VECTO

Autonomous Intelligence. Human Control.

Security & Privacy

Datasheet for procurement, security teams, and DPOs

vecto-os.com · 5 May 2026

At a Glance

Platform	Vecto — graph-native projectplatform
Headquartered	Belgium (EU)
Data residency	EU by default; customer-controlled for private deployments
Encryption (transit)	TLS 1.3
Encryption (at rest)	AES-256-GCM, per-tenant key
Access control	RBAC (Root → Tenant Admin → Member)
Audit trail	Immutable, per-actor, exportable
Tenant isolation	DB-level workspace filter
Authentication	JWT + scopes; SSO/SAML on roadmap
Compliance posture	GDPR-aligned · NIS2-aware · AI Act-ready · ISO 27001 roadmap

1. Architecture

Vecto is a multi-tenant graph platform built on three architectural principles:

- **Defense in depth** — every API call passes through middleware checks before any data is touched
- **Least privilege by default** — RBAC scopes are explicit, never inherited beyond their boundary
- **Per-tenant cryptographic isolation** — every tenant has its own encryption key; the platform operator cannot read tenant data without that key

Deployment options:

Mode	Where it runs	Who manages
Public SaaS	Vecto-managed cloud (EU)	Vecto
Private SaaS	Customer cloud (AWS/Azure/GCP)	Vecto manages, customer controls infra
On-premise	Customer datacenter	Customer
Appliance	Customer hardware (planned)	Customer

For all non-public-SaaS deployments: **trust-but-verify** licensing — no telemetry, customer self-reports usage, Vecto retains audit rights. Privacy-conscious enterprise procurement choice.

---

2. Encryption

2.1 In transit

- **TLS 1.3** for all client connections
- **HTTPS only**; HTTP redirects to HTTPS
- Certificate management via Let's Encrypt (public SaaS) or customer PKI (on-prem)
- mTLS available for service-to-service in private deployments

2.2 At rest

- **AES-256-GCM** for all database content
- **Per-tenant encryption keys** managed by `cryptoService`
- Keys stored separately from data; rotatable without service downtime
- Backups encrypted with the same per-tenant key

2.3 Key management

- Roadmap: **HSM/cloud-KMS integration** for tenant-key resolution (Phase 2)
- Roadmap: **tenant-managed keys** (BYOK) for customers requiring exclusive control (Phase 3)
- Current: env-managed master key, per-tenant DEKs in DB (Phase 1)

---

3. Access Control

3.1 Authentication

- **JWT-based** session tokens, signed with rotating secret
- **Argon2id** password hashing (no bcrypt, no MD5/SHA1)
- Token lifetime: configurable per tenant; default 8 hours
- Refresh tokens: rotated on each use (replay-safe)
- **SSO / SAML / OIDC** on roadmap for enterprise tier

3.2 Authorization (RBAC)

Three-tier role hierarchy:

Scope	Capabilities
Platform / Founder	Cross-tenant operations, system config, infrastructure
Tenant Admin	All operations within own tenant; user mgmt
Member	Project-scoped operations as granted
Viewer	Read-only within scope

Every API endpoint declares its required scope via `requireScope( ... )` middleware. **No request bypasses authorization** — there is no "admin only" backdoor.

3.3 Per-project access (PAT scoping)

Personal Access Tokens (e.g., for MCP/agent use) can be scoped to:

- Tenant-wide (humans, default for personal tokens)
- **Per-project restricted** (recommended for agents, default for MCP tokens)

This limits blast radius if a token leaks.

4. Tenant Isolation

4.1 Database-level

Every query in Vecto includes a `tenant_id` filter, enforced at the data-access layer (`db.ts`). **No agent or user can read or write outside their workspace** — not via prompt injection, not via API parameter manipulation.

4.2 Process-level

Multi-tenant operations run with explicit tenant context. Background jobs carry their initiating tenant; no shared state between tenants.

4.3 Storage-level

Per-tenant encryption keys mean that even at the disk level, tenant A's data cannot be decrypted with tenant B's key.

---

5. Audit Trail

Every action emits a structured audit event:

```
{
  event: 'category.action',
  scope: 'infra|tenant|project|user',
  sensitivity: 'normal|security',
  severity: 'info|warn|error|critical',
  actor: { type: 'human|agent', id, ... },
  scope_vid: <which workspace>,
  timestamp: ISO 8601,
  payload: { ... },
  signature: HMAC-SHA256
}
```

Properties

- **Immutable** — append-only log; no UPDATE or DELETE on audit rows
- **Cryptographically signed** — HMAC over event payload
- **Per-actor identity** — distinguishes human vs agent, names the agent

- **Exportable** — JSON / CSV for SIEM integration
- **Sensitivity-tagged** — security-sensitive payloads protected by additional `auditor` capability check

Notification linkage

User-facing notifications (`notifications` table) **must** reference an audit row via `triggered_by_audit_vid`. No notification can exist without its audit provenance.

