International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

PhishDetectPro: A Servlet-Based Smart Wallet Simulation and Approval Phishing Detection Framework Using Intent Validation

H. Jalandhara, Department of Computer Science and Engineering, GNITC, 22wj1a05a8@gniindia.org

I.V.N Sai Nitin, Department of Computer Science and Engineering, GNITC, 22wj1a05a9@gniindia.org

K. Shashidhar, Department of Computer Science and Engineering, GNITC, kaparaboinashashi@gmail.com

Rajashree Sutrawe, Department of Computer Science and Engineering, Assistant Professor, GNITC, raj.sutrawe@gniindia.com

---

Abstract

This project examines a deceptive blockchain scam called **approval phishing**, where users unknowingly grant malicious contracts access to their wallet funds. The attack uses a malicious Smart Contract Architecture (SCA) that tricks users into approving a contract controlled by an attacker. Once approval is granted, a malicious External Owned Account (EOA) triggers the contract to transfer tokens without the user's knowledge. The threat is worsened by weaknesses in wallet user interfaces that fail to clearly warn users about risky approval requests. The scam infrastructure typically operates through a secured web server hidden behind CDN layers, hosting fake investment platforms and a management system for scammers. This project simulates and analyzes such attacks using a Servlet–JSP web application, evaluates wallet vulnerabilities, detects scam behaviors, and suggests safer wallet mechanisms inspired by EIP-4337 account abstraction.

1.INTRODUCTION

Blockchain-based cryptocurrencies provide greater transparency and security than traditional financial systems due to their decentralized structure and cryptography. However, the absence of a central authority and the complexity of blockchain technologies introduce new risks for users. The rapid growth of cryptocurrency markets attracts both new investors and malicious actors who exploit system vulnerabilities. Smart contracts, which execute predefined logic on the blockchain, expand functionality but also increase security challenges. Many platforms support standardized tokens such as ERC-20, TRC-20, and BSC-20, which can be transferred or traded like digital assets. Unlike native cryptocurrencies, tokens allow delegated permissions through the **approve** function, enabling others to transfer assets. Attackers exploit this feature through approval phishing, using social engineering to trick users into granting malicious permissions that allow unauthorized token transfers.

2. LITERATURE REVIEW

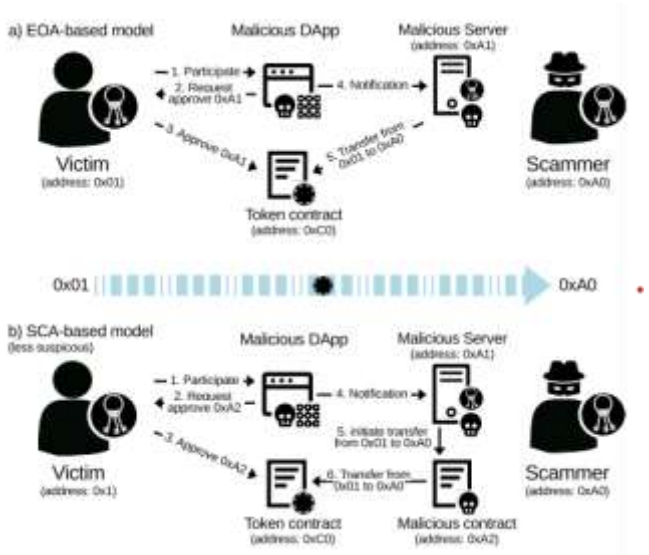
In recent research Ethereum security focuses on detecting and preventing blockchain-based fraud such as phishing attacks, Ponzi schemes, and unsafe token approvals. Studies like **"Fishing for Fraudsters" by J. Liu et al. (2024)** analyse blockchain transaction flows and approval-phishing patterns to uncover organized phishing gangs. By examining wallet behaviour, transaction clustering, and attacker-controlled contracts, the research identifies how scammers operate coordinated ecosystems involving phishing websites, fake investment portals, and malicious smart contracts. Similarly, **PonziGuard by R. Liang et al. (2024)** introduces a detection framework based on Contract Runtime Behaviour Graphs (CRBG), which monitors smart contract execution paths to identify Ponzi schemes through behavioural signatures such as deposit-redistribution loops and abnormal fund flows.

