

loki

github.com/grafana/loki

Scanned 24 September 2026 at 07:08 · Report generated 29 October 2026 · Go · 28,940 stars

OVERALL ASSESSMENT

Critical

Combined score 535 · 337 findings after filtering

Summary

This repository has two major security issues that are serious enough to prevent safe use. The most critical problem is that the code contains severe insecure coding patterns, specifically missing mutex locks and unencrypted gRPC connections, which means attackers could easily crash the system or steal data without any encryption. While there are also signs of potential malicious intent, such as suspicious URLs and obfuscated strings, these do not appear to be intentional attacks but rather common mistakes found in vendor code. The scan reported a vulnerability risk score of 514.0, which indicates a high level of danger due to the critical security flaws present in the application logic.

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 Critical (514) Insecure coding an attacker could exploit. Usually mistakes, and usually fixable in place.	MALICIOUS PATTERN Medium (21) 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	24	225	26	275
yara	Signatures of malicious behaviour in source files	35	0	18	53

DETECTOR	LOOKS FOR	HIGH	MEDIUM	LOW	TOTAL
guarddog	Malicious package and install-time behaviour	0	3	3	6
bandit	Insecure coding patterns in Python source	0	0	3	3

180 of 337 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	Found an insecure gRPC server without 'grpc.Creds()' or options with credentials. This allows for a connection without encryption to this server. A malicious attacker could tamper with the gRPC message, which could compromise the machine. Include credentials derived from an SSL certificate in order	1	/pkg/logql/ internal/ logqltest/ exec_query_f rontend.go:99
HIGH	semgrep	Missing mutex unlock (`m.mut` variable) before returning from a function. This could result in panics resulting from double lock operations	1	/pkg/ruler/ storage/ instance/ manager.go:343
HIGH	semgrep	Missing mutex unlock (`r.userManagerMtx` variable) before returning from a function. This could result in panics resulting from double lock operations	1	/pkg/ruler/ base/ manager.go:282
HIGH	semgrep	Missing mutex unlock (`o.mu` variable) before returning from a function. This could result in panics resulting from double lock operations	2	/pkg/engine/ internal/ obslock/ obslock.go:122
HIGH	semgrep	Missing `RUnlock` on an `RWMutex` (`m.mtx` variable) lock before returning from a function	1	/pkg/storage/ stores/ shipper/ indexshipper/ downloads/ util.go:68

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
HIGH	semgrep	A <code>`run:`</code> step pipes the output of <code>`curl`</code> or <code>`wget`</code> directly into a shell interpreter. This is the "curl bash" install pattern — if the remote server is compromised or the URL is hijacked, an attacker can execute arbitrary code in your CI runner. Consider downloading the file first, verifying its c	1	<code>/.github/actions/setup-k3d/action.yaml:10</code>
HIGH	semgrep	Untrusted input could be used to tamper with a web page rendering, which can lead to a Cross-site scripting (XSS) vulnerability. XSS vulnerabilities occur when untrusted input executes malicious JavaScript code, leading to issues such as account compromise and sensitive information leakage. To preve	1	<code>/pkg/ingester/ingester.go:984</code>
HIGH	semgrep	Using variable interpolation <code>`\${{...}}`</code> with <code>`github`</code> context data in a <code>`run:`</code> step could allow an attacker to inject their own code into the runner. This would allow them to steal secrets and code. <code>`github`</code> context data can have arbitrary user input and should be treated as untrusted. Instead, use	3	<code>/.github/workflows/logql-bench.yml:70</code>
HIGH	semgrep	Generic Secret detected	2	<code>/operator/hack/addon_grafana_gateway_oauth.yaml:188</code>
HIGH	semgrep	bcrypt hash detected	2	<code>/production/helm/loki/values.yaml:399</code>
HIGH	semgrep	By not specifying a USER, a program in the container may run as 'root'. This is a security hazard. If an attacker can control a process running as root, they may have control over the container. Ensure that the last USER in a Dockerfile is a USER other than 'root'.	9	<code>/clients/cmd/fluent-bit/Dockerfile:24</code>

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
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	4	/vendor/modernc.org/libc/capi_windows_arm64.go
HIGH	yara	YARA rule matched: detect_download_and_execute — Downloads a remote resource AND executes it - the actually dangerous combination, not just downloading alone	10	/vendor/github.com/sahilm/fuzzy/Makefile
HIGH	yara	YARA rule matched: detect_obfuscated_string_execution — Reconstructs a string character-by-character or from hex escapes, then immediately executes it - fromCharCode/chr() alone are common in ordinary text-processing code, so this requires both the reconstruction and an execution call together	19	/vendor/modernc.org/libc/ccgo_linux_ppc64le.go
HIGH	yara	YARA rule matched: detect_persistence_mechanism — Modifies a startup/scheduling location - legitimate application code essentially never does this	2	/vendor/github.com/alecthomas/chroma/v2/lexers/embedded/bash.xml
MEDIUM	semgrep	Service 'docs' 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 service to	1	/cmd/logql-analyzer/docker-compose.yaml:11
MEDIUM	semgrep	Variable `version` is likely modified and later used on error. In some cases this could result in panics due to a nil dereference	1	/pkg/querier/queryrange/querystarding.go:181
MEDIUM	semgrep	Variable `resp` is likely modified and later used on error. In some cases this could result in panics due to a nil dereference	1	/pkg/logcli/client/client.go:736

