CZU: 004.056.55

TRANSACTIONS TRACKING AND VISUALIZATION TOOL IN THE BLOCKCHAIN

Veaceslav IURCO, IPLT M. Eminescu
https://orcid.org/0000-0001-7595-3330
Olga CERBU, State University of Moldova
https://orcid.org/0000-0002-6278-7115
Nichita CRAVCENCO, State University of Moldova
https://orcid.org/0000-0002-9105-5116
Vadim GONȚA, CEITI
https://orcid.org/0000-0002-7429-4262
Roman JALBA, CEITI
https://orcid.org/0000-0003-2238-2946

Abstract. In this paper is described how the Visual Explorer works. This application is used as a guide for users who look at the transactions in crypto space. It is intended to facilitate the search for usual transactions, stolen funds as well as typical attacks in the blockchain.
Keywords: BC – BlockChain; ETH – Ether; EVM – Ethereum Virtual Machine; attacks on blockchain; cryptocurrency; mining; proof-of-work; Web3.

INSTRUMENT DE URMĂRIRE ȘI VIZUALIZARE A TRANZACȚIILOR ÎN BLOCKCHAIN

Rezumat. În această lucrare este descris modul în care funcționează Visual Explorer. Această aplicație este utilizată ca ghid pentru utilizatorii care se uită la tranzacțiile din spațiul criptografic. Acesta este destinat să faciliteze căutarea tranzacțiilor obișnuite, a fondurilor furate, precum și a atacurilor tipice în blockchain.
Cuvinte cheie: BC-BlockChain; ETH-ether; EVM – Ethereum mașină virtuală; atacuri asupra blockchain; Criptomonedă; minerit; proof-of-work; Web3.

Introduction

In this paper, we talk about an application for displaying a graph of transaction history in a blockchain based on an Ethereum Virtual Machine (EVM).

Blockchain is a shared, immutable ledger that facilitates the process of recording transactions and tracking assets in a business network. Blockchain has gained great popularity and is used in systems for various purposes such as cryptocurrencies and control systems. Blockchain provides decentralized, secure and at the same time open data storage (open database). To automate the implementation of agreements in the systems that are used in the blockchain, smart contracts will be used. The concept of the blockchain is that the network validators, which each user can become, can create new blocks with transactions, and receive remuneration for this in the currency of this blockchain, for example Ether (ETH) – the coin of the Ethereum blockchain. Validators also check the history of the blockchain for the presence of attacks or all kinds of malicious actions.

Smart contracts are simply programs stored on a blockchain that run when predetermined conditions are met. They typically are used to automate the execution of an agreement so that all participants can be immediately certain of the outcome, without any intermediary's involvement or time loss [3].

Solidity is a statically typed programming language designed for developing smart contracts that run on the Ethereum Virtual Machine (EVM).

About the project

Visual Explorer - is a tool that is primarily created for specialists whose specialization is to investigate the blockchain. Blockchain itself is a secure platform, but the problem often arises in the code of smart contracts, when even a Boolean operation can leak millions of dollars. Hackers very often use these vulnerabilities to get a tidbit and at the same time remain anonymous. But, like hackers who use their knowledge to the detriment of others, benefiting themselves, there are "white" hackers who conduct investigations, helping project "victims", which often leads to a partial refund, and sometimes a full refund.

