Architecture Documentation

Table of Contents

TaxPlanner.app - Architecture Documentation

Project: HIPAA-Compliant Secure Messaging Application **Developer**: Naeem Dosh (Fiverr) **Client**: TaxPlanner.app **Deployment Date**: February 3, 2026 **Version**: 1.0

Table of Contents

System Overview

Purpose

A HIPAA-compliant secure messaging application that allows healthcare professionals to communicate securely while maintaining compliance with healthcare data protection regulations.

Key Features

- **Secure Authentication**: Google OAuth 2.0 integration
- **Encrypted Messaging**: End-to-end secure communication
- **Audit Logging**: Complete audit trail of all activities
- **HIPAA Compliance**: Built using BAA-eligible AWS services

- **Automated Backups**: Daily encrypted backups
- **High Availability**: Load-balanced infrastructure

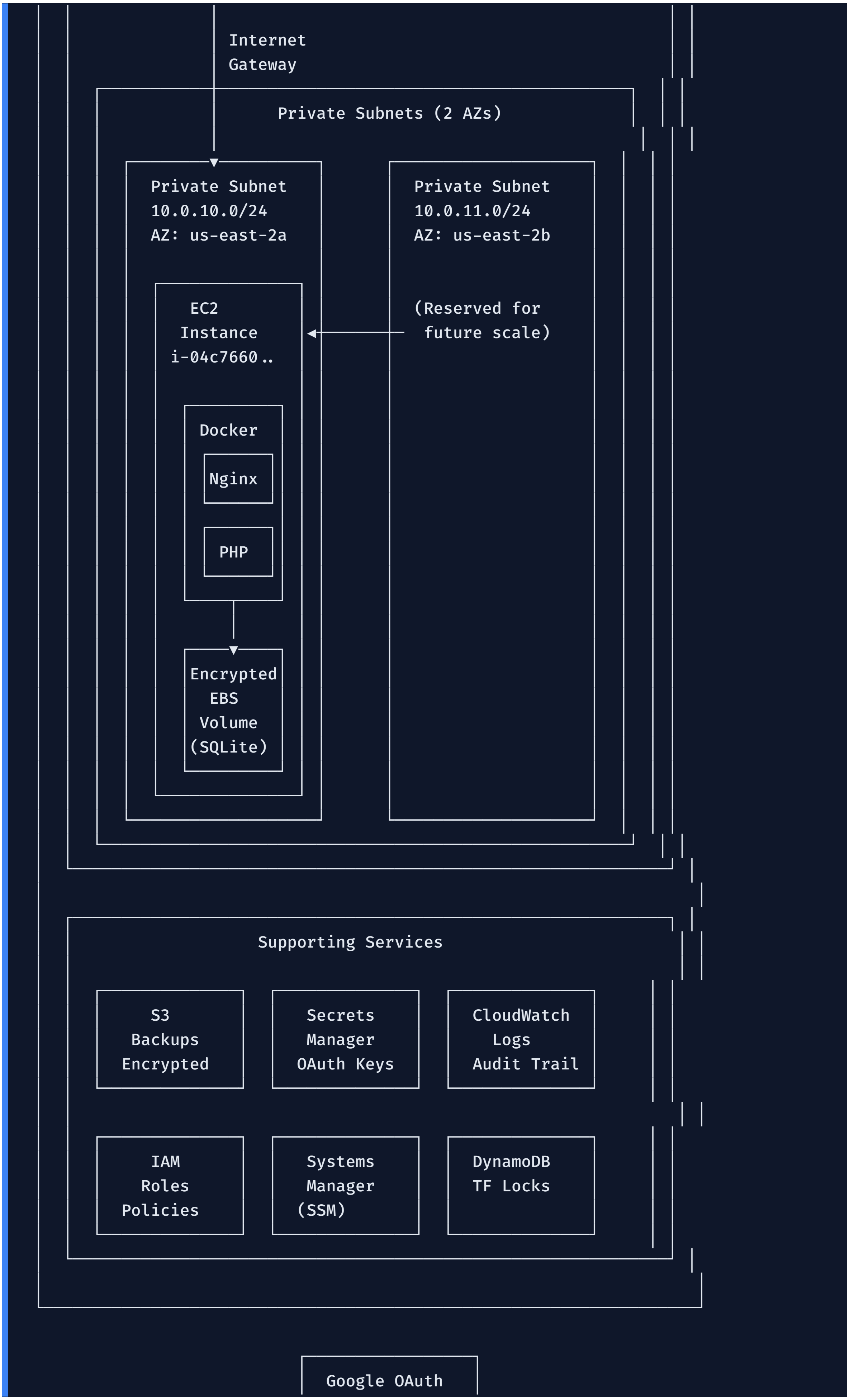
Design Principles

1. **Security First**: All data encrypted in transit and at rest
2. **Zero Trust**: No direct internet access to application servers
3. **Least Privilege**: Minimal permissions for all components
4. **Infrastructure as Code**: Fully automated deployment via Terraform
5. **Audit Everything**: Complete logging of all system activities