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	Service 'loki' 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 service to	1	/debug/docker-compose.yaml:7
MEDIUM	semgrep	Service 'nginx' 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 service t	1	/clients/cmd/docker-driver/docker-compose.yaml:3
MEDIUM	semgrep	Service 'promtail' 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 servic	1	/debug/docker-compose.yaml:19
MEDIUM	semgrep	Variable `res` is likely modified and later used on error. In some cases this could result in panics due to a nil dereference	1	/pkg/querier/queryrange/querysharding.go:230
MEDIUM	semgrep	Variable `member` is likely modified and later used on error. In some cases this could result in panics due to a nil dereference	1	/pkg/ui/cluster.go:76
MEDIUM	semgrep	Service 'grafana' 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 service	1	/debug/docker-compose.yaml:32
MEDIUM	semgrep	This code uses a 224-bit hash function, which is deprecated or disallowed in some security policies. Consider updating to a stronger hash function such as SHA-384 or higher to ensure compliance and security.	3	/pkg/dataobj/uploader/uploader.go:67

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	Detected SSL that will accept an unverified connection. This makes the connections susceptible to man-in-the-middle attacks. Use 'OpenSSL::SSL::VERIFY_PEER' instead.	2	/clients/cmd/ fluentd/lib/ fluent/ plugin/ out_loki.rb:26
MEDIUM	semgrep	Iteration over a possibly empty map `sources`. This is likely a bug or redundant code	1	/pkg/engine/ internal/ worker/ worker.go:521
MEDIUM	semgrep	Iteration over a possibly empty map `sinks`. This is likely a bug or redundant code	1	/pkg/engine/ internal/ worker/ worker.go:522
MEDIUM	semgrep	Iteration over a possibly empty map `filterCols`. This is likely a bug or redundant code	1	/pkg/dataset/ layout/ codec_struct. go:222
MEDIUM	semgrep	Iteration over a possibly empty map `distinctUsers`. This is likely a bug or redundant code	1	/pkg/storage/ stores/ shipper/ indexshipper/ downloads/ table_manager. go:351
MEDIUM	semgrep	The RSA key size 1024 is insufficient by NIST standards. It is recommended to use a key length of 2048 or higher.	1	/clients/cmd/ fluentd/spec/ gems/fluent/ plugin/ loki_output_ spec.rb:381
MEDIUM	semgrep	Service 'promtail' is explicitly disabling seccomp confinement. This runs the service in an unrestricted state. Remove 'seccomp:unconfined' to prevent this.	1	/debug/ docker- compose.yaml: 19
MEDIUM	semgrep	Service 'loki' is explicitly disabling seccomp confinement. This runs the service in an unrestricted state. Remove 'seccomp:unconfined' to prevent this.	1	/debug/ docker- compose.yaml: 7
MEDIUM	semgrep	Service 'loki' allows for privilege escalation via setuid or setgid binaries. Add 'no-new-privileges:true' in 'security_opt' to prevent this.	1	/debug/ docker- compose.yaml: 7

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	Service 'nginx' allows for privilege escalation via setuid or setgid binaries. Add 'no-new-privileges:true' in 'security_opt' to prevent this.	1	/clients/cmd/docker-driver/docker-compose.yaml:3
MEDIUM	semgrep	Service 'promtail' allows for privilege escalation via setuid or setgid binaries. Add 'no-new-privileges:true' in 'security_opt' to prevent this.	1	/debug/docker-compose.yaml:19
MEDIUM	semgrep	Service 'docs' allows for privilege escalation via setuid or setgid binaries. Add 'no-new-privileges:true' in 'security_opt' to prevent this.	1	/cmd/logql-analyzer/docker-compose.yaml:11
MEDIUM	semgrep	Service 'grafana' allows for privilege escalation via setuid or setgid binaries. Add 'no-new-privileges:true' in 'security_opt' to prevent this.	1	/debug/docker-compose.yaml:32
MEDIUM	semgrep	Calling `wg.Wait()` inside a loop blocks the call to `wg.Done()`	1	/pkg/compactor/compactor.go:466
MEDIUM	guarddog	Detects unicode homoglyphs and uncommon characters used for obfuscation [threat-runtime-obfuscation-unicode]	1	/vendor/github.com/go-openapi/strfmt/duration.go:40
MEDIUM	guarddog	Detects URLs to suspicious domains often used for exfiltration or C2 [threat-network-exfiltration]	2	/vendor/github.com/aliyun/credentials-go/credentials/ecs_ram_role_credentials_provider.go:14
MEDIUM	semgrep	Semgrep could not determine that the argument to 'template.HTML()' is a constant. 'template.HTML()' and similar does not escape contents. Be absolutely sure there is no user-controlled data in this template. If user data can reach this template, you may have a XSS vulnerability. Instead, do not use	1	/cmd/logline-util/dump_report.go:122

