Proceedings of the 2nd International Conference on Business and Policy Studies
DOI: 10.54254/2754-1169/7/20230246

Delving into the Security Side of Decentralized Finance Applications

Wenhao Xie[1,a,*]

[1] *Case Western Reserve University, Cleveland OH 44106, USA*
a. xxwwhh529@gmail.com
**corresponding author*

Abstract: This article is a study on the security of decentralized finance. We summarize the background and application of decentralized finance, as well as the development history and latest research of decentralized finance. Moreover, we propose a classification framework for existing various DeFi DApps. We also manually collected attacks against decentralized financial applications and classified them by category. Through some case studies, we give some best practices for practitioners in decentralized finance industries to avoid or defend against attacks.

Keywords: decentralized finance, security, cryptocurrencies, finance

1. Introduction

Decentralized Finance (DeFi) is a brand-new financial system for everyone, without the need for trusted intermediaries such as banks. Although there is no specific birth date for decentralized finance, some important historical events have made the emergence of DeFi possible.

The first important event was Nakamoto's invention of bitcoin in 2009. The Bitcoin platform adopts peer-to-peer (p2p) and blockchain technology. And Bitcoin, the official token, is not issued by a centralized monetary institution, but is generated by massive calculations of specific algorithms, dubbed as mining. Bitcoin is also traded through the p2p network rather than through a centralized bank, which significantly reduces the transaction fee and improve the anonymity. As such, Bitcoin's market cap was around 550 billion USD [1]. Nevertheless, due to Bitcoin's consensus algorithm, i.e., Proof of Work (PoW), problems emerged, e.g., network congestion and slower transaction speed. Although it is feasible for people to send bitcoins around the world, finance is more than that. Every mature financial system needs a series of other important services, such as lending, trading, or derivatives. Though Bitcoin has a script system, it is non-Turing-complete, which means it is not practical for developing the above applications.

Ethereum was proposed by in 2013 [2], and has a relatively higher transaction-per-second (TPS) than Bitcoin due to its improvement on the consensus algorithm. Moreover, Ethereum has proposed smart contracts, which can be deployed and executed on the Ethereum platform. As smart contracts are able to interact with each other by transactions, Ethereum began to attract more and more developers who want to build a variety of decentralized applications (DApps), from games to financial applications. Moreover, through the ERC-20 interface, Ethereum allows users to issue tokens that can be freely flowed across the network. These characteristics jointly makes Ethereum

DOI: 10.54254/2754-1169/7/20230246

stand under the spotlight, whose final market value reached 346 billion USD. Finally, it gave birth to the decentralized financial (DeFi) applications.

DeFi is a combination of cryptography, blockchain, smart contract, etc., in which smart contract is an important part. DeFi projects use smart contracts to implement various functions of traditional financial institutions, such as derivatives, lending, trading, financial management, asset management, and insurance. However, like any emerging technology, DeFi also suffers risks. For instance, cryptocurrency platform Poly Network was hit by a major attack in Aug. 2021, where the hacker has stolen more than 600M USD worth of tokens. Therefore, the security issues of DeFi have been placed under the spotlight. For example, Bekemeier discussed the systematic risks at the technical level of blockchain and economic level, and provided empirical analysis [3].

In this paper, we first introduce some basic knowledge about blockchain and smart contracts. Then, we propose a classification framework for existing various DeFi DApps. We also introduce some security issues against DeFi DApps, and conduct some case studies. Finally, we give some best practices for DeFi developers and users.

2. Background

In this section, we will briefly introduce some basic information about blockchain technology, smart contracts, and decentralized finance applications.

2.1. Blockchain

Blockchain technology originated from Bitcoin and can be regarded as a decentralized database. Specifically, Bitcoin dynamically adjusts the generating speed of a new block, which will be appended to the end of the existing block every 10 minutes. Such a continuously increasing data structure is called “blockchain”. The blockchain distributes the “ledgers”, which should be kept by the original centralized institutions, to all participating client nodes in the world, where each node participates in bookkeeping. Therefore, a blockchain contains all historical transaction records since the start of its corresponding platform.