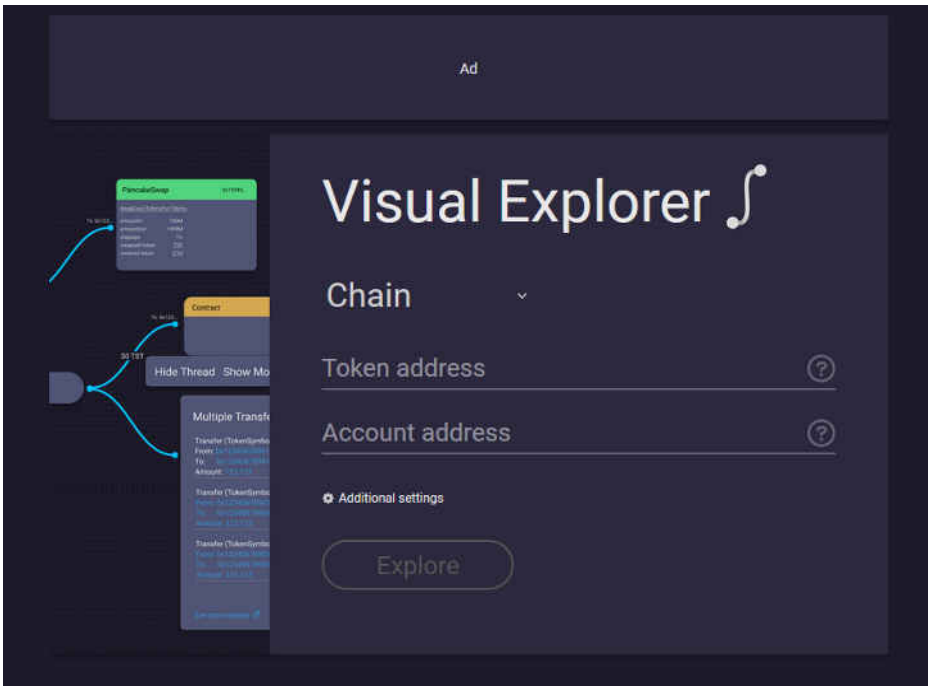

Figure 1. The interface of application

The interface of the web application is quite simple, but it will be difficult for an ordinary user, and in some cases even impossible to understand what transaction means. On the main page there are attributes such as a list of blockchains, a field for entering cryptocurrencies, and a field for entering an account whose sequence of transactions needs to be tracked. From the list of blockchains, select the necessary one (EVM-compatible blockchains are currently available) to track transactions in it, in the first input field, the address of the cryptocurrency contract is provided, for example, Fig. 2, the USDT (cryptocurrency pegged to 1 USD; in

other words USDT is a crypto dollar; $1 = 1 USDT) token is tracked. In the last input field, must be written the account address, which is a public key, in the form of a hash [1].

Since, first of all, this application is necessary to track fraudsters transactions, in the first field it is assumed that the specialist will enter the address of the cryptocurrency that was created by the attacker, and in the second field the account of the attacker himself.

When the “Explore” button is clicked, the user is shown a graph of the transaction sequence, from the moment of the fraudster's first interaction with this smart contract to the last interaction transaction, as well as everyone with whom the fraudster interacted. using some filtering, it will be possible to track all accounts that may be presumably members of the fraudster's team or his additional accounts.