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	Downcasting or changing sign of an integer with `int32` method	3	/operator/internal/manifests/networkpolicy.go:490
MEDIUM	semgrep	An HTTP redirect was found to be crafted from user-input `r`. This can lead to open redirect vulnerabilities, potentially allowing attackers to redirect users to malicious web sites. It is recommend where possible to not allow user-input to craft the redirect URL. When user-input is necessary to cra	1	/pkg/ui/handler.go:98
MEDIUM	semgrep	Detected 'Fprintf' or similar writing to 'http.ResponseWriter'. This bypasses HTML escaping that prevents cross-site scripting vulnerabilities. Instead, use the 'html/template' package to render data to users.	2	/pkg/loki/status.go:15
MEDIUM	semgrep	Downcasting or changing sign of an integer with `uint32` method	3	/pkg/storage/chunk/chunk.go:192
MEDIUM	semgrep	ReverseProxy can remove headers added by Director. Consider using ReverseProxy.Rewrite instead of ReverseProxy.Director.	2	/pkg/ui/handler.go:106
MEDIUM	semgrep	Detected SHA1 hash algorithm which is considered insecure. SHA1 is not collision resistant and is therefore not suitable as a cryptographic signature. Use SHA256 or SHA3 instead.	4	/operator/internal/manifests/config.go:30
MEDIUM	semgrep	Container or pod is running in privileged mode. This grants the container the equivalent of root capabilities on the host machine. This can lead to container escapes, privilege escalation, and other security concerns. Remove the 'privileged' key to disable this capability.	2	/operator/hack/addons_ocp.yaml:99
MEDIUM	semgrep	String-formatted SQL query detected. This could lead to SQL injection if the string is not sanitized properly. Audit this call to ensure the SQL is not manipulable by external data.	4	/pkg/goldfish/storage_mysql.go:487

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	<code>`MinVersion`</code> is missing from this TLS configuration. By default, as of Go 1.22, TLS 1.2 is currently used as the minimum. General purpose web applications should default to TLS 1.3 with all other protocols disabled. Only where it is known that a web server must support legacy clients with unsupport	5	/pkg/storage/chunk/cache/redis_client.go:77
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>	97	/.github/workflows/minor-release-pr.yml:670
MEDIUM	semgrep	The profiling 'pprof' endpoint is automatically exposed on /debug/pprof. This could leak information about the server. Instead, use <code>`import "net/http/pprof"`</code> . See https://www.farsightsecurity.com/blog/txt-record/go-remote-profiling-20161028/ for more information and mitigation.	1	/clients/cmd/docker-driver/main.go:43
MEDIUM	semgrep	When working with web applications that involve rendering user-generated content, it's important to properly escape any HTML content to prevent Cross-Site Scripting (XSS) attacks. In Go, the <code>`text/template`</code> package does not automatically escape HTML content, which can leave your application vuln	9	/operator/internal/manifests/internal/gateway/build.go:7
MEDIUM	semgrep	Deserializing into <code>`interface{}`</code> allows arbitrary data structures and types, which can lead to security vulnerabilities (CWE-502). Use a concrete struct type instead.	1	/clients/cmd/fluent-bit/loki.go:226
MEDIUM	semgrep	The special variable IFS affects how splitting takes place when expanding unquoted variables. Don't set it globally. Prefer a dedicated utility such as 'cut' or 'awk' if you need to split input data. If you must use 'read', set IFS locally using e.g. 'IFS="," read -a my_array'.	2	/.github/workflows/scripts/common.sh:56

SEVERITY	DETECTOR	FINDING	SEEN	EXAMPLE LOCATION
MEDIUM	semgrep	Checks for disabling of TLS/SSL certificate verification. This should only be used for debugging purposes because it leads to vulnerability to MTM attacks.	1	/pkg/storage/chunk/client/gcp/gcs_object_client.go:365
MEDIUM	semgrep	Found an HTTP server without TLS. Use 'http.ListenAndServeTLS' instead. See https://golang.org/pkg/net/http/#ListenAndServeTLS for more information.	2	/clients/cmd/docker-driver/main.go:43
MEDIUM	semgrep	Detected a possible denial-of-service via a zip bomb attack. By limiting the max bytes read, you can mitigate this attack. `io.CopyN()` can specify a size.	2	/pkg/storage/bloom/v1/archive.go:129
MEDIUM	semgrep	Detected directly writing or similar in 'http.ResponseWriter.write()'. This bypasses HTML escaping that prevents cross-site scripting vulnerabilities. Instead, use the 'html/template' package and render data using 'template.Execute()'.	20	/pkg/ruler/base/api.go:372

Showing the 60 most significant of 70 distinct findings. The counts in the previous section cover all of them.

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.