2.2. Smart Contract

The concept of smart contract is defined by Nick Szabo, which can be regarded as a programmable contract [4]. Smart contracts allow the setting of transaction conditions, and the execution of transactions does not require trust in a third party. However, because there is no trusted execution environment, the smart contract itself has never been applied to the practical industry. The birth of blockchain technology provides the feasibility for the use of this technology. Specifically, a smart contract corresponds to an account in the blockchain, integrating both code and data. Once users invoke these smart contracts by transactions, their targeted functions will be executed, whose behavior is hard-coded in the smart contract.

2.3. Decentralized Finance (DeFi)

DeFi projects use smart contract technology to realize various functionalities derived from traditional financial institutions, e.g., derivatives and exchanges.

To be specific, similarly to the conventional derivatives, derivative DeFi application means financial contracts which derive their value from the performance of underlying assets. It is generally an agreement between two or more subjects, including traditional derivatives written in contracts, like future contracts and option contracts. As for the exchanges, which can be divided into centralized exchanges (CEX) and decentralized exchanges (DEX). In CEX, the deposited digital assets from

DOI: 10.54254/2754-1169/7/20230246

users are centrally kept and controlled by an institution or a company. Contrarily, in DEX, all the funds are kept in the user's wallet address or trading smart contract, which is completely controlled by the user's private key. When a user initiates a transaction to a DEX, it will execute the smart contract to complete the transaction, and the asset transfer is completed on-chain fully, which is open and transparent.

Besides, DeFi has other practical applications, like stable coin, lending market and insurance. We will introduce the whole DeFi ecosystem according to their classification combined with representative applications in Section IV.

3. Related Work

3.1. Smart Contract Security Analysis

Smart contract is one of the most important functions in blockchain applications. It realizes trusted transactions without a third party. However, with the rapid development of blockchain smart contracts, many security problems have also been exposed, and the attacks caused by some contract vulnerabilities have led to terrible losses.

As the number of smart contracts on the chain continues to increase, making it more difficult for manual audit, various research institutions have begun to study the methods of automated audit of smart contracts. He et al. studied the security vulnerabilities of Ethereum smart contract, its defense mechanism and some security audit methods [5]. Their research included three analytical tools: oyente, porosity, and mythril. surveyed 27 smart contract analysis tools [6]. Durieux et al. used 9 automatic analysis tools to analyze 47587 Ethereum smart contracts, and found that 97% of the contracts had vulnerabilities [7]. Tang et al. surveyed 15 analysis tools and their related vulnerabilities. They reviewed tools in three categories such as static analysis, dynamic analysis, and formal analysis [8].

3.2. Decentralized Finance (DeFi)

Related work on our primary research topic, exploitation of DeFi protocols, is relatively limited due to the short existence of the industry. For example, Chen et al. have focused on blockchain technology which can reduce transaction costs, expand transaction scope, and empower peer-to-peer transactions, creating a new paradigm for decentralized business models [9]. Werner et al. considered DeFi from two points of view, the DeFi Optimist and the DeFi Pessimist, and examined the workings of DeFi systematically [10]. Oosthoek enumerated security threats to DeFi projects based on in-the-wild attacks, as well as countermeasures to inform mitigation [11].

4. Taxonomy of DeFi

In this section, we will illustrate the taxonomies of DeFi categories, as well as corresponding representative applications. Compared with traditional centralized finance, DeFi projects have the following characteristics:

- *Open-source and transparent*. Since most mainstream projects are open-source for public review, bugs should be found quickly. At least 29 cybersecurity firms, e.g., OpenZeppelin [12], Trail of Bits [13], and ConsenSys Diligence [14], currently provide auditing services for cryptocurrency projects. Moreover, many DeFi projects offer bug bounty programs for white hats to encourage active participation in finding vulnerabilities.
- *Decentralized operation and supervision*. DeFi treats any participant indiscriminately. Meanwhile, DeFi projects cannot be supervised by any centralized institutions after the launching. In other words, these DeFi applications are supervised by Decentralized Autonomous Organizations,

DOI: 10.54254/2754-1169/7/20230246

a.k.a. DAO, which is an organization whose behaviors are controlled jointly by recognized smart contracts. Any community member can initiate a proposal, and users can vote to determine the development direction of the project based on their holdings.

4.1. Derivative

There are four types of derivatives: future, forward, swap, and option. In the field of DeFi, futures derivatives are very important for traders to hedge, since futures let the buyer purchase the asset at an agreed price on a fixed date in the future. In traditional asset trading markets, we can buy and sell stocks, real estate, precious metals or bulk commodities. However, at the current stage of DeFi, we do not have a direct way to trade them without the help of oracles, which inevitably leads to centralized concern from users. Therefore, the project side put forward the synthetic assets: a financial instrument composed of one or more assets or derivatives. For example, USDT is a synthetic asset introduced into the blockchain to anchor the USD.