Other studies focus on infrastructure and security improvements. **Cornelius et al. (2024)** explore the Waku Network as a decentralized messaging infrastructure that enables secure and scalable communication for decentralized applications. Meanwhile, **ERC-7674** proposes temporary token approvals to reduce risks associated with permanent ERC-20 allowances. Research by **Wu et al. (2024)** applies network-embedding techniques to detect phishing scammers by analysing wallet interaction graphs. Finally, **Wang et al. (2024)** quantify the risks of unlimited ERC-20 approvals, demonstrating how forgotten allowances expose users to large financial losses. Together, these works highlight the need for improved blockchain analytics, safer token approval mechanisms, and better wallet security.

3. System Architecture

PhishDetectPro uses a layered servlet-based architecture to detect approval phishing. JSP pages form the presentation layer, while Java Servlets handle requests and sessions. A smart wallet simulation and SIVRD engine analyze approvals and detect risks. Attack simulations demonstrate token drainage, while a blockchain ledger module records transactions and a database stores system data securely.This project having the following 4 modules:

Module	Description
1. Smart Wallet Simulation	It manages wallet creation, token balances, approvals, transfers, and state updates without deploying real smart contracts.
2 Approval Phishing Detection	This module detects approval phishing attacks by correlating approval parameters with predefined risk patterns.
3. Token Drain Simulation	This module simulates malicious token draining using previously granted approvals.
4. Blockchain Ledger & Block Creation	This module emulates a blockchain ledger by grouping transactions into blocks with hashes, timestamps, and previous block references.



4. PROPOSED METHODOLOGY

The PhishDetectPro system is designed with multiple modules that simulate blockchain wallet operations and detect approval phishing attacks in a controlled environment. The **Smart Wallet Simulation module** reproduces the behavior of a blockchain wallet without deploying real smart contracts. It manages wallet creation, token balances, approvals, and token transfers using Java Servlets and database storage. This module follows ERC-20 style approval mechanisms, allowing users to grant spending permissions to other addresses.

The **Approval Phishing Detection module** identifies suspicious approval requests that may lead to unauthorized token access. It analyzes parameters such as spender identity, approval amount, and transaction frequency. If risky patterns or unknown addresses are detected, the system generates alerts and warns the user before approval is completed.

The **Token Drain Simulation module** demonstrates how attackers exploit previously granted approvals. When a risky approval exists, the system simulates malicious transfers that drain tokens from the user wallet without further confirmation. This helps users understand the consequences of approval phishing attacks.

The **Blockchain Ledger and Block Creation module** emulates blockchain behavior by grouping transactions into blocks containing timestamps, hashes, and references to previous blocks. This ensures a tamper-evident transaction history and demonstrates how blockchain records approvals and transfers.

The **JSP Dashboard module** provides a user interface to display transaction details, phishing alerts, approved transfers, and wallet balances. Finally, the **MySQL Database module** stores user credentials, logs, and metadata, supporting system operations while maintaining organized and secure data management.

In addition to these modules, the system includes a User Interface and Navigation module that provides secure access to all functionalities. Users must register with details such as username, password, email, city, and country before accessing the platform. After authentication, users can navigate to different modules through the main dashboard. This interface ensures that only authorized users can perform wallet operations and view transaction information.

The system also defines clear input and output operations for each module. User actions such as login, wallet transactions, and approval requests act as inputs. Based on these inputs, the system generates outputs like wallet creation, token balance updates, phishing alerts, and blockchain transaction records. The dashboard displays real-time transaction data, approved permissions, and detected phishing activities. This structured flow helps users understand how approvals are processed, how attacks occur, and how blockchain records maintain transparency.

Overall, the architecture integrates simulation, detection, visualization, and storage components to demonstrate approval phishing risks and improve user awareness and wallet security.

5. RESULTS AND DISCUSSION

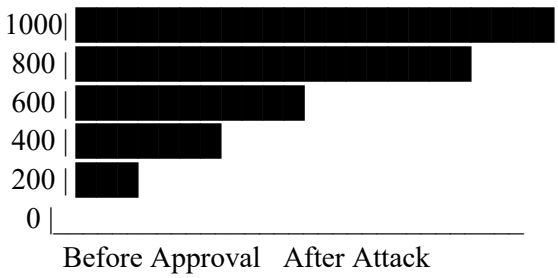
The implementation of **PhishDetectPro** demonstrates how approval phishing attacks occur in blockchain wallets and how they can be detected through a simulation framework. The system integrates smart wallet simulation, phishing detection, token drain demonstration, and blockchain ledger visualization using a servlet–JSP architecture. The **Smart Wallet Simulation module** successfully replicates ERC-20 approval mechanisms, enabling users to grant permissions and perform token transactions in a controlled environment.

