

SAMPLE REPORT · illustrative findings on a fictional client · shows Kloof's output shape, tiers, and reviewed calibration – not a real customer

SAMPLE – MAY BE SHARED · RIDGE & TITLE (FICTIONAL) · RADS ENTERPRISES / KLOOF

Kloof/ V1.2 · KLF-RPT · STATIC

Prepared by RADS Enterprises, LLC
Client Ridge & Title (sample)
Report date 2026-09-06

APPLICATION SECURITY EVIDENCE REPORT · SUPABASE (POSTGRES · RLS · STORAGE · EDGE FUNCTIONS)

Ridge & Title — Kloof Evidence Report

A static Kloof battery over one repository: Postgres/RLS/migrations, storage config, **every edge function's authorization**, and a client-bundle secret scan. This is a reviewed artifact — a human read every High before this PDF was signed. The engine runs in minutes; the review is the product. **It is not a penetration test or a SOC 2 report.**

SCOPE	RUN TYPE	ENGINE	SKU
Repo: DB + RLS + 41 edge fns	Static (no production access)	Kloof v1.2 · named battery	KLF-RPT

Verdict — a strong data layer with two high-severity edge-function flaws to fix now.

The Postgres/RLS foundation is well-built: row-level security on every application table, private document buckets, SECURITY DEFINER functions that self-authorize, and no server secret in the client bundle. The material risk is not in the data layer — it is that several edge functions treat the platform's `verify_jwt` gateway flag as authorization. Because the project's anon key is public, that is effectively unauthenticated. The headline (**EF-1, HIGH**) is an open, brand-signed email relay; the second high (**EF-6 / AUTH-2**) is a queue worker that trusts a locally-decoded token, so any token — including the public anon key — can trigger it. Three findings we **reviewed and downgraded** are shown too — that calibration is the point of a reviewed report. No PHI (HIPAA N/A); the AI assistant reads no stored content (AI tier N/A).

70

SECURITY

data layer strong · EF tier open

65

SOC 2

CC6.x technical themes

N/A

HIPAA

no PHI in scope

N/A

AI / AGENTIC

assistant reads no stored data

High 2

Medium 4

Low 1

Reviewed-clear 3

Pass 14

How to read this report

VERIFIED

A probe ran against the staging API and confirmed it. Proof, not inference.

CONFIG

Read from schema, policies & migrations. A strong signal to confirm live.

REVIEW

Guided review item. Excluded from the coverage score.

EXTERNAL

A named third-party tool asserted it. Ingested and triaged, not Kloof-proven; excluded from the Kloof score.

Coverage is evidence, not a certificate. Findings carry a confidence tier. CONFIG is read from migrations and edge-function source (this was a static engagement — nothing ran against production). EXTERNAL is third-party tool output, triaged but never counted in the Kloof score. Three items the automated battery raised were reviewed and cleared — shown as REVIEWED-CLEAR so you can see the calibration a signed report adds over a raw scan.

Priority findings

Ranked by real risk after human review. Fix EF-1 first.

EF-1 Open, brand-signed email relay (broken function-level authz)

CONFIG

HIGH

OBSERVATION

The transactional-email function has **no in-function authorization** — it relies on the platform's `verify_jwt=true` gateway flag. But the project's **anon key is public**, so any internet caller satisfies that gate, then supplies the recipient address and template data. The email templates render a caller-supplied link straight into the button target.

IMPACT

Anyone can send a convincing, brand-signed email from the company's own domain, to any address, carrying an attacker-chosen link — phishing under a trusted domain, email-bombing, and sender-reputation damage. The template registry and suppression list do not prevent it.

REPRODUCTION

```
# with only the PUBLIC anon key:
POST /functions/v1/send-transactional-email
Authorization: Bearer <public-anon-key>
{ "recipientEmail":"victim@example.com",
  "template":"deal-status",
  "templateData":{"ctaUrl":"https://attacker.example/x" } }
# → the gateway accepts the call (the public anon key satisfies verify_jwt) and the
# CTA renders the attacker link. CONFIG finding: message delivery was NOT fired on this
# static run – the send is proven only under the Report + staging SKU.
```

REMEDIATION

Gate it with the service-role pattern already used by the cron mailers: require `Authorization: Bearer <service-role key>` (constant-time compare) and route user flows through a server enqueue path, not direct calls. Allow-list the CTA link to the canonical origin; rate-limit per recipient. Then apply the same gate to every privileged function currently behind `verify_jwt` only.

