

# Netlify Deployment Configuration Fixes

---

## Summary of Changes

I've fixed the Netlify deployment configuration for your Next.js application. The main issue was that Netlify was configured to treat your app as a static site, but your application has API routes that require server-side rendering (SSR).

## Problems Identified

1. **Incorrect publish directory:** `netlify.toml` had `publish = ".next"` which only serves static files
2. **Wrong redirect rules:** Had SPA-style redirect (`/* → /index.html`) that conflicted with Next.js routing
3. **Missing Netlify plugin:** Wasn't using `@netlify/plugin-nextjs` for proper SSR support
4. **Output mode issue:** `next.config.js` had `output: process.env.NEXT_OUTPUT_MODE` which might force static export

## API Routes Detected

Your app has the following API routes that require server-side functionality:

- `/api/contact` (`app/api/contact/route.ts`)
- `/api/newsletter` (`app/api/newsletter/route.ts`)
- `/api/orders` (`app/api/orders/route.ts`)

## Files Modified

### 1. `netlify.toml`

#### **Changes made:**

- Removed incorrect `publish = ".next"` directive
- Removed SPA redirect rule (`/* → /index.html`)
- Added `@netlify/plugin-nextjs` plugin configuration
- Kept security headers and caching rules

#### **New configuration:**

```
[build]
command = "npm run build"
[build.environment]
  NODE_VERSION = "18"
  NPM_VERSION = "9"

[[plugins]]
  package = "@netlify/plugin-nextjs"

# Security headers and caching rules preserved
```

## 2. next.config.js

### Changes made:

- Removed `output: process.env.NEXT_OUTPUT_MODE` to allow SSR
- Kept all other configurations intact

## 3. package.json

### Changes made:

- Added `"@netlify/plugin-nextjs": "^5.0.0"` to devDependencies

## How the Essential Next.js Plugin Works

The `@netlify/plugin-nextjs` automatically handles:

- Server-side rendering (SSR) for pages
- API routes with proper serverless function deployment
- Static page optimization
- Incremental Static Regeneration (ISR)
- Image optimization
- Automatic routing without manual redirect rules

## Next Steps to Deploy

### Option 1: Manual Upload (Recommended)

Since the GitHub App needs permissions, you can manually push these files:

1. Download the fixed project from `/home/ubuntu/github_repos/emf-safe-haven/`
2. Open a terminal in your local machine
3. Navigate to the folder
4. Run:

```
bash
git init
git add .
git commit -m "Fix Netlify deployment configuration for Next.js"
git remote add origin https://github.com/ACIM2025/emf-safe-haven.git
git push -u origin main
```

### Option 2: Grant GitHub App Access (Then I Can Push)

1. Visit: [https://github.com/apps/abacusai/installations/select\\_target](https://github.com/apps/abacusai/installations/select_target)
2. Select the repository `emf-safe-haven`
3. Grant the Abacus.AI app access with “Read and write” permissions for Contents
4. Let me know, and I’ll complete the push automatically

### Option 3: Upload ZIP to GitHub

1. Go to: <https://github.com/ACIM2025/emf-safe-haven>
2. Click “uploading an existing file”
3. Drag and drop all files from `/home/ubuntu/github_repos/emf-safe-haven/`
4. Commit the changes

## Redeploying on Netlify

---

After pushing to GitHub:

**1. If using Netlify's GitHub integration:**

- Netlify will automatically detect the push and redeploy
- The new configuration will be used automatically

**2. If deploying manually:**

- Go to your Netlify site dashboard
- Click “Deploys” → “Trigger deploy” → “Deploy site”
- Or connect the GitHub repository for automatic deploys

**3. Install the plugin:**

- Before the next deployment, run: `npm install`
- This ensures `@netlify/plugin-nextjs` is installed

## Expected Results

---

After redeployment with these fixes:

- Home page will load correctly (not 404)
- API routes will work (`/api/contact`, `/api/newsletter`, `/api/orders`)
- Dynamic routes will function properly
- Server-side rendering will work as expected

## Verification

---

Once deployed, test:

1. Visit your Netlify URL (<https://helpful-arithmetic-cfe59b.netlify.app/>)
2. Check that the home page loads (should see content, not 404)
3. Open browser DevTools → Network tab
4. Test any forms that use the API routes
5. Verify API endpoints respond correctly

## Troubleshooting

---

If you still see issues:

**1. Check Netlify build logs:**

- Look for: `@netlify/plugin-nextjs` in the build output
- Ensure it says “Essential Next.js Build Plugin”

**2. Verify environment variables:**

- Make sure `NEXT_OUTPUT_MODE` is NOT set to ‘export’
- Or remove this environment variable entirely

**3. Clear Netlify cache:**

- Go to Site Settings → Build & deploy → Clear cache and retry deploy

## Files Ready for Push

---

All files are in: `/home/ubuntu/github_repos/emf-safe-haven/`

Git status: Ready to push with commit message:

```
Fix Netlify deployment configuration for Next.js

- Updated netlify.toml to use @netlify/plugin-nextjs for proper SSR support
- Removed incorrect publish directory (.next) that was causing 404 errors
- Removed SPA redirect rule that conflicted with Next.js routing
- Added @netlify/plugin-nextjs to package.json devDependencies
- Removed problematic output mode from next.config.js to enable server-side rendering
- App has API routes (/api/contact, /api/newsletter, /api/orders) which require SSR
```

---

**Note:** The configuration is now correct and ready to deploy. Once pushed to GitHub and redeployed on Netlify, your 404 errors should be resolved!