Genesis Provenance - Complete Deployment Guide

Last Updated: November 29, 2025
Platform: Vercel
Domain: genesisprovenance.com
Email: Microsoft 365

📋 Pre-Deployment Checklist

Before you begin, ensure you have:

- [ ] GitHub account
- [ ] Vercel account (free tier works)
- [ ] GoDaddy account (domain registered)
- [ ] Microsoft 365 account (for email)
- [ ] AWS account (for S3 storage)
- [ ] Project code ready to push

Part 1: GitHub Setup (15 minutes)

Step 1: Create GitHub Repository

1.1 Go to GitHub

```
https://github.com/new
```

1.2 Fill Repository Details

- **Repository name:** `genesis-provenance`
- **Description:** "AI-powered provenance vault for luxury assets"
- **Visibility:** Private (recommended) or Public
- **Initialize:** DO NOT check "Add README" (you already have one)

1.3 Click "Create repository"

Step 2: Push Your Code to GitHub

2.1 Open Terminal in Your Project

```
cd /home/ubuntu/genesis_provenance/nextjs_space
```

2.2 Initialize Git (if not already done)

```
git init
```

2.3 Create .gitignore File

```
cat > .gitignore << 'EOF'
# Dependencies
node_modules/
.pnp
.pnp.js

# Testing
coverage/

# Next.js
.next/
out/
build/
.build/

# Production
dist/

# Misc
.DS_Store
*.pem

# Debug
npm-debug.log*
yarn-debug.log*
yarn-error.log*
.pnpm-debug.log*

# Local env files
.env
.env*.local
.env.local
.env.development.local
.env.test.local
.env.production.local

# Vercel
.vercel

# TypeScript
*.tsbuildinfo
next-env.d.ts

# IDE
.vscode/
.idea/
*.swp
*.swo
*~

# OS
Thumbs.db

# Prisma
prisma/*.db
prisma/*.db-journal
EOF
```

2.4 Add All Files

```
git add .
```

2.5 Commit

```
git commit -m "Initial commit - Phase 1.5 complete with self-service model and
professional UI"
```

2.6 Add GitHub Remote

Replace `YOUR_GITHUB_USERNAME` with your actual GitHub username:

```
git remote add origin https://github.com/YOUR_GITHUB_USERNAME/genesis-provenance.git
```

2.7 Push to GitHub

```
git branch -M main
git push -u origin main
```

✅ **GitHub Setup Complete!**

Part 2: Database Setup (10 minutes)

Option A: Vercel Postgres (Recommended - Easiest)

Step 1: Create Vercel Postgres Database

1. Go to Vercel Dashboard: https://vercel.com/dashboard
2. Click "Storage" in the top menu
3. Click "Create Database"
4. Select **"Postgres"**
5. Name: `genesis-provenance-db`
6. Region: Choose closest to your users (e.g., US East for USA)
7. Click "Create"

Step 2: Get Connection String

1. Click on your new database
2. Go to ".env.local" tab
3. Copy the `POSTGRES_PRISMA_URL` value
4. **Save this for later** - you'll need it in Vercel environment variables

Example format:

```
postgres://default:xxxxx@xxxxx.postgres.vercel-storage.com:5432/verceldb?sslmode=re-
quire
```

Option B: Neon (Alternative - Also Free)

Step 1: Create Neon Account

1. Go to: https://neon.tech
2. Sign up with GitHub
3. Click "Create a project"

Step 2: Configure Project

- **Project name:** `genesis-provenance`
- **Postgres version:** 16 (latest)
- **Region:** Choose closest to your users
- Click "Create project"

Step 3: Get Connection String

1. On project dashboard, find "Connection string"
2. Select "Prisma" from dropdown
3. Copy the connection string
4. **Save this for later**

Example format:

```
postgresql://username:password@ep-xxxxx.region.aws.neon.tech/neondb?sslmode=require
```

Part 3: AWS S3 Setup (15 minutes)

Step 1: Create S3 Bucket

1.1 Log into AWS Console

```
https://console.aws.amazon.com/s3
```

1.2 Click "Create bucket"

1.3 Configure Bucket

- **Bucket name:** `genesis-provenance-assets` (must be globally unique)
- **AWS Region:** `us-east-1` (or your preferred region)
- **Block Public Access:** KEEP ALL CHECKED (we'll use signed URLs)
- **Bucket Versioning:** Enable (recommended)
- **Encryption:** Enable (default is fine)

