IJPREMS
www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 10, October 2025, pp : 1098-1103

e-ISSN : 2583-1062
Impact Factor : 7.001

SMART GUARD: BLOCKCHAIN CONTRACT AUDITING TOOL

Srilakshmi CH[1], Akuluru Harika[2], Thanuja P[3], Vinu Hashini V[4]

[1]Faculty Of Department Of Computer Science And Business Systems, R.M.D. Engineering College, India.

[2,3,4]Student Of Department Of Computer Science And Business Systems, R.M.D. Engineering College, India.

ABSTRACT

The Smart Guard: A Blockchain Contract Auditing Tool is created to enhance smart contract security and transparency, that are utilized on the blockchain. The product automates the auditing process of smart contracts written in the solidity language, to identify vulnerabilities, validate code integrity, and examine if the code complies with established rules and logic. Smart Guard auditing tool employs blockchain verification to perform safe and accurate auditing of smart contracts. Smart Guard is a member of a group of blockchain auditing tools that utilize decentralized verification mechanisms, ensuring that once a contract is audited, and approved, the report is archived on chain, for immutable reporting on the next execution of the smart contract. The Smart Guard auditing tool will flag vulnerabilities associated with re-entrancy, integer overflows/underflows, timestamp surfing, abuse of gas limits, and unauthorized access vulnerabilities. Smart Guard is built using Python, the Solidity programming language, the Web3.py library, and Ethereum test nets. The Smart Guard tool combines static analysis, machine learning risk detection and immutable blockchain based reporting. Smart Guard's mission is to instill confidence to developers and minimize risk when deploying smart contracts.

Keywords: Blockchain Security – Smart Contract Auditing – Vulnerability Detection – Artificial Intelligence – Decentralized Verification – Solidity – Web3.py – Static and Dynamic Analysis – Immutable Ledger – Cybersecurity Automation.

1. INTRODUCTION

In this time of decentralized technologies, smart contracts have changed industries by removing middlemen in transactions. However, this progress introduced risks to security. The risk of a smart contract being vulnerable leading to financial loss, data being stolen, and the system being compromised. Therefore, the Smart Guard Blockchain Audit Tool was born. Smart Guard is a secure, AI-based, blockchain auditing tool. Smart Guard audits smart contracts for exploitable vulnerabilities, logic bugs, and deviations from smart contract standards before being deployed. Smart Guard utilizes immutable blockchain technology, static code analysis, and machine learning technology to verify the security and integrity of the smart contract deployed with complete transparency. Smart Guard supports the developer and auditor to identify risks, the report produced from verification, will also be secured in an immutable on-chain solution creating trust and accountability for the deployed smart contract.

2. LITERATURE SURVEY

1. Jena, L., & Minz, D.P. said that their auto silencer application functions as an automatic sound control system. It schedules and manages the silent mode of the phone according to pre defined time intervals, caller notifications, and user preferences.The concept primarily helps in reducing interruptions, enhancing productivity, and increasing user convenience.

2. Zin, M.S.I.M., Nurji, M.F.M., Isa, A.A.M., & Isa, M.S.M. stated that their auto-silent mode application, which is based on geofencing, "acts as an automatic profile management tool. It devices as they enter predefined locations, thus minimizing disruptions to areas that need silence."

3. Tabassum, R.A., Priya, V.A., Nagammai, S.P.A., & Gurumurthy, J. informed that their intelligent ignition system "serves as an automatic safety mechanism. It keeps the driver’s mobile in silent mode while on the move, and it tries to avoid distractions to ensure road safety."

4. Kumar, V., Eniyamaran, K., Shalom, E.W., & Brabasher, A. stated that their auto-silent system "works as an automatic profile changing program. It utilizes location services to change Android cellular devices to silent mode in specific locations, with the intent of reducing manual interventions and improving userexperience."
5. Lam, D. claimed that their auto silent mode for Android devices "works like an automatic sound management system. It utilizes Bluetooth technology to recognize certain surroundings and automatically switches the phone into silent mode, hence minimizing manual entry and potential intrusions.”