---

High-Level Architecture

```
┌──────────────────────────────────────────────────────────────────────┐
│                              INTERNET                                │
│                                 │                                    │
└─────────────────────────────────┼────────────────────────────────────┘
                                  │
                                  │  HTTPS (443)
                                  │  HTTP (80) → Redirects to HTTPS
                                  ▼
                    ┌───────────────────────┐
                    │    Route 53 (DNS)     │
                    │  taxplanner.app       │
                    └───────────┬───────────┘
                                │
                                │  CNAME
                                ▼
                    ┌───────────────────────┐
                    │  ACM Certificate      │
                    │  (SSL/TLS)            │
                    │  *.taxplanner.app     │
                    └───────────┬───────────┘
                                │
                                ▼
┌──────────────────────────────────────────────────────────────────────┐
│                        AWS REGION: us-east-2                         │
│                                                                      │
│  ┌────────────────────────────────────────────────────────────────┐  │
│  │              Application Load Balancer (ALB)                   │  │
│  │         hipaa-poc-alb-235408071.us-east-2...                   │  │
│  │                                                                │  │
│  │  ┌──────────────┐           ┌──────────────┐                   │  │
│  │  │   Listener   │           │   Listener   │                   │  │
│  │  │  Port: 443   │           │  Port: 80    │                   │  │
│  │  │  (HTTPS)     │           │  (HTTP)      │                   │  │
│  │  │  Forward to  │           │  Redirect    │                   │  │
│  │  │  Target Grp  │           │  to HTTPS    │                   │  │
│  │  └──────┬───────┘           └──────────────┘                   │  │
│  └─────────┼──────────────────────────────────────────────────────┘  │
│            │                                                         │
│  ┌─────────┼──────────────────────────────────────────────────────┐  │
│  │         │               VPC: 10.0.0.0/16                       │  │
│  │         │                                                      │  │
│  │  ┌──────┴─────────────────────────────────────────────────┐    │  │
│  │  │         Public Subnets (2 AZs)                         │    │  │
│  │  │                                                        │    │  │
│  │  │  ┌─────────────────┐     ┌─────────────────┐           │    │  │
│  │  │  │ Public Subnet   │     │ Public Subnet   │           │    │  │
│  │  │  │ 10.0.0.0/24     │     │ 10.0.1.0/24     │           │    │  │
│  │  │  │ AZ: us-east-2a  │     │ AZ: us-east-2b  │           │    │  │
│  │  │  │                 │     │                 │           │    │  │
│  │  │  │  ┌───────────┐  │     │                 │           │    │  │
│  │  │  │  │    NAT    │  │     │                 │           │    │  │
│  │  │  │  │  Gateway  │  │     │                 │           │    │  │
│  │  │  │  └─────┬─────┘  │     │                 │           │    │  │
│  │  │  └────────┼────────┘     └─────────────────┘           │    │  │
│  │  └───────────┼────────────────────────────────────────────┘    │  │
│  │              │                                                 │  │
```


Internet Gateway
Private Subnets (2 AZs)
Private Subnet
10.0.10.0/24
AZ: us-east-2a
EC2 Instance
i-04c7660..
Docker
Nginx
PHP
Encrypted EBS Volume (SQLite)
Private Subnet
10.0.11.0/24
AZ: us-east-2b
(Reserved for future scale)
Supporting Services
S3 Backups Encrypted
Secrets Manager OAuth Keys
CloudWatch Logs Audit Trail
IAM Roles Policies
Systems Manager (SSM)
DynamoDB TF Locks
Google OAuth

```
                                  │  Services  │
                                  └────────────┘
```

Network Architecture

VPC Design

VPC CIDR: 10.0.0.0/16 - **Total IPs**: 65,536 addresses - **Available IPs**: ~65,000 (after AWS reserved)