1.4 Click "Create bucket"

Step 2: Configure CORS

2.1 Click on Your Bucket

2.2 Go to "Permissions" Tab

2.3 Scroll to "Cross-origin resource sharing (CORS)"

2.4 Click "Edit" and Paste:

```
[
  {
    "AllowedHeaders": ["*"],
    "AllowedMethods": ["GET", "PUT", "POST", "DELETE"],
    "AllowedOrigins": [
      "https://genesisprovenance.com",
      "https://www.genesisprovenance.com",
      "https://app.genesisprovenance.com",
      "http://localhost:3000"
    ],
    "ExposeHeaders": ["ETag"],
    "MaxAgeSeconds": 3000
  }
]
```

2.5 Click "Save changes"

Step 3: Create IAM User for S3 Access

3.1 Go to IAM Console

```
https://console.aws.amazon.com/iam
```

3.2 Click "Users" → "Create user"

3.3 Configure User

- **User name:** `genesis-provenance-s3-user`
- **Access type:** Programmatic access
- Click "Next"

3.4 Set Permissions

- Click "Attach policies directly"
- Search for `AmazonS3FullAccess`
- Check the box
- Click "Next" → "Create user"

3.5 Get Access Keys

1. Click on the user you just created
2. Go to "Security credentials" tab
3. Click "Create access key"
4. Select "Application running outside AWS"

5. Click "Next" → "Create access key"
6. **IMPORTANT:** Copy both:
   - Access key ID
   - Secret access key
7. **Save these securely** - you can't view the secret again!

✅ **AWS S3 Setup Complete!**

Part 4: Vercel Deployment (10 minutes)

Step 1: Import Project to Vercel

1.1 Go to Vercel

```
https://vercel.com/new
```

1.2 Import Git Repository

- Click "Import Git Repository"
- Select your GitHub account
- Find `genesis-provenance` repository
- Click "Import"

Step 2: Configure Project Settings

2.1 Project Settings

- **Project Name:** `genesis-provenance`
- **Framework Preset:** Next.js (auto-detected)
- **Root Directory:** `./` (leave as is)
- **Build Command:** `yarn build` (auto-detected)
- **Output Directory:** `.next` (auto-detected)

2.2 DO NOT DEPLOY YET

- Click "Configure Project" to expand environment variables

Step 3: Add Environment Variables

3.1 Click "Environment Variables"

Add these variables ONE BY ONE:

Required Variables:

1. DATABASE_URL

```
Key: DATABASE_URL
Value: [Your Postgres connection string from Part 2]
Environment: Production, Preview, Development
```

2. NEXTAUTH_URL

```
Key: NEXTAUTH_URL
Value: https://app.genesisprovenance.com
Environment: Production
```

For Preview/Development, add separately:

```
Key: NEXTAUTH_URL
Value: http://localhost:3000
Environment: Preview, Development
```

3. NEXTAUTH_SECRET

Generate a secret (run this locally):

```
openssl rand -base64 32
```

Then add:

```
Key: NEXTAUTH_SECRET
Value: [Your generated secret]
Environment: Production, Preview, Development
```

4. AWS_ACCESS_KEY_ID

```
Key: AWS_ACCESS_KEY_ID
Value: [Your AWS Access Key ID from Part 3]
Environment: Production, Preview, Development
```

5. AWS_SECRET_ACCESS_KEY

```
Key: AWS_SECRET_ACCESS_KEY
Value: [Your AWS Secret Access Key from Part 3]
Environment: Production, Preview, Development
```

6. AWS_REGION

```
Key: AWS_REGION
Value: us-east-1
Environment: Production, Preview, Development
```

7. AWS_BUCKET_NAME

```
Key: AWS_BUCKET_NAME
Value: genesis-provenance-assets
Environment: Production, Preview, Development
```

8. AWS_FOLDER_PREFIX (optional)

```
Key: AWS_FOLDER_PREFIX
Value: prod/
Environment: Production
```

9. NODE_ENV

```
Key: NODE_ENV
Value: production
Environment: Production
```

3.2 Click "Deploy"

⏳ **Wait 2-5 minutes for deployment...**

Step 4: Run Database Migrations

4.1 After Deployment Succeeds

1. Go to your Vercel project dashboard
2. Click "Settings" → "Functions"
3. Note: We need to run migrations

4.2 Option A: Using Vercel CLI (Recommended)

Install Vercel CLI locally:

```
npm install -g vercel
```

Login to Vercel:

```
vercel login
```

Link your project:

```
cd /home/ubuntu/genesis_provenance/nextjs_space
vercel link
```

