Building an Example of the DeFi Presale Cross-chain Application

Rita Tsepeleva and Vladimir Korkhov(✉)

Saint Petersburg State University, St. Petersburg, Russia
st062153@student.spbu.ru, v.korkhov@spbu.ru

Abstract. Decentralized finance (DeFi) is now a massive aggregator of many financial blockchain protocols. The overall value locked in them is currently valued at around 80 billion USD. Every day, a growing number of new users bring their investments to DeFi. Decentralized finance entails the creation of a single ecosystem of many blockchains that interact with one another. To enable DeFi, the problem of combining and interacting blockchains becomes critical. In this paper we describe the concept of a DeFi protocol, which employs cross-blockchain interface technologies, and show how a prototype pre-sale application based on the proposed concept can be created.

Keywords: Blockchain · Distributed ledger technologies · Solidity · Smart-contracts · Decentralized finances · Cross-chain

1 Introduction

Blockchain is a promising data storing technology that is still in its infancy. Unlike a traditional backend, this system may provide complete transaction transparency to its users. This means that every user is able to witness every action that takes place within a certain protocol. Furthermore, unlike a traditional backend, the blockchain distributes data storage. It prevents data loss and corruption in the event of a node failure. Furthermore, each user's data is completely secure on the blockchain which is especially important for financial data, mainly cryptocurrencies. All of these benefits make blockchain a reliable technology. Users are no longer frightened to invest in various blockchain systems thanks to them.

Consequently, we now have a massive collection of protocols, each of which is part of a larger ecosystem. The question is how to find a space for your own idea in all of these projects and how to make your crypto-product a huge success. In this paper, we'll look at a DeFi protocol that can be used as a launchpad, providing users a platform to place their own-developed tokens for presale. To expand the project's user base, we propose creating a multichain platform with a cross-chain interacting protocol rather than limiting ourselves to one blockchain network; this issue was considered in our previous work [1].

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2022
O. Gervasi et al. (Eds.): ICCSA 2022 Workshops, LNCS 13377, pp. 229–241, 2022.
https://doi.org/10.1007/978-3-031-10536-4_16

2 Related Work

The issue of integrating and interacting blockchains is a hot topic right now. The article [2] explains the fundamentals of cross-chain transactions and divides existing research into three categories: public connectors, blockchains of blockchains, and hybrid connectors. Based on established criteria, each category is further broken into sub-categories. The authors show that cross-blockchain transactions are impossible to carry out in practice without the involvement of a trusted third party. They offer the Blockchain Interoperability Methodology (BIF), a framework for evaluating blockchain interoperability solutions that defines criteria. The authors focus on the cross-chain communication protocol (CCCP), which describes how a pair of homogenous blockchains interact to appropriately synchronize cross-chain transactions.

The cross-chain transaction processing based on version control is proposed by W. Wang et al. [3]. In contrast to conventional locking-based cross-chain transaction systems, the authors suggest optimistic approaches in which modified data can be used instantly with a rolling back procedure that ensures atomicity.

The article [4] provides a variety of solutions and approaches for dealing with this issue. The authors provide a theoretical cross-communication model and compare it to the so-called mother blockchain model in order to handle cross communication across blockchain-based systems without the use of an intermediate.

Li et al. suggest a blockchain architecture to suit industrial norms that frequently require interoperability in their paper [5]. To increase the system's scalability, they introduce the concept of satellite chains, which can privately run several consensus algorithms in parallel. This approach also includes a regulator that oversees the entire network and uses smart contracts to enforce certain policies. Hyperledger Fabric is used in the prototype implementation.

H. Wang et al. present a blockchain router that allows blockchains to communicate and link [6]. To enable different blockchains in the network to communicate with each other in a similar way to the Internet, a special economic model is established. One of the blockchains in the system acts as a router, analyzing and transmitting communication requests while dynamically preserving the blockchain network's topology structure according to the communication protocol.

Zamyatin et al. present a systematic overview of cross-chain communication protocols and formalize the research topic to demonstrate that cross-chain communication is impossible without the assistance of a trusted third party [7]. The authors create a methodology for developing new CCC protocols and evaluating existing ones, focusing on the inherent trust assumptions. A centralized or decentralized trusted third party is possible. The Hyperledger Cactus proposes trustworthy validatorscitemontgomery as an example of centralized trusted parties. Another blockchain can serve as a decentralized trusted party, with members agreeing on the global ledger state via a consensus process.

