AuditAce

FROM INCEPTION TO SUCCESS

I-Stable

Smart Contract Audit Report

ABOUT AUDITACE

Audit Ace is built, to combat financial fraud in the cryptocurrency industry, a growing security firm that provides audits, Smart contract creation, and end-to-end solutions to all crypto-related queries.

Website - https://auditace.tech/

Telegram - https://t.me/Audit_Ace

Twitter - https://twitter.com/auditace_

Github - https://github.com/Audit-Ace

Overview

AUDITACE team has performed a line-by-line manual analysis and automated review of smart contracts. Smart contracts were analyzed mainly for common contract vulnerabilities, exploits, and manipulation hacks.

Audit Result: Passed - Medium Risk

Audit Date: November 26, 2022

KYC:Not done till date of Audit

Audit Team: TEAM AUDITACE

Result details: no centralization or logical issues found in the contract, token launched on a local blockchain and all functionalities were tested.

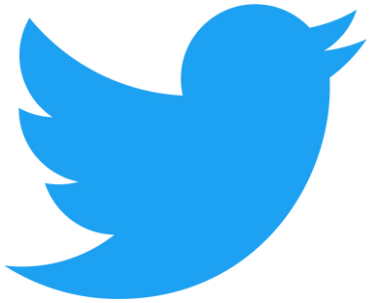
Disclaimer

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment. Team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed. The Auditace team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Auditace receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token. The Auditace team disclaims any liability for the resulting losses.

I-Stable

Social Media Overview

https://t.me/iStableOfficial



twitter.com/istableofficial

http://i-Stable.com

Token Summary

Parameter	Result
Address	0x0573780eB18D5c847D89e745e94149B9E9d0cdE8
Token Type	BEP 20
Decimals	18
Supply	100,000,000
Platform	Binance Smart Chain
Compiler	**v0.8.17+commit.8df45f5f**
Contract checksum	debd8d927f8951e077a12890dc0306ec0b27621c2 b16b832247eeebce198561a

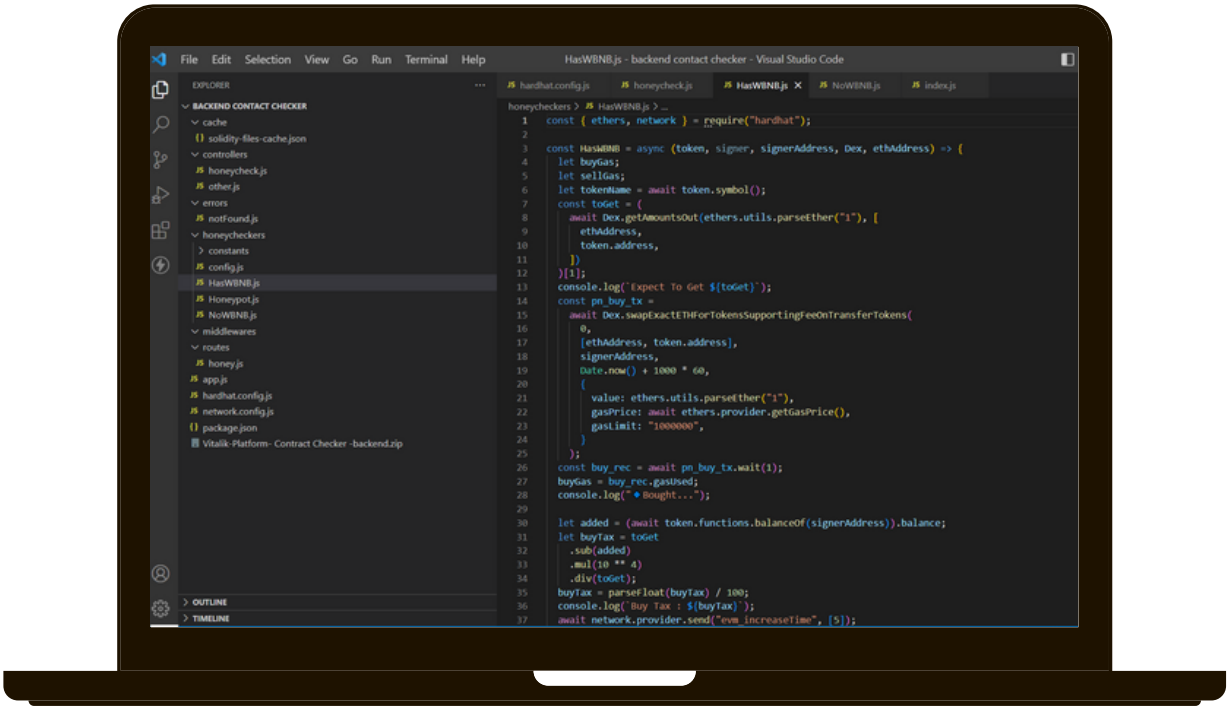
Used Tools

Manual Review - Forked Pancakeswap V2 on local blockchain

Tests:

- **adding liquidty with WBNB and BUSD**
- **buying and selling right after launch**
- **taxes get collected inside the contract**
- **taxes are sent marketing wallet (BNB) and autoliquidity works**

CONTRACT FUNCTION SUMMARY



Can edit Tax?

DETECTED

Can take back Ownership?

NOT DETECTED

Is Blacklisted?

NOT DETECTED

Is Whitelisted?

NOT DETECTED

Is Mintable?

NOT DETECTED

Disable Trade?

DETECTED

Is Trading with CooldownTime?

NOT DETECTED

AUDIT METHODOLOGY

The auditing process will follow a routine as special considerations by Auditace:

- Review of the specifications, sources, and instructions provided to Auditace to make sure the contract logic meets the intentions of the client without exposing the user's funds to risk.

- Manual review of the entire codebase by our experts, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.

- Specification comparison is the process of checking whether the code does what the specifications, sources, and instructions provided to Auditace describe.

- Test coverage analysis determines whether the test cases are covering the code and how much code isexercised when we run the test cases.

- Symbolic execution is analysing a program to determine what inputs cause each part of a program to execute.

- Reviewing the codebase to improve maintainability, security, and control based on the established industry and academic practices.

Issues Checking Status

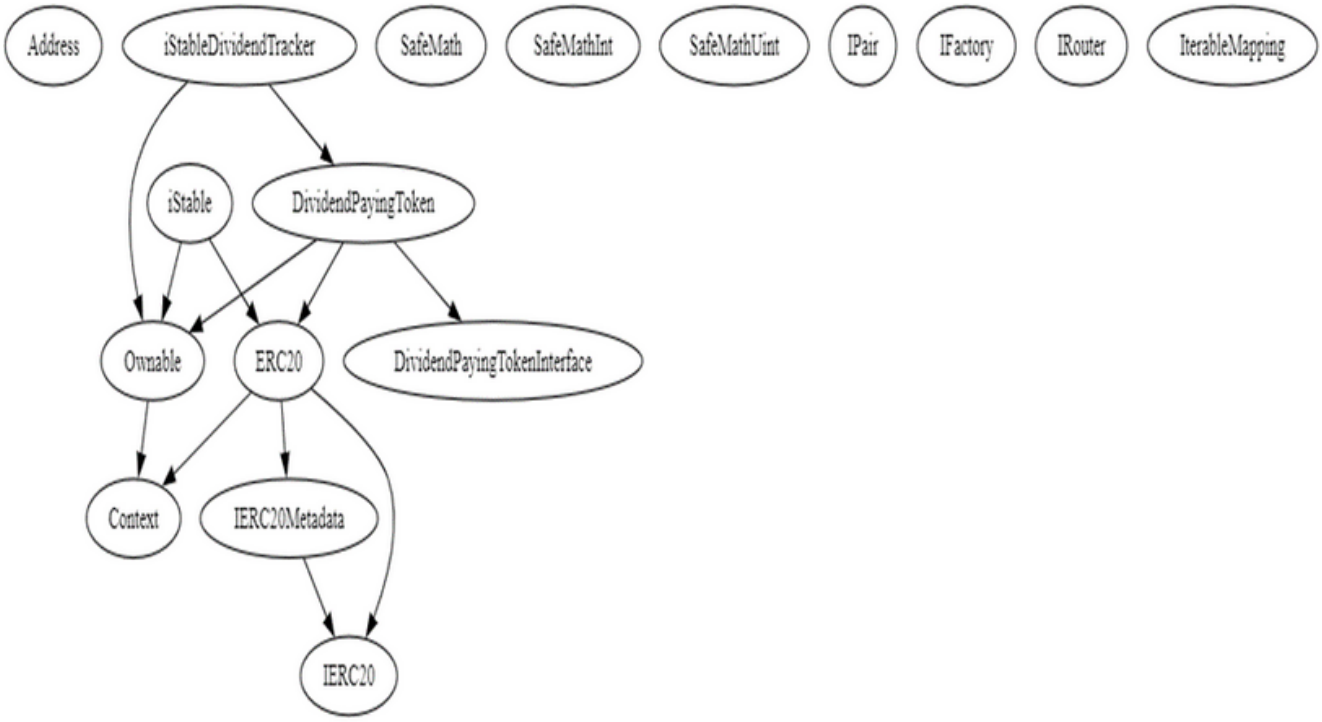
No	Issue Description	