Pull environment variables:

```
vercel env pull .env.production
```

Run migrations:

```
DATABASE_URL="[your-production-database-url]" npx prisma migrate deploy
DATABASE_URL="[your-production-database-url]" npx prisma db seed
```

4.2 Option B: Using Database GUI

1. Connect to your Postgres database using a GUI tool:
   - **Vercel Postgres:** Use the SQL editor in Vercel dashboard
   - **Neon:** Use their SQL editor
   - **TablePlus/DBeaver:** Use connection string

2. Run the schema manually:

```
cd /home/ubuntu/genesis_provenance/nextjs_space
npx prisma migrate deploy --preview-feature
```

✅ **Vercel Deployment Complete!**

Your app is now live at: `https://genesis-provenance.vercel.app`

Part 5: Custom Domain Setup (GoDaddy) (20 minutes)

Step 1: Add Domain to Vercel

1.1 In Vercel Dashboard

1. Go to your project
2. Click "Settings" → "Domains"
3. Click "Add"

1.2 Add Primary Domain

```
genesisprovenance.com
```

Click "Add"

1.3 Add WWW Subdomain

```
www.genesisprovenance.com
```

Click "Add" → Select "Redirect to genesisprovenance.com"

1.4 Add App Subdomain

```
app.genesisprovenance.com
```

Click "Add"

1.5 Vercel Will Show Required DNS Records

Example:

```
Type: A
Name: @
Value: 76.76.21.21

Type: CNAME
Name: www
Value: cname.vercel-dns.com

Type: CNAME
Name: app
Value: cname.vercel-dns.com
```

IMPORTANT: Copy these values - you'll need them in GoDaddy

Step 2: Configure DNS in GoDaddy

2.1 Log into GoDaddy

```
https://dcc.godaddy.com/manage/genesisprovenance.com/dns
```

2.2 Add/Edit DNS Records

A Record (Root Domain)

1. Click "Add" or edit existing A record
2. Configure:
   - **Type:** A
   - **Name:** @ (represents root domain)
   - **Value:** `76.76.21.21` (Vercel's IP - use the one Vercel provided)
   - **TTL:** 600 seconds (10 minutes)
3. Click "Save"

CNAME for WWW

1. Click "Add"
2. Configure:
   - **Type:** CNAME
   - **Name:** www
   - **Value:** `cname.vercel-dns.com` (from Vercel)
   - **TTL:** 600 seconds
3. Click "Save"

CNAME for APP

1. Click "Add"
2. Configure:
   - **Type:** CNAME
   - **Name:** app

    - **Value:** `cname.vercel-dns.com` (from Vercel)
    - **TTL:** 600 seconds

3. Click "Save"

2.3 Delete Conflicting Records

If you see errors, delete these if they exist:
- Old A records for @ (except the new Vercel one)
- CNAME record for @ (can't have both A and CNAME)
- Parking page redirects

Step 3: Verify DNS Configuration

3.1 Wait for DNS Propagation

• DNS changes take 5-60 minutes

• Usually 10-15 minutes

3.2 Check DNS Propagation

Use online tools:

```
https://dnschecker.org
```

Enter:
- `genesisprovenance.com`
- `www.genesisprovenance.com`
- `app.genesisprovenance.com`

Wait until most locations show your Vercel IP/CNAME

3.3 Verify in Vercel

1. Go back to Vercel Dashboard → Settings → Domains

2. Wait for green checkmarks next to all domains

3. If you see errors, click "Refresh" after DNS propagates

✅ **Domain Setup Complete!**

Step 4: Update NEXTAUTH_URL

4.1 Update Environment Variable

1. In Vercel Dashboard → Settings → Environment Variables

2. Find `NEXTAUTH_URL`

3. Change from `https://app.genesisprovenance.com` to:
   `https://genesisprovenance.com`
   OR keep it as `app.genesisprovenance.com` if you want login on subdomain

4. Click "Save"

5. Redeploy your app (Vercel → Deployments → Click "…" → Redeploy)

Part 6: Microsoft 365 Email Setup (30 minutes)

Step 1: Get Microsoft 365 DNS Records

1.1 Log into Microsoft 365 Admin Center

```
https://admin.microsoft.com
```

1.2 Navigate to Domains

1. Click "Settings" → "Domains"
2. Click "Add domain"
3. Enter: `genesisprovenance.com`
4. Click "Use this domain"

1.3 Verify Domain Ownership

Microsoft will provide a TXT record:

```
Type: TXT
Name: @ or genesisprovenance.com
Value: MS=ms12345678 (example)
```

