDomains.json
MEDIUM	semgrep	GitHub Actions step uses a mutable tag or branch reference. Tags and branch names can be silently repointed by the action owner, enabling supply-chain attacks — as seen in the trivy-action and kics-github-action compromises. Pin the reference to a full 40-character commit SHA instead, e.g. <code>`uses: ac</code>	6	/.github/workflows/run-tests.yml:54

How to read this report

- **This is a static scan.** The code was read, never executed. Behaviour that only appears at runtime is outside what any of these detectors can see.
- **Findings are not all equally real.** A classifier rates each finding on how likely it is to be genuine rather than noise, and that rating weights the scores. Measured against a hand-reviewed sample it reached precision 0.52 and recall 0.62 — useful for ranking, not a verdict on any single finding.
- **A high score is not proof of malice.** Security tools, penetration-testing frameworks and malware research repositories contain the same patterns as real malware by design. The two risk types above exist to keep that distinction visible.
- **Absence of findings is not proof of safety.** A detector that is not installed, or a language none of them parse, produces silence rather than a warning.
- **This scan is a snapshot.** The repository may have changed since 24 September 2026.