Example: Synthetix. Synthetix is a decentralized derivative application issued on Ethereum. On the Synthetix, users can issue and trade synthetic assets, i.e., tokenized derivatives. Synthetic assets track changes in the value of underlying assets and allow exposure to assets without holding actual assets. In order to issue specific synthetic assets, users must provide collateral in the form of SNx tokens. The agreement requires excess collateral, 500% at the current mortgage rate, which is mainly to cope with any sharp price changes of synthetic assets. This means that only $100 worth of synthetic assets can be issued once every $500 SNx locked in the system.

4.2. Loans

There are many similarities between decentralized lending and traditional financial lending. However, the biggest difference is that decentralized lending takes digital currency as collateral, and the lending agreement is based on smart contracts, not based on specific objects or centralized institutions. Lending and borrowing of on-chain assets are facilitated through protocols for loanable funds (PLFs), which refer to DeFi lending protocols that establish distributed ledger-based markets for loanable funds of crypto assets by pooling deposited funds in smart contracts [15].

Example: Makerdao. Makerdao is one of the most popular lending platforms. It allows users to pledge assets such as Ether as collateral to obtain stable currency DAI, which can be used for storage or lending. MakerDAO adopts a dual currency model, one is the stable currency Dai, which is the currency you can eventually borrow. The other is the equity token and management token MKR, which is the interest that users need to pay when redeeming the mortgaged Ethereum. Through the dual currency mechanism, MakerDAO enables the entire decentralized collateral loan system to operate.

4.3. Stable Coin

Stable coin is a type of cryptocurrency designed to maintain a stable market price. Collateralized stable coin companies are expected to actually hold assets (e.g., USD or gold) pegged to their tokens. Therefore, they issue new units based on the value of their holdings, which is the basis for most stable coins. Prominent examples include USD Coin (USDC), Paxos (PAX), and TrueUSD (TUSD), where each token is backed 1:1 with USD held in a bank account. The problem of these stable currencies is that they are too centralized, and the issuing company needs to hold dollars or other assets of the same value. Besides, some stable coins are pegged to other cryptocurrencies rather than fiat currencies or commodities, and these are often referred to as crypto-collateralized stable coins. The pegs for these coins are maintained through over-collateralization and stabilization mechanisms. A well-known example is DAI, a stable coin minted in the MakerDAO ecosystem. There also exist non-

DOI: 10.54254/2754-1169/7/20230246

collateralized stable coins that utilize algorithms to control the supply of tokens in order to fix the price at a predetermined level. The goal of these tokens is to maintain a stable value by algorithmically expanding and contracting their circulating supply based on market behavior.

Example: Ampleforth. This is the first algorithmic stable coin based on the concept of rebase to balance supply and demand. When the price of AMPL exceeds $1, the protocol will perform a deliberate inflation on AMPL through increasing everyone's balance to make sure each AMPL is only worth $1.

4.4. Exchanges

Decentralized exchanges (DEX) allow users to trade digital currencies. Using DEX requires a public key, and users are in charge of their own private key, so centralized hosting is no longer required. There are many types of DEX, of which there are two main principles: automated market maker (AMM) and order books. Specifically, the former one is found on CEX and keeps a record of all of the ongoing trading activities, but may encounter insufficient liquidity, while the later one provides liquidity algorithmically through simple pricing rules with on-chain liquidity pools in place of order books.

Example: Uniswap. Uniswap has become the leading DEX in the DeFi ecosystem. The core is its first automatic market maker system. Unlike the traditional CEX order book trading mode, anyone in Uniswap can increase the liquidity by adding token pairs. Through providing token pairs without counterparty, it greatly reduces the transaction threshold and is closer to the essence of DeFi and blockchain, i.e., decentralization.