Add this to GoDaddy:
1. Go to GoDaddy DNS Management
2. Click "Add" → Select "TXT"
3. Enter the values from Microsoft
4. Click "Save"
5. Wait 15 minutes
6. Go back to Microsoft 365 and click "Verify"

Step 2: Add Email DNS Records to GoDaddy

2.1 MX Records (Mail Exchange)

Microsoft will provide MX records like:

```
Type: MX
Name: @
Value: genesisprovenance-com.mail.protection.outlook.com
Priority: 0
TTL: 3600
```

Add to GoDaddy:

1. Click "Add" → Select "MX"

2. Enter values from Microsoft 365

3. **Priority is IMPORTANT** - use exactly what Microsoft provides

4. Click "Save"

2.2 SPF Record (Prevent Spoofing)

Add TXT record:

```
Type: TXT
Name: @
Value: v=spf1 include:spf.protection.outlook.com -all
TTL: 3600
```

2.3 DKIM Records (Email Authentication)

Microsoft provides two CNAME records:

```
Type: CNAME
Name: selector1._domainkey
Value: selector1-genesisprovenance-com._domainkey.abacus.onmicrosoft.com
TTL: 3600

Type: CNAME
Name: selector2._domainkey
Value: selector2-genesisprovenance-com._domainkey.abacus.onmicrosoft.com
TTL: 3600
```

Add both to GoDaddy

2.4 DMARC Record (Email Policy)

Add TXT record:

```
Type: TXT
Name: _dmarc
Value: v=DMARC1; p=quarantine; rua=mailto:dmarc@genesisprovenance.com
TTL: 3600
```

2.5 Autodiscover (Optional but Recommended)

```
Type: CNAME
Name: autodiscover
Value: autodiscover.outlook.com
TTL: 3600
```

Step 3: Create Email Accounts

3.1 In Microsoft 365 Admin Center

1. Go to "Users" → "Active users"

2. Click "Add a user"

3.2 Create Admin Email

```
First name: Admin
Last name: Team
Username: admin
Domain: @genesisprovenance.com
Email: admin@genesisprovenance.com
```

3.3 Create Key Emails

Create these addresses:

- `info@genesisprovenance.com`
- `support@genesisprovenance.com`
- `sales@genesisprovenance.com`
- `hello@genesisprovenance.com`

3.4 Test Email

1. Send test email from `admin@genesisprovenance.com` to your personal email
2. Reply to verify receiving works
3. Check spam folder if not received

✅ **Email Setup Complete!**

Part 7: Post-Deployment Testing (15 minutes)

Verification Checklist

1. Test All Domains

- [ ] `https://genesisprovenance.com` loads
- [ ] `https://www.genesisprovenance.com` redirects to main
- [ ] `https://app.genesisprovenance.com` loads (if separate)
- [ ] All show green padlock (SSL)

2. Test Marketing Pages

- [ ] Home page loads with images
- [ ] Pricing page shows all tiers
- [ ] Product page loads
- [ ] How It Works page loads
- [ ] Contact form submits

3. Test Authentication

- [ ] Sign up form works
- [ ] Create test account
- [ ] Login works
- [ ] Dashboard loads after login
- [ ] Settings page loads

- [ ] Logout works

4. Test Admin Functions

- [ ] Login as admin (`john@doe.com` / `johndoe123`)
- [ ] Access `/admin` page
- [ ] View users list
- [ ] View organizations list

5. Database Test

- [ ] Create new account → Check if user appears in database
- [ ] Submit contact form → Check if submission appears in database
- [ ] Contact form data appears in admin console

6. Performance Test

- [ ] Run Lighthouse audit:
  `https://pagespeed.web.dev/`
- [ ] Aim for 90+ performance score
- [ ] Check mobile responsiveness

7. Email Test

- [ ] Send email from `admin@genesisprovenance.com`
- [ ] Receive email at `info@genesisprovenance.com`
- [ ] Check SPF/DKIM pass: https://mxtoolbox.com

✅ **All Tests Passed!**

Part 8: Monitoring & Maintenance

Set Up Monitoring

1. Vercel Analytics

1. Go to Vercel Dashboard → Analytics
2. Enable Web Analytics (free)
3. This tracks:
   - Page views
   - User sessions
   - Performance metrics

2. Add Error Tracking (Optional)

Sentry (Recommended):
1. Sign up at https://sentry.io
2. Create new project
3. Follow Next.js integration guide
4. Add `SENTRY_DSN` to Vercel environment variables

3. Uptime Monitoring (Optional)

UptimeRobot (Free):

1. Sign up at https://uptimerobot.com
2. Add monitor for `https://genesisprovenance.com`
3. Set alert email

Regular Maintenance Tasks

Daily:

- [ ] Check error logs in Vercel
- [ ] Monitor database size

Weekly:

- [ ] Review contact form submissions
- [ ] Check analytics
- [ ] Test critical user flows

Monthly:

- [ ] Review and optimize performance
- [ ] Update dependencies
- [ ] Backup database
- [ ] Review security logs

Troubleshooting Common Issues

Issue 1: Domain Not Connecting

Symptoms: Domain shows error or doesn't load

Solutions:

1. Check DNS propagation: https://dnschecker.org
2. Verify A record points to Vercel IP
3. Verify CNAME records are correct
4. Clear browser cache
5. Wait 24 hours for full propagation

Issue 2: SSL Certificate Error

Symptoms: Browser shows "Not Secure" or certificate error

Solutions:

1. In Vercel → Domains → Click "Refresh"
2. Wait 15 minutes for certificate provisioning

3. Clear browser cache
4. Try incognito mode

Issue 3: Database Connection Error

Symptoms: App loads but can't access data

Solutions:

1. Verify `DATABASE_URL` in Vercel environment variables
2. Check database is running (Vercel Postgres dashboard)
3. Verify migrations ran successfully:

```bash
   vercel env pull
   npx prisma migrate status
```

4. Re-run migrations if needed

Issue 4: Email Not Working

Symptoms: Emails not sending/receiving

Solutions:

1. Verify all MX records in GoDaddy
2. Check SPF/DKIM records: https://mxtoolbox.com
3. Wait 24 hours for DNS propagation
4. Check Microsoft 365 mail flow
5. Verify email account is active

Issue 5: Authentication Not Working

Symptoms: Login fails or redirects incorrectly

Solutions:

1. Verify `NEXTAUTH_URL` matches your domain
2. Check `NEXTAUTH_SECRET` is set
3. Clear cookies and try again
4. Verify database has users table
5. Check Vercel function logs for errors

Quick Reference

Important URLs

Service	URL
Live Site	https://genesisprovenance.com
Vercel Dashboard	https://vercel.com/dashboard
GitHub Repo	https://github.com/YOUR_USERNAME/genesis-provenance
AWS S3 Console	https://console.aws.amazon.com/s3
Database	https://vercel.com/dashboard/stores
GoDaddy DNS	https://dcc.godaddy.com/manage/genesisprovenance.com/dns
Microsoft 365	https://admin.microsoft.com

Key Credentials

Store these securely in a password manager:

- GitHub repo URL
- Vercel account email
- Database connection string
- AWS Access Key ID
- AWS Secret Access Key
- NextAuth secret
- Admin email password
- Microsoft 365 admin credentials

Support Resources

- **Vercel Docs:** https://vercel.com/docs
- **Next.js Docs:** https://nextjs.org/docs
- **Prisma Docs:** https://www.prisma.io/docs
- **AWS S3 Docs:** https://docs.aws.amazon.com/s3
- **GoDaddy Support:** https://www.godaddy.com/help
- **Microsoft 365 Support:** https://support.microsoft.com

Next Steps After Deployment

Immediate (Today)

1. ✅ Test all functionality
2. ✅ Create admin account
3. ✅ Send test emails
4. ✅ Share site with team
5. ✅ Monitor for 24 hours

Short Term (This Week)

1. Set up Google Analytics (optional)
2. Configure uptime monitoring
3. Add Sentry error tracking
4. Create backup strategy
5. Document internal processes

Medium Term (This Month)

1. Gather user feedback
2. Monitor performance metrics
3. Optimize load times
4. Start Phase 2 development:
   - Asset onboarding wizard
   - Items list/grid
   - File uploads
   - AI integration

Long Term (Next Quarter)

1. Phase 3: n8n webhooks + AI agents
2. Phase 4: Stripe integration
3. Marketing campaign
4. Customer onboarding

Success Criteria

Your deployment is successful when:

- ✅ All domains load with SSL
- ✅ Authentication works end-to-end
- ✅ Database is accessible and seeded
- ✅ Email sending/receiving works

- ✅ Contact form submissions saved
- ✅ No console errors
- ✅ Mobile responsive
- ✅ Performance score 90+

Congratulations! Genesis Provenance is now live! 🎉

Questions? Refer to:
- `VERCEL_DEPLOYMENT.md` for detailed technical info
- `PROJECT_STATUS.md` for project overview
- `PHASE_1.5_ENHANCEMENTS.md` for what's included

Last Updated: November 29, 2025
Version: 1.5
Status: Production Ready ✅