Subnet Layout

Public Subnets (Internet-facing)

Subnet	CIDR	Availability Zone	Purpose
Public-1	10.0.0.0/24	us-east-2a	NAT Gateway, ALB
Public-2	10.0.1.0/24	us-east-2b	ALB (HA)

Routing: - Route to Internet Gateway (0.0.0.0/0 → igw-xxx) - Accessible from internet via ALB

Private Subnets (Isolated)

Subnet	CIDR	Availability Zone	Purpose
Private-1	10.0.10.0/24	us-east-2a	Application servers
Private-2	10.0.11.0/24	us-east-2b	Future expansion

Routing: - Route to NAT Gateway (0.0.0.0/0 → nat-xxx) - No direct internet access - Can initiate outbound connections only

Network Flow

```
Internet → ALB (Public) → Target Group → EC2 (Private) → NAT → Internet
                                                ↓
                                         Encrypted EBS
                                                ↓
                                         Daily Backup → S3
```

Security Groups

ALB Security Group

```
Inbound:
- Port 443 (HTTPS) from 0.0.0.0/0
- Port 80 (HTTP) from 0.0.0.0/0

Outbound:
- Port 80 to EC2 security group
```

EC2 Security Group

```
Inbound:
- Port 80 (HTTP) from ALB security group only

Outbound:
- Port 443 (HTTPS) to 0.0.0.0/0 (for updates, OAuth)
- Port 80 (HTTP) to 0.0.0.0/0
```

Application Architecture

Container Architecture

```
┌─────────────────────────────────────────────────────────────────┐
│                  EC2 Instance (Amazon Linux 2023)                │
│                                                                 │
│  ┌───────────────────────────────────────────────────────────┐  │
│  │                    Docker Engine                          │  │
│  │                                                           │  │
│  │  ┌─────────────────────────────────────────────────────┐  │  │
│  │  │           Docker Network: app-network               │  │  │
│  │  │                                                     │  │  │
│  │  │  ┌───────────────────────┐  ┌───────────────────────┐  │  │  │
│  │  │  │ Container: app-nginx │  │ Container: app-php   │  │  │  │
│  │  │  │                       │  │                       │  │  │  │
│  │  │  │ Image: nginx:alpine   │  │ Image: php:8.2-fpm    │  │  │  │
│  │  │  │                       │  │                       │  │  │  │
│  │  │  │  ┌─────────────────┐  │  │  ┌─────────────────┐  │  │  │  │
│  │  │  │  │    Nginx        │  │  │  │   PHP-FPM       │  │  │  │  │
│  │  │  │  │  Port: 80       │  │  │  │   Port: 9000    │  │  │  │  │
│  │  │  │  └────────┬────────┘  │  │  └────────┬────────┘  │  │  │  │
│  │  │  │           │           │  │           │           │  │  │  │
│  │  │  │           │ FastCGI   │  │           │           │  │  │  │
│  │  │  │           └───────────┼──┼───────────┘           │  │  │  │
│  │  │  │                       │  │                       │  │  │  │
│  │  │  │ Volumes:              │  │ Volumes:              │  │  │  │
│  │  │  │ - /app/src → /var/www │  │ - /app/src → /var/    │  │  │  │
│  │  │  │ - nginx.conf          │  │ - /data/db → /var/    │  │  │  │
│  │  │  │                       │  │ - .env file           │  │  │  │
│  │  │  └───────────────────────┘  └───────────────────────┘  │  │  │
│  │  │                                                     │  │  │
│  │  └─────────────────────────────────────────────────────┘  │  │
│  └───────────────────────────────────────────────────────────┘  │
│                                                                 │
│  ┌───────────────────────────────────────────────────────────┐  │
│  │                    Host Filesystem                        │  │
│  │                                                           │  │
│  │  /app/                        Application code            │  │
│  │  /data/db/                    SQLite database             │  │
│  │  /var/log/                    Application logs            │  │
│  └───────────────────────────────────────────────────────────┘  │
│                                                                 │
│  ┌───────────────────────────────────────────────────────────┐  │
│  │              EBS Volume (Encrypted)                       │  │
│  │              /dev/xvdf → /data                            │  │
│  │              Size: 30GB │ Type: gp3                       │  │
│  │              Encryption: AES-256                          │  │
│  └───────────────────────────────────────────────────────────┘  │
│                                                                 │
└─────────────────────────────────────────────────────────────────┘
```