Specifically, a liquidity pool has two kinds of tokens. The first liquidity provider (LP) will set the initial price of the pool once a liquidity pool is created. All LPs have to maintain the stable value of the two assets, as an arbitrage space will be created if the initial price is out of line with the global market price. In exchange for the liquidity, LPs receive a special LP token, representing the proportion of liquidity they provide. If a LP wants to withdraw the liquidity they provide, the LP token will be destroyed. Note that, once LP tokens are destroyed, the price adjustment is initiated according to the deterministic algorithm. Uniswap adopts the constant product market maker algorithm:

$$x * y = k$$

Specifically, the product of the number of token X and the number of token Y is always equal to the constant K. Therefore, for a token, if the demand increases, the algorithm will always increase its price. However, such an intuitive approach has an intrinsic problem. If the transaction volume is relatively large to the liquidity pool, the price of the token will fluctuate dramatically, leading to attackers exploiting or manipulating the price of a token to achieve arbitrage. Therefore, Uniswap V2 and subsequent updates have introduced the weight of time parameters, which allow time-weighted average pricing based on token pair prices at each block [16].

4.5. Insurance

Due to the potential huge expenditure scenario in the DeFi ecosystem, tokens pledged in smart contracts are vulnerable to attacks. Although smart contracts of most projects are subject to code audit, the possibility of being hacked always exists, which will lead to capital loss. For example, bZx had three high-profile DeFi security attacks in 2020, resulting in enormous financial loss [17]. Therefore, the insurance DeFi can provide some compensation in case of loss of capital due to specific scheduled events, including attacks on DeFi protocols and smart contract failures.

DOI: 10.54254/2754-1169/7/20230246

Example: Nexus Mutual. Nexus Mutual is a decentralized insurance protocol based on Ethereum. At present, it can provide security for smart contracts on Ethereum. Specifically, its members bear insurance risks and enjoy premium income. This relationship is realized through its token, named NXM, which is not allowed to go on the exchange avoiding the risk of price manipulation. A user can become an insured or an insurer. As an insured, the user can choose a DApp, set the value and time to be insured, and then produce a quotation of the DApp. The user can use ETH, Dai or NXM to pay for the policy. In this way, if he has assets in the contract, once a vulnerability causes his property loss, Nexus Mutual will make compensation. If you want to be an insurer, you need to buy NXM with Dai, then select a DApp you want to protect and pledge a certain amount of NXM.

5. Attack

We can divide attacks into two categories, one is *arbitrage* by exploiting flaws in business logic through arbitrage attack, and the other is *exploiting smart contracts' vulnerabilities* to steal or transfer property. Table 1 illustrates the latest representative attacks against DeFi DApps. We will conduct case studies in this section.

Table 1: Representative attack events against DeFi platforms.

Attack category	Hacking Style	Event	Financial Loss
Arbitrage	Flash Loan Attack	Yearn. Finance [18]	$11m
	Flash Loan Attack	bZx Hack [19]	$8m
	Flash Loan Attack	DODO DEX Exploit [20]	$3.8m
	Flash Loan Attack	Cheese bank [21]	$3.3m
	Flash Loan Attack	Harvest finance [22]	$34m
	Flash Loan Attack	Balancer [23]	$500k
Exploiting Smart Contract	"evil contract" exploit	Furucombo Drain [24]	$14m
	Vulnerable contract calls	PolynetWork [25]	$600m
	Reentrancy attack	dForce & Lendf.Me [26]	$25m
	Smart contract vulnerability	Bancor [27]	$23.5m

5.1. Arbitrage on bZx

On February 15, 2020, the bZx team said that a hacker attacked the bZx protocol by an arbitrage attack, leading to the suspension on most of its functions. The attack flow is shown in Fig. 1. Specifically, in step 1, the attacker borrowed 10,000 ETH by calling the dYdX flash loan function in the contract. The attacker deposits 5,500 ETH of it into Compound as collateral in step 2, borrowing 112 WBTC. Step 3 takes advantage of the bZx margin trade feature to short ETH in favor of WBTC. In this transaction, bZx forwards the order to KyberSwap. KyberSwap basically queries its reserves and finds the best exchange rate. As only Uniswap can provide such liquidity, this transaction essentially drove the price of WBTC in Uniswap to three times higher. After the WBTC price surged in Uniswap, the attacker sold all the 112 WBTC borrowed from Compound in step 2 to Uniswap and returned the corresponding WETH. At this step, the attacker gains the profit.The attacker repays the 10,000 ETH of the flash loan to dYdX in step 5, completing the flash loan repayment.

