

Coreenna Security Whitepaper

Version 1.0 — June 2026

Coreenna builds and signs mobile apps on behalf of developers. This document describes the controls we use to protect source code, signing materials, build artifacts, and customer data.

1. Architecture & Trust Boundary

Coreenna does not clone customer source code to its own servers. All builds execute on GitHub Actions runners inside the customer's own repository. The Coreenna control plane orchestrates workflows via the GitHub App API and never gains direct filesystem access to source.

2. Signing Materials

- Android keystores, key passwords, Apple .p12 certificates, and provisioning profiles are encrypted at rest with authenticated encryption (AES-GCM).
- Key passwords are stored separately from keystore blobs; compromise of one does not compromise the other.
- Secrets are pushed to GitHub Actions Secrets using GitHub's libsodium sealed-box encryption and decrypted only inside ephemeral runners.
- Materials are never written to logs, build output, or artifact metadata.

3. Authentication & Access Control

- All API endpoints require an authenticated session (JWT).
- Database row-level security (RLS) restricts every read and write to the owning user.
- File storage is scoped to the user's folder (`storage.foldername(name)[1] = auth.uid()`).
- Realtime broadcast/presence channels are deny-all by default; only RLS-filtered `postgres_changes` are exposed.

4. Build Integrity

- Each build is issued a per-build HMAC token used to update status from the GitHub Actions runner.
- Terminal-state guards prevent status replay: once a build is success / failed / cancelled, further updates return 409.
- Artifact URLs are validated server-side: HTTPS-only, 2048-character limit, no javascript:, data:, or file: schemes.
- Client-side rendering wraps all artifact URLs in a `safeHref()` filter as defense-in-depth.

5. Log Handling

Build logs are streamed from GitHub Actions and stored for retrieval. Known secret patterns are redacted server-side before display. Customers are advised never to paste plaintext secrets into build configuration files.

6. Customer Control

- Rotate keystores and certificates at any time from Settings.
- Export project and build data on demand.
- Delete account — all signing materials are purged within 7 days.

7. Independent Validation

Corenna's backend has been subjected to a structured penetration test covering token replay, workflow_dispatch abuse, and cross-tenant data access. The most recent test returned zero findings. Test scope and methodology are available on request.

8. Responsible Disclosure

Report suspected vulnerabilities to security@corenna.app with reproduction steps. Please do not publicly disclose issues until we have had a reasonable opportunity to investigate and respond.