STATUS

● OPEN · OWASP A01 · A04 · confused deputy

Queue worker trusts a Bearer prefix + a locally-decoded token

CONFIG

HIGH

OBSERVATION

An internal email-queue worker checks only that the `Authorization` header *starts with* "Bearer", then base64-decodes the token locally and trusts its role claim — no verified `getUser()` and no constant-time service-role compare.

IMPACT

Any holder of any token (including the public anon key) can trigger queue processing — premature drains, send-timing manipulation, denial of service. A locally-decoded JWT is attacker-forgeable; it is not authentication.

REMEDIATION

Constant-time compare the header against the service-role key on this cron-only endpoint, or verify the caller with `getUser()` ; reject user JWTs. Never make an authorization decision on a locally-decoded token.

STATUS

● OPEN · OWASP A01

GR-1b EXECUTE granted to anon on SECURITY DEFINER functions

CONFIG

MEDIUM

OBSERVATION

Three SECURITY DEFINER RPCs are granted `EXECUTE` to the `anon` role (token-lookup and eligibility helpers). A DEFINER function runs with its owner's privileges; granting it to anon widens the unauthenticated surface even when RLS is on elsewhere.

IMPACT

Unauthenticated callers can invoke owner-privileged logic. Whether that leaks or mutates depends on each function body — but anon `EXECUTE` on a DEFINER function is a surface that should be deliberate, not default.

REMEDIATION

Revoke `EXECUTE` from anon on these functions unless a specific unauthenticated flow requires them; if one does, ensure the function self-authorizes and returns only non-sensitive data.

STATUS

● OPEN · OWASP A01 · SOC2 CC6.1

EF-9 Provisioning endpoint mints users from a caller-supplied password

[CONFIG](#)[MEDIUM](#)

OBSERVATION

A sandbox-provisioning function creates real auth users with a caller-supplied password, gated only by a static shared-secret header (no expiry, no rotation, no admin role check).

IMPACT

If the static token leaks, unauthenticated account creation follows.

REMEDIATION

Put provisioning behind a verified signed-in admin (role check), and if a machine token must remain, rotate/scope/expire it and compare it in constant time.

STATUS

● OPEN · OWASP A07

BILL-1 Entitlement quotas ship in shadow mode (not enforced)

[REVIEW](#)[MEDIUM](#)

OBSERVATION

The entitlement system defaults to a `shadow / warn` mode: it logs a would-block but does not enforce, and the create path checks seat presence rather than the plan's limits.

IMPACT

Plan limits are advisory today; and when enforcement is switched on, the current count-then-insert pattern will be race-exposed under concurrency.

REMEDIATION

Enforce the quota inside the mutating function in the same transaction as the insert, using a row lock or a counting constraint — never an edge-function read then a separate write.

STATUS

● OPEN · OWASP A04 · abuse / revenue

EF-8 Utility functions run without JWT verification or a signature

[CONFIG](#)[MEDIUM](#)

OBSERVATION

Five scheduled/utility functions run with `verify_jwt=false` and verify no signature or shared secret in-function.

IMPACT

If any is reachable off the schedule, it is unauthenticated. Review each: it should verify a signature, compare a service-role secret, or be removed from public routing.

REMEDIATION

Add a signature/secret check to each, or confirm it is invoked only by the scheduler and not exposed.

STATUS

● OPEN · OWASP A01

T1 Four queue functions omit a pinned search_path

[CONFIG](#)

LOW

OBSERVATION

Four SECURITY DEFINER queue helpers omit `SET search_path`. Currently mitigated — they are service-role-only and call schema-qualified objects — so the classic hijack path is closed.

REMEDIATION

Add `SET search_path` to match the other 70 definer functions and stay safe if EXECUTE is ever re-granted.

STATUS

● OPEN · OWASP A01

Reviewed and cleared (what a signed report adds over a raw scan)

The automated battery raised these; human review downgraded them. Showing them is the calibration you're paying for.

