

PMAT-labs

github.com/HuskyHacks/PMAT-labs

Scanned 18 September 2026 at 19:54 · Report generated 29 October 2026 · HCL · 1,158 stars

OVERALL ASSESSMENT

Critical

Combined score 349 · 44 findings after filtering

Summary

The repository shows two very different kinds of problems. The most serious issue is that the code contains malware signatures, including ransomware and trojans, which means the software could be used to steal data or damage systems if someone runs it. These malicious patterns are flagged by security tools with a critical score of 346.0, indicating a high likelihood that the code was intentionally created for harm rather than being a mistake. On the other hand, the developer's own codebase has only a low vulnerability risk score of 3.0, meaning it does not contain obvious insecure coding practices that an attacker could exploit from outside. This suggests that while the external code is dangerous, the internal logic used to generate or analyze it appears safe and follows standard security guidelines. In practice, you should treat this repository as unsafe for any production use because of the clear evidence of malicious intent in its source files.

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 (3)

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

MALICIOUS PATTERN

Critical (346)

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
clamav	Known malware signatures	33	0	0	33
yara	Signatures of malicious behaviour in source files	10	0	0	10
semgrep	Insecure coding patterns across many languages	0	1	0	1

34 of 44 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	clamav	ClamAV detected: Win.Trojan.MSShellcode-6360728-0	24	/labs/ 5-1.Automation/ BlueJupyter/ FORTRIAGE_extr acted/out25.exe
HIGH	clamav	ClamAV detected: Win.Trojan.MSShellcode-7	4	/labs/ 1-1.BasicStati cAnalysis/ Malware.Packed AndNotPacked.e xe.malz/ Malware.Packed AndNotPacked.e xe_extracted/ Malware.Packed AndNotPacked.e xe/ Malware.NotPac ked.exe.malz
HIGH	clamav	ClamAV detected: Doc.Downloader.Pwshell-10001336-0	1	/labs/ 3-1.GonePhishi ng- MaldocAnalysis/ Word/docm/ bookReport_ext racted/ bookReport.docm
HIGH	clamav	ClamAV detected: Eicar-Signature	1	/labs/ 5-1.Automation/ BlueJupyter/ FORTRIAGE_extr acted/out19.exe

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
HIGH	clamav	ClamAV detected: Win.Ransomware.Wanna-9769986-0	1	/labs/ 4-1.Bossfight- wannacry.exe/ Ransomware.wan nacry.exe.malz_ extracted/ Ransomware.wan nacry.exe.malz
HIGH	clamav	ClamAV detected: Win.Trojan.MSShellcode-6360730-0	1	/labs/ 5-1.Automation/ BlueJupyter/ FORTRIAGE_extr acted/out14.exe
HIGH	clamav	ClamAV detected: Win.Trojan.Swrort-5710536-0	1	/labs/ 1-1.BasicStati cAnalysis/ Malware.Packed AndNotPacked.e xe.malz/ Malware.Packed AndNotPacked.e xe_extracted/ Malware.Packed AndNotPacked.e xe/ Malware.Packed. exe.malz
HIGH	yara	YARA rule matched: detect_process_injection — Windows process-injection API sequence	2	/labs/ 2-1.AdvancedSt aticAnalysis/ Malware.stage0. exe.malz/ Malware.stage0. exe.malz_extra cted/ Malware.stage0. exe.malz
HIGH	yara	YARA rule matched: detect_anti_analysis — Checks for a debugger, VM, or sandbox before continuing - a common malware evasion technique, essentially never present in ordinary application code	8	/labs/ 2-1.AdvancedSt aticAnalysis/ Dropper.Downlo adFromURL.exe. malz/ Dropper.Downlo adFromURL.exe_ extracted/ Dropper.Downlo adFromURL.exe. malz

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	Resources in the AWS subnet are assigned a public IP address. Resources should not be exposed on the public internet, but should have access limited to consumers required for the function of your application. Set <code>`map_public_ip_on_launch`</code> to false so that resources are not publicly-accessible.	1	/aws-lab/network.tf:21

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 18 September 2026.