

home-generative-agent

github.com/goruck/home-generative-agent

Scanned 24 September 2026 at 09:58 · Report generated 29 October 2026 · Python · 308 stars

OVERALL ASSESSMENT

Medium

Combined score 61 · 37 findings after filtering

Summary

This repository has a medium vulnerability score of 61.0, which indicates that there are several insecure coding patterns in the code that an attacker could potentially exploit. The scan found two services running with writable root filesystems, meaning they could theoretically be used to download or modify files if someone gains access to them. Additionally, the code contains multiple hardcoded passwords and secrets that should never be stored in plain text, which is a common mistake in insecure coding rather than evidence of malicious intent. While the scan did not find any signs of intentional malware, these security issues need to be addressed before the code can be considered safe for production use.

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 (61)

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	0	24	0	24

DETECTOR	LOOKS FOR	HIGH	MEDIUM	LOW	TOTAL
bandit	Insecure coding patterns in Python source	0	0	13	13

36 of 37 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
MEDIUM	semgrep	Service 'pgvector_db' is running with a writable root filesystem. This may allow malicious applications to download and run additional payloads, or modify container files. If an application inside a container has to save something temporarily consider using a tmpfs. Add 'read_only: true' to this ser	1	/.devcontainer/docker-compose.yml:17
MEDIUM	semgrep	Service 'devcontainer' is running with a writable root filesystem. This may allow malicious applications to download and run additional payloads, or modify container files. If an application inside a container has to save something temporarily consider using a tmpfs. Add 'read_only: true' to this se	1	/.devcontainer/docker-compose.yml:3
MEDIUM	semgrep	Detected a python logger call with a potential hardcoded secret "Token count for %s (approx fallback): %d" being logged. This may lead to secret credentials being exposed. Make sure that the logger is not logging sensitive information.	1	/custom_components/home_generative_agent/agent/token_counter.py:351
MEDIUM	semgrep	Detected a python logger call with a potential hardcoded secret "Token counts from metadata: %s" being logged. This may lead to secret credentials being exposed. Make sure that the logger is not logging sensitive information.	1	/custom_components/home_generative_agent/agent/graph.py:2779
MEDIUM	semgrep	Service 'pgvector_db' allows for privilege escalation via setuid or setgid binaries. Add 'no-new-privileges:true' in 'security_opt' to prevent this.	1	/.devcontainer/docker-compose.yml:17

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	Service 'devcontainer' allows for privilege escalation via setuid or setgid binaries. Add 'no-new-privileges:true' in 'security_opt' to prevent this.	1	/.devcontainer/docker-compose.yaml:3
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.yaml:4
MEDIUM	semgrep	Detected a python logger call with a potential hardcoded secret "Exact token counting for %s failed; using approximate counts for the " "next %.0f seconds. Conversation continues. Cause: %s" being logged. This may lead to secret credentials being exposed. Make sure that the logger is not	1	/custom_components/home_generative_agent/agent/token_counter.py:65
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	14	/.github/workflows/validate.yaml:23
MEDIUM	semgrep	Untrusted user input in `importlib.import_module()` function allows an attacker to load arbitrary code. Avoid dynamic values in `importlib.import_module()` or use a whitelist to prevent running untrusted code.	1	/custom_components/home_generative_agent/helpers.py:93
LOW	bandit	Possible hardcoded password: 'ha_password'	1	/custom_components/home_generative_agent/const.py:100

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
LOW	bandit	Possible hardcoded password: 'normal'	1	/ custom_components/home_generative_agent/sentinel/auth_inventor.py:65
LOW	bandit	Possible hardcoded password: 'system'	1	/ custom_components/home_generative_agent/sentinel/auth_inventor.py:66
LOW	bandit	Possible hardcoded password: 'type_ha_new_admin_or_token'	1	/ custom_components/home_generative_agent/sentinel/notifier.py:794
LOW	bandit	Possible hardcoded password: 'type_ha_long_lived_token_stale'	1	/ custom_components/home_generative_agent/sentinel/notifier.py:795
LOW	bandit	Possible hardcoded password: 'long_lived_access_token'	1	/ custom_components/home_generative_agent/sentinel/auth_inventor.py:67
LOW	bandit	Possible hardcoded password: 'sentinel_ha_token_stale_days'	1	/ custom_components/home_generative_agent/const.py:390

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
LOW	bandit	Possible hardcoded password: 'Starý dlouhodobý přístupový token'	1	/ custom_components/home_generative_agent/sentinel/notifier_messages.py:143
LOW	bandit	Possible hardcoded password: 'Nový administrátor nebo přístupový token'	1	/ custom_components/home_generative_agent/sentinel/notifier_messages.py:142
LOW	bandit	Possible hardcoded password: 'Old long-lived access token'	2	/ custom_components/home_generative_agent/sentinel/notifier_messages.py:61
LOW	bandit	Possible hardcoded password: 'New administrator or access token'	2	/ custom_components/home_generative_agent/sentinel/notifier_messages.py:60

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.