T2 Three tables have RLS on but no policy → deny-all (safe)

[CONFIG](#)[PASS](#)

OBSERVATION

The raw battery flags a table with row-level security enabled and no policy as a coverage gap. On review, these three are backend-only tables: RLS-on + no client policy = **deny-all to anon/authenticated**, reachable only by the service role. That is the intended locked-down state, not a vulnerability.

REMEDIATION

None. Reviewed-clear. (This is a calibration the reviewer applies; a raw scanner would mis-rank it as critical.)

STATUS

● WELL-BUILT · OWASP A01 · CC6.1

GR-1 anon SELECT on the pricing catalog

[REVIEW](#)[PASS](#)

OBSERVATION

anon can read the public `plans` pricing table. Reviewed as intentional — it is non-tenant reference data shown on the marketing/pricing page.

REMEDIATION

None, assuming pricing is meant to be public (confirm).

STATUS

● WELL-BUILT · OWASP A01

EF-7 Caller-influenced link — same finding as EF-1

[REVIEW](#)[PASS](#)

OBSERVATION

The caller-controlled link flag resolves to the EF-1 relay above, not a separate defect. Folded into EF-1's remediation (allow-list the CTA origin).

REMEDIATION

Covered by EF-1.

STATUS

● WELL-BUILT · OWASP A03

Data layer & AI tier (summary)

RLS/secrets

Tenant isolation, storage, secrets, DEFINER authz

CONFIG

PASS

OBSERVATION

Row-level security on all application tables; private document buckets with size/MIME limits; no service-role/secret in the client bundle (only the expected public anon key); mutating DEFINER RPCs self-authorize; append-only audit tables. Statically, a strong tenant boundary.

REMEDIATION

To convert this from CONFIG to VERIFIED, add the staging SKU (a live cross-tenant probe).

STATUS

●

 PASS · OWASP A01/A05 · CC6.1

AI-1.11

AI / agentic tier

REVIEW

N/A

OBSERVATION

One AI surface (a chat assistant). It receives only the live, length-capped conversation; it reads no stored deal content, holds no tools, has no DB access. Stored-content injection does not apply.

REMEDIATION

Re-run the AI-11 gate before any future feature lets the assistant read stored records.

STATUS

●

 N/A · OWASP LLM Top 10

General security layer — Semgrep · gitleaks · AI code review (ingested, not Kloof-proven)

Kloof folds best-in-class commodity tools into the same report — marked as theirs, never counted in the Kloof score.

What this layer is. This is the **general-purpose layer**: SAST (Semgrep), secret scanning (gitleaks), and AI code-review tools such as /security-audit or Snyk. They cover **breadth** across the whole repository — generic injection, dependency CVEs, code smells — fast and cheaply. We run the ones reachable in this engagement and fold their output in here, clearly marked as theirs. Keep running them; they are useful. They are simply **not** where the Supabase tenant-boundary risk lives, and their output is a starting point, not a verdict.

TOOL	CATEGORY	STATUS	RESULT
gitleaks	Secret scan	RAN	1 hit, triaged to INFO (below). Value redacted.
Semgrep	SAST	RAN	6 informational items — all duplicates of CONFIG findings or framework noise; no material finding beyond the Kloof tiers.
AI code review (e.g. /security-audit)	General LLM SAST	NOT RUN	Same layer as above; a general LLM reviewer produces a non-deterministic worry list, not a signed artifact. Run it in your IDE — we don't reproduce a third-party tool's output as ours.
OWASP ZAP	DAST (frontend)	N/A	No running frontend target in a static engagement.

GITLEAKS:jwt JWT in committed .env

GITLEAKSGITLEAKSINFO

LOCATION

.env : PUBLISHABLE_KEY

KLOOF TRIAGE

Expected-public. The publishable/anon key, shipped in the client bundle by design. Not a secret leak. No server secret is committed.

Third-party output. gitleaks flags every JWT it sees; the reviewer distinguishes an expected-public client key from a real server secret. The useful signal: no service_role or server secret is committed.

