ENGINEERING REFERENCE ARCHITECTURE & SAFEGUARDS WHITEPAPER

HIPAA-Compliant Production Cloud Architecture & Private GenAI Pipeline

Application: AetheraOS Clinical EHR Platform

Architected By: New Paradigm Systems (NPSys)

Infrastructure Target: Amazon Web Services (AWS)

Security Standard: HIPAA Security Rule (45 CFR Part 164)

Workload Baseline: 25 Practices / ~750 DAUs

Delivery Framework: Infrastructure as Code (Terraform)

---

1. Executive & Technical Overview

AetheraOS is an Electronic Health Record (EHR) and clinical workflow SaaS developed for functional, integrative, and naturopathic medical clinics. The platform orchestrates workflows including botanical and nutritional dispensary tracking, detailed intake evaluations, and an integrated clinical AI assistant capable of evaluating patient records, structuring SOAP notes, and screening for complex herb-drug-nutrient interactions.

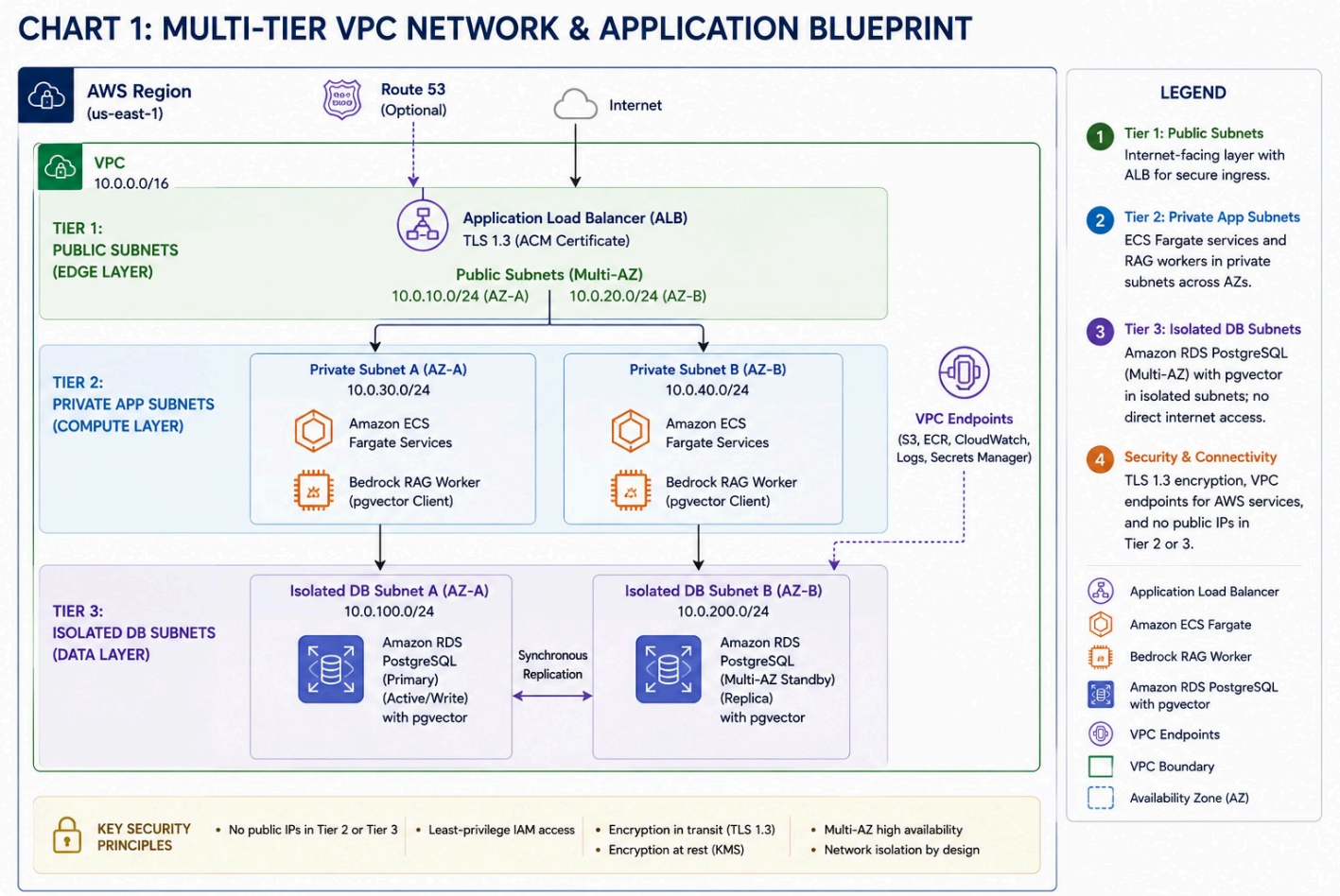
Following prototype validation in containerized local development environments, New Paradigm Systems engineered a multi-tenant, production-grade cloud platform on Amazon Web Services (AWS). This architecture fulfills technical safeguards mandated under the HIPAA Security Rule, ensures strict tenant data segregation, and provisions low-latency, private AI inference without incurring heavy GPU server maintenance or idle infrastructure debt.