The **Approval Phishing Detection module** analyzes spender identity, approval limits, and transaction behavior to identify suspicious approval requests. When risky patterns are detected, the system generates alerts and warns users before completing the approval. The **Token Drain Simulation module** illustrates the impact of malicious approvals by simulating unauthorized token transfers without further user consent.

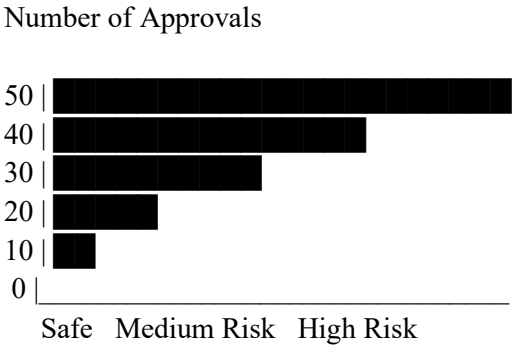
Additionally, the **Blockchain Ledger module** records transactions into blocks with timestamps and hashes, ensuring transparency. The **JSP Dashboard** displays transactions, alerts, and balances, helping users understand risks and encouraging safer wallet usage.

Phishing Detection Result Graph

Token Drain Simulation Result



Wallet Approval Risk Distribution



6. Quantitative Security Analysis

The PhishDetectPro system was evaluated using metrics such as approval risk detection accuracy, transaction processing time, and token drain simulation results. The Approval Phishing Detection module successfully identified suspicious approvals by analyzing spender identity and approval limits. The system generated alerts with minimal delay, ensuring real-time monitoring. Token drain simulations demonstrated how quickly funds can be transferred after unsafe approvals. Additionally, blockchain ledger testing confirmed secure, tamper-evident transaction recording through hashed blocks.

7.CONCLUSION

The PhishDetectPro project successfully demonstrates a servlet-based framework for detecting approval phishing attacks in smart wallet environments. The system simulates real blockchain wallet behavior, allowing users to understand how token approvals can be misused. By integrating Smart Intent Validation, the project effectively identifies risky approval requests before execution. The approval phishing detection module warns users about token drain possibilities and enforces informed decision-making. The smart wallet simulation provides transparency into token balances and approval flows. Blockchain ledger and block creation modules

ensure traceability and immutability of simulated transactions. Token drain simulation highlights real-world fraud scenarios in a controlled environment. The system enhances user awareness of common Web3 scams such as approval phishing and investment fraud. Modular architecture improves maintainability and scalability. Secure servlet-based communication ensures controlled transaction handling. The project bridges the gap between theory and practical security implementation. Overall, PhishDetectPro delivers an effective, educational, and security-focused solution for mitigating approval phishing risks in decentralized systems.

8. FUTURE SCOPE

In the future, the system can be extended to integrate **real blockchain networks** for live transaction analysis and phishing detection. Advanced **AI and machine learning models** can be added to identify complex scam patterns automatically. The platform can also include **real-time wallet plugins**, improved approval-warning mechanisms, and cross-chain monitoring tools to enhance security and protect users from emerging blockchain fraud techniques.

REFERENCES

[1] Badawi and G.-V. Jourdan, ‘‘Cryptocurrencies emerging threats and defensive mechanisms: A systematic literature review,’’ IEEE Access, vol. 8, pp. 200021–200037, 2020. [Online]. Available:https://ieeexplore.ieee.org/document/9243940/

[2] J. Wu, Q. Yuan, D. Lin, W. You, W. Chen, C. Chen, and Z. Zheng,‘‘Who are the phishers? Phishing scam detection on Ethereum via network embedding,’’ IEEE Trans. Syst., Man, Cybern., Syst., vol. 52, no. 2,pp. 1156–1166, Feb. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9184813/

[3] M. Bartoletti, S. Lande, A. Loddo, L. Pompianu, and S. Serusi,‘‘Cryptocurrency scams: Analysis and perspectives,’’ IEEE Access, vol. 9,pp. 148353–148373, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9591634/

[4] D. Wang, H. Feng, S. Wu, Y. Zhou, L. Wu, and X. Yuan, ‘‘Penny wise and poundfoolish: Quantifying the risk of unlimited approval of ERC20 tokens on Ethereum,’’ in Proc. 25th Int. Symp. Res. Attacks, Intrusions Defenses. Limassol, Cyprus: ACM, Oct. 2022, pp. 99–114. [Online]. Available:https://dl.acm.org/doi/10.1145/3545948.3545963