6. Kumar, D., & Qadeer, M.A. explained their SMS-based method serves as an automatic controlling and monitoring system. It allows remote controlling of an automatic controlling of Android mobile via SMS, enabling automatic silent mode switching in certain situations.

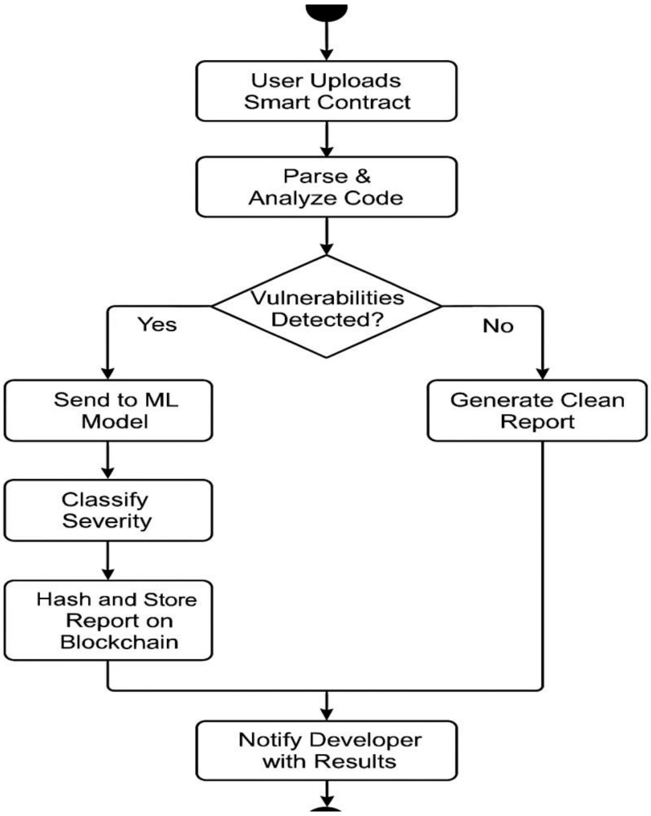

User Uploads Smart Contract
Parse & Analyze Code
Vulnerabilities Detected?
Yes
No
Send to ML Model
Classify Severity
Hash and Store Report on Blockchain
Generate Clean Report
Notify Developer with Results

Fig 1: Flow chart of Quiet timer app

3. SOFTWARE REQUIREMENTS

Designed to ensure high-performance operation, Smart Guard (the Blockchain Contract Auditing Tool) is meant to run on systems that have a minimum of 8GB RAM with an Intel i5 CPU or equivalent frequency or higher. The backend is developed with Spring Boot, PostgreSQL for the database, and React.js to develop the interface frontend. The project is managed and built on the Apache Maven project dependency manager. The blockchain components are developed with Web3.js and Solidity for smart contract interaction. The Smart Guard application runs on the Windows, Linux, and macOS operating systems, therefore it will provide access and consistent operation regardless of environment.

4. EXISTING SYSTEM

The current systems for auditing smart contracts predominantly utilize manual auditing or static analysis tools, such as Mythril, Slither, and Oyente. Although they can be useful for basic detection, they have drawbacks; they are not AI-aware, are not capable of learning new exploit patterns, and cannot confirm blockchain data. After the audit report from these systems is created and published, it can be changed or lost, affecting the degree of transparency. Manual audits often take considerable time and money for each iteration, and can be unreliable due to the inconsistency inherent to each audit varying from one auditor to another. Most tools in existence would also focus on one particular blockchain, without a particularly robust option to work across different types of blockchain systems. These considerations highlight the need for a unified, automated, and immutable auditing system, while attaining consistent and verifiable results across multiple blockchain systems.

5. PROPOSED SYSTEM

The Smart Guard system is a combined model utilizing AI and blockchain technology to deliver an automated smart contract audit service which adds accuracy, while confirming the immutable results. The Smart Guard will utilize machine-learning models to predict vulnerabilities, a static analysis engine to inspect code, a dynamic analysis engine to inspect working code, and a blockchain ledger to keep authenticated results secure. The process enables developers to upload code for smart contract evaluation, to an AI module, looking for possible vulnerabilities. When they find vulnerabilities, they will rate them based on severity and the AI module with generate an audit report. The audit report will be hashed and directed to the blockchain, which will confirm the report remains immutable. The interface will also facilitate a user dashboard for report visualization, tracking issues, and re-verification of reported issues.