Primary Architectural Goals

- **Managed Serverless Compute:** Abstract operating system maintenance and automate horizontal scaling using AWS ECS on AWS Fargate fronted by an Application Load Balancer and AWS WAF.
- **HIPAA Technical Safeguards:** Implement end-to-end data isolation with customer-managed encryption keys (AES-256 via KMS), isolated private networking, and write-once immutable audit logging (S3 Object Lock in Compliance Mode).
- **Private AI & Semantic Retrieval (RAG):** Deploy an asynchronous Retrieval-Augmented Generation pipeline leveraging Amazon RDS PostgreSQL (`pgvector`) and AWS Bedrock over AWS PrivateLink endpoints—guaranteeing zero clinical prompt traversal over the public internet.
- **Automated Infrastructure as Code (IaC):** Maintain full system reproducibility, environment parity, and rapid disaster recovery via modular Terraform deployments.

2. System Architecture & Infrastructure Blueprint

The architecture implements a defense-in-depth, three-tier network configuration distributed across multiple Availability Zones (AZs) in a dedicated Virtual Private Cloud (VPC). Compute runtimes and persistence layers remain entirely isolated from direct public ingress routes.



Component Specifications

- **Serverless Compute (AWS ECS on Fargate):** Containers run in private subnets across two AZs. Fargate eliminates underlying server host management, provides automatic security patching, natively assumes IAM execution roles, and auto-scales task instances based on real-time traffic demand.
- **Unified Relational & Vector persistence (Amazon RDS PostgreSQL 16 + `pgvector`):** Practice management records, clinical charting, and high-dimensional botanical vector embeddings reside within a single managed database cluster. This avoids the synchronization lag, failure surface, and operational overhead of maintaining a disparate third-party vector database service.
- **Private Foundation Model Orchestration (AWS Bedrock):** Clinical note evaluations leverage Anthropic Claude models accessed via managed API under an executed AWS Business Associate Agreement (BAA) with zero-retention policies. Utilizing managed models eliminates the high fixed idle compute cost of hosting custom open-weight models on dedicated GPU nodes (e.g., EC2 g5 instances).
- **AWS PrivateLink Endpoints:** Communication between ECS compute tasks, RDS, Bedrock, Secrets Manager, and S3 traverses private VPC Interface Endpoints over AWS internal backbone fiber.

- **Edge Defense (AWS WAF & Route 53):** Manages ingress routing, enforces TLS 1.3 termination at the Application Load Balancer, and blocks SQL injection, cross-site scripting (XSS), and malicious request floods.

3. Application Topology, Multi-Tenancy & AI Data Flow

Multi-Tenant Isolation via Row-Level Security (RLS)

AetheraOS employs a pooled multi-tenant database pattern. To prevent multi-tenant data cross-talk while avoiding the resource overhead of provisioning isolated database instances per clinic, tenant isolation is programmatically enforced at the PostgreSQL engine level:

1. **Cryptographic Context Injection:** Authenticated sessions carry signed JSON Web Tokens (JWT) specifying the verified `user_id` and `clinic_id`.
2. **Session State Setting:** When a pooled connection is claimed, the API runtime issues a localized session command before executing transactional queries:

   ```
   SET LOCAL app.current_clinic_id = 'clinic_oak_wellness';
   ```

3. **Engine-Level RLS Policies:** All clinical schemas enforce active Row-Level Security policies:

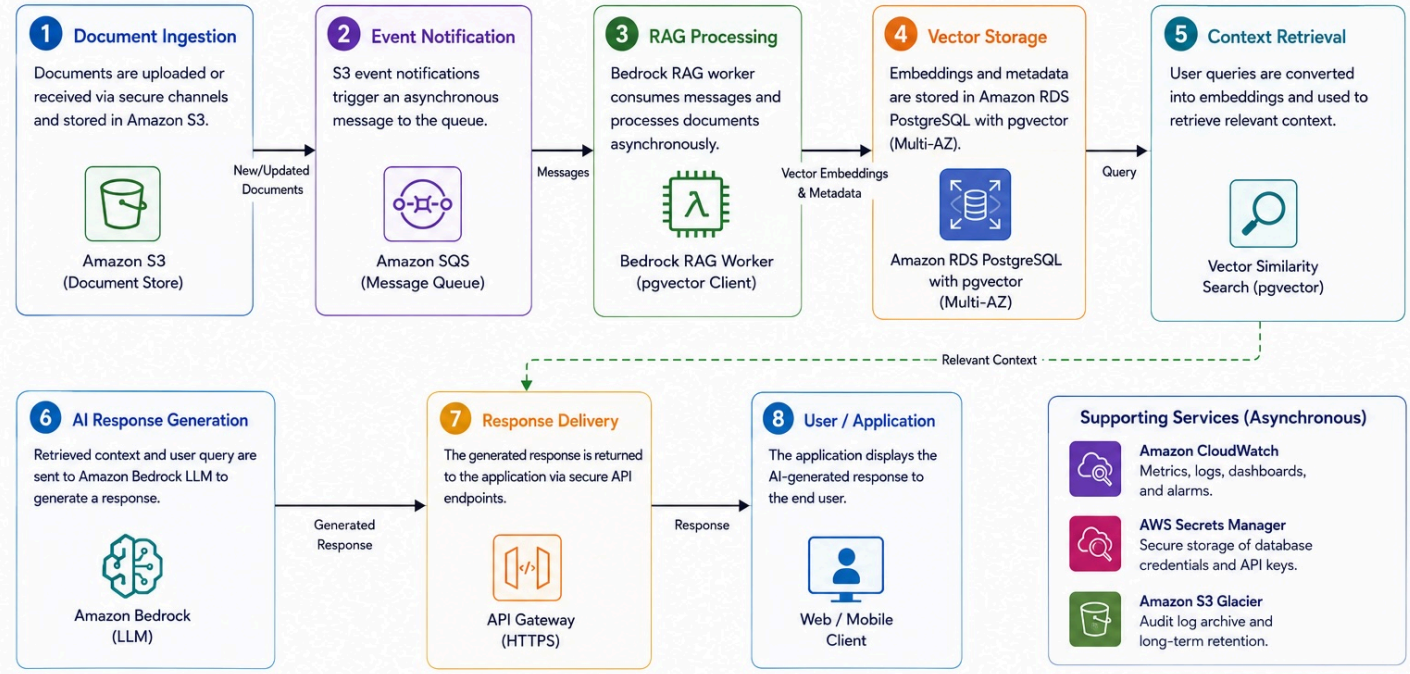
   ```
   ALTER TABLE patient_records ENABLE ROW LEVEL SECURITY;

   CREATE POLICY clinic_isolation_policy ON patient_records
   USING (clinic_id = current_setting('app.current_clinic_id'));
   ```

   Even if an application-layer endpoint executes a broad query without a localized filter (e.g., `SELECT * FROM patient_records`), the database engine limits the return set exclusively to records matching the active tenant context.

Asynchronous Retrieval-Augmented Generation (RAG) Flow

To prevent complex clinical evaluations and vector searches from blocking practitioner interactions, all AI-augmented inference jobs process asynchronously:

CHART 2: ASYNCHRONOUS AI (RAG) WORKFLOW & DATA FLOW



FLOW LEGEND

- Primary Asynchronous Flow
- Query & Context Retrieval Flow
- Response Flow
- Supporting / Monitoring Flow
- Workflow Step

DATA STORES

- Amazon S3 (Object Storage)
- Amazon SQS (Message Queue)
- Amazon RDS PostgreSQL with pgvector (Vector DB)

KEY CHARACTERISTICS

- Fully asynchronous ingestion and processing
- Scalable, decoupled architecture
- Multi-AZ high availability for data layer
- Secure, private, and HIPAA-aligned design

4. Security, Privacy & HIPAA Compliance Matrix

Under the HIPAA Shared Responsibility Model, the platform implements comprehensive technical safeguards across all core requirements of 45 CFR Part 164, Subpart C:

TABLE 2: HIPAA SECURITY RULE TECHNICAL SAFEGUARDS MATRIX

HIPAA SECURITY STANDARD	TECHNICAL SAFEGUARD IMPLEMENTATION	AWS PLATFORM MECHANISM
Transmission Security (§ 164.312(e)(1))	TLS 1.3 enforced at ALB; private subnet routing for all compute; zero internet traversal for AI queries via VPC Endpoints.	AWS ACM, AWS PrivateLink
Encryption at Rest (§ 164.312(a)(2)(iv))	All database volumes, vector embeddings, and Amazon S3 document buckets are encrypted using Customer-Managed Keys (CMKs) with AES-256.	AWS Key Management Service (KMS)
Audit Controls (§ 164.312(b))	Comprehensive database query logging via pgaudit. System, access, and pipeline logs exported to immutable, write-locked log archives.	PostgreSQL pgaudit, Amazon CloudWatch, AWS CloudTrail
Log Immutability (SOC 2 / HIPAA § 164.312)	Compliance audit logs archived to dedicated S3 buckets configured with S3 Object Lock in Compliance Mode (WORM: Write Once, Read Many) for 6 years.	Amazon S3 Object Lock
Data Integrity & DR (§ 164.312(c)(1))	Continuous Write-Ahead Log (WAL) archiving to secondary availability zones. Point-in-time recovery target: RPO < 5 minutes, RTO < 30 minutes.	Amazon RDS Automated Backups
Zero-Trust Admin (Privacy Rule § 164.502)	Master Operations interfaces access system telemetry and container health only. Connection proxies strictly forbid queries against PHI tables.	AWS IAM Least-Privilege Policies

Identity, Secrets & Least Privilege

- **Zero Long-Term IAM Keys:** ECS Fargate tasks utilize temporary AWS STS credentials via IAM Task Execution Roles, eliminating stored access keys.
- **Automated Secret Rotation:** Database credentials, third-party tokens, and JWT secrets are stored in AWS Secrets Manager and injected into task runtime memory upon startup.
- **Separation of Operational Duties:** Platform maintenance and system health monitors access infrastructure telemetry and CloudWatch metrics only. Operational proxies strictly forbid maintenance accounts from running queries against tables containing Protected Health Information (PHI).

5. Implementation Roadmap & Validation Gates

Platform deployment was executed across four continuous two-week engineering sprints, enforcing technical acceptance criteria at each boundary:

TABLE 3: PHASED IMPLEMENTATION ROADMAP & DELIVERY GATES

PHASE	MILESTONE (DELIVERY GATE)	KEY DELIVERABLES	ACCEPTANCE CRITERIA (GATE)	ESTIMATED DURATION
Phase 1 Foundation	1. Infrastructure Readiness (Gate 1)	• Multi-AZ VPC, subnets, routing, and SGs • Core IAM roles and least-privilege policies • AWS KMS keys and encryption configured • Network connectivity and endpoints validated	Infrastructure validated; security baselines approved; connectivity tests successful.	Weeks 1–2
Phase 2 Data Layer	2. Data Platform Deployment (Gate 2)	• Amazon RDS PostgreSQL (Multi-AZ) with pgvector • Database security, backups, and monitoring • Connection pooling and performance tuning • Data seed scripts and migration framework	Database operational; backups verified; performance benchmarks within targets.	Weeks 3–4
Phase 3 Application Layer	3. Application Platform Deployment (Gate 3)	• ECS Fargate services and auto-scaling • Bedrock RAG pipeline (SQS, worker, pgvector) • CI/CD pipelines and container registry • Audit logs immutably archived to S3	Application stack deployed; RAG pipeline validated; audit logging confirmed.	Weeks 5–6
Phase 4 Security & Resilience	4. Security Hardening & DR Validation (Gate 4)	• Security hardening and vulnerability remediation • WAF, TLS, and access controls validated • DR failover tests (RPO < 5 min, RTO < 30 min) • Backup restore and integrity validation	Security review passed; DR tests successful; rollback and recovery validated.	Weeks 7–8
Phase 5 Production Cutover	5. Production Launch & Handover (Gate 5)	• Production deployment and DNS cutover • Monitoring dashboards and alerting enabled • Runbooks, documentation, and team training • Final sign-off and knowledge transfer	Production cutover complete; monitoring active; client sign-off obtained.	Week 9
TOTAL PROJECT DURATION From project kickoff to production cutover				9 WEEKS

6. Cloud Economics & Operational Efficiency

By avoiding over-provisioned Kubernetes clusters and dedicated 24/7 GPU instances, the serverless cloud design maintains a lean cost profile while retaining high reliability and compliance guarantees:

> **Baseline Operational Cost Profile (25 Clinics, ~750 DAUs):**
>
> - **Serverless Compute (ECS Fargate Web, API, & Workers):** ~$60.00 / month
> - **Relational & Vector Data Store (Multi-AZ RDS PostgreSQL 16):** ~$112.00 / month
> - **Network & Edge Protection (ALB, Route 53, NAT Gateways):** ~$92.00 / month
> - **Managed AI & Embeddings (AWS Bedrock & Titan):** ~$45.00 / month
> - **Compliance Archival, Storage & Telemetry (S3 WORM, CloudWatch):** ~$47.00 / month
> - **Total Baseline Cloud Expenditure: ~$356.00 / month** (averaging **~$14.25 per clinic/month**)

This model allows digital health startups to scale efficiently, onboarding clinics incrementally without fixed infrastructure penalties or runaway compute overhead.