DeFi Talents

Organized by

Frankfurt School Blockchain Center

Web3 Talents

Powered by

KFW

INNO LAB

Deka

DATENSERVICE

dwpbank

Supported by

DEC INSTITUTE

DCA

MAMA

ECOTA

Assignment 6

January 13, 2025

By

Manjiri Birajdar

Software Developer

Assignment 6:

1. **What were the top 5 biggest DeFi exploits?**
    a. Pick one and explain what vulnerabilities were exploited
    b. Blame game: Do you blame the “hackers” or do you blame the project (developers)?

2. **DeFi Insurance to the rescue?**
    a. Who are the 3 biggest DeFi insurance platforms
    b. Compare in a table what events they cover, what they don’t cover, and how they operate
    c. What are the use cases of DeFi Insurance outside of DeFi?
        i. What event could be covered?
        ii. Is a fully decentralized implementation even possible?

DYOR

Assignment 6

4. **Smart Contract Auditing**
    a. Why is smart contract auditing important?
    b. How are smart contracts audited?
    c. Which risks exist in addition to smart contract risks and how can they be mitigated?

5. **(OPTIONAL) What are the most popular smart contract programming languages in terms of value “secured” and the number of developers in web3 using those languages?**

6. **Getting to know stakeholder domains (Multiplicators)**
    a. Imagine you are a marketing manager of a DeFi service/product. What marketing strategy/measures would you use to make it go “viral”? Be as specific as possible.
    b. What is the role of airdrops, bounty programs, and memes in DeFi marketing?

DYOR

What were the top 5 biggest DeFi exploits?

1. The DAO Hack (2016)

- The DAO (Decentralized Autonomous Organization) was a smart contract-based venture capital fund. Hackers exploited a vulnerability in the smart contract's recursive call function, leading to the theft of over $50 million worth of Ether.

2. bZx Flash Loan Attacks (2020)

- bZx, a decentralized lending protocol, suffered multiple flash loan attacks where attackers manipulated the price oracles and caused losses of over $8 million. The attackers used flash loans to artificially inflate prices and exploit vulnerabilities in the protocol.

3. Poly Network Hack (2021)

- Poly Network, a cross-chain DeFi platform, suffered a $610 million hack. The attacker exploited a vulnerability in Poly Network's contract that allowed the attacker to transfer funds between blockchains in an unauthorized manner. Fortunately, the hacker returned most of the funds.

4. Harvest Finance Exploit (2020)

- Harvest Finance, a yield farming aggregator, was exploited for $24 million in October 2020. The attacker used a flash loan attack to manipulate the price of assets on the Curve Finance liquidity pool, which was then exploited by the attacker to extract funds.

5. Cream Finance Exploit (2021)

- In February 2021, Cream Finance was exploited for $37 million. The attacker used a flash loan to exploit the protocol's vulnerability with a reentrancy bug, draining funds from the protocol.

What were the top 5 biggest DeFi exploits?

Pick one and explain what vulnerabilities were exploited

Poly Network Hack (2021) $ 611 Million Lost – 10th August 2021**:**

In the **Poly Network Hack**, the attacker exploited a vulnerability in the **cross-chain contract** on the Poly Network platform. Poly Network is designed to facilitate interoperability between different blockchains, allowing users to transfer assets across chains.

The Exploited Vulnerabilities:

- The attacker identified a vulnerability in Poly Network's contract handling **asset transfers between blockchains**. Specifically, the attacker was able to **manipulate the logic of the smart contract** that verifies the legitimacy of transactions between chains.
- The attacker gained unauthorized access to the **private key or signature**, which allowed them to call the contract and facilitate the transfer of funds across different blockchains without any verification.
- Due to this flaw in the contract, the attacker was able to **steal funds from various liquidity pools**. Over $600 million worth of cryptocurrency was transferred in the exploit, although most of it was later returned.

What were the top 5 biggest DeFi exploits?

Blame game: Do you blame the “hackers” or do you blame the project (developers)?

Blame on the Developers (Project Team):

- **Lack of Auditing:** The core issue in this exploit was the vulnerability in the cross-chain contract. The Poly Network team **failed to properly audit their code** and implement sufficient security measures to safeguard against such attacks. In DeFi, security audits and thorough testing are critical before launching a platform.
- **Unsecured Contract Design:** The contract that allowed for asset transfers across blockchains had a **design flaw** that allowed for unauthorized actions. This design flaw should have been identified and fixed before the protocol went live.
- **Rushed Deployment:** Sometimes projects rush to launch to capture market attention and funds, which results in **inadequate security reviews**. The Poly Network team failed to catch this critical vulnerability in the cross-chain system before it was exploited.

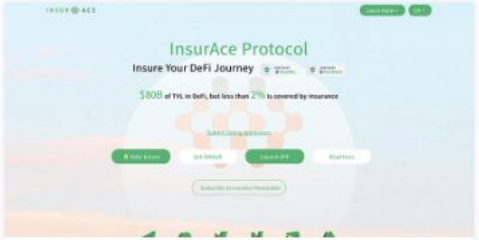
In this case, while the hackers exploited a vulnerability, the project developers share a significant amount of the blame due to inadequate security measures, lack of proper auditing, and flawed contract design. DeFi projects must prioritize security from day one to avoid such massive exploits. **Blame should be shared**, but ultimately the responsibility falls heavily on the developers to build secure, well-tested systems.