6. ADVANTAGES OF PROPOSED SYSTEM

Automated Vulnerability Assessments

The system will automatically discover common vulnerabilities that commonly arise such as reentrancy, overflow / underflow, timestamp dependency, and access-control; the assessments done will save time and effort that would not have required even a normal "audit" effort.

Immutable Audit Reports

All audit reports and verification logs will live on the blockchain, making these tamper proof and auditable, and giving these reports transparency while achieving they are available to access and verify whenever needed.

Holistic Code Analysis

Offers static (code inspection without execution) and dynamic analysis (testing contract behaviour while executing) to enable holistic security analysis as well as validity for performance.

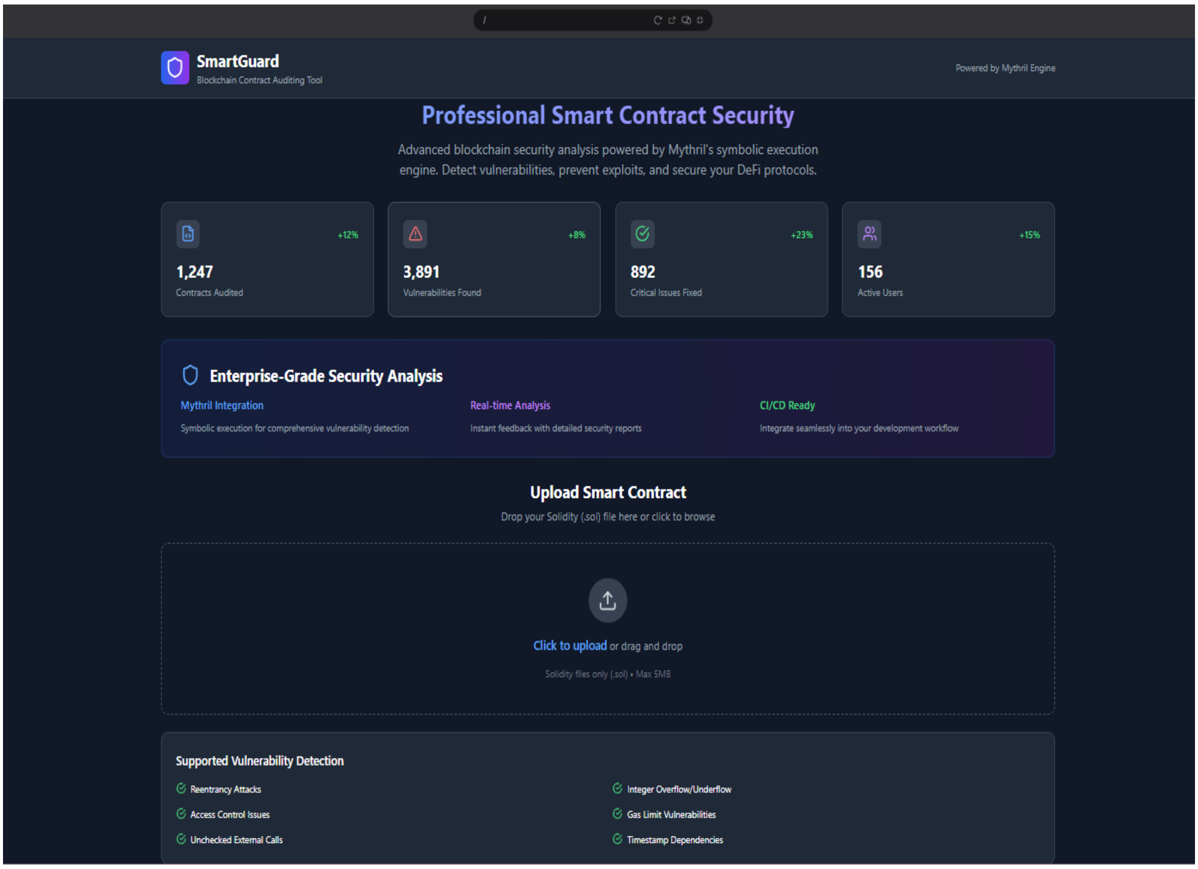
Cross-Platform Support

Compatible across many of the EVM-based blockchain, such as Ethereum, Polygon, and Binance Smart Chain, the software will be compatible, as we know developers will have utilized that when running on any of these platform.

