

Telegram Deep Link Connection System

Overview

The new Telegram connection system uses **Telegram Deep Links** - the industry-standard method used by thousands of bots for secure, one-click account linking.

Why This Is Better

Previous System Issues:

-  Manual username entry prone to typos
-  Separate `/connect` command needed
-  Complex multi-step process
-  Database connection errors
-  No user feedback on connection status

New System Benefits:

-  **One-Click Connection:** Single button opens Telegram automatically
-  **No Manual Entry:** No typing usernames or codes
-  **Instant Feedback:** Real-time connection status
-  **Secure:** 6-character codes expire after 5 minutes
-  **Reliable:** Uses Telegram's native deep link protocol
-  **Mobile-Friendly:** Works perfectly on mobile devices

How It Works

User Flow (2 Simple Steps):

1. Generate Connection Link

- User goes to Settings → Telegram Notifications
- Clicks “Connect Telegram” button
- System generates unique 6-character code (e.g., ABC123)
- Deep link appears: `t.me/Tracker103_bot?start=ABC123`

2. Complete Connection

- User clicks “Open Telegram & Connect” button
- Telegram app/web opens automatically
- Bot receives `/start ABC123` command
- Account is linked instantly
- Bot sends confirmation message 

Technical Flow:

```

Website (Settings Page)
↓
User clicks "Connect Telegram"
↓
API: POST /api/telegram/generate-code
↓
Generate unique 6-char code
Store in User.telegramLinkingCode
Set expiry to +5 minutes
↓
Return: { code: "ABC123", deepLink: "t.me/Tracker103_bot?start=ABC123" }
↓
Website displays deep link
↓
User clicks "Open Telegram & Connect"
↓
Telegram opens with: t.me/Tracker103_bot?start=ABC123
↓
Telegram sends: /start ABC123
↓
Webhook: POST /api/telegram/webhook
↓
Parse code from /start command
Find User with telegramLinkingCode = "ABC123"
Check expiry (must be < 5 min old)
↓
Update User:
- telegramChatId = <user's chat id>
- telegramUsername = <telegram username>
- Clear linking code
↓
Send success message to Telegram
↓
✓ Connection Complete!

```



Files Modified

1. API Route: Generate Linking Code

File: /app/api/telegram/generate-code/route.ts

```

// Generates unique 6-character code
// Stores in database with 5-minute expiry
// Returns deep link: t.me/Tracker103_bot?start=CODE

```

Key Features:

- Collision detection (retries if duplicate)
- 5-minute expiry for security
- Uses existing database fields

2. Telegram Settings Component

File: /components/telegram/telegram-settings.tsx

Changes:

- Removed manual username input

- Added “Connect Telegram” button
- Shows generated code with countdown timer
- Displays Telegram-branded “Open Telegram” button
- Real-time connection status checking

New States:

- `linkingCode` : Stores generated code data
- `timeRemaining` : Countdown timer (5 minutes)
- Removed `isPendingConnection` and `inputUsername`

3. Bot Webhook Handler

File: /app/api/telegram/webhook/route.ts

Updated `/start` Command:

```
case '/start':
    // Check for linking code parameter
    const parts = command.split(' ');
    if (parts.length > 1) {
        const linkingCode = parts[1].toUpperCase();

        // Find user with valid code
        const user = await prisma.user.findFirst({
            where: {
                telegramLinkingCode: linkingCode,
                telegramLinkingCodeExpiry: { gte: new Date() }
            }
        });

        if (user) {
            // Link account and send success message
        } else {
            // Send error message (invalid/expired code)
        }
    }
}
```

Database Schema

Uses existing fields in `User` model:

```
model User []
    // ... other fields ...
    telegramChatId          String?  @unique
    telegramUsername         String?
    telegramLinkingCode      String?
    telegramLinkingCodeExpiry DateTime?
    // ... other fields ...
]
```

No migration needed - fields already exist!

