

Security & Compliance Posture

AI Operations Center — Controls, Audit, and Data Protection

Document Owner: AiSO Platform Team
Last Updated: May 2026
Audience: CISO, IT Security, Internal Audit, Compliance

Executive Summary

The AI Operations Center implements enterprise-grade security controls across 5 layers: data protection, access control, audit logging, LLM safety, and operational governance. This document maps each control to relevant compliance frameworks and identifies gaps that must be addressed before production deployment.

1. Data Protection

Financial Data Precision

Control	Implementation	Framework
Decimal precision for financial fields	All monetary values stored as PostgreSQL <code>Decimal</code> type, serialized as strings across API boundaries	SOX §302/404
Currency tracking	All financial records include explicit currency field	GAAP
Immutable financial records	Deduction amounts, invoice totals, and value ledger entries cannot be modified after creation	SOX §302

PII Handling

Control	Implementation	Status
Email/phone redaction	Redaction rules configured for LLM prompt sanitization	✓ Designed, needs production hardening
Customer name handling	Customer names visible to authorized roles only	⚠ Requires RBAC implementation
Data residency	Database location configurable per deployment	✓
Encryption at rest	Delegated to database provider (RDS encryption, Azure encryption)	Provider-dependent
Encryption in transit	HTTPS enforced for all client-server communication	✓

Data Classification

Data Type	Classification	Storage	Retention
Customer master	Confidential	PostgreSQL	Permanent
Financial transactions	Restricted	PostgreSQL (Decimal)	7 years (SOX)
Agent reasoning traces	Internal	PostgreSQL (JSON)	2 years recommended
LLM prompts/responses	Internal	In-memory (not persisted in full)	Session only
Audit log entries	Restricted	PostgreSQL (immutable)	7 years (SOX)
Override reasons / SME notes	Confidential	PostgreSQL	Permanent

3. Audit Logging

SOX-Compliant Audit Trail

Feature	Implementation	Table
Immutable event log	All agent decisions, approvals, and system actions logged with timestamps	<code>AuditLogEntry</code>
Who-what-when-why	Every entry includes actor, action, evidence references, and correlation ID	<code>AuditLogEntry</code>
Agent run traces	Full reasoning chain persisted per agent execution	<code>AgentRun</code>
MCP tool call audit	Every external system call logged with request, response, latency	<code>MCPToolCall</code>
Writeback approval chain	Multi-tier approval with signer identity and timestamps	<code>WritebackRequest</code>
HITL decision capture	Human approvals/rejections with notes and override reasons	<code>HITLQueueItem</code> , <code>OverrideReason</code>
Value attribution	Financial impact tracked from projection through realization	<code>ValueLedgerEntry</code>

Audit Log Integrity

- Entries are append-only (no UPDATE or DELETE operations on audit tables)
 - Each entry includes a `signature` field (placeholder for HMAC signing in production)
 - Correlation IDs link related events across tables
 - Timestamps are server-generated (not client-supplied)
-

4. LLM Safety Controls

Prompt Security

Control	Implementation	Status
Prompt injection screening	Input scanning for jailbreak patterns (“ignore previous instructions”, role-play injection)	✓ Designed
Output scanning	Response scanning for leaked secrets, customer-data spill, policy violations	✓ Designed
System prompt isolation	System prompts never exposed to end users	✓ Enforced
Structured output enforcement	All agent outputs are JSON-parsed with schema validation	✓ Enforced

LLM Data Flow

User Input → [PII Redaction] → [Injection Screen] → System Prompt + Context
 → LLM API (HTTPS) → [Output Scan] → [JSON Parse] → Application Logic

- No customer data is used for LLM fine-tuning
- LLM API calls use HTTPS with API key authentication
- Prompts include explicit instructions to not reveal customer PII in summaries
- Reasoning traces store agent decisions but not raw LLM token streams

Model Governance

Control	Implementation
Model versioning	<code>modelVersion</code> tracked on every <code>AgentRun</code>
Prompt versioning	<code>PromptVersion</code> table with semantic versioning
A/B testing	<code>ABTestRun</code> with statistical significance tracking
Drift detection	<code>DriftMonitor</code> tracks input/output distribution changes
Cost controls	<code>CostMeterEntry</code> tracks token usage with per-MVP budgets

5. Operational Governance

Decision Control Framework

Control	How It Works
Autonomy ratchet	Agents start with low autonomy; thresholds raised via boundary conditions after proving agreement rates
HITL routing matrix	Configurable rules determine which decisions require human approval based on confidence, amount, risk
Escalation triggers	Auto-escalation for aged items, circuit breaker trips, SLA breaches, credit score drops
Separation of duties	Business approval + IT/SOX approval for high-value writebacks
Value accounting	Every agent action tracked with projected → realized → unrealized lifecycle

Incident Response

Scenario	Automated Response
LLM provider outage	Agents use fallback heuristics, HITL queue absorbs workload
ERP system down	MCP circuit breaker trips, cached data used, manual fallback
Classification drift detected	Alert generated, re-eval triggered, prompt version review
Agreement rate drops below bar	Automatic block on autonomy expansion, SME review triggered

6. Compliance Gap Analysis

Must Address Before Production

#	Gap	Severity	Remediation
1	No user authentication	 Critical	Implement OAuth 2.0 / SAML SSO with identity provider
2	No API authorization middleware	 Critical	Add JWT validation + RBAC middleware to all routes
3	Audit log signatures not HMAC-signed	 Medium	Implement HMAC signing with rotating keys
4	PII redaction not hardened	 Medium	Expand regex patterns, add ML-based NER for edge cases
5	No data retention automation	 Low	Implement automated purge for records past retention period
6	No penetration testing	 Medium	Conduct OWASP Top 10 assessment before go-live
7	Database backup strategy	 Medium	Configure automated backups with point-in-time recovery

Already Satisfied

Requirement	How
SOX §302 — Financial accuracy	Decimal precision, immutable records, approval chains
SOX §404 — Internal controls	Dual-lane approval, audit trail, separation of duties
Data integrity	Foreign keys, unique constraints, cascade rules
Change management	Prompt versioning, A/B testing, boundary condition audit trail
Operational monitoring	Agent traces, drift detection, cost metering, eval harness