SOLIDProof

Bring trust into your projects

Blockchain Security | Smart Contract Audits | KYC Development | Marketing

MADE IN GERMANY

JetBolt

UNIT TESTS

SECURITY ASSESSMENT

31 July, 2024

for

CONTENTS

Introduction

SolidProof.io is a brand of the officially registered company FutureVisions Deutschland, based in Germany. We're mainly focused on Block-chain Security such as Smart Contract Audits and KYC verification for project teams. Solidproof.io assess potential security issues in the smart contracts implementations, review for potential inconsistencies between the code base and the whitepaper/documentation, and provide suggestions for improvement.

Disclaimer

SolidProof.io reports are not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. These reports are not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team. SolidProof.io do not cover testing or auditing the integration with external contract or services (such as Unicrypt, Uniswap, Pancake-Swap etc'…)

SolidProof.io Audits do not provide any warranty or guarantee regarding the absolute bug- free nature of the technology analyzed, nor do they provide any indication of the technology proprietors. SolidProof Audits should not be used in any way to make decisions around investment or involvement with any particular project. These reports in no way provide investment advice, nor should be leveraged as investment advice of any sort.

SolidProof.io Reports represent an extensive auditing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology. Blockchain technology and cryptographic assets present a high level of ongoing risk. SolidProof's position is that each company and individual are responsible for their own due diligence and continuous security. SolidProof in no way claims any guarantee of security or functionality of the technology we agree to analyze.

Project Overview

Summary

Project Name	JetBolt Wallet
Website	https://jetbolt.io/
About the Project	JetBolt aims to power the next generation of crypto apps. Join now and be part of the future of crypto.
Chain	N/A
Language	Solidity
Codebase	as Zip File
Commit	N/A
Unit Tests	N/A

Social Medias

Telegram	N/A
Twitter	https://x.com/JetBoltOfficial
Facebook	N/A
Instagram	N/A
GitHub	N/A
Reddit	N/A
Medium	N/A
Discord	N/A
YouTube	N/A
TikTok	N/A
LinkedIn	N/A
CoinMarketCap	N/A

Audit Summary

Version	Delivery Date	Change Log
v1.0	31 July, 2024	• Layout Project • Automated/Manual-Security Review • Summary

Note - The following audit report presents a comprehensive security analysis of the smart contract utilized in the project. This analysis did not include functional testing (or unit testing) of the contract's logic. We cannot guarantee 100% logical correctness of the contract as it was not functionally tested by us.

File Overview

The Team provided us with the files that should be tested in the security assessment. This audit covered the following files listed below with a SHA-1 Hash.

1. Contracts.zip
   (558228e0669538ddb36ed383e637b4c43461849d)

Please note: Files with a different hash value than in this table have been modified after the security check, either intentionally or unintentionally. A different hash value may (but need not) be an indication of a changed state or potential vulnerability that was not the subject of this scan.

Imported packages

Used code from other Frameworks/Smart Contracts (direct imports).
1. @openzeppelin/contracts-upgradeable/access/OwnableUpgradeable.sol
2. @openzeppelin/contracts-upgradeable/proxy/utils/Initializable.sol
3. @openzeppelin/contractsupgradeable/token/ERC20/ERC20Upgradeable.sol

Please note: Files with a different hash value than in this table have been modified after the security check, either intentionally or unintentionally. A different hash value may (but need not) be an indication of a changed state or potential vulnerability that was not the subject of this scan.

Unit Tests Results

Token.sol

Token.t.sol:TokenTest

[PASS] testFail_updateInterfacesEOA() (gas: 351231)
[PASS] testFail_updateInterfacesNotOwner() (gas: 355432)
[PASS] test_initialize() (gas: 348728)
[PASS] test_updateInterfaces() (gas: 401995)

   Suite result: ok. 4 passed; 0 failed; 0 skipped; finished in 4.75ms (508.30µs CPU time)

TokenAdmin.t.sol:TokenAdminTest

[PASS] testFail_enableTransfersNonOwner() (gas: 14049)
[PASS] testFail_endPresaleNonOwner() (gas: 14091)
[PASS] testFail_restricUserNoOwnerBan() (gas: 14428)
[PASS] testFail_restricUserNoOwnerUnban() (gas: 14335)
[PASS] test_enableTransfers() (gas: 33360)
[PASS] test_endPresale() (gas: 33514)
[PASS] test_getBanned() (gas: 52313)
[PASS] test_getNetWorth() (gas: 1081423)
[PASS] test_restricUserBan() (gas: 47270)
[PASS] test_restricUserUnban() (gas: 47554)