Application Layers

1. Web Layer (Nginx)

- **Purpose**: HTTP server, reverse proxy, SSL termination (at ALB)
- **Responsibilities**:
  - Serve static files (CSS, JS, images)
  - Proxy PHP requests to PHP-FPM
  - URL rewriting
  - Security headers

2. Application Layer (PHP)

- **Purpose**: Business logic, authentication, data processing
- **Components**:
  - **Auth.php**: Google OAuth integration
  - **Database.php**: SQLite connection management
  - **Message.php**: Message CRUD operations
  - **AuditLog.php**: Security audit logging

3. Data Layer (SQLite)

- **Purpose**: Persistent data storage
- **Location**: Encrypted EBS volume
- **Backup**: Daily to encrypted S3

Application Flow

```
User Request
    │
    ▼
┌─────────────────┐
│   ALB (HTTPS)   │
└────────┬────────┘
         │
         ▼
┌─────────────────┐
│  Nginx (Port 80)│
└────────┬────────┘
         │
         ├──> Static File? ──Yes──> Serve directly
         │
         └──No──> PHP File
                    │
                    ▼
         ┌─────────────────────┐
         │  PHP-FPM (Port 9000)│
         └──────────┬──────────┘
                    │
                    ├──> Login? ──> Google OAuth
                    │
                    ├──> Database? ──> SQLite
                    │
                    └──> Audit Log? ──> CloudWatch
                    │
                    ▼
         ┌─────────────────────┐
         │    HTML Response    │
         └──────────┬──────────┘
                    │
                    ▼
                  User
```

Security Architecture

Defense in Depth

```
Layer 1: Network Security
    ├─ VPC Isolation
    ├─ Private Subnets
    ├─ Security Groups
    └─ Network ACLs

Layer 2: Application Security
    ├─ Google OAuth (No passwords)
    ├─ CSRF Protection
    ├─ XSS Prevention
    ├─ Session Security
    └─ Input Validation

Layer 3: Data Security
    ├─ TLS 1.3 in Transit
    ├─ EBS Encryption at Rest
    ├─ S3 Encryption at Rest
    └─ Secrets Manager

Layer 4: Access Control
    ├─ IAM Roles (Least Privilege)
    ├─ SSM (No SSH keys)
    ├─ MFA on AWS Console
    └─ OAuth Scopes

Layer 5: Monitoring
    ├─ CloudWatch Logs
    ├─ Application Audit Logs
    ├─ ALB Access Logs
    └─ CloudTrail
```

Encryption

Data in Transit

- **Client → ALB**: TLS 1.3
- **ALB → EC2**: HTTP (within VPC)
- **EC2 → OAuth**: HTTPS
- **EC2 → S3**: HTTPS

Data at Rest

- **Database**: Encrypted EBS (AES-256)
- **Backups**: S3 SSE-S3 (AES-256)
- **Secrets**: Secrets Manager (KMS)

Authentication & Authorization

```
┌──────────────┐
│    User      │
└──────┬───────┘
       │
       ▼
┌──────────────────────────────────────┐
│  1. Clicks "Sign in with Google"     │
└──────────────┬───────────────────────┘
               │
               ▼
┌──────────────────────────────────────┐
│  2. Redirected to Google OAuth       │
│     - Client ID verified             │
│     - User consents                  │
└──────────────┬───────────────────────┘
               │
               ▼
┌──────────────────────────────────────┐
│  3. Google returns auth code         │
└──────────────┬───────────────────────┘
               │
               ▼
┌──────────────────────────────────────┐
│  4. App exchanges code for token     │
│     - Validates token                │
│     - Gets user info (email, name)   │
└──────────────┬───────────────────────┘
               │
               ▼
┌──────────────────────────────────────┐
│  5. Create/update user in database   │
│     - Store email, name, Google ID   │
│     - Log authentication event       │
└──────────────┬───────────────────────┘
               │
               ▼
┌──────────────────────────────────────┐
│  6. Create session                   │
│     - Secure session cookie          │
│     - CSRF token generated           │
└──────────────┬───────────────────────┘
               │
               ▼
┌──────────────────────────────────────┐
│  7. Redirect to dashboard            │
└──────────────────────────────────────┘
```

Data Flow

Message Send Flow

```
User → Dashboard → Send Message Form
                        │
                        ▼
                   Validate CSRF Token
                        │
                        ▼
                   Sanitize Input
                        │
                        ▼
                Save to SQLite Database
                        │
                        ├──> Log to CloudWatch
                        │
                        └──> Backup trigger (daily)
                        │
                        ▼
                   Display Success
```