What this layer does not cover — and Kloof does. None of these tools can fire a real cross-tenant request to **prove** RLS holds at runtime (Kloof's VERIFIED tier), reason about whether an edge function's verify_jwt gate is real authorization when the anon key is public (EF-1 above), or hand you a dated artifact a human signed. A general LLM reviewer is also non-deterministic and can hallucinate a *pass* on a real bug — and can be prompt-injected by stored content into a clean report (the exact risk Kloof's AI-11 check exists for). Breadth from this layer; depth and proof from the tiers above.

Coverage & towers

TOWER	COVERAGE	PASSED / APPLICABLE	BASIS
Security (OWASP / ASVS)	~70% (reviewed)	data layer strong · 2 EF highs open (EF-1, EF-6)	OWASP A01/A04/A05 · ASVS
SOC 2 — Common Criteria	~65% (reviewed)	CC6.x technical themes present	org controls out of repo
HIPAA — Technical Safeguards	N/A	no PHI — commercial real estate	re-scope only if health data is ingested

Coverage = passed ÷ (passed + failed) among mapped, applicable checks. Review/NA items are excluded. This is evidence contributed toward the framework — never a compliance certification.

Appendix — battery composition

Kloof v1.2 ran a named battery across three confidence tiers (CONFIG / VERIFIED / REVIEW) and an EXTERNAL tool tier, plus a per-function review of all 41 edge functions. Scores shown are the **reviewed** verdict; a human read every High before signing. The battery is validated by a seeded-regression harness (planted vulnerabilities caught, zero false positives on clean input).

Scan provenance: config tier read from committed migrations (AST-parsed) + edge-function source; external tier — gitleaks executed; live probes (cross-tenant VERIFIED, EF-1 send-proof) are available under the Report + staging SKU against a staging project only, never production.

Report identity

Product	Kloof v1.2
Recipient	Ridge & Title (sample client)
Snapshot	commit 9f2c1ab · repo hash (sample)
SKU	KLF-RPT — Report
Run type	Static (repository source)
Production accessed	No
Findings	High 2 · Medium 4 · Low 1 · Reviewed-clear 3 · Pass 14

Disclaimer

DISCLAIMER

This document is a Kloof evidence report prepared by RADS Enterprises, LLC ("Kloof") for the named recipient only.

Nature of work. Kloof runs a named battery of checks against repository source (migrations, schema, storage config, edge-function source, and client bundles in scope) and, if purchased, against a staging project the recipient designated. A static or CONFIG finding is an inference from source or schema. A VERIFIED finding is a probe that ran in the named staging environment. REVIEW and EXTERNAL items are not included in coverage scores. Coverage is passed / (passed + failed) among applicable scored checks. It is not a grade of the whole company.

Not a certification. This report is not a penetration test, not a SOC 2 Type I or Type II report, not a HIPAA, PCI DSS, or ISO/IEC 27001 attestation, and not a legal opinion. References to SOC 2 Common Criteria (for example CC6.x) describe thematic overlap with in-repo technical controls. Organizational controls (access reviews, change management, vendor management, incident response) are out of scope unless explicitly listed.

No warranty of security. Absence of a finding is not proof that a vulnerability does not exist. Presence of a finding is not proof of production exploitability unless marked VERIFIED. Source and staging can diverge from production. The recipient is responsible for remediation, for deciding whether to commission a scoped penetration test, and for all representations to customers, insurers, and auditors.

Authorization and data. Work was performed only on systems and repositories the recipient authorized. Kloof does not require a production service_role key for a static Report. Staging credentials, if any, are used solely for the purchased probes and are not retained as a standing admin grant. Kloof is not a Business Associate under HIPAA by delivery of this report.

Confidentiality. This report is confidential to the recipient. It may be shown to the recipient's counsel, auditors, and due-diligence counterparties under NDA. It may not be published, excerpted as a marketing "score," or altered without the cover, scores, and this disclaimer intact.

Limitation. To the extent permitted by law, Kloof's liability arising from this engagement is limited to the fees paid for the SKU that produced this report. Kloof is not liable for lost profits, lost data, or third-party claims arising from the recipient's use of the software described herein.

Engine version and date are printed on the cover. Findings describe the snapshot as of that date only.