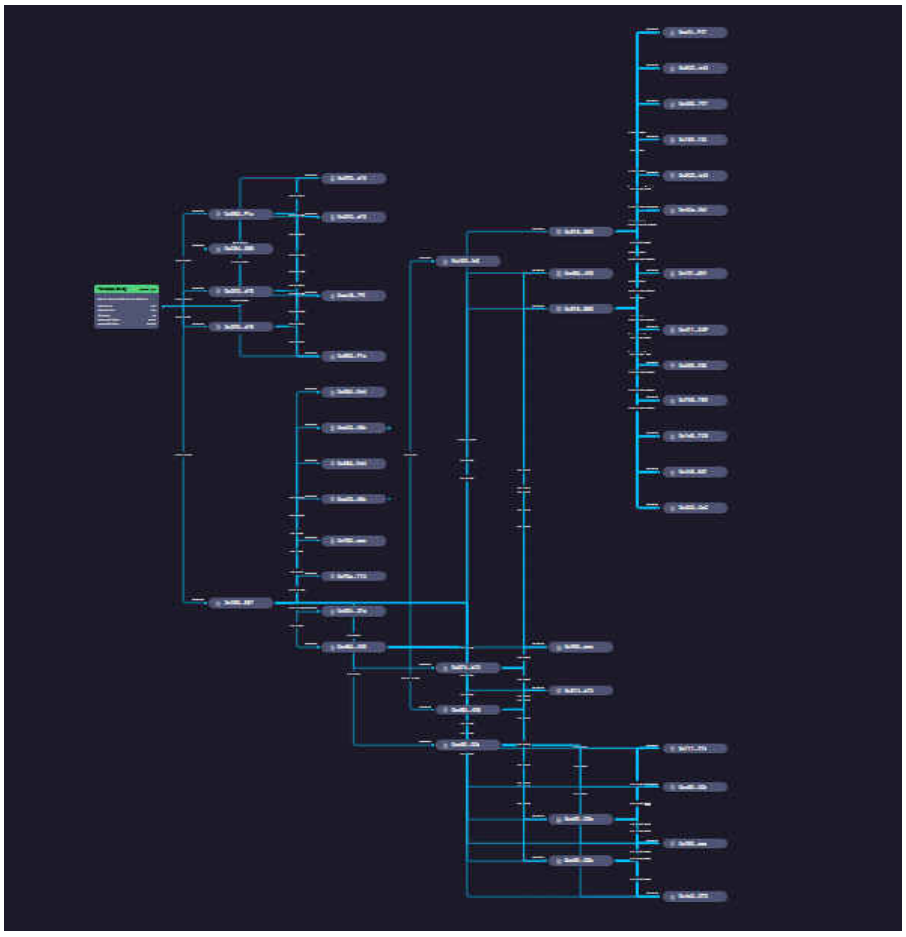

Figure 2. Transaction tracking chart

Consider an example, on a smaller number of transactions.

The first node is a root node of an account which transactions we want to monitor. From the root node the two curves originate with the value of the transaction's hash. The two nodes are representing so-called multisend transactions (multisend transactions - complex transaction in which participate 3 or more accounts or in which transfers goes from address to address until it reaches the final account) [2]. Then from these two nodes another curve originates, first nodes of which represent the addresses who did the transfers.

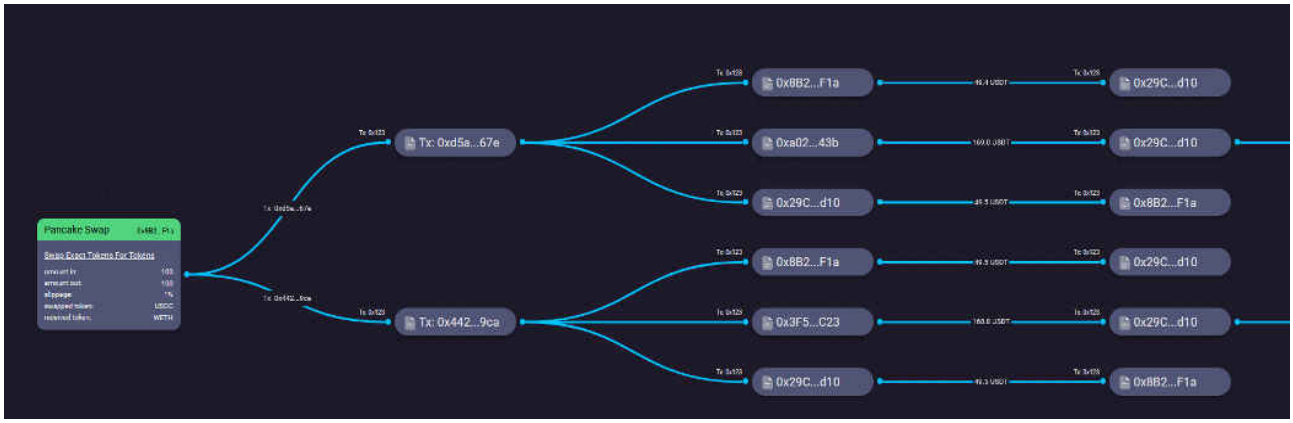

Figure 3. Chart display fragment

In order to display the graph, I use an algorithm in which we generate the position **x**, then arrange the positions by **y**.

1) Generate position by x; $x = k_i + x + n$**;**
2) Generate position by **y;**

```
While i != 0:
  if(layer == this.layers - 1):
       k = y + n;
  else:
       firstChildren = children[0];
       lastChildren = children[children.length - 1];
       y = (lastChildren.y + firstChildren.y) / 2;
  i = i - 1.
```

Conclusion

This application is designed to lower the entry threshold for people who are interested in blockchain and web3 applications, as well as simplify the work for companies conducting investigations inside the blockchain.

This moment the application is in development and the algorithm of representing transactions upgrades every day to improve user experience on it.

References

1. CONTI, M., KUMAR, S., LAL, C., RUJ, S. *A Survey on Security and Privacy Issues of Bitcoin* // arXiv:1706.00916v3 [cs.CR]. 2017. URL: https://arxiv.org/pdf/1706.00916.pdf
2. IBM. What are smart contracts on blockchain? URL https://www.ibm.com/topics/smart-contracts
3. ТРУБАЧ, Г.Г. Виды атак на блокчейн и умные контракты: https://elib.bsu.by/bitstream/123456789/216685/1/278-281.pdf