DOI: 10.54254/2754-1169/7/20230246

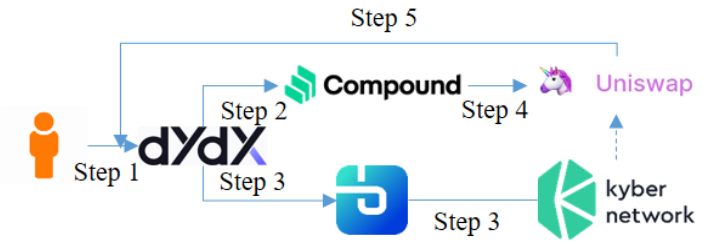

Figure 1: Five steps to attack bZx.

5.2. Exploiting Smart Contract

Exploiting Smart Contract on PolyNetwork. On August 10, 2021, PolyNetwork suffered a cross-chain attack and $600 million valued encrypted assets were transferred. In PolyNetwork, users can create cross-chain transactions on the source chain. Once the transaction is confirmed, the source chain Relayer, the Poly Chain and the target chain's Relayer will consecutively relay the corresponding block's header. If the head is verified on the target chain, the cross-chain transaction initiated by the user is achieved.

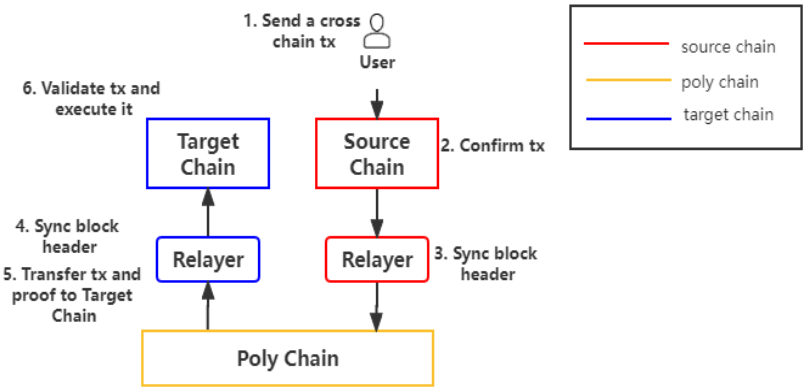

Figure 2: Structure of Poly Network.

This attack involves two contracts, which are dubbed as ECCD and ECCM, respectively. The former one verifies the block headers synchronized by Poly Chain, while the latter one stores cross-chain data, as well as the public key of the relayer validator, Keeper. The hacker initiated a malicious transaction that should have been invalid on the public chain. Then, the source chain's relayer incorporates this transaction into the Merkle tree of Poly Chain and signs it without sufficient inspection, and then publishes it in the Poly Chain block. After that, the hacker uses the valid Merkle proof on Ethereum to call the ECCM contract of Poly Network and change the keepers to the public key controlled by the hacker. Finally, after obtaining the keepers permission, hackers can unlock assets arbitrarily on multiple public chains.

Exploiting Smart Contract on Lendf.Me. On April 19, 2020, Lendf.Me was attacked and $25 million worth of crypto assets were drained from the contract. The attacker used the reentrancy vulnerability in the Lend.Me account to steal all the assets. In the same week, Uniswap was also attacked in a similar way, which was jointly caused by the reentrancy vulnerability in DeFi and ERC-777.

DOI: 10.54254/2754-1169/7/20230246

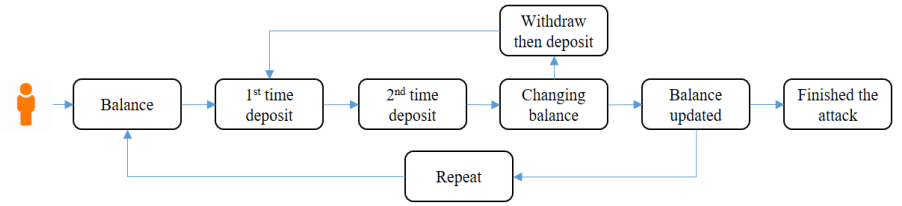

Figure 3: Flow diagram of lendf.me attack.

Specifically, as the ERC-777 protocol does not stipulate how to implement functions and DeFi contract developers should not make any assumptions about the implementation of participants. In the attack smart contract against Lendf.me, the hacker called the function defi.withdraw () in the tokensToSend. The attacker made two calls to the supply () function of Lendf.Me, but these two calls were independent, not calling the supply () function again in the previous supply () function. During the second call of the supply () function, the attacker calls the withdraw () function of Lendf.Me in his own contract, and finally withdraws the money. The attacker's withdraw () call occurs in the transferFrom function, that is, it is called when Lendf.Me calls the user's tokensToSend() hook function through transferFrom. The attacker re-entered the Lendf.Me contract through the supply () function, causing a reentrancy attack.

