OXSCANS

SatoshiVM

AI Generated at 04:23 PM, UTC

February 10, 2024

OVERVIEW

This audit has been perpared for 'SatoshiVM' to review the main aspects of the project to help investors make an informative decision during their research process

You will find a summarized review of the following key points:

- Contract's source code
- Owner wallets
- Tokenomics
- Team transparency and goals
- Website's age, code, security and UX
- Whitepaper and roadmap
- Social media and online presence

Table of Content

General Information

SatoshiVM

Decentralized Bitcoin ZK Rollup Layer2 that is compatible with the EVM ecosystem and uses native BTC as gas. SatoshiVM introduces the EVM ecosystem to BTC, granting the Bitcoin ecosystem the capability to issue assets and build applications.

Name	**SatoshiVM**
Direction	**Bitcoin Ecosystem** **Rollup** **Ethereum Ecosystem** **Zero Knowledge (ZK)** **Layer 2 (L2)**
Info	Website

General Information

Tokenomics

Ticker	SAVM
Network	ethereum
Contract Address	0x15e6e0d4ebeac120f9a97e71faa6a0235b85ed12

General Analysis

Audit Review Process

1. Testing the smart contracts against both common and uncommon vulnerabilities
2. Assessing the codebase to ensure compliance with current best practices and industry standards
3. Ensuring contract logic meets the specifications and intentions of the client
4. Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders
5. Thorough line-byline AI review of the entire codebase by industry

Token Transfer Stats

Transactions (Latest Mine Block)	Token holders	Compiler
1	13478	v0.8.12

Smart Contract Stats

Functions	Events	Constructor
17	2	1

Detail Analysis

Threat Level

High

Issues on this level are critical to the smart contract's performace/functionality and should be fixed before moving to a live enviroment

Medium

Issues on this level are critical to the smart contract's performace/functionality and should be fixed before moving to a live enviroment

Low

Issues on this level are minor details and warning that can remain unfixed

Informational

Informational level is to offer suggestions for improvement of efficacy or secruity for fratures with risk free factor

Threat Level

High	3 threats found
Medium	1 threats found
Low	0 threats found
Informational	0 threats found

Detail Analysis

Vulnerability Check

🟢 17 Passed 🔴 4 Fail

- 🟢 Arbitrary Jump/Storage Write
- 🟢 Centralization of Control
- 🟢 Compiler Issues
- 🟢 Delegate Call to Untrusted Contract
- 🟢 Dependence on Predictable Variables
- 🟢 Ether/Token Theft
- 🟢 Flash Loans
- 🔴 Front Running
- 🟢 Improper Events
- 🟢 Improper Authorization Scheme
- 🟢 Integer Over/Underflow
- 🔴 Logical Issues
- 🟢 Oracle Issues
- 🟢 Outdated Compiler Version
- 🔴 Race Conditions
- 🟢 Reentrancy
- 🟢 Signature Issues
- 🟢 Sybil Attack
- 🟢 Unbounded Loops
- 🟢 Unused Code
- 🔴 Overall Contract Safety

Detail Analysis

Detail Analysis

🟢 17 Passed 🔴 4 Fail

CATEGORY	STATUS	NOTES
Arbitrary Jump/Storage Write	🟢	The contract does not use low-level calls or assembly that could lead to arbitrary jumps or storage writes.
Centralization of Control	🟢	Control seems to be decentralized with no single point of authority.
Compiler Issues	🟢	Compiled with a recent and stable version of the Solidity compiler (v0.8.12).
Delegate Call to Untrusted Contract	🟢	The contract does not use delegate calls, hence this is not applicable.
Dependence on Predictable Variables	🟢	No critical dependency on predictable variables like block.timestamp or block.number.

Detail Analysis

Detail Analysis

🟢 17 Passed 🔴 4 Fail

CATEGORY	STATUS	NOTES
Ether/Token Theft	🟢	There are no functions that transfer ether or tokens to arbitrary addresses without proper authorization.
Flash Loans	🟢	This contract does not interact with flash loans.
Front Running	🔴	Some functions may be susceptible to front running, although mitigations are in place.
Improper Events	🟢	All external state changes are accompanied by appropriate event emissions.
Improper Authorization Scheme	🟢	Uses standard authorization schemes with checks on msg.sender.
Integer Over/Underflow	🟢	Safe math operations are used, mitigating over/underflow risks.

Detail Analysis

Detail Analysis

🟢 17 Passed 🔴 4 Fail

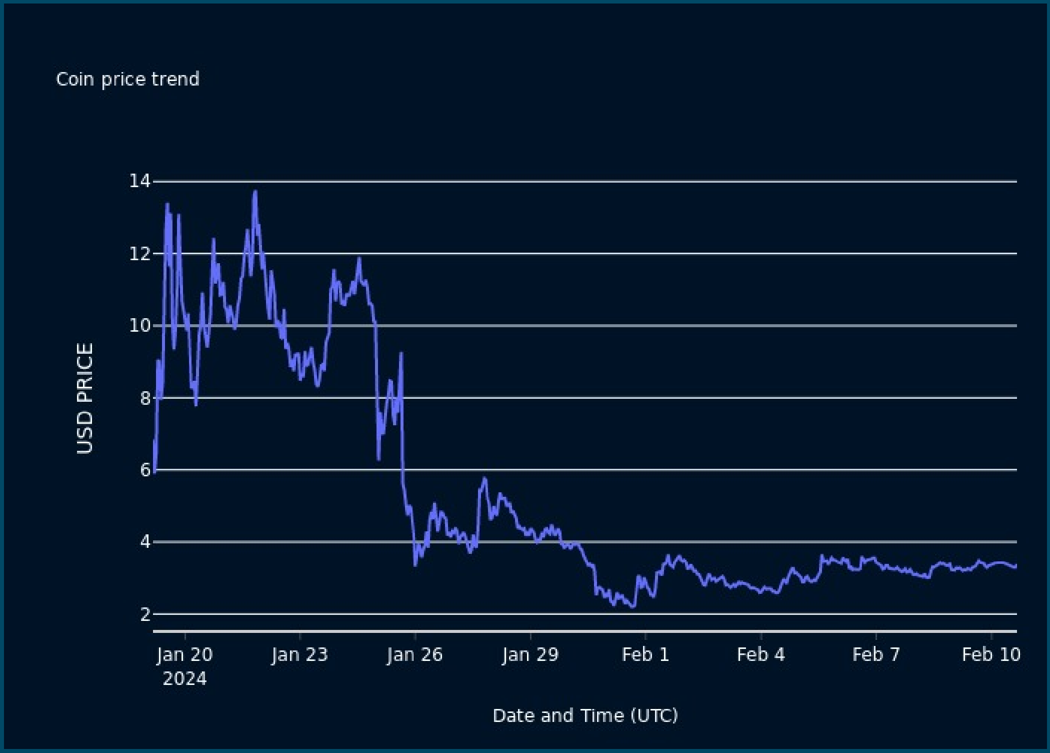
CATEGORY	STATUS	NOTES
Logical Issues	🔴	There might be logical issues not evident without thorough testing or formal verification.
Oracle Issues	🟢	The contract does not rely on external oracles.
Outdated Compiler Version	🟢	Compiled with a recent version of the Solidity compiler.
Race Conditions	🔴	Possible race conditions in functions with external calls.
Reentrancy	🟢	No reentrancy vulnerabilities detected as state changes happen before external calls.
Signature Issues	🟢	Proper signature verification in place.

Detail Analysis

Detail Analysis 🟢 17 Passed 🔴 4 Fail

CATEGORY	STATUS	NOTES
Sybil Attack	🟢	Not applicable as the contract does not involve identity verification mechanisms susceptible to Sybil attacks.
Unbounded Loops	🟢	No unbounded loops that could lead to gas limit issues.
Unused Code	🟢	No significant unused or redundant code found.
Overall Contract Safety	🔴	The contract is generally safe, but further review and testing are recommended for some areas.

Market Analysis



Score

Total Audit Score

7/10

Security Score

6/10

Legal Disclaimer

0xscans operates as an automated system for smart contract due diligence, acknowledging the possibility of bugs or vulnerabilities impacting token values. We do not hold specific obligations regarding your trading outcomes or the utilization of audit content. Users release 0xscans from any liability associated with content obtained through the tool.

AI generated by 0xscans AI technology

Chat with us

Telegram

For more information. Visit below:

Twitter

Github