Security Architecture

Version: 2026-04 · **Owner:** Engineering · **Distributed as:** `Votriz_Security_Architecture.pdf`

Scope

This document describes the technical controls Votriz uses to protect customer data, isolate tenants, and resist common attack classes. Everything below maps to runtime evidence in the platform — auditors can confirm each claim against `/security/isolation-proof` and `scripts/pre_prod_validation.sh`.

1. Data isolation (three layers)

Layer 1 — Application

- Every authenticated request derives `org_id` from the JWT, never from request body or query parameters.
- Every database query filters by `org_id`; this is enforced by code review and by `pre_prod_validation.sh` which provisions two isolated orgs and asserts that org B cannot read, write, or infer any of org A's data across every tenant-scoped endpoint.
- The architectural invariant is documented in `CLAUDE.md` (section "Security Invariants") and treated as inviolable.

Layer 2 — Database (RLS)

- PostgreSQL Row-Level Security is enabled on the nine tenant-scoped tables: `brands`, `channels`, `content_queue`, `published_posts`, `email_subscribers`, `email_campaigns`, `email_sends`, `mentions`, `presence_incidents`.
- Each table carries an isolation policy of the form `org_id::text = current_setting('app.current_org_id', true)`.
- The connection role currently bypasses RLS (`bypassrls=t`). The policies are deployed inert until the application moves to a non-superuser role and sets `app.current_org_id` per request. This is honestly reflected in the `/security/isolation-proof` output.

Layer 3 — Infrastructure (Enterprise option)

- Dedicated database schemas or separate PostgreSQL instances are available for Enterprise customers as part of the contract.
- Per-org encryption keys (planned) — currently the platform uses a single AES-256 Fernet key for OAuth tokens.

2. Encryption

Layer	Standard	Key custody
In transit (public)	TLS 1.3	Cloudflare-managed
In transit (internal)	Isolated container network	n/a
At rest (database)	Volume-level encryption	Infrastructure layer
OAuth tokens (per-row)	Fernet (AES-128-CBC + HMAC-SHA256)	Hardware-backed secure storage (HSM)
JWT signing	HS256	Hardware-backed secure storage (HSM)
Backups	Storage-layer encryption	Provider

All API keys (Anthropic, OpenAI, fal.ai, Resend, Stripe, etc.) live in hardware-backed secure storage and are namespaced per service. They are never stored in code, configuration files, or anywhere reachable from the runtime container's writable filesystem.

3. Access control (RBAC)

Five built-in roles ranked by privilege: **owner > admin > manager > editor > viewer**. The `member` legacy alias maps to editor's permission set for back-compat.

- 40+ wildcard-aware permissions across 20 resource types (`brands.*`, `email.campaigns.send`, `audit_log.export`, etc.).
- Brand-level scoping for managers and editors via `users_brand_access`. Owner and admin have implicit access to every brand in the org.
- Per-user permission overrides via `user_permission_overrides` for fine-grained exceptions.
- Custom roles for Enterprise tier via `custom_roles`.

The role-based layer is enforced through the `require_permission()` FastAPI dependency on every mutating route. Live coverage is inspectable in the `services/permissions.py` ROLE_PERMISSIONS map.

4. Audit logging

- Append-only `security_audit_log` table with an immutability trigger (`prevent_audit_modification()`) that raises on UPDATE and DELETE — even from the database superuser.
- Every authenticated action records: org_id, user_id, user_email, user_role, ip_address, user_agent, action, resource_type, resource_id, details (JSONB), request_id, session_id, created_at.
- Queryable via `GET /audit/log`, exportable via `/audit/log/export?format=csv|json`, summarised via `/audit/log/summary`.
- 7-year retention target, aligned with SOC 2.

5. AI safety

- Human-approval gate on every published piece of content. Ghost Presence (autonomous mode) is opt-in per org and uses a configurable confidence threshold; "critical" brand-monitor incidents always route to human review regardless of score.

- Customer data is **not** used to train, fine-tune, or improve any AI model. All inference goes through Anthropic's API, whose contract excludes API data from training corpora.
- Prompt-injection defense via `services/prompt_guard.py` (pattern-matching on common jailbreak markers) for the support chatbot and any other free-form user-input → LLM path.
- PII redaction layer (`services/pii_redactor.py`) for chatbot and email-personalization paths. **Not** applied to the lead generator, where extracting public business contact information is the explicit job.

6. Operational security

- Container images rebuilt from source on every deploy (no long-lived snapshots that drift from `git HEAD`).
- Secrets injected at container start from hardware-backed secure storage. The secure-storage layer is the *only* authoritative source of production credentials.
- Operator access is gated by an encrypted VPN tunnel; no port-forwarded shell access on the public internet.
- `pre_prod_validation.sh` runs before every production deploy and blocks merges if multi-tenant isolation regresses.
- `daily_metrics.sh` provides a one-glance health/usage snapshot for morning operations.

7. Incident response

See `AI_INCIDENT_RESPONSE.md` for the full severity ladder and playbook. Headline: P1 (data breach) → 15-minute notification to AI Safety Officer, 4-hour customer notification, 72-hour public report.

8. Verification endpoints

Endpoint	Auth	Purpose
`GET /security/isolation-proof`	any user	Live three-layer isolation evidence
`GET /security/posture`	owner / admin	Compact dashboard summary
`GET /audit/log`	owner / admin	Searchable audit feed
`GET /audit/log/export`	owner	Bulk CSV / JSON download
`GET /audit/log/summary`	owner / admin	Aggregates for compliance card

9. Roadmap

- Q3 2026: third-party penetration test
- Q4 2026: SOC 2 Type I attestation
- Q4 2026: Move application connection role off `votriz` superuser → activates RLS enforcement live in production
- Q2 2027: SOC 2 Type II attestation
- 2027: ISO 27001 + ISO 42001 (AI management systems)