Token Gate Implementation for Wallet Tracker

Overview

The Wallet Tracker feature now requires users to connect their wallet and hold at least **3,000 $DEFIDASH tokens** to access the feature. This provides exclusive access to premium wallet tracking capabilities.

Token Details

- **Contract Address**: `0xd6df108d516a5dc83f39020a349085c79d4edf0d`
- **Required Balance**: 3,000 $DEFIDASH
- **Network**: Base (Chain ID: 8453)

Implementation Components

1. Token Gate API Endpoint

File: `/app/api/token-gate/check/route.ts`

Features:
- Checks ERC20 token balance using ethers.js
- Validates wallet address format
- Supports multiple chains (Ethereum, Base, BSC)
- Returns detailed balance information and access status

API Usage:

```
GET /api/token-gate/check?address=0x...&chainId=8453
```

Response:

```
{
  "success": true,
  "address": "0x...",
  "chain": "Base",
  "chainId": "8453",
  "contractAddress": "0xd6df108d516a5dc83f39020a349085c79d4edf0d",
  "balance": 5000,
  "required": 3000,
  "hasAccess": true,
  "message": "Access granted! You hold 5000.00 $DEFIDASH tokens."
}
```

2. TokenGate Component

File: `/components/wallet/TokenGate.tsx`

A comprehensive React component that handles:
- Wallet connection prompts
- Balance verification
- Loading states
- Error handling
- Success/failure UI states

UI States:

1. **Not Connected**: Shows wallet connect interface with feature highlights
2. **Checking Balance**: Loading spinner while verifying tokens
3. **Insufficient Balance**: Shows current vs required balance with purchase links
4. **Access Granted**: Success banner with protected content displayed

3. Wallet Tracker Page Integration

File: `/app/wallet-tracker/page.tsx`

The entire wallet tracker page is wrapped with the TokenGate component:

```
<TokenGate
  requiredTokens={3000}
  contractAddress="0xd6df108d516a5dc83f39020a349085c79d4edf0d"
  chainId="8453"
>
  {/* Protected wallet tracker content */}
</TokenGate>
```

User Experience Flow

Step 1: Initial Visit (Not Connected)

Users see:
- Token gate explanation
- Required balance information (3,000 $DEFIDASH)
- Feature benefits list
- Wallet connection button

Step 2: Wallet Connection

Users can:
- Connect via Coinbase Wallet (already integrated)
- View supported networks
- See connection status

Step 3: Balance Verification

System automatically:
- Checks $DEFIDASH balance on connected wallet
- Verifies against required amount (3,000 tokens)
- Shows loading state during verification

Step 4: Access Decision

If Balance >= 3,000 tokens:
- Green success banner displays

- Full wallet tracker features unlock
- User can track wallets across 6+ chains

If Balance < 3,000 tokens:
- Yellow warning banner displays
- Shows current balance vs required balance
- Provides links to purchase $DEFIDASH
- Offers "Refresh Balance" button

Features Unlocked by Token Gate

With 3,000+ $DEFIDASH tokens, users gain access to:

✅ **Multi-Chain Wallet Tracking**
- Ethereum, BNB Chain, Polygon, Base, Optimism, Arbitrum, Solana

✅ **Real-Time Balance Monitoring**
- Native currency balances
- ERC-20 token holdings
- USD valuations

✅ **Transaction History**
- Recent transactions
- Token transfers
- Historical activity

✅ **Watchlist Management**
- Add wallets to watchlist
- Get alerts for whale movements
- Track specific token holdings

✅ **Telegram Notifications**
- Real-time transaction alerts
- Whale activity notifications
- Custom watchlist alerts

Technical Details

Dependencies Added

- `ethers` v6.15.0 - For Web3 interactions and token balance checking

API Provider Support

The wallet tracker uses multiple API providers with fallback:
1. **Alchemy** (Primary)
2. **Moralis** (Fallback)
3. **Etherscan** (Final fallback)

Wallet Connection

Uses existing `WalletContext` with:
- Coinbase Wallet SDK integration

- Multi-chain support
- Persistent connection state

Security Features

- Read-only wallet access
- No private key exposure
- Client-side balance verification
- Server-side API validation

Configuration

Environment Variables Required

```
ALCHEMY_API_KEY=your_alchemy_key
MORALIS_API_KEY=your_moralis_key
ETHERSCAN_API_KEY=your_etherscan_key
```

Customization Options

Change required balance:

```
<TokenGate requiredTokens={5000}>
```

Change network:

```
<TokenGate chainId="1"> // Ethereum
```

Change contract:

```
<TokenGate contractAddress="0x...">
```

Testing the Token Gate

Test with Your Wallet

1. Visit `/wallet-tracker`
2. Click “Connect Coinbase Wallet”
3. Approve connection in wallet
4. System automatically checks balance
5. Access granted if >= 3,000 $DEFIDASH

Manual API Test

```
curl "https://defidashtracker.com/api/token-gate/check?address=YOUR_ADDRESS&chainId=8453"
```

Purchase $DEFIDASH Tokens

Users can purchase tokens via:
- Decentralized exchanges (DEX) on Base network
- View token info: https://basescan.org/token/0xd6df108d516a5dc83f39020a349085c79d4edf0d
- Ensure wallet is on Base network before purchasing

Future Enhancements

Possible improvements:
- [ ] NFT-based access (in addition to token holding)
- [ ] Tiered access (different features for different balances)
- [ ] Staking rewards for token holders
- [ ] Token burning mechanism for premium features
- [ ] Multi-token support (hold X or Y tokens)

Build Status

✅ TypeScript compilation: Success
✅ Next.js build: Success
✅ API routes: Functional
✅ Token gate integration: Complete
✅ Wallet connection: Working

Files Modified/Created

New Files

- `/app/api/token-gate/check/route.ts` - Balance checking API
- `/components/wallet/TokenGate.tsx` - Token gate UI component
- `/TOKEN_GATE_IMPLEMENTATION.md` - This documentation

Modified Files

- `/app/wallet-tracker/page.tsx` - Added TokenGate wrapper
- `package.json` - Added ethers.js dependency

Notes

- The token gate checks balance in real-time when wallet connects
- Balance is verified on the Base network by default
- Users can refresh balance at any time
- Connection state persists across sessions
- No backend authentication required - purely wallet-based access

Support

If users report token gate issues:
1. Verify they’re on the correct network (Base)
2. Confirm contract address is correct

3. Check Alchemy/Moralis API keys are valid
4. Ensure wallet has sufficient gas for transactions
5. Verify token balance on block explorer

Implementation Date: November 14, 2025
Status: ✅ Complete and Deployed
Network: Base (Chain ID: 8453)
Contract: 0xd6df108d516a5dc83f39020a349085c79d4edf0d