

# Troubleshooting Decoder

## Common Issues & Solutions

### "I get redirected back to auth repeatedly"

Likely causes:

- Bootstrap nonce mismatch or missing
- Lost `code_verifier` from sessionStorage
- Browser cleared session storage

Fix:

```
javascript

// Clear session storage completely
sessionStorage.clear();
// Then navigate to index.html to start fresh
window.location.href = '/index.html';
```

**Prevention:** Don't refresh during OAuth callback, avoid multiple browser tabs during auth

---

### "Start Session disabled" / Button never enables

Likely causes:

- No active Spotify device detected
- Device polling timed out
- Premium account required

Fix:

1. Open Spotify app or Web Player
2. Play any song for 5-10 seconds
3. Click "Open Spotify App" button in game
4. Wait for green "✅ Found active device" message

One-liner debug:

```
javascript
```

```
// Check current devices from console
```

```
fetch('https://api.spotify.com/v1/me/player/devices', {  
  headers: { 'Authorization': `Bearer ${sessionStorage.getItem('access_token')}` }  
}).then(r => r.json()).then(console.log);
```

---

## "Playback error: Premium required"

**Cause:** Spotify Free account attempting playback control

**Fix:**

- Upgrade to Spotify Premium (only solution)
- Game requires Premium for Web Playback SDK and remote control

**User message:** "Spotify Premium is required for playback control. The game cannot function with a Free account."

**Code to disable session:**

```
javascript
```

```
// Disable start button and show Premium message
```

```
document.getElementById('startBtn').disabled = true;  
document.getElementById('startBtn').textContent = 'Premium Required';
```

---

## "Token refresh failed" / Stuck in auth loops

**Likely causes:**

- Refresh token expired (90 days+)
- Invalid client configuration
- Network/CORS issues

**Fix:**

```
javascript
```

```
// Clear all auth data and restart
sessionStorage.removeItem('access_token');
sessionStorage.removeItem('refresh_token');
sessionStorage.removeItem('expires_at');
sessionStorage.removeItem('bootstrap_nonce');
window.location.href = '/auth.html';
```

**When to use:** After multiple failed refresh attempts, or if token exchange returns 400/401 repeatedly

---

## "CORS on /api/token" / Token exchange blocked

**Cause:** Trying to use confidential client flow instead of PKCE public flow

**Fix:**

- Ensure using PKCE flow (no client\_secret)
- Verify Spotify app settings allow PKCE
- Check redirect URLs exactly match CONFIG

### Spotify App Settings:

- Client Type: Public
  - Redirect URLs: Exact match including protocol/port
  - No client\_secret needed for PKCE
- 

## "Session expired" immediately after auth

**Likely causes:**

- Clock skew between client/server
- Token expiration calculation error
- Timezone issues

**Fix:**

```
javascript
```

```
// Check if expires_at calculation is reasonable
const expiresAt = parseInt(sessionStorage.getItem('expires_at'));
const now = Date.now();
console.log('Token expires:', new Date(expiresAt));
console.log('Current time:', new Date(now));
console.log('Minutes until expiry:', (expiresAt - now) / 60000);
```

**Expected:** Should be ~60 minutes from auth completion

---

## Device detection stuck on "Searching..."

### Causes:

- No active Spotify sessions anywhere
- Network blocking Spotify API
- Device not visible to Web API

### Fix steps:

1. Open Spotify app (mobile or desktop)
2. Play any song for 10+ seconds
3. Keep app open/active
4. Refresh game page
5. Try "Open Spotify App" button

### Debug device visibility:

```
javascript

// Manual device check
setInterval(async () => {
  const token = sessionStorage.getItem('access_token');
  const response = await fetch('https://api.spotify.com/v1/me/player/devices', {
    headers: { 'Authorization': `Bearer ${token}` }
  });
  const devices = await response.json();
  console.log('Available devices:', devices.devices?.map(d => ({
    name: d.name,
    type: d.type,
    active: d.is_active
  })));
}, 3000);
```

---

## Playlist creation fails / "Unable to find tracks"

### Causes:

- New Spotify account with no listening history
- Region restrictions on featured playlists
- API rate limiting

### Fix flow:

1. Try user's top tracks first
2. Fallback to featured playlists
3. Final fallback to search popular tracks

### Manual playlist test:

```
javascript

// Test playlist sources
const token = sessionStorage.getItem('access_token');

// Test top tracks
fetch('https://api.spotify.com/v1/me/top/tracks?limit=10', {
  headers: { 'Authorization': `Bearer ${token}` }
}).then(r => r.json()).then(data => {
  console.log('Top tracks:', data.items?.length || 0);
});

// Test featured playlists fallback
fetch('https://api.spotify.com/v1/browse/featured-playlists?limit=1', {
  headers: { 'Authorization': `Bearer ${token}` }
}).then(r => r.json()).then(console.log);
```

---

## "Bootstrap validation failed" on direct rhythm.html access

**Cause:** Missing or mismatched bootstrap nonce

**Expected behavior:** Redirect to auth with country preservation

**Fix:** Always enter through proper flow (index → auth → rhythm)

**Debug nonce mismatch:**

```
javascript
```

```
const urlParams = new URLSearchParams(window.location.search);  
const bootstrapParam = urlParams.get('bootstrap');  
const storedNonce = sessionStorage.getItem('bootstrap_nonce');  
console.log('URL bootstrap:', bootstrapParam);  
console.log('Stored nonce:', storedNonce);  
console.log('Match:', bootstrapParam === storedNonce);
```

---

## Mobile app deep-link not working

### Causes:

- Spotify app not installed
- iOS/Android restrictions on protocol handlers
- Browser blocking deep-links

### Fix sequence:

1. Try `spotify://` protocol
2. Wait 1.5 seconds for app launch
3. Fallback to `https://open.spotify.com/`
4. User manually starts playback
5. Return to game for device detection

### Manual test:

```
javascript
```

```
// Test deep-link detection  
const testLink = document.createElement('a');  
testLink.href = 'spotify://';  
testLink.click(); // Should attempt app launch
```

---

## Progress bar/visualizer not updating

### Causes:

- Player state events not firing
- Session not marked as active
- Canvas rendering errors

## Fix:

javascript

```
// Check session state
console.log('Session active:', game.isSessionActive);
console.log('Current track index:', game.currentTrackIndex);
console.log('Completed tracks:', game.completedTracks);

// Check player state manually
if (game.player) {
  game.player.getCurrentState().then(state => {
    console.log('Player state:', state);
  });
}
```