6. Best Practice

DeFi is attractive and profitable for users and investors. But from the above-mentioned attacks, we need to know that solving our own risk control is the top priority. Thus, we will give some best practices for DeFi developers and users.

For developers, we should always be vigilant about possible vulnerabilities in contracts and make continuous improvements. For example, after the attack of Lendf.me, we should know that when it comes to the connection of multiple contracts, we need to ensure their consistency in security. Besides, setting up an insurance mechanism is necessary when using the Oracle machine to obtain the external price. If the fluctuation is too large, the transaction should be suspended in time. Prevent the market from being maliciously manipulated and causing losses. Failure to establish such a mechanism is one of the reasons why bZx is under attack.

For users, we don't need to lose confidence in the DeFi industry. For instance, after the Lendf.me attack, some users do not trust ERC-777, but ERC-777 is a good token standard, which can greatly improve the user experience of Defi applications. By using the hook callback mechanism, two or more transactions are required in ERC-20, while using ERC-777 a single transaction can be completed. At the same time, ERC-777 can avoid reentrancy attacks, such as adding reentrancy restrictions to Defi contracts.

7. Conclusion

We propose a detailed DeFi taxonomy framework, which includes derivatives, loans, stable coins, exchanges, and insurance. We roughly divide attacks against DeFi into two categories, one is arbitrage and the other is exploiting smart contracts' vulnerabilities, and listed some attack events. We conduct case studies on representative DeFi attack events. Finally, we provide best practices for avoiding or defending against attacks for both developers and users.

References

[1] *Cryptocurrency prices, charts and market capitalizations. CoinMarketCap. (n.d.). Retrieved May 22, 2022, from https://coinmarketcap.com.*

DOI: 10.54254/2754-1169/7/20230246

[2] Tapscott, D., & Tapscott, A. (2016). Blockchain revolution: how the technology behind bitcoin is changing money, business, and the world. Penguin.
[3] Bekemeier, F. Deceptive assurance? A conceptual view on systemic risk in decentralized finance (DEFI). 2021 4th International Conference on Blockchain Technology and Applications (2021).
[4] Buterin, V. Ethereum whitepaper. ethereum.org. Retrieved May 20, 2022, from https://ethereum.org/en/whitepaper/ (2014).
[5] He, D., Deng, Z., Zhang, Y., Chan, S., Cheng, Y., & Guizani, N. Smart contract vulnerability analysis and security audit. IEEE Network, 34(5), 276–282 (2020).
[6] di Angelo, M., & Salzer, G. A survey of tools for analyzing Ethereum Smart Contracts. 2019 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPCON) (2019).
[7] Durieux, T., Ferreira, J. F., Abreu, R., & Cruz, P. Empirical Review of Automated Analysis Tools on 47,587 Ethereum Smart contracts. Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering (2020).
[8] Tang, X., Zhou, K., Cheng, J., Li, H., & Yuan, Y. The vulnerabilities in smart contracts: A survey. Advances in Artificial Intelligence and Security, 177–190. (2021).
[9] Chen, H., Pendleton, M., Njilla, L., & Xu, S. A survey on Ethereum Systems Security. ACM Computing Surveys, 53(3), 1–43 (2021).
[10] Werner, S. M., Perez, D., Gudgeon, L., Klages-Mundt, A., Harz, D., & Knottenbelt, W. J. (2021, September 26). Sok: Decentralized finance (DEFI). arXiv.org. (Retrieved May 20, 2022).
[11] Oosthoek, K. Flash crash for cash: Cyber threats in decentralized finance. arXiv.org. Retrieved July 18, 2022, from https://arxiv.org/abs/2106.10740 (2021, June 20).
[12] OpenZeppelin. Retrieved July 26, 2022, from https://www.openzeppelin.com/ (n.d.).
[13] Trail of Bits. Retrieved July 26, 2022, from https://www.trailofbits.com/ (n.d.).
[14] Fuzzing. ConsenSys Diligence. Retrieved July 26, 2022, from https://consensys.net/diligence/fuzzing/?utm_source=google&utm_medium=cpc&utm_campaign=Diligence_Search_Brand&gclid=Cj0KCQjwof6WBhD4ARIsAOi65agPK_rXXvLDnsBK3TrAbCN3mVot5Mqw1dNBM7x21l1hoI1d1Mp9orwaAojjEALw_wcB (n.d.).
[15] Gudgeon, L., Werner, S., Perez, D., & Knottenbelt, W. J. DEFI protocols for loanable funds. Proceedings of the 2nd ACM Conference on Advances in Financial Technologies (2020).
[16] Uniswap V2 Overview. Uniswap Protocol. Retrieved July 26, 2022, from https://uniswap.org/blog/uniswap-v2 (2020, March 23).
[17] Rubin, J. Defi lender bZx suffers hack for reported $55m. CoinDesk Latest Headlines RSS. Retrieved May 20, 2022, from https://www.coindesk.com/business/2021/11/05/defi-lender-bzx-suffers-hack-for-reported-55m/ (2021, November 5).
[18] DEFIYIELD.App. Yearn Finance Exploit explained. Medium. Retrieved May 20, 2022, from https://blog.defiyield.app/yearn-finance-exploit-explained-a10b07c280c8 (2021, May 27).
[19] Chipolina, S. DEFI protocol bZx hacked for third time, loses $8 million. Decrypt. Retrieved May 20, 2022, from https://decrypt.co/41718/defi-protocol-bzx-hacked-for-third-time-loses-8-million (2020, September 14).
[20] Behnke, R. Explained: The dodo dex hack (March 2021). halborn. Retrieved May 20, 2022, from https://halborn.com/explained-the-dodo-dex-hack-march-2021/ (2021, March 12).
[21] Ivan. Defi Deep Dive - Top Defi Hacks of 2020. Moralis Academy. Retrieved May 20, 2022, from https://academy.moralis.io/blog/defi-deep-dive-top-defi-hacks-of-2020 (2021, January 9).
[22] Crystal analytics team. Defi hacks case study: Harvest finance protocol. Crystal Blockchain Analytics for Crypto Compliance. Retrieved May 20, 2022, from https://crystalblockchain.com/investigations/defi-hacks-case-study-harvest-finance-protocol/#:~:text=What%20happened%20to%20the%20stolen%20funds%20taken%20from%20Harvest%20Finance%3F&text=As%20a%20result%20of%20this,ERC%2D20%20token%20for%20another. (2021, April 14).
[23] Shevchenko, A. DEFI protocol balancer hacked through exploit it seemingly knew about. Cointelegraph. Retrieved May 20, 2022, from https://cointelegraph.com/news/defi-protocol-balancer-hacked-through-exploit-it-seemingly-knew-about (2020, June 29).
[24] Thurman, A. Transaction batching protocol furucombo suffers $14 million 'evil contract' hack. Cointelegraph. Retrieved May 20, 2022, from https://cointelegraph.com/news/transaction-batching-protocol-furucombo-suffers-14-million-evil-contract-hack (2021, February 27).
[25] Browne, R. Hacker behind $600 million crypto heist returns final slice of stolen funds. CNBC. Retrieved July 18, 2022, from https://www.cnbc.com/2021/08/23/poly-network-hacker-returns-remaining-cryptocurrency.html (2021, August 24).
[26] Zhao, W. DForce hacker returns almost all of stolen $25m in crypto. CoinDesk Latest Headlines RSS. Retrieved May 20, 2022, from https://www.coindesk.com/markets/2020/04/21/dforce-hacker-returns-almost-all-of-stolen-25m-in-crypto/ (2020, April 21).

DOI: 10.54254/2754-1169/7/20230246

[27] *Russell, J. The crypto world's latest hack sees Bancor lose $23.5m. TechCrunch. Retrieved May 20, 2022, from https://techcrunch.com/2018/07/10/bancor-loses-23-5m/?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAACBLO8pTf7jn_hUXNrTEzeOUatd0SXO8fyaJ-aumGSJCB6mDXDUV8V992ewu0abR11OURNg5YvfrSmeRvRZJMis1K_qxxcBIbldqhNhYUwlzh9ZmJG-O46vsw3rXS1mp6Rev78i-pysEBQiS9VP-EndchKfSf60TlSNxdPyKJ5QU (2018, July 11).*