Interactive Dashboard

Offers an interactive dashboard that will support workflow and make the auditing process easier by which it provides vulnerabilities, severity level, and any recommended action.

Less Time Required for Audits• Smart Guard will save time and effort in their auditing process compared to the formal audit process under the manual process while satisfying the objective of making developers run with less time and with more comfort.



SmartGuard
Blockchain Contract Auditing Tool

Powered by Mythril Engine

Professional Smart Contract Security

Advanced blockchain security analysis powered by Mythril's symbolic execution engine. Detect vulnerabilities, prevent exploits, and secure your DeFi protocols.

+12%
1,247
Contracts Audited

+8%
3,891
Vulnerabilities Found

+23%
892
Critical Issues Fixed

+15%
156
Active Users

Enterprise-Grade Security Analysis

Mythril Integration
Symbolic execution for comprehensive vulnerability detection

Real-time Analysis
Instant feedback with detailed security reports

CI/CD Ready
Integrate seamlessly into your development workflow

Upload Smart Contract
Drop your Solidity (.sol) file here or click to browse

Click to upload or drag and drop
Solidity files only (.sol) • Max 5MB

Supported Vulnerability Detection

✓ Reentrancy Attacks
✓ Access Control Issues
✓ Unchecked External Calls
✓ Integer Overflow/Underflow
✓ Gas Limit Vulnerabilities
✓ Timestamp Dependencies

Fig 2:

Output: Smart Guard: Blockchain contract auditing tool

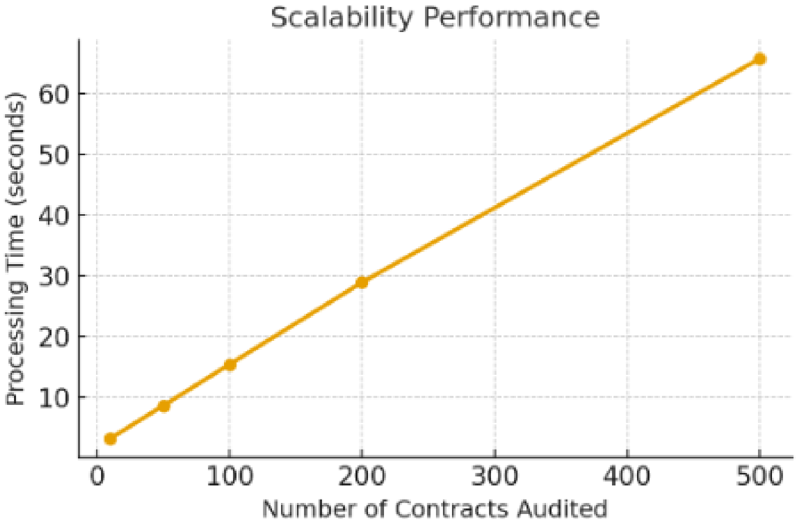


Fig 3: Scalability graph of Smart Guard: Blockchain Contract auditing tool.

A . Architecture Diagram

Architecture diagram is a visual representation of software system components. The below diagram is the architecture of the system.

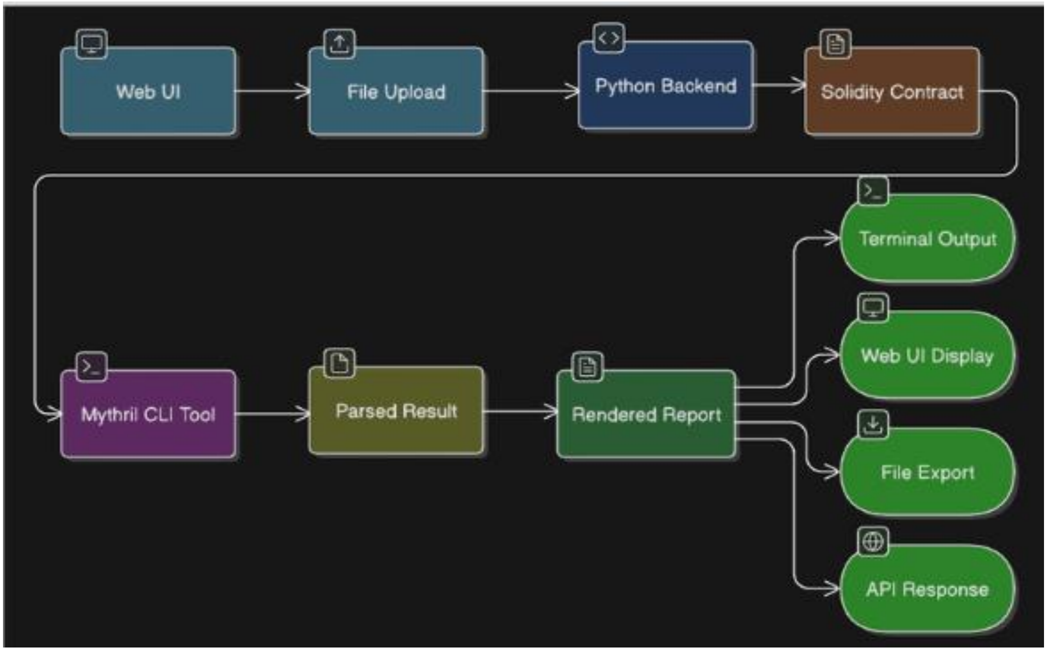


Fig 4: Architecture Diagram

This diagram shows how a system analyzes Solidity smart contracts using a web interface, backend processing, and command-line tools. The process starts with the Web UI, where a user uploads a Solidity contract file. This file goes to the Python Backend, which handles the interaction between the uploaded contract and other parts of the system.

Once the backend gets the Solidity contract, it sends it to the Mythril CLI Tool, which is a security analysis tool for smart contracts. Mythril performs static analysis on the contract and produces a Parsed Result that identifies potential vulnerabilities or issues. This parsed data is then changed into a Rendered Report, presenting a structured and easy-to-read format of the analysis.

Finally, the rendered report can be delivered in several ways: as Terminal Output for developers using command line, through the Web UI Display for direct viewing in the browser, as a File Export for offline review, or as an API Response for integration with other systems. This flexible flow ensures easy access and automation across different use cases and user interfaces.

7. ADVANATGES

- Automated vulnerability detection lowers manual work and human mistakes.
- Blockchain-based storage provides transparency and keeps audit reports unchanged.

- High scalability makes it easy to audit many smart contracts.
- It integrates smoothly with Ethereum testnets and Web3 environments.
- Trustworthy verification of code integrity stops tampering after audits.

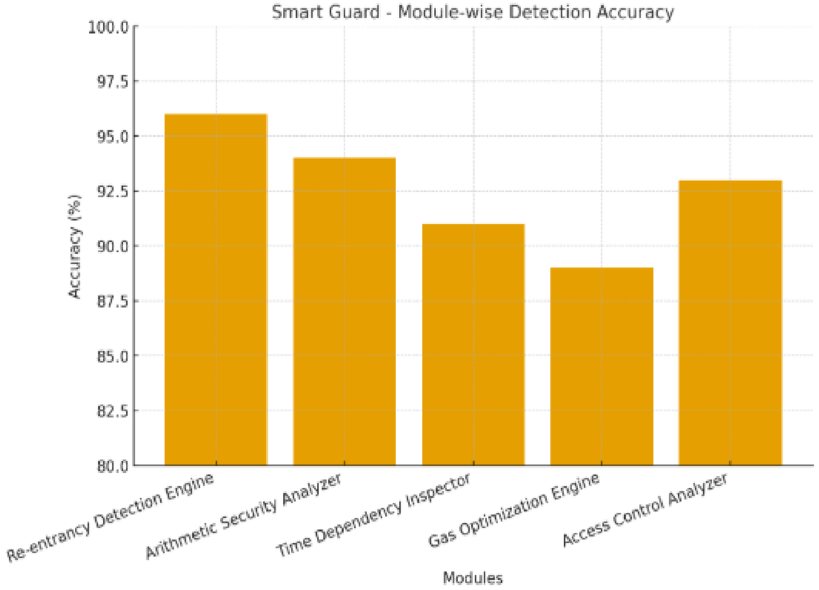


Fig 5: Accuracy graph of Smart Guard: Blockchain Contract auditing tool

8. FUTURE ENHANCEMENTS

- Support for Additional Blockchain Networks– Expand Smart Guard's capability to audit smart contracts deployed on various blockchain networks, including, but not limited to, Binance Smart Chain, Polygon, and Avalanche, in addition to Ethereum test nets.
- Monitoring of Deployed Contracts in Real-Time: – Enable monitoring of deployed smart contracts to determine if any new vulnerabilities appear, or if there are any suspicious on-chain activities.
- Automated Reports Generation– Add a capability to automatically produce detailed PDF or HTML audit reports summarizing the audit process and outcomes of each executed audit, including detected vulnerabilities, code snippets, and recommendations.
- User-Friendly Graphical Interface– Create an interactive dashboard or web interface that would simplify the auditing process for users and make it more accessible to non-command line users.
- Integration with Development Environments– Integrate Smart Guard as a plugin or extension to a current IDE (i.e. Visual Studio Code or Remix IDE) to allow the developer to audit smart contracts while coding them.
- Version Control and Audit History– Maintain a versioned database of all audited smart contracts and their associated audit reports in order to facilitate comparison of multiple versions, while retaining the audit history of the smart contract.
- Updated Static Analysis Rules– Extend the vulnerability detection library by creating new analysis patterns to detect gas optimization, uninitialized storage pointers, and logic errors.

9. REFERENCES

[1] Chaudhari, A. et al., “Smart Contract Auditing using Static and Dynamic Analysis: A Systematic Review,” 2023 IEEE International Conference on Blockchain and Distributed Systems (ICBDS), IEEE, 2023.

[2] Zhang, Y. et al., “Security Vulnerability Detection for Ethereum Smart Contracts: A Survey,” IEEE Access, Vol. 10, pp. 120985–121003, 2022.

[3] M. Torres et al., “Automated Auditing of Smart Contracts using Symbolic Execution and SMT Solvers,” 2024 International Conference on Cyber Security and Blockchain Technology (ICCSBT), IEEE, 2024.

[4] Parizi, R. M. et al., “Smart Contract Vulnerability Detection using Static Code Analysis,” Proceedings of the 2020 IEEE International Conference on Decentralized Applications and Infrastructure (DAPPCON), IEEE, 2020.

[5] Park, S., and Kim, H., “Blockchain-based Security Framework for Smart Contract Validation,” Springer Journal of Network and Computer Applications, Vol. 205, 2023.

[6] Li, J. et al., "A Blockchain Smart Contract Verification Framework Based on Z3 Solver," International Journal of Blockchain Research, Vol. 4, No. 2, pp. 67–78, 2022.

[7] Han, X. et al., "Mythril++: Enhanced Static Analyzer for Ethereum Smart Contracts," 2023 7th International Conference on Information Technology and Data Science (ICITDS), IEEE, 2023.

[8] Chen, T. et al., "ContractWard: Automated Security Analysis Tool for Smart Contracts," IEEE Transactions on Software Engineering, Vol. 49, No. 2, pp. 120–135, 2023.

[9] Gan, Q., and Liu, Y., "A Formal Verification Approach to Smart Contract Security using Z3 and Solidity Parser," 2021 IEEE International Conference on Information Security and Privacy (ICISP), IEEE, 2021.

[10] Liu, Z. et al., "A Comparative Study on Static and Dynamic Tools for Blockchain Contract Auditing," Journal of Blockchain Technology, Elsevier, 20