Cross-Chain Transaction Detector Documentation

Project Overview

The Cross-Chain Transaction Detector is a Go-based backend system designed to monitor and detect cross-chain transactions between Bitcoin and Ethereum networks, with integration for the THORChain DEX. It monitors specified addresses and identifies patterns such as bridge transactions, wrapped tokens, and DEX swaps.

System Architecture

Core Components

1. **Monitor Service**
    - Real-time transaction monitoring
    - Pattern detection algorithms
    - Transaction processing and deduplication
    - Blockchain client coordination
2. **Blockchain Clients**
    - Ethereum Client (via Alchemy API)
    - Bitcoin Client (via BlockCypher API)
    - THORChain Client (via THORChain API)
3. **Transaction Processing Pipeline**
    - Fetches transactions across chains
    - Analyzes patterns and cross-chain activities
    - Reports and logs activities in real-time

Features

1. Multi-Chain Support

- **Bitcoin Network**
    - Monitor transactions via BlockCypher API
    - Track address balances
    - Retrieve historical transactions
- **Ethereum Network**
    - Real-time monitoring via Alchemy API
    - ETH and ERC token transfer support
    - Detects smart contract interactions

2. Cross-Chain Detection Patterns

- **Bridge Contracts**
    - Detects known bridges (e.g., Wormhole, Polygon, Avalanche)
    - Analyzes transaction flows and target chains
- **Wrapped Tokens**
    - Tracks WBTC and similar wrapped assets
    - Detects wrapping/unwrapping events and identifies original chains
- **THORChain Integration**
    - Monitors swaps and ongoing cross-chain transactions

3. Monitoring Capabilities

- Real-time transaction tracking (30-second refresh interval)
- Automatic deduplication
- Detailed logging of detected cross-chain activities

Technical Implementation

Environment Configuration

Ethereum Configuration

ETH_API_KEY=<Your-Ethereum-API-Key>

ETHEREUM_RPC=https://eth-mainnet.g.alchemy.com/v2/<Your-Alchemy-Key>

Bitcoin Configuration

BTC_API_KEY=<Your-Bitcoin-API-Key>

BITCOIN_RPC=https://api.blockcypher.com/v1/btc/main

THORChain Configuration

THORCHAIN_API=https://thorchain.net

Monitoring Configuration

MONITOR_ADDRESS=<Target-Address-To-Monitor>

Installation and Setup

1. **Prerequisites**
    - Go 1.19+
    - Git
    - Blockchain API access

2. **Steps**
3. # Clone the repository
4. git clone https://github.com/shrxyeh/cross-chain-detector.git
5. cd cross-chain-detector
6. # Install dependencies
7. go mod download
8. # Configure environment
9. cp .env.example .env
10. # Update .env with your API keys and settings
11. # Run the detector
12. go run cmd/main.go

Usage Examples

1. **Start Monitoring**
2. export MONITOR_ADDRESS=0x742d35Cc6634C0532925a3b844Bc454e4438f44e
3. go run cmd/main.go
4. **Sample Output**
5. CROSS-CHAIN TRANSACTION DETECTED ✓
6. ----------
7. Transaction Details:
8. - Hash: 0x1234...
9. - From: 0x742d...
10. - To: 0x3ee1...
11. - Value: 1.5 ETH
12. Cross-Chain Information:
13. - Source Chain: Ethereum
14. - Target Chain: Bitcoin
15. - Target Address: bc1q...
16. - Protocol: Wormhole
17. - Status: Pending
18. ----------

Security Considerations

1. **API Key Management**
    - Use environment variables for API keys.
    - Never store private keys.
2. **Rate Limiting**
    - Respect API rate limits.
    - Use backoff mechanisms for retries.
3. **Error Handling**
    - Comprehensive error logging.
    - Graceful shutdown support.

Limitations and Future Improvements

Current Limitations

- Limited to Bitcoin and Ethereum networks.
- Basic THORChain integration.
- Limited bridge contract support.

Potential Enhancements

- Support additional blockchains.
- Expand DEX integrations.
- Implement advanced pattern detection.
- Add analytics and history storage.
- Develop API endpoints for external queries.

Maintenance and Support

- Regular updates for API compatibility.
- Real-time monitoring for API health.
- Error reporting and optimization.

Conclusion

The Cross-Chain Transaction Detector offers a reliable solution for monitoring cross-chain activities between Bitcoin and Ethereum, with modularity for future enhancements. It's an essential tool for developers and researchers in blockchain interoperability.