   Suite result: ok. 10 passed; 0 failed; 0 skipped; finished in 3.41ms (1.75ms CPU time)

TokenMint.t.sol:TokenMintTest

[PASS] testFail_mintTokensBannedUser() (gas: 109736)
[PASS] testFail_mintTokensExtraLiquidity() (gas: 113667)
[PASS] testFail_mintTokensMoreThanCapped() (gas: 17972540)
[PASS] testFail_mintTokensNoOwner() (gas: 109956)
[PASS] testFail_mintTokensPresaleEnded() (gas: 133130)
[PASS] testFail_mintTokensSigner() (gas: 113461)
[PASS] testFail_mintTokensTeam() (gas: 113871)
[PASS] test_getExchangeRate() (gas: 18591)
[PASS] test_mintTokensMarkerFive() (gas: 910978)
[PASS] test_mintTokensMarkerFour() (gas: 908768)
[PASS] test_mintTokensMarkerOne() (gas: 901616)
[PASS] test_mintTokensMarkerThree() (gas: 906646)
[PASS] test_mintTokensMarkerTwo() (gas: 904370)

Suite result: ok. 13 passed; 0 failed; 0 skipped; finished in 21.75ms (25.04ms CPU time)

TokenTransfer.t.sol:TokenTransferTest

[PASS] testFail_transferByUsernameDisabled() (gas: 109382)
[PASS] testFail_transferByUsernameUserBanned() (gas: 297674)
[PASS] testFail_transferByUsernameUserNotExist() (gas: 48987)
[PASS] testFail_transferDisabled() (gas: 14739)
[PASS] testFail_transferFromDisabled() (gas: 39155)
[PASS] testFail_transferFromUserBanned() (gas: 100240)
[PASS] testFail_transferUserBanned() (gas: 75288)
[PASS] test_transfer() (gas: 240115)
[PASS] test_transferByUsername() (gas: 325618)
[PASS] test_transferFrom() (gas: 534000)

Suite result: ok. 10 passed; 0 failed; 0 skipped; finished in 3.13ms (1.67ms CPU time)

TokenStake.t.sol:TokenStakeTest

[PASS] testFail_claimRewardsTwiceADay() (gas: 629630)
[PASS] testFail_decreasseStakeNoStake() (gas: 346984)
[PASS] testFail_decreasseStakeZeroAmount() (gas: 346510)
[PASS] testFail_increaseStakeBannedUser() (gas: 52094)
[PASS] testFail_increaseStakeOverBalance() (gas: 18741)
[PASS] testFail_increaseStakeStakeZero() (gas: 14697)
[PASS] test_claimRewardsWithFavoriteBonus() (gas: 2212454)
[PASS] test_decreaseStakeWithRewards() (gas: 703360)
[PASS] test_decreasseStake() (gas: 305830)
[PASS] test_increaseStake() (gas: 351760)
[PASS] test_increaseStakeWithRewards() (gas: 707975)

Suite result: ok. 11 passed; 0 failed; 0 skipped; finished in 4.75ms (4.85ms CPU time)

Friends.sol

Friends.t.sol:FriendsTest

[PASS] testFail_updateInterfacesEOA() (gas: 60993)
[PASS] testFail_updateInterfacesNotOwner() (gas: 63043)
[PASS] test_initialize((gas: 58675)
[PASS] test_updateInterfaces() (gas: 84797)

   Suite result: ok. 4 passed; 0 failed; 0 skipped; finished in 1.29ms (315.66μs CPU time)

FriendsUsername.t.sol:FriendsUsernameTest

[PASS] testFail_setUsernameMixed() (gas: 17991)
[PASS] testFail_setUsernameTooLong() (gas: 15550)
[PASS] testFail_setUsernameTooShort() (gas: 15646)
[PASS] testFail_setUsernameUnderscore() (gas: 17991)
[PASS] testFail_setUsernameUpperrcase() (gas: 15939)
[PASS] test_checkUsernameAvailability() (gas: 19063)
[PASS] test_checkUsernameAvailabilityNotAvailable() (gas: 107195)
[PASS] test_getAddress() (gas: 103236)
[PASS] test_getAddressNoUsername() (gas: 15812)
[PASS] test_getUsername() (gas: 104563)
[PASS] test_setUsername() (gas: 101023)

   Suite result: ok. 11 passed; 0 failed; 0 skipped; finished in 4.75ms (1.27ms CPU time)

FriendsFriendFunctions.t.sol:FriendsFriendFunctionsTest

[PASS] testFail_acceptFriendNoRequest() (gas: 15699)
[PASS] testFail_addFriendAddressZero() (gas: 13203)
[PASS] testFail_addFriendByUsernameNonExistant() (gas: 16090)
[PASS] testFail_addFriendPendingRequest() (gas: 169984)
[PASS] testFail_addFriendSelf() (gas: 12585)
[PASS] testFail_addFriendTwice() (gas: 319728)
[PASS] testFail_declineFriendNoRequest() (gas: 14561)
[PASS] testFail_removeFriendAddressZero() (gas: 13154)
[PASS] testFail_removeFriendNoFriend() (gas: 15374)
[PASS] test_acceptFriend() (gas: 249685)
[PASS] test_acceptFriendAddedByUsername() (gas: 313752)
[PASS] test_addFriend() (gas: 167422)
[PASS] test_addFriendByUsername() (gas: 247732)
[PASS] test_addFriendMultiple() (gas: 682404)
[PASS] test_declineFriend() (gas: 143570)
[PASS] test_declineFriendAddedByUsername() (gas: 207838)

[PASS] test_removeFriendRequestAcceptor() (gas: 265473)
[PASS] test_removeFriendRequestSender() (gas: 265958)

   Suite result: ok. 18 passed; 0 failed; 0 skipped; finished in 4.75ms (2.68ms CPU time)

FriendsProfile.t.sol:FriendsProfileTest

[PASS] test_getFavorites() (gas: 1201372)
[PASS] test_getFriendOutgoingRequests() (gas: 544798)
[PASS] test_getFriendRequests() (gas: 547268)
[PASS] test_getFriends() (gas: 1007710)
[PASS] test_getProfile() (gas: 755211)
[PASS] test_getProfileFrom() (gas: 876062)
[PASS] test_getProfileFromFavorite() (gas: 987856)
[PASS] test_getProfileFromFriend() (gas: 989749)
[PASS] test_getProfileFromIncoming() (gas: 988317)
[PASS] test_getProfileFromPending() (gas: 989938)

   Suite result: ok. 10 passed; 0 failed; 0 skipped; finished in 4.76ms (7.49ms CPU time)

FriendsFavorites.t.sol:FriendsFavoritesTest

[PASS] testFail_addFavoriteAddressZero() (gas: 11765)
[PASS] testFail_addFavoriteAlreadyFavorit() (gas: 462835)
[PASS] testFail_addFavoriteNonFriend() (gas: 18933)
[PASS] testFail_addFavroite11Favorites() (gas: 4079529)
[PASS] testFail_unfavoriteAddressZero() (gas: 3828669)
[PASS] testFail_unfavoriteNoFavorite() (gas: 3708721)
[PASS] test_addFavorite() (gas: 365213)
[PASS] test_addFavroite10Favorites() (gas: 3057415)
[PASS] test_getFavoritesAddresses() (gas: 3069084)
[PASS] test_unfavorite() (gas: 3069218)

   Suite result: ok. 11 passed; 0 failed; 0 skipped; finished in 4.74ms (10.54ms CPU time)

Sessions.sol

[PASS] test_createSession() (gas: 130993)
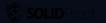**[PASS]** test_deleteSession() (gas: 159041)
[PASS] test_initilize() (gas: 56356)
[PASS] test_renewSession() (gas: 156083)

Suite result: ok. 4 passed; 0 failed; 0 skipped; finished in 685.62μs (665.03μs CPU time)

Unit Tests Summary

Ran 11 test suites in 22.88ms (58.74ms CPU time):
106 tests passed,
0 failed,
0 skipped
(106 total tests)

Blockchain Security | Smart Contract Audits | KYC Development | Marketing

MADE IN GERMANY