

apartment

github.com/rails-on-services/apartment

Scanned 24 September 2026 at 10:11 · Report generated 29 October 2026 · Ruby · 512 stars

OVERALL ASSESSMENT

Medium

Combined score 53 · 22 findings after filtering

Summary

The repository has a medium vulnerability score of 53.0, which indicates that some code contains insecure patterns an attacker could exploit, but the scan did not find any signs of intentional malicious behavior. The most critical issues are related to missing security features like cross-site request forgery protection and mass assignment limits in your controllers and models. While one finding suggests a manual ERB template was created, this is likely just standard development practice rather than a deliberate attack attempt. Additionally, the code does not use the latest dependency updates with a cooldown period, which could theoretically introduce new risks from unstable packages. Overall, these are typical coding mistakes that should be fixed to improve security, but they do not suggest the project is currently compromised or trying to harm others.

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

Medium (53)

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

MALICIOUS PATTERN

Safe (0)

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	19	0	21

DETECTOR	LOOKS FOR	HIGH	MEDIUM	LOW	TOTAL
yara	Signatures of malicious behaviour in source files	1	0	0	1

21 of 22 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	Detected controller which does not enable cross-site request forgery protections using 'protect_from_forgery'. Add 'protect_from_forgery :with => :exception' to your controller class.	1	/spec/dummy/app/controllers/application_controller.rb:3
HIGH	semgrep	Checks for models that do not use attr_accessible. This means there is no limiting of which variables can be manipulated through mass assignment. For newer Rails applications, parameters should be allowlisted using strong parameters. For older Rails versions, they should be allowlisted using strong_	1	/spec/unit/adapters/abstract_adapter_spec.rb:8
HIGH	yara	YARA rule matched: detect_persistence_mechanism — Modifies a startup/scheduling location - legitimate application code essentially never does this	1	/README.md
MEDIUM	semgrep	This Dependabot configuration does not set a cooldown period. Newly published packages can be malicious or unstable. Add a `cooldown` block with `default-days: 7` to each `package-ecosystem` entry under `updates` to wait 7 days before proposing updates to newly published package versions. Reference:	2	/.github/dependabot.yml:12
MEDIUM	semgrep	Found that the setting for providing detailed exception reports in Rails is set to true. This can lead to information exposure, where sensitive system or internal information is displayed to the end user. Instead, turn this setting off.	1	/spec/dummy/config/environments/test.rb:6

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	Detected manual creation of an ERB template. Manual creation of templates may expose your application to server-side template injection (SSTI) or cross-site scripting (XSS) attacks if user input is used to create the template. Instead, create a '.erb' template file and use 'render'.	1	/spec/integration/v4/support.rb:173
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. `uses: ac	15	/.github/workflows/ci.yml:62

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.