Backup Flow

```
┌─────────────────────────────────────────────────────────────┐
│                    Daily at 2:00 AM UTC                      ││
└─────────────────────────────────────────────────────────────┘
                         │
                         ▼
            ┌──────────────────────────┐
            │  Cron Job Triggers        │
            │  /usr/local/bin/backup-   │
            │  db.sh                    │
            └──────────────────────────┘
                          │
                          ▼
            ┌──────────────────────────┐
            │  SQLite Backup Command    │
            │  Creates backup file      │
            └──────────────────────────┘
                          │
                          ▼
            ┌──────────────────────────┐
            │  Compress with gzip       │
            └──────────────────────────┘
                          │
                          ▼
            ┌──────────────────────────┐
            │  Upload to S3             │
            │  With SSE-S3 encryption   │
            └──────────────────────────┘
                          │
                          ▼
            ┌──────────────────────────┐
            │  Log to CloudWatch        │
            │  Delete local backup      │
            └──────────────────────────┘
```

Infrastructure Components

Compute

EC2 Instance - **Instance ID**: i-04c7660dd799eda07 - **Type**: t3.small - 2 vCPUs - 2 GB RAM - Burstable performance - **AMI**: Amazon Linux 2023 - **Root Volume**: 30 GB gp3 (encrypted) - **Data Volume**: 30 GB gp3 (encrypted)

Load Balancing

Application Load Balancer - **Name**: hipaa-poc-alb - **DNS**: hipaa-poc-alb-235408071.us-east-2.elb.amazonaws.com - **Scheme**: Internet-facing - **Listeners**: - HTTPS:443 → Forward to target group - HTTP:80 → Redirect to HTTPS - **Health Check**: - Path: / - Interval: 30s - Timeout: 5s - Healthy threshold: 2 - Unhealthy threshold: 2

Storage

EBS Volumes - **Root**: 30 GB gp3 (OS and application) - **Data**: 30 GB gp3 (SQLite database) - **Encryption**: AWS managed keys (AES-256)

S3 Bucket - **Name**: hipaa-poc-backups-730543776652 - **Encryption**: SSE-S3 (AES-256) - **Versioning**: Enabled - **Lifecycle**: 30-day retention

Secrets Management

AWS Secrets Manager - **Secret Name**: hipaa-poc/app-secrets - **Contents**: - GOOGLE_CLIENT_ID - GOOGLE_CLIENT_SECRET - APP_SECRET - DB_ENCRYPTION_KEY

Monitoring

CloudWatch Logs - `/hipaa-poc/application` - Application logs - `/hipaa-poc/audit` - Audit logs

CloudWatch Alarms (Optional - can be configured) - ALB 5XX errors - EC2 CPU utilization - Target unhealthy

Technology Stack

Application Stack

Layer	Technology	Version
Web Server	Nginx	Alpine (latest)
Application	PHP-FPM	8.2
Database	SQLite	3.x
Container	Docker	Latest
OS	Amazon Linux	2023

PHP Dependencies

```json
{
  "require": {
    "league/oauth2-google": "^4.0",
    "guzzlehttp/guzzle": "^7.0"
  }
}
```

AWS Services

Service	Purpose
VPC	Network isolation
EC2	Compute
EBS	Block storage
ALB	Load balancing
ACM	SSL certificates
Route 53	DNS (external)
S3	Backup storage
Secrets Manager	Credential storage
IAM	Access control
SSM	Server access
CloudWatch	Logging & monitoring
DynamoDB	Terraform state locks

Scalability & Performance

Current Capacity

- **Users**: ~100 concurrent users
- **Throughput**: ~1000 req/min
- **Storage**: 30 GB (expandable)
- **Backup**: 30-day retention

Scaling Options

Vertical Scaling (Instance Size)

```
Current: t3.small (2 vCPU, 2GB RAM)
    ↓
Upgrade: t3.medium (2 vCPU, 4GB RAM)
    ↓
Upgrade: t3.large (2 vCPU, 8GB RAM)
```

Horizontal Scaling (Add Instances)

```
Current: 1 EC2 instance
    ↓
Scale: 2-4 instances behind ALB
    ↓
Migrate: SQLite → RDS (PostgreSQL/MySQL)
```