The Cosmos [8] project is a fast growing ecosystem of interconnected, autonomous blockchains constructed with specific application components

(the Cosmos SDK) and connected via the Inter-Blockchain Communication (IBC) protocol. The Tendermint Byzantine Fault Tolerance (BFT) consensus method [9], which ensures finality, order consistency, and optional availability, is at the heart of Cosmos. The shared Cosmos Hub, which is powered by the Tendermint consensus and keeps track of the number of tokens in each connected chain and manages transfers between them, allows for interoperability. The transport layer (TAO), which provides the infrastructure for establishing secure connections and authenticating data packets between chains, and the application layer, which specifies how these data packets should be packaged and interpreted by the sending and receiving chains, are the two layers that make up IBC. The IBC application layer can be used to build a wide range of cross-chain applications, including token transfers, interchain accounts (delegate calls between two chains), non-fungible token transfers and oracle data feeds [10].

3 The Launchpad Application

A presale app is one of the most popular sorts of DeFi apps nowadays. A launchpad platform is another name for this type. The essence of this project type is to provide a platform on which users can pre-sell their own tokens.

There are reasons why such platforms are so popular:

- The advantage for investment users is that a large number of various assets are gathered in one place.
- The platform community is an advantage for token creators. Token holders are no longer required to find their own audience.
- Additionally, token owners are relieved of the necessity to complete numerous complex transactions on their own (token sale transfers, liquidity allocation, etc.)
- The benefit to future token holders is transparency and confidence. Each investor can view the public price of the token, the percentage of tokens guaranteed to go to the liquidity pool, the period of liquidity lock (the creator has no capacity to withdraw liquidity and collapse the liquidity pool), and other information by opening the contract code.
- The ability to collect fees from customers for various sorts of services is a benefit for the platform owner.

Problem to Solve

A platform of this type can be built within a single blockchain. However, the number of supported assets and users will be limited to the network chosen. The target platform is capable of combining assets from various blockchains. The more blockchains the platform owner adds, the more likely the platform's success and earnings will be.

However, keep in mind that when developing such a platform, you must consider the architecture, scalability, security, performance, and other important factors.

3.1 The Components of the Presale Platform

First and foremost, it is critical to distinguish platform users from other cryptocurrency wallet owners. It will assist the platform's owners in protecting their community and users from malicious spam and other information. For example, preventing the publication of a "bad" token for sale, which may be unsecure or cost nothing.

It is proposed to create the platform's own token and introduce a staking smart contract into the system to differentiate platform users from the rest of the network participants. Staking is a generic term for one type of decentralized financial protocol. In general, it works by users sending a certain amount of assets to a smart contract in order to increase them (to get rewards for it). It functions similarly to a bank deposit.

All of these protocols differ in their reward accrual systems for deposits. There are numerous staking protocols, ranging from simple annual interest rates (APR, APY) to a "share" system (in which the smart contract calculates each participant's share of the total deposit and awards him a reward based on the calculated share).

As a result, users will invest the created platform token in the staking protocol in order to participate in all of the other events and receive even more platform tokens. It validates users' intentions toward the platform and encourages them to behave appropriately in the platform community.

On the platform, a ranking system will be created. The more money a user deposits, the higher his rank will be. A high rank indicates that a specific user has some advantages (guaranteed token allocation, a high pool weight, an ability to buy tokens earlier than the other participants, etc.). A system like this will encourage users to invest and participate in presales more and more. Users will also receive rewards for their deposits, allowing their rank to rise in the future. All of these logic details can boost the platform's profit.

After making a minimum deposit and becoming a full-rights member of the platform, the user gains the ability to create presales and participate in existing presales.

If the user owns a new token and wishes to conduct an IDO, he can fill out the form and create a presale contract. What kind of information does the creator provide as a result of completing the form?