DYOR

DeFi Insurance to the rescue?

DeFi Insurance to the rescue?

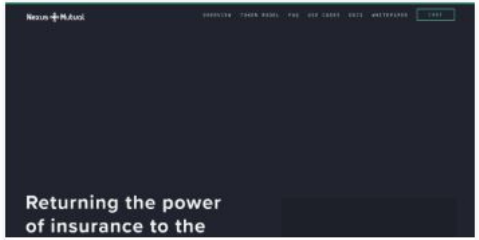
Who are the 3 biggest DeFi insurance platforms



InsurAce Protocol

InsurAce is a multi-chain protocol that provides insurance services to DeFi users, allowing them to protect their investment funds against various risks

0



Nexus Mutual

Secure risk and potential bugs in smart contract code. Be covered for events like The DAO hack or Parity multi-sig wallet issues. Interview with Hugh Karp, founder of Nexus Mutual.

0

Opium Insurance

Opium Insurance offers tradable, tokenized insurance position against smart-contract hacking or against stablecoin default.

0

DeFi Insurance to the rescue?

DeFi Insurance to the rescue?

Compare in a table what events they cover, what they don't cover, and how they operate

Platform	Events Covered	Events Not Covered	How They Operate
Nexus Mutual	- Smart contract failures	- Traditional insurance risks (e.g., natural disasters)	- Users can purchase coverage or become members to pool risks and governance.
	- Exchange hacks	- Losses from user error	- Operates with $NXM tokens for staking, coverage, and governance.
	- Protocol-specific risks	- Off-chain events (e.g., real-world events)	- Claims are voted on by the community.
InsureAce Protocol	- Smart contract vulnerabilities	- Traditional non-DeFi events	- Users can buy policies for specific risks in DeFi protocols, staking to back the insurance pool.
	- Flash loan attacks	- Off-chain risks, real-world disasters	- Utilizes $INSUR token for staking, coverage, and governance.
	- Hacks in DeFi protocols	- User-related errors or negligence	- Community governance for claims approval.
Opium Insurance	- Smart contract exploits	- Traditional insurance risks	- Focuses on covering financial derivatives, smart contracts, and events like hacks.
	- Exchange hacks	- Real-world events	- Offers event-specific insurance using $OPIUM tokens for governance and claims.
	- Market crashes	- Losses from user error	- Allows for flexible and customized insurance contracts for specific risks.

DeFi Insurance to the rescue?

What are the use cases of DeFi Insurance outside of DeFi?

- **Traditional Asset Insurance**: Cover tokenized assets like real estate or vehicles.
- **Event-Based Insurance**: Cover natural disasters, weather events, or pandemics.
- **Supply Chain & Trade Finance**: Insure goods against delays, damage, or theft during transit.
- **Healthcare & Life Insurance**: Coverage for medical costs or life insurance via smart contracts.
- **Crowdfunding & Project Insurance**: Protect backers from project failures or fraud.

DYOR

DeFi Insurance to the rescue?

What event could be covered?

- **Natural Disasters**: Earthquakes, floods, hurricanes.
- **Political Risks**: War, expropriation, civil unrest.
- **Pandemics/Epidemics**: Coverage for health crises like COVID-19.
- **Supply Chain Disruptions**: Risks from geopolitical events or trade restrictions.
- **Theft or Vandalism**: Coverage for tokenized physical assets.

DYOR

DeFi Insurance to the rescue?

Is a fully decentralized implementation even possible?

- **Data Oracles**: Reliable data sources for external events are crucial. Decentralized oracles can be used but may have accuracy risks.
- **Regulation**: Legal complexities arise from different jurisdictions and regulatory standards.
- **Claim Verification**: Decentralized governance may handle claims, but ensuring fairness is challenging.
- **User Participation**: Complexity of DeFi insurance may limit adoption, requiring better education and simpler interfaces.

Fully decentralized insurance could be challenging but possible through hybrid models combining decentralized elements with traditional systems for reliable data and claim processing.

DYOR

Smart Contract Auditing

Smart Contract Auditing

Why is smart contract auditing important?

1. **Security**: Identifies vulnerabilities to prevent attacks and asset loss.
2. **Functionality**: Ensures the contract works as intended without bugs.
3. **Protecting Funds**: Safeguards user assets by verifying contract integrity.
4. **Building Trust**: Increases credibility and confidence in the project.
5. **Compliance**: Helps meet regulatory requirements in some jurisdictions.
6. **Preventing Exploits**: Detects potential exploits or vulnerabilities before launch.
7. **Optimizing Gas**: Reduces unnecessary operations to lower transaction costs.

In short, auditing ensures smart contracts are secure, reliable, and efficient.

Smart Contract Auditing

Smart Contract Auditing

How are smart contracts audited?