---

6. Secrets Management

- **No plaintext secrets** in database, env, or logs
- API keys and credentials encrypted with tenant-specific keys via `cryptoService`
- Keys rotatable without service downtime
- `.env` files outside repo, never committed
- Secrets in CI/CD via runner-secrets, scoped per environment

---

7. Release Integrity

Every Vecto release is **HMAC-signed** end-to-end:

1. CHANGELOG entry created
2. Release notes drafted
3. Sign-off commit with `Release-Version`, `Release-Doc-Ref`, `Security-Review-Verdict` trailers
4. HMAC signature appended via `scripts/sign-release.mjs`
5. Build gate (`scripts/check-release-sign-off.mjs`) refuses any commit without valid signature
6. `POST /api/release/signoff` verifies signature against tenant-specific HKDF-derived key
7. Audit event `release.signoff` emitted; release node created in graph
8. Git tag `vX.Y.Z` applied to signed commit

Result: any deployed release can be traced back through cryptographic chain-of-custody to the originating idea, decision, and review.

Read more: `decisions/release-flow.md` (public).

8. AI Governance

8.1 Autonomy spectrum

Every AI agent in Vecto operates at a configurable autonomy level (Observe / Advise / Assist / Delegate / Automate). Per project, per user, per task type. **No AI can elevate its own authority.**

8.2 Separation of duties

- An agent that builds something **cannot** sign off its own security review
- Human and AI commits are distinguished in git history
- Critical sign-offs require human cryptographic signature

8.3 LLM-provider flexibility

Vecto supports multiple LLM backends (Ollama local, Google AI, OpenAI, Anthropic). Choice per tenant, per use case. **Local Ollama is first-class** — privacy-first deployments require zero external LLM calls.

8.4 Data sent to LLMs

- Tenant data sent to external LLMs only when tenant explicitly enables external provider
- Audit trail records which LLM provider received which payload
- Prompt-cache and storage policies documented per provider in tenant settings

9. GDPR Posture

Requirement	Vecto position
Data minimization	Customer chooses what data enters Vecto graph
Right of access	Per-tenant data export available on request
Right of rectification	All graph nodes editable by appropriately-scoped users
Right of erasure	Soft-delete with archival; hard-delete on documented request (audit-trail retention rules apply)
Data portability	JSON / Markdown export of complete tenant graph
Records of processing	Audit trail satisfies Art. 30 record-keeping for processing operations
DPIA support	Architecture documentation supports customer DPIA exercises
Sub-processor list	Public on `vecto-os.com/trust` (planned); included in DPA
Data residency	EU by default; private deployments give customer full control

10. Standards & Compliance Roadmap

Standard	Status
GDPR	✅ Aligned by architecture
OWASP Top 10	✅ Implemented across the stack
TLS 1.3	✅ All connections
EU AI Act	✅ Architecture-ready (audit, autonomy, oversight); formal conformity per customer use case
NIS2	🟡 Aware; targeted readiness for 2026-2027
DORA	🟡 Aware (FinServ deployments); architectural fit assessed per case
eIDAS 2.0	🟡 Crypto/PKI capability; integration on roadmap
ISO 27001	🟡 Internal controls aligned; formal certification planned
SOC 2 Type II	🟡 Roadmap
Cyber Resilience Act	🟡 Aware; architectural fit

11. Sub-processors

Public list maintained on `vecto-os.com/trust` (planned). Currently:

- **Ollama** (self-hosted by Vecto / customer) — no external data flow
- **Google AI / OpenAI / Anthropic** (only if tenant enables) — data flow per tenant configuration
- **Cloudflare** (public SaaS only) — DDoS + edge proxy
- **Hetzner / customer-cloud** — infrastructure (depending on deployment)

Full DPA available on request.

12. Vulnerability Reporting & Incident Response

- **Responsible disclosure**: `security@vecto-os.com`
- **Acknowledgment SLA**: 48 hours
- **Critical patch SLA**: 7 days for tier-1 issues
- **Public security advisories** for CVEs affecting Vecto users
- **Customer notification** within 72 hours of confirmed breach affecting customer data (GDPR Art. 33)

---

13. Contact

For procurement / security questionnaires, vendor risk assessments, or DPA discussions:

security@vecto-os.com **vecto-os.com/trust**

This datasheet describes architectural posture as of 5 May 2026. Specific certification status updates as our compliance program matures.