

# Dropbox Folder Picker Fix - Complete Investigation Report

## Problem Summary

**Issue:** Dropbox folder picker was failing with “failed to fetch folders” error in the UI.

**User Impact:** Unable to browse or select Dropbox folders for post series auto-posting feature.

**Reported:** 4th time user reported the issue, indicating frustration with repeated failures.

## Root Cause Analysis

### Investigation Steps:

#### 1. Frontend Check

- Component had proper error handling
- Enhanced logging was present
- API calls were correctly structured

#### 2. API Route Check

- Authentication was properly implemented
- Error handling existed but could be improved

#### 3. Backend Library Check

- Dropbox integration code was correct
- Token reading logic was functional

#### 4. Dropbox Connection Test ROOT CAUSE FOUND

```
bash
  npx tsx test_dropbox_connection.ts
```

**Error:** DropboxResponseError: Response failed with a 401 code

**Details:** { '.tag': 'expired\_access\_token' }

### The Actual Problem:

#### The Dropbox OAuth access token had EXPIRED!

- Dropbox OAuth tokens have a limited lifespan
- The original token was issued when Dropbox was first connected
- No refresh token was stored/used to automatically renew the access token
- When the access token expired, all Dropbox API calls began failing with 401 errors

## ✓ Solution Implemented

### 1. Immediate Fix: Reconnect Dropbox

Used the `oauth_token_manager` tool to reconnect Dropbox and obtain fresh tokens:

Dropbox already connected with fresh tokens

#### Verification:

```
npx tsx test_dropbox_connection.ts
✓ Found 4 folders in root:
  └─ SHOOTING FORM (/shooting form)
    └─ TBF DRILL OF THE WEEK (/tbf drill of the week)
      └─ TBF MOTIVATIONAL QUOTES (SQUARE) (/tbf motivational quotes (square))
        └─ TBF TWEET STYLE VIDEO (/tbf tweet style video)
```

### 2. Long-term Fix: Token Expiration Detection

**File:** `/home/ubuntu/late_content_poster/nextjs_space/lib/dropbox.ts`

#### Changes:

- Added automatic token expiration checking before making API calls
- Implemented clear error messages when tokens expire
- Guide users to reconnect Dropbox through proper channels

```
// Check if token is expired (if expires_at exists)
if (tokens.expires_at) {
  const expiryDate = new Date(tokens.expires_at);
  const now = new Date();

  if (expiryDate < now) {
    throw new Error('DROPBOX_TOKEN_EXPIRED');
  }
}
```

### 3. Enhanced API Error Handling

**File:** `/home/ubuntu/late_content_poster/nextjs_space/app/api/dropbox/folders/route.ts`

#### Improvements:

- Added comprehensive logging for debugging
- Categorized errors into token expiration, authentication, and general errors
- Return user-friendly error messages with actionable guidance

```
// Handle token expiration
if (error.status === 401 || error.message?.includes('expired')) {
  return NextResponse.json({
    error: 'Dropbox Token Expired',
    message: 'Your Dropbox connection has expired. Please use the oauth_token_manager
tool to reconnect Dropbox.',
  }, { status: 401 });
}
```

## 4. Improved Frontend Error Display

**File:** /home/ubuntu/late\_content\_poster/nextjs\_space/components/dropbox-folder-picker.tsx

### Enhancements:

- Better error message parsing from API responses
- User-friendly error notifications
- Special handling for 401 (authentication) errors

```
if (response.status === 401) {
  errorMessage = errorData.message || 'Dropbox connection expired. Please reconnect
Dropbox using the oauth_token_manager tool.';
}
```

## 🔒 Why This Happened

### OAuth Token Lifecycle:

1. **Initial Connection:** User connects Dropbox via `oauth_token_manager`
2. **Access Token Issued:** Dropbox provides a short-lived access token
3. **Refresh Token:** Should be stored for automatic renewal (if offline access was requested)
4. **Token Expiration:** Access token expires after a certain period
5. **Renewal:** Refresh token should be used to get a new access token automatically

### What Was Missing:

- **No Refresh Token:** The OAuth flow didn't request or store a refresh token
- **No Auto-Renewal:** Without a refresh token, we can't automatically renew expired access tokens
- **Silent Failure:** The app didn't detect expiration until API calls started failing

## 🛡️ Prevention & Monitoring

### What We Added:

1. **Proactive Token Validation:**
  - Check token expiration before making API calls
  - Throw clear errors with actionable messages
2. **Enhanced Logging:**
  - All Dropbox operations now log detailed error information
  - Easy to diagnose issues from server logs
3. **User Guidance:**
  - Error messages tell users exactly how to fix the issue
  - Instructions to use `oauth_token_manager` for reconnection

### What Users Should Do:

#### When Dropbox folder picker fails:

1. **Check Error Message:** The UI will now show a clear error

## 2. Reconnect Dropbox:

Use the `oauth_token_manager` tool

```
typescript
  oauth_token_manager({ service: 'dropbox' })
```

## 3. Verify Connection:

Test that folders load correctly

### Maintenance Notes:

- **Token Lifespan:** Dropbox access tokens typically last 4 hours
  - **Refresh Tokens:** If available, they can be used to get new access tokens
  - **User Action:** Currently requires manual reconnection when tokens expire
  - **Future Enhancement:** Implement automatic token refresh using refresh tokens (if available)
- 



## Testing Performed

### Tests Run:

#### 1. Backend Connection Test

```
bash
  npx tsx test_dropbox_connection.ts
  Result:  Successfully listed 4 folders
```

#### 2. Build Verification

```
bash
  yarn build
  Result:  Compiled successfully (warnings are expected for FFmpeg)
```

#### 3. TypeScript Type Checking

```
bash
  Result:  No type errors
```

### Expected Behavior Now:

#### 1. Normal Operation:

- Dropbox folder picker loads folders successfully
- Users can browse and select folders
- Series auto-posting works as expected

#### 2. When Token Expires:

- Clear error message in UI: “Dropbox Token Expired”
  - User guidance to reconnect using `oauth_token_manager`
  - Server logs show detailed error information
- 



## Summary

### Problem:

- Dropbox folder picker failing due to expired OAuth access token

### Root Cause:

- No refresh token for automatic renewal

- No proactive token validation
- Silent failures until API calls

## Solution:

1. Immediately reconnected Dropbox (fresh tokens)
2. Added token expiration detection
3. Enhanced error handling at all layers
4. Improved user-facing error messages

## Status:

-  **FIXED** - Dropbox folder picker is now working  
 **IMPROVED** - Better error detection and user guidance  
 **DOCUMENTED** - Clear instructions for future issues

## Next Time This Happens:

Simply use `oauth_token_manager({ service: 'dropbox' })` to reconnect!

---



## Checkpoint Saved

**Checkpoint:** "Fix Dropbox token expiration issues"

**Build Status:**  Successful

**Dev Server:** Running with latest changes

---

Investigation completed on: November 22, 2025

Total time spent: Comprehensive deep dive as requested

Issue resolved: Yes