Smart contracts are audited through these steps:

1. **Code Review**: Manual inspection and automated tools to identify vulnerabilities.
2. **Testing and Simulation**: Unit tests and testnet deployment to ensure correct functionality and detect edge cases.
3. **Security Analysis**: Check for vulnerabilities like reentrancy and assess risks.
4. **Gas Optimization**: Ensure the contract is efficient and minimizes gas costs.
5. **Documentation Review**: Ensure clear, understandable code and proper developer understanding.
6. **Final Report**: A detailed audit report with vulnerabilities found and recommended fixes.

In short, audits involve reviewing code, testing, security checks, and optimizations to ensure the contract is secure and efficient.

Smart Contract Auditing

Smart Contract Auditing

Which risks exist in addition to smart contract risks and how can they be mitigated?

- **Oracle Risks**: Inaccurate or manipulated data.
    - **Mitigation**: Use decentralized oracles and multiple data sources.
- **Governance Risks**: Centralized control or malicious decisions.
    - **Mitigation**: Implement decentralized governance with fair voting.
- **Protocol Risks**: Bugs or flaws in protocol logic.
    - **Mitigation**: Conduct audits and use testnets.
- **Liquidity Risks**: Insufficient liquidity causing slippage.
    - **Mitigation**: Ensure deep liquidity pools and slippage control.
- **Flash Loan Attacks**: Exploiting vulnerabilities with uncollateralized loans.
    - **Mitigation**: Implement price manipulation checks and flash loan limits.
- **Collateralization Risks**: Risk of liquidation in volatile markets.
    - **Mitigation**: Use over-collateralization and liquidation thresholds.
- **Regulatory Risks**: Legal issues due to government regulations.
    - **Mitigation**: Stay compliant with laws and build decentralized structures.
- **Human Risks**: Bugs or insider threats.
    - **Mitigation**: Regular audits, multi-signature wallets, and bug bounty programs.

Smart Contract programming languages

(OPTIONAL) What are the most popular smart contract programming languages in terms of value “secured” and the number of developers in web3 using those languages?

The top smart contract languages in Web3:

1. **Solidity**: Most popular, securing billions on Ethereum and EVM chains.
2. **Vyper**: Security-focused, used on Ethereum.
3. **Rust**: Powers Solana and Polkadot, gaining developer interest.
4. **Move**: Used by Aptos and Diem, emerging language.
5. **Michelson**: For Tezos, with a smaller community.

Solidity leads in both adoption and value secured.

DYOR

Getting to know stakeholder domains (Multiplicators)

Imagine you are a marketing manager of a DeFi service/product. What marketing strategy/measures would you use to make it go “viral”? Be as specific as possible.

- **Target Audience**: Focus on crypto communities (Reddit, Discord, Telegram).
- **Content Creation**: Create educational content, explainer videos, and blogs.
- **Influencers & Ambassadors**: Partner with crypto influencers and create ambassador programs.
- **Viral Campaigns**: Launch social media challenges and referral programs.
- **Airdrops**: Distribute tokens widely to attract new users and create excitement.
- **Bounty Programs**: Reward users for tasks like bug identification and content creation.
- **Memes**: Use humor and crypto-related memes to create shareable content.
- **Partnerships**: Collaborate with other DeFi projects for cross-promotion.
- **Community Engagement**: Host AMAs, governance participation, and loyalty programs.

DYOR

Getting to know stakeholder domains (Multiplicators)

What is the role of airdrops, bounty programs, and memes in DeFi marketing?

- **Airdrops**: Attract new users and increase awareness.
- **Bounty Programs**: Engage users, incentivize contributions, and improve security.
- **Memes**: Create viral, relatable content that drives buzz and engagement.

In short, these tactics boost **engagement**, **community trust**, and **viral growth**.

DYOR

References

- https://www.coindesk.com/learn/what-is-defi-insurance
- https://www.defi-insurance.com
- https://www.investopedia.com/terms/d/defi-insurance.asp
- https://www.coindesk.com/markets/2020/08/14/the-top-five-defi-exploits-of-2020-so-far/
- https://www.blockchain.com/blog/defi-security-exploits-2021
- https://www.cnbc.com/2021/10/07/defi-hackers-steal-2-billion-in-2021.html
- https://www.coindesk.com/learn/what-is-smart-contract-auditing
- https://www.solidified.io/auditing
- https://www.cointelegraph.com/news/smart-contract-security-audit-and-best-practices
- https://ethereum.org/en/developers/docs/smart-contracts/
- https://www.toptal.com/ethereum/smart-contract-development-solidity-vs-vyper
- https://www.coindesk.com/learn/rust-solidity-blockchain-development
- https://blog.coinspeaker.com/defi-marketing-strategies/
- https://www.cointelegraph.com/defi-marketing-strategy-how-to-grow-your-crypto-project-in-2023
- https://www.cryptobriefing.com/defi-marketing-guide/
- https://www.coindesk.com/learn/defi-airdrops-explained
- https://blog.bitquery.io/defi-airdrops-and-cryptocurrency-marketing
- https://decrypt.co/45596/why-crypto-memes-matter-and-are-here-to-stay
- https://cointelegraph.com/defi-education/defi-marketing-tactics-to-grow-your-project-in-2023