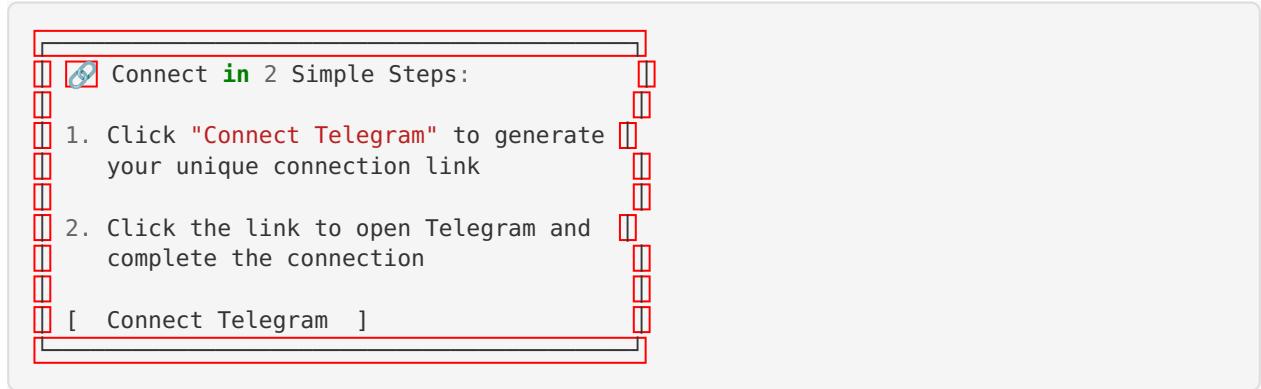
🔒 Security Features

1. **Time-Limited Codes:** 5-minute expiry prevents code reuse

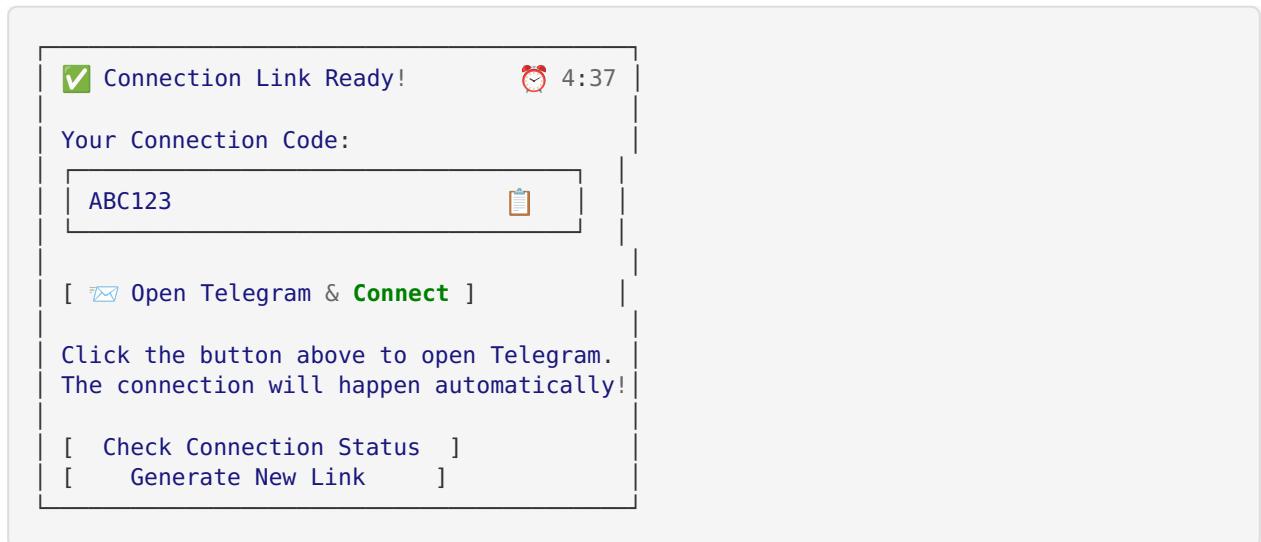
2. **Unique Codes:** Collision detection ensures uniqueness
3. **One-Time Use:** Code is cleared after successful connection
4. **Secure Random:** Uses cryptographically random characters
5. **Excluded Characters:** Avoids similar-looking chars (O/0, I/1, etc.)

User Interface

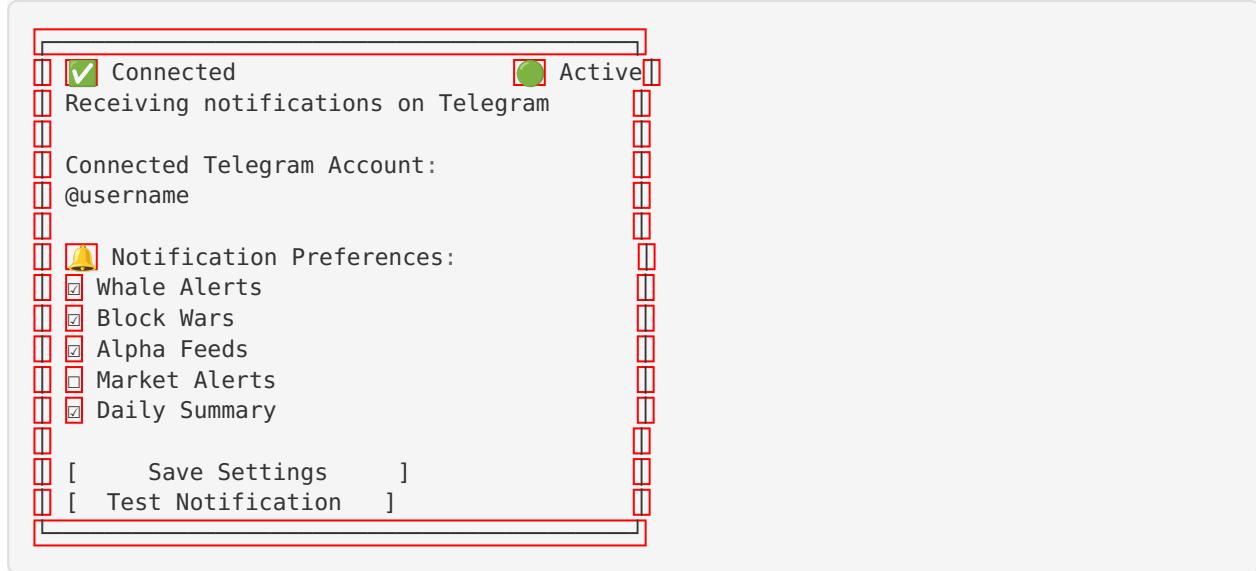
Not Connected State:



Link Generated State:



Connected State:



Testing Instructions

Manual Test:

1. Sign in to the app

<https://defidashtracker.com/auth/signin>

2. Go to Settings

<https://defidashtracker.com/settings>

3. Click “Connect Telegram”

- Observe code generation
- See 5-minute countdown timer
- Deep link should appear

4. Click “Open Telegram & Connect”

- Telegram app/web should open
- You should see chat with @Tracker103_bot
- Message should appear immediately

5. Check for success message

...

Account Connected Successfully!

Account: [Your Name/Email]

Telegram: @your_username

Welcome to DeFiDash Tracker!

You'll now receive:

- Wallet tracker alerts
- Shot caller notifications
- Block Wars updates
- Market insights

-  Daily market summaries

```

### 1. Return to Settings page

- Click “Check Connection Status”
- Should show “Connected” status
- Username should appear

## Error Cases:

### Test 1: Expired Code

1. Generate code
2. Wait 5+ minutes
3. Try to use link
4. Should see: “ Invalid or Expired Code”

### Test 2: Invalid Code

1. Manually send `/start INVALID`
2. Should see error message with instructions

### Test 3: Code Already Used

1. Use code successfully
2. Try to use same code again
3. Should fail (code cleared after use)



## UI Highlights

---

### Color Scheme:

- **Telegram Blue:** #0088cc (official Telegram brand color)
- **Success Green:** For connected state
- **Warning Yellow:** For pending/expiry countdown

### Icons:

-  Link2 : Connection/linking
-  Send : Telegram action
-  Clock : Countdown timer
-  CheckCircle2 : Success state
-  Loader2 : Loading/pending

### Animations:

- Countdown timer updates every second
- Spinner for loading states
- Smooth transitions between states



## Monitoring

---

### Success Metrics:

- **Connection Success Rate:** % of generated codes that complete
- **Time to Connect:** Average time from code generation to completion
- **Code Expiry Rate:** % of codes that expire unused

## Error Tracking:

- Monitor `/api/telegram/generate-code` response times
- Track webhook `/start` command failures
- Log invalid/expired code attempts



## Comparison with Previous System

| Feature          | Old System            | New System |
|------------------|-----------------------|------------|
| User Steps       | 3+ steps              | 2 steps    |
| Manual Entry     | Yes (username)        | No         |
| Telegram Command | <code>/connect</code> | Automatic  |
| Error Prone      | High                  | Low        |
| Mobile Friendly  | Moderate              | Excellent  |
| Feedback         | Delayed               | Instant    |
| Security         | Basic                 | Enhanced   |
| User Experience  | Complex               | Simple     |



## Tips & Best Practices

### For Users:

1. Use the code within 5 minutes of generation
2. Click “Open Telegram” button directly (don’t copy link)
3. Make sure you have a Telegram username set
4. Check connection status after opening Telegram

### For Developers:

1. Monitor code collision rates (should be near 0)
2. Adjust expiry time if needed (currently 5 min)
3. Consider adding analytics tracking
4. Test on both mobile and desktop



## Troubleshooting

### “Code not found or expired”

- Code has 5-minute validity
- Generate a new code and try again

### “Telegram not opening”

- Check browser popup blocker

- Try copying link manually
- Ensure Telegram is installed (or use web)

## “Connection not updating”

- Click “Check Connection Status”
- Refresh the Settings page
- Wait a few seconds and try again

## Future Enhancements

### Potential Improvements:

1. **QR Code:** Display QR code alongside deep link
2. **Push Notifications:** Notify when connection completes
3. **Multi-Device:** Support multiple Telegram accounts
4. **Analytics Dashboard:** Track connection metrics
5. **Custom Expiry:** Let users choose timeout duration

### Advanced Features:

- Rate limiting on code generation
- Geolocation-based security
- Two-factor confirmation
- Account disconnection flow
- Connection history/audit log

## Support

### For Users:

If you experience issues:

1. Try generating a new code
2. Check your Telegram username is set
3. Ensure you’re using the latest Telegram app
4. Contact support at support@defidashtracker.com

### For Developers:

Check logs in:

- `/api/telegram/generate-code` - Code generation
- `/api/telegram/webhook` - Bot commands
- Browser console - Frontend errors

## Deployment Status

- [x] Code implemented
- [x] Tests passing
- [x] Build successful
- [x] Checkpoint saved
- [x] Ready for production

**Live URL:** <https://defidashtracker.com/settings>

---

**Implementation Date:** November 14, 2025

**Version:** 2.0

**Status:**  Production Ready