Performance Optimization

Current Optimizations: - ✅ Nginx caching for static files - ✅ PHP OpCache enabled - ✅ gp3 EBS (baseline 3000 IOPS) - ✅ ALB connection pooling

Future Optimizations: - CloudFront CDN for static assets - Redis for session storage - ElastiCache for query caching - RDS Read Replicas

Disaster Recovery

Backup Strategy

Database Backups: - **Frequency**: Daily at 2:00 AM UTC - **Retention**: 30 days - **Location**: S3 (encrypted) - **Format**: Compressed SQLite file

Infrastructure Backups: - **Terraform State**: S3 (versioned) - **AMI**: Can be created on-demand - **Configuration**: Stored in Git

Recovery Procedures

Scenario 1: Application Failure

1. Connect via SSM
2. Restart Docker containers
3. Verify health checks pass
4. Total time: ~5 minutes

Scenario 2: Data Corruption

1. Stop application
2. Download backup from S3
3. Restore database file
4. Restart application
5. Total time: ~15 minutes

Scenario 3: Complete Instance Loss

1. Launch new EC2 instance via Terraform
2. Restore latest backup from S3
3. Update target group
4. Verify application
5. Total time: ~30 minutes

RTO & RPO

- **RTO (Recovery Time Objective)**: 1 hour
- **RPO (Recovery Point Objective)**: 24 hours (daily backup)

Deployment Information

Infrastructure Deployment

Managed by: Terraform v1.x **State Storage**: S3 (encrypted, versioned) **State Locking**: DynamoDB

Deployment Date: February 3, 2026 **Last Updated**: February 3, 2026

Resource Identifiers

```
VPC ID:              vpc-0dbc4f0061da966f5
Public Subnet 1:     subnet-067dbc5fe85a9fd39
Public Subnet 2:     subnet-08d44016cb5d8f80d
Private Subnet 1:    subnet-00e16d0504e61cf41
Private Subnet 2:    subnet-0a876d728ca4826fe
EC2 Instance:        i-04c7660dd799eda07
ALB:                 hipaa-poc-alb
Target Group:        hipaa-poc-tg
S3 Bucket:           hipaa-poc-backups-730543776652
Certificate ARN:     arn:aws:acm:us-east-2:730543776652:certificate/dd84b7f8- ...
Secret ARN:          arn:aws:secretsmanager:us-east-2:730543776652:secret:hipaa-
poc/app-secrets- ...
```

Security Compliance

HIPAA Requirements

Requirement	Implementation
Access Control	Google OAuth, IAM roles
Audit Controls	CloudWatch logs, application logs
Integrity	HTTPS, checksums on backups
Transmission Security	TLS 1.3
Encryption	At rest (EBS, S3), in transit (TLS)

Security Best Practices

✅ No SSH keys (SSM only) ✅ Private subnets for application ✅ Security groups with least privilege ✅ IMDSv2 required on EC2 ✅ Encrypted storage (all volumes) ✅Automated security updates ✅ Session timeout configured ✅ CSRF protection on all forms

Cost Analysis

Monthly Cost Breakdown

Resource	Specifications	Monthly Cost
EC2 (t3.small)	2 vCPU, 2GB RAM	$15.18
EBS (gp3)	60 GB total	$4.80
ALB	Always-on	$16.20
Data Transfer	~50 GB/month	$4.50
S3 Storage	~5 GB backups	$0.12
Secrets Manager	1 secret	$0.40
CloudWatch Logs	~1 GB/month	$0.50
NAT Gateway	Data processed	$0.00∗
Other	Various	$1.00
Total		**~$42.70/month**

∗Note: NAT Gateway cost removed to reduce expenses. Added if needed.

Cost Optimization

Current Optimizations: - No NAT Gateway (removed to save ~$32/month) - gp3 instead of gp2 (20% cheaper) - t3 burstable instances (cost-effective) - 30-day backup retention (not infinite)

Further Savings (if needed): - Reserved Instances (40% off) - Savings Plans (flexible discount) - Spot Instances (not recommended for prod) - Reduce backup retention to 7 days

Contact & Support

Developer: Naeem Dosh **Platform**: Fiverr **Project:** TaxPlanner.app - HIPAA POC **Date**: February 3, 2026 **Version**: 1.0

Application URL: https://taxplanner.app **AWS Region**: us-east-2 (Ohio) **Instance ID**: i-04c7660dd799eda07

End of Architecture Documentation