[5] J. Liu, J. Chen, J. Wu, Z. Wu, J. Fang, and Z. Zheng, ‘‘Fishing for fraudsters: Uncovering Ethereum phishing gangs with blockchain data,’’ IEEE Trans. Inf. Forensics Security, vol. 19, pp. 3038–3050, 2024.[Online]. Available: https://ieeexplore.ieee.org/document/10415200

[6] Y. Zhang, W. Yu, Z. Li, S. Raza, and H. Cao, ‘‘Detecting Ethereum Ponzi schemes based on improved LightGBM algorithm,’’ IEEE Trans. Computat. Social Syst., vol. 9, no. 2, pp. 624–637, Apr. 2022. [Online].Available: https://ieeexplore.ieee.org/document/9479748/

[7] R. Liang, J. Chen, K. He, Y. Wu, G. Deng, R. Du, and C. Wu,‘‘PonziGuard: Detecting Ponzi schemes on Ethereum with contract runtime behavior graph (CRBG),’’ in Proc. IEEE/ACM 46th Int. Conf.Softw. Eng., Lisbon, Portugal, Feb. 2024, pp. 1–12. [Online]. Available: https://dl.acm.org/doi/10.1145/3597503.3623318

[8] M.Bartoletti, S. Carta, T. Cimoli, and R. Saia, ‘‘Dissecting Ponzi schemes on Ethereum: Identification, analysis, and impact,’’ Future Gener. Comput.Syst., vol. 102, pp. 259–277, Jan. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X18301407

[9] A. Holub and J. O’Connor, ‘‘COINHOARDER: Tracking a Ukrainian Bitcoin phishing ring DNS style,’’ in Proc. APWG Symp. Electron. Crime Res. (eCrime), May 2018, pp. 1–5. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/8376207/

[10] E. Badawi, G.-V. Jourdan, G. Bochmann, and I.-V. Onut,‘‘An automatic detection and analysis of the Bitcoin generator scam,’’in Proc. IEEE Eur. Symp. Secur. Privacy Workshops (EuroS&PW),Sep. 2020, pp. 407–416. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9229769/

[11] etherscan.io. Token Tracker Etherscan. Accessed: Nov. 20, 2024. [Online].Available: https://etherscan.io/tokens

[12] C. Shier, M. I. Mehar, A. Giambattista, E. Gong, G. Fletcher, R. Sanayhie, M. Laskowski, and H. M. Kim, ‘‘Understanding a revolutionary and flawed grand experiment in blockchain: The DAO attack,’’SSRN Electron. J., vol. 21, no. 1, pp. 19–32, 2019. [Online]. Available:https://www.igi-global.com/article/understanding-a-revolutionary-andflawed-grand-experiment-in-blockchain/216950

[13] L. Swartz, ‘‘Theorizing the 2017 blockchain ICO bubble as a network scam,’’ New Media Soc., vol. 24, no. 7, pp. 1695–1713,Jul. 2022. [Online]. Available: https://journals.sagepub.com/doi/10.1177/14614448221099224

[14] F. Torres, M. Steichen, and R. State, ‘‘The art of the scam:Demystifying honeypots in Ethereum smart contracts,’’ in Proc. 28th USENIX Secur. Symp., Jan. 2019, pp. 1591–1607. [Online]. Available:https://www.usenix.org/conference/usenixsecurity19/presentation/ferreira

[15] M. La Morgia, A. Mei, F. Sassi, and J. Stefa, ‘‘The doge of wall street:Analysis and detection of pump and dumpcryptocurrency manipulations,’’ACMTrans.InternetTechnol.,vol. 23,no.1,pp. 1–28,Feb.2023.[Online].Available: https://dl.acm.org/doi/10.1145/3561300

[16]A.Czajkowski.(2021).TokenApprovalSCAM.JakSięPrzedNimUstrzec?| CrypS.[Online].Available:https://cryps.pl/artykul/token-approval-scamjak-sie-przed-nim-ustrzec/

[17] M. Young. (2021). Scammers Stealing Chainlink By Abusing Token Approval Transactions. [Online]. Available: https://beincrypto.com/scammers-stealing-chainlink-abusing-token-transactions/

[18] Chainalysis. (2024). The Chainalysis 2024 Crypto Crime Report. [Online].Available: https://go.chainalysis.com/crypto-crime-2024.html