- the token contract's address
- amount of tokens for the presale
- the price per one token

That is one of the most important factors to consider when starting a sale. A soft-cap and a hard-cap are also important parameters. These parameters denote a soft-minimum and a hard-maximum of funds raised. So, a soft-cap border is the presale's minimum goal, and a hard-cap border is the maximum amount of funds raised (the presale creator can not earn more funds than the hard-cap value). For instance, suppose the creator decides to sell one million tokens. The cost of one token is 0.0001ETH. As a result, the maximum amount raised will be 1.000.000

* 0.0001ETH = 100ETH (a hard-cap). A soft-cap value should be less than or equal to the hard-cap value. This position is provided by the presale's creator and reflects the success of the presale campaign. If no funds in the amount of a soft-cap or more were collected during the sale process, the presale is usually deemed a failure. As a result, the soft cap is known as the minimum boundary. In this case, it is critical to maintain the integrity and transparency of the contract logic: users must be able to withdraw their investments if the outcome is unsuccessful, and token owners must be able to return all of their tokens.

DEX parameters are also required for carrying out an IDO-process. We will conduct a small educational program to avoid incomprehensible locations. The initial placement of liquidity on the exchange is implied by the IDO process. It means that the owner of a newly created token has decided to begin selling his token. The owner should select a decentralized exchange, which serves as a platform for token sales. In the crypto-community, there are numerous such platforms. There are platforms that are very similar to one another, which allows developers to easily interact with them all. However, there are platforms that are ideologically opposed. It's intriguing, but for a basic understanding, only the standard approach will suffice.

3.2 The Principle of Operation of a Decentralized Exchange (DEX)

To begin, we will only consider two assets: token-1 and token-2. Assume that user-1 is the creator of token-1. He wishes to begin selling token-1. Standard exchanges work on the basis of pools and reserves. The price between pool assets is defined by the reserves (or, more precisely, the reserve ratio) of the current liquidity pool. So, in order to begin selling token-1, user-1 must first create a pool on DEX with such token and some other asset. For example, user-1 is the owner of token-2. As a result, user-1 can create a pool using tokens 1 and 2. Assume that user-1 decides to provide x token-1 and y token-2 (token-1 reserve of liquidity pool equals x, token-2 reserve of liquidity pool equals y. As a result, in order to purchase one token-1 from this pool, the buyer must pay y/x token-2.

However, what happens to the liquidity pool in an exchange transaction? At the start, we have x amount of token-1 and y amount of token-2. Assume we want to purchase 1 token-1. As previously discussed, we should provide y/x token-2. And how will the reserves change? Because we only purchased one token, the reserve of token-1 will become $x-1$ and the reserve of token-2 will become $y+y/x$ (because we have provided such payment amount according to the price). It is easy to see that the reserve ratio has shifted. This implies that the price has shifted as well. The token-1 is now more expensive than before, with the price per token-1 equaling $(x(y+1)+y)/x2$. That is a significant disadvantage of such a system. Everything is dependent on the willingness of both sellers and buyers. It is assumed that demand will fluctuate around the initial price from time to time, but no one is immune from a critical situation in which one of the reserves runs out and the second becomes worthless.

But what are the benefits of using such platforms? Previously, order books were used to exchange one cryptocurrency for another. The principle of order

book operation is the creation of an order for the sale or purchase of a specific amount of an asset. When someone finds your offer appealing, he accepts it and the transaction is completed. However, it should be noted that such a person had to be waited for a very long time. Furthermore, there may not be anyone who would agree to the deal at all. In this instance, your offer has "burned out." Thus, the main advantage of decentralized exchanges is that they allow for quick transactions. Users can bring one asset and exchange it for another at any time. Furthermore, if there is no pool on the exchange with the two assets you are interested in, but there are other pools with these assets, the exchange will still find a way to exchange the tokens you are interested in. For example, user-2 owns token-3 and wishes to purchase token-2. However, there is no token-2 – token-3 pool on the DEX. However, token-1 – token-2, and token-1 – token-3 exist. As a result, the DEX will exchange token-3 for token-1 (based on the price in the 1–3 pool), and then token-1 for token-2 (according the 1–2 pool price). Token-1 serves as a connecting link in this case. Such chains can be much longer, but they all work on the same principle as described above.

