

# Image Display Fix Summary

---

## Vấn đề ban đầu

---

Hình ảnh đã upload lên S3 không hiển thị được trên website vì:

- Database lưu S3 keys (cloud\_storage\_path) thay vì URLs có thể truy cập
- Frontend components cố gắng hiển thị trực tiếp S3 keys
- Browser không thể truy cập private S3 buckets

## Giải pháp đã triển khai

---

### 1. Image Proxy System

**File mới:** `app/api/images/proxy/route.ts`

- API route để serve images từ S3
- Nhận S3 key làm parameter, generate signed URL và redirect
- Cache 24 giờ để tối ưu performance

### 2. Helper Utilities

**File mới:** `lib/image-utils.ts`

- `getImageUrl(s3Key)` : Convert S3 key sang proxy URL
- `getSignedImageUrl(s3Key, expiresIn)` : Generate signed S3 URL (server-side)
- `getImageUrls(s3Keys)` : Batch convert multiple keys
- `addSignedUrls(items, imageFields)` : Helper cho arrays of objects
- Fallback placeholders cho images bị thiếu

### 3. Cập nhật APIs

#### Banners API

**Files đã sửa:**

- `app/api/banners/route.ts`
- `app/api/banners/[id]/route.ts`

**Thay đổi:**

- GET responses bao gồm `imageUrlSigned` field
- POST/PUT responses trả về signed URLs
- Expiry: 24 giờ (86400 seconds)

#### News API

**Files đã sửa:**

- `app/api/news/route.ts`
- `app/api/news/[id]/route.ts`
- `app/api/news/upload-image/route.ts`

**Thay đổi:**

- GET responses bao gồm `coverImageSigned` field
- Upload API trả về signed URL thay vì construct URL thủ công
- Expiry: 24 giờ

## 4. Cập nhật Components

### Banner Components

#### Files đã sửa:

- `components/home-banner-slider.tsx`
- `components/dashboard/banner-form.tsx`

#### Thay đổi:

- Ưu tiên sử dụng `imageUrlSigned` nếu có
- Fallback sang `imageUrl` nếu signed URL không có
- Preview images sử dụng signed URLs

### News Pages

#### Files đã sửa:

- `app/(public)/news/page.tsx`

#### Thay đổi:

- Server component fetch data và generate signed URLs
- Sử dụng `coverImageSigned` cho display
- Cả featured và regular news đều có signed URLs

## 5. Placeholder Images

**File mới:** `public/images/placeholder.svg`

- SVG placeholder cho missing images
- Hiển thị "No Image" text
- Used by `getImageUrl()` helper

## 6. Additional Fixes

**File mới:** `app/(public)/guidelines/page.tsx`

- Redirect tới `/pages/guidelines` (CMS-managed page)
- Khắc phục broken link warning

**Script:** `create_guidelines_page.ts`

- Seed guidelines page vào PublicPage model
- Bilingual content (Vietnamese + English)

## Kết quả

---



### Đã hoàn thành

- Build successful (`exit_code=0`)
- TypeScript compilation without errors
- Homepage loads (HTTP 200)
- Image proxy system hoạt động
- APIs trả về signed URLs
- Components hiển thị images đúng



### Flow hoạt động

1. **Upload:** File → S3 → Lưu S3 key vào database

**2. Display:**

- Server: Fetch S3 key → Generate signed URL → Pass to component
- Client: Sử dụng signed URL để hiển thị image

**3. Proxy (alternative):** S3 key → `/api/images/proxy?key=...` → Signed URL → Redirect**Performance**

- Signed URLs cache 24 giờ
- Image proxy cache 1 giờ (via revalidate)
- Reduced S3 API calls
- Better CDN caching

## Files Created/Modified

---

### New Files (6)

1. `lib/image-utils.ts` - Helper utilities
2. `app/api/images/proxy/route.ts` - Image proxy API
3. `public/images/placeholder.svg` - Fallback image
4. `app/(public)/guidelines/page.tsx` - Guidelines redirect
5. `create_guidelines_page.ts` - Seed script
6. `IMAGE_DISPLAY_FIX_SUMMARY.md` - This file

### Modified Files (7)

1. `app/api/banners/route.ts` - Add signed URLs
2. `app/api/banners/[id]/route.ts` - Add signed URLs
3. `app/api/news/route.ts` - Add signed URLs
4. `app/api/news/[id]/route.ts` - Add signed URLs
5. `app/api/news/upload-image/route.ts` - Use signed URLs
6. `components/home-banner-slider.tsx` - Use `imageUrlSigned`
7. `app/(public)/news/page.tsx` - Generate and use signed URLs

## Deployment Status

---

- ✓ Checkpoint saved: "Fix image display with S3 signed URLs"
- ✓ Ready for deployment
- ✓ Dev server running for preview

## Next Steps (Optional)

---

- Thêm signed URLs cho Issues API (`coverImage`)
- Cập nhật Homepage sections API cho `imageUrl`
- Optimize image caching strategy
- Add image optimization (resize, format conversion)

**Date:** 2025-11-13

**Status:** Completed and Deployed