These are not the only benefits of decentralized exchanges. After all, we were able to consider only the user's perspective, but we have not yet seen everything from the liquidity provider's perspective. What is the advantage of a user investing his funds in a liquidity pool? Decentralized exchanges are designed in such a way that a commission is naturally charged for each exchange transaction (otherwise, why create such a platform?) This fee is shared by all liquidity providers whose funds are currently in the pool. Furthermore, the commission is proportional to the pool's ownership share. So, simply by owning two crypto-assets, you can profit from them by investing in a liquidity pool.

At this point, we will conclude a brief overview of the work of decentralized exchanges, as we have already determined how useful it is and why people use it. Returning to the question of what parameters are required to carry out the IDO process on the launchpad.

DEX parameters fields include:

- the current DEX-platform that will be used in the liquidity allocation process;
- the listing price - from this price, token sales on the exchange will begin in the future;
- liquidity percentage allocation - the percentage of earned funds that will go to the liquidity pool in the specified ratio. For example, the presale raised funds equivalent to 0.01ETH. The creator of this presale specified a percentage value of 30% and a listing price of 0,0008ETH per token. This means that we should subtract 30% from 0.01ETH (resulting in 0.03ETH going to the liquidity pool) and divide the result by the price (0,03ETH/0,0008ETH = 37,5 tokens going to the liquidity pool). This enables the token owner to conduct the initial placement of his token on the exchange automatically by specifying only a few parameters;
- liquidity lock duration - the period of time for which the provided liquidity is "frozen." This procedure protects token buyers from falling prices (the token

owner cannot withdraw liquidity from the pool and collapse the price during this time period);
– liquidity allocation time - the timestamp after which the function of adding liquidity becomes available;

Rather than researching the entire issue and conducting a large number of transactions, the token owner can simply enter these 5 values and the launchpad platform will handle the rest.

The vesting parameters are another interesting set of parameters. The vesting process enables the presale creator to send tokens to buyers in equal parts over time. As a result, the vesting parameters are as follows:

– the percentage of purchased tokens that is available immediately after the completion of sales;
– the percentage of tokens that will be released gradually during the vesting process;
– vesting period – the period of time, each time after which the percentage of tokens is released;

Consider the following example of a simple mechanism: Assume user-1 invested 0.01ETH in token-1 during the presale period (propose that the price equals 0,0005ETH per one token). It means that user-1 should take 20 tokens. However, the creator determines the vesting percentages (suppose 20 percent the first one and 15 percent the second one). As a result, user-1 has the right to take 20 * 20% = 4 tokens at once, and the remaining 16 tokens during the vesting process. If the vesting period is one month, then user-1 will receive his funds for six months (for five months user-1 will take 3 tokens per month, and for the sixth he will receive the remaining one token from the entire amount allotted to him). Such a token release system will assist the creator in smoothly distributing the asset in order to avoid rash transactions by token holders.

3.3 The Crosschain-Part of the Application

As previously stated, it is necessary to enable interaction with various blockchains on the platform in order to increase the number of potential users and investments in the project. Everyone understands that each blockchain network is an autonomous, independent unit that has no idea how to communicate with other blockchain networks. We will use the method described in the article to make this possible on our platform [1].

But where will the cross-chain methods be used in the application? Remember that in order to participate in presales and create your own, you must first make a deposit of project tokens on the staking smart contract. This deposit confirms that you are a full platform member. To make it easier to track platform users, it is suggested that staking contracts be published only on the most popular blockchain networks. These are an Ethereum blockchain [11] and the Binance Smart Chain [12].

We use a cross-chain backend and a signature system to transfer data on the size of user deposits from these networks to other networks to the platform contracts. The signature system implies that the backend role is added to the contract (the smart contract remembers an address of our backend), which will transmit valid information received from outside to the contract (in our case from another network). The backend encrypts this information, formats it so that the contract can recognize it, and signs it with its private key. In turn, when the contract receives a message, it can determine whether the backend truly owns the private key used to sign the message. The message is accepted if the answer is correct. However, if the answer is incorrect, the transaction fails.

The most important piece of information in our staking contract to compile signatures is the deposit size of a specific user. Second, the user's wallet address associated with this deposit. We must require that the information be up to date in addition to the address and the amount of tokens deposited to the contract. As a result, the third parameter will be the time the data was read by the backend. An example is shown in Fig. 1.

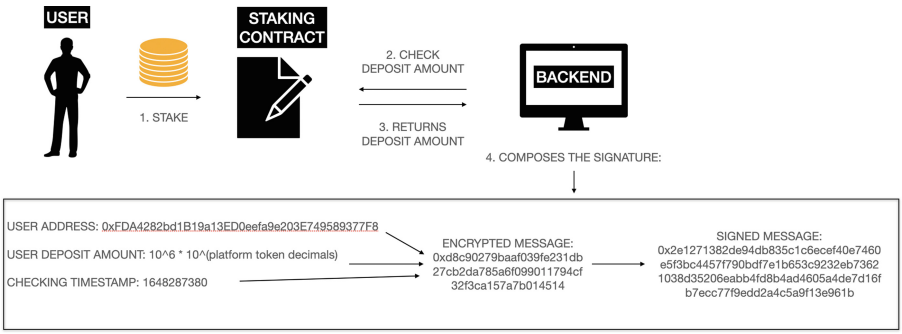

Fig. 1. Signature composing.

The figure depicts how the backend generates the signature. First, the backend receives a request from the frontend to create it, which provides it with the user's address. For this user, the backend polls the staking contract and receives the amount of his deposit at the time, as well as a timestamp of when he received this data. The three parameters are then combined into a single hash, which is used to sign the backend with its private key.

A system like this was developed to provide valid data received from outside (in our case, from another blockchain network) to smart contracts. The contract is assumed to receive the same three parameters and the final signature for the transaction input. The contract then hashes the parameters in the same way, receives a hash message, and sends the message and signature to the recover-function. The output of this function is the public address of the account that signed this message. Because the contract remembers all public addresses that can provide valid information to the contract, it will be easy for the contract to

compare the received address from the recover-function with all available ones and determine whether this information is truly from a reliable source. This type of operation allows users to prevent the contract from processing incorrect data from the outside.

3.4 Project Architecture

The architecture of the project is shown in Fig. 2.

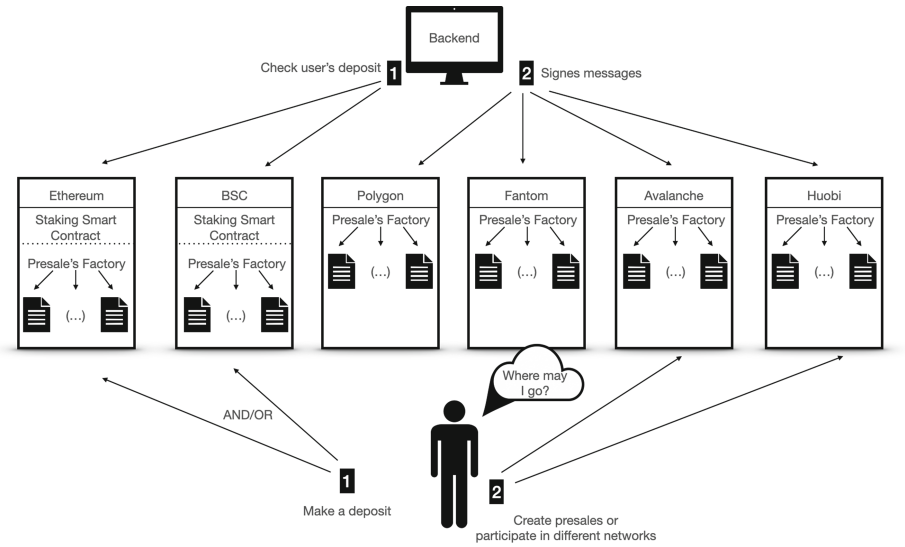

Fig. 2. Project architecture.

The following is provided in different networks:

- Staking smart contracts only available on the Ethereum and Binance Smart Chain networks. Contracts accept user deposits and send corresponding messages to the backend;
- Presale factories - smart contracts that receive messages from the backend and deploy presale contracts with the parameters listed in Sect. 3.1;
- The backend that checks information in the Ethereum and Binance Smart Chain networks and sends messages to other networks.

The backend sends messages to the "main networks" as well. The backend extends its actions to absolutely all networks of the project. Likewise, the user can connect to any network. The most important thing is to make a small deposit in one (or both) of the major networks.

3.5 The Presale's Lifecycle

Further in this section we will go through the entire presale lifecycle to better understand each step of the user on the platform:

- The user must own the platform token (for example, the token can be purchased on a DEX) and approve a staking contract on the Ethereum/BSC blockchain to use his tokens;
- To be able to create presales and participate in others, the user should submit a minimum deposit to the staking contract on the major blockchains.
- If the user wishes to build own presale, he need do the following steps:
  - fill the form with all general necessary presale's parameters:
    1. the presale token address
    2. a hardcap and a softcap (max/min boundaries for the earning funds process)
    3. an open timestamp and a close timestamp (time boundaries for the earning funds process)
    4. a token price
  - fill the form with all DEX presale's parameters:
    1. the DEX address
    2. a listing price (the price per one token on current DEX)
    3. a liquidity allocation timestamp (the time after which adding liquidity function will be available; it is necessary to be lower than close presale timestamp)
    4. a liquidity percentage allocation (a percent from raised funds that will be allocated on current DEX in pair with presale token)
  - make an approve to presales factory contract (allows smart-contract send current amount of user's tokens). The amount of approved tokens is calculated according to the following formula:

$$\frac{hardcap}{price_t} + \frac{hardcap * \frac{listing_percent}{100}}{price_l},$$

    where $price_t$ is a token price, and $price_l$ is a listing price
  - sign a transaction with these parameters for creating own presale contract;
  - waiting for investment's time passed;
  - Sign a transaction with a function that will provide liquidity to the exchange in the previously defined proportion if the investment procedure ends with "Success" (presale did not collect a soft cap payment amount);
  - If the investment process finishes with a "Fail" (presale did not collect a soft cap payment amount), sign a transaction with a function that returns presale-tokens (it is required to consider multiple scenarios of investment stage completion). In a worst-case scenario, the creator must return the tokens, and investors must return their funds;

  - When the investment collection status is “Success” and liquidity is added to the DEX, the presale creator opens the option to withdraw earned cash, and investors open vesting (investors can take the acquired tokens in portions, according to the conditions provided by the campaign creator);
  - among other things, when adding liquidity, as mentioned earlier, liquidity is blocked on the presale contract for the period specified by the creator. When this deadline is reached, the owner can take this liquidity back;
- if the user wants to participate in a presale-campaign, he should follow these steps:
  - Select a current presale contract to investigate the price per token and the amount of payment tokens already earned;
  - Decide on the investment amount and the value of the token you will receive at the end;
  - If these values appeal to the user, he can sign an investment transaction (but first, send an approve transaction to authorize the presale contract to receive your payment tokens);
  - As a result, you can make many investment transactions until the presale campaign ends or the hardcap is reached;
  - Vesting begins when the investing process is completed: Every investor can claim tokens only once during each vesting period, and the number of tokens claimed equals the total amount of his investment multiplied by the price and multiplied by the vesting percentage;
  - if the presale status is “Fail” after the presale campaign ends, every investor can withdraw his investments, and vesting does not open in this situation;

User action strategies may differ from each other. Only the main stages of user activity are presented above.

3.6 Analysis of the Results Obtained

Comparing the resulting solution with similar projects

	Our solution	Starter	PinkSale
Verified contracts (security)	Yes	No	No
Crosschain	Yes	No	No
Different strategies	Yes	Yes	Yes
Scalability	Yes	Yes	Yes
Fees	10% from investment	Not found	1BNB/0.2ETH

For greater clarity, Fig. 3 shows histograms that display quantitative estimates of the performance of the compared services.

According to the histograms it is clear that our solution meets the security requirements, and also has prospects for development when adding new blockchains, such as, for example, Tron, HuobiChain, PulseChain, Waves, and adding new DEXes in different networks.

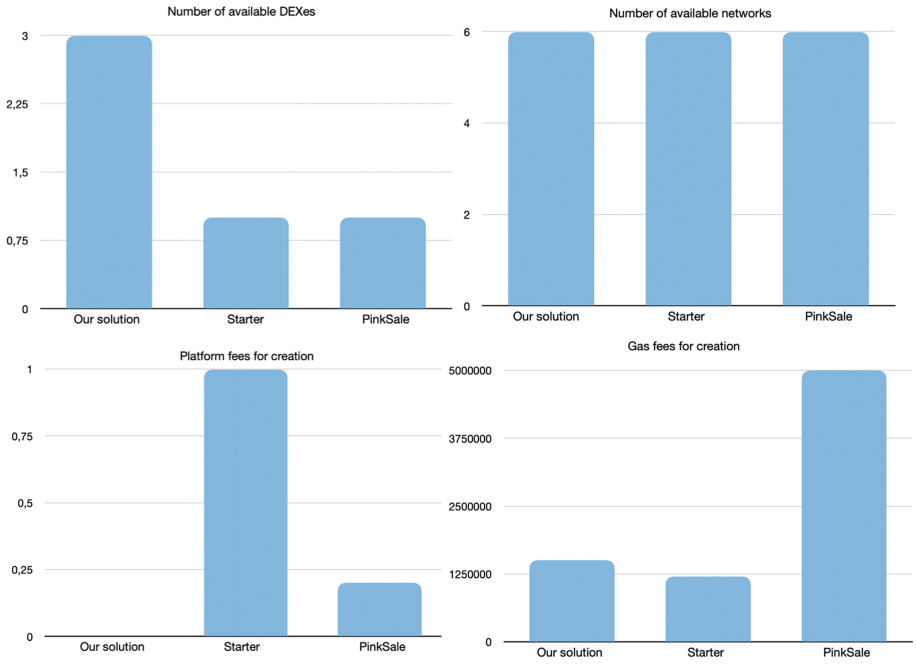

Fig. 3. Comparison of our solution with another ones.

4 Conclusions

Transferring tokens or any other information from one blockchain network to another was impossible just a few years ago. However, we have demonstrated through our experience that it is possible and necessary to develop useful and absolutely secure solutions in this area in order to give users with as much freedom as possible and eliminate all possible barriers to the adoption of blockchain technology.

We've constructed one of the most popular types of decentralized applications, the launchpad, based on the cross-chain asset transfer protocol that we proposed earlier. We've demonstrated that such apps may be scalable, simple to use, and, most significantly, not limited to a single network. DApps with these features will be able to get a lot more out of their blockchain platforms.

References

1. Tsepeleva, R., Korkhov, V.: Implementation of the cross-blockchain interacting protocol. In: Gervasi, O., et al. (eds.) ICCSA 2021. LNCS, vol. 12952, pp. 42–55. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86973-1_4
2. Belchior, R., Vasconcelos, A., Guerreiro, S., Correia, M.: A survey on blockchain interoperability: past, present, and future trends. ACM Comput. Surv. **54**(8) (2022). https://doi.org/10.1145/3471140. Article no. 168

3. Wang, W., Zhang, Z., Wang, G., Yuan, Y.: Efficient cross-chain transaction processing on blockchains. Appl. Sci. **12**(9), 4434 (2022). https://doi.org/10.3390/app12094434
4. Pillai, B., Biswas, K., Muthukkumarasamy, V.: Cross-chain interoperability among blockchain-based systems using transactions. Knowl. Eng. Rev. (2020). https://doi.org/10.1017/S0269888920000314. Article no. e23
5. Li, W., Sforzin, A., Fedorov, S., Karame, G.O.: Towards scalable and private industrial blockchains. In: Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts, April 2017, pp. 9–14 (2017). https://doi.org/10.1145/3055518.3055531
6. Wang, H., Cen, Y., Li, X.: Blockchain router: a cross-chain communication protocol. In: Proceedings of the 6th International Conference on Informatics, Environment, Energy and Applications (2017)
7. Zamyatin, A., et al.: SoK: communication across distributed ledgers. Technical report (2019). https://eprint.iacr.org/2019/1128.pdf
8. Cosmos. https://cosmos.network
9. Kwon, J.: Draft v. fall. Tendermint: Consensus without mining (2014). https://tendermint.com/static/docs/tendermint.pdf
10. IBC Protocol. https://ibcprotocol.org/
11. Ethereum [Electronic resource]. https://ethereum.org
12. Binance Chain Documentation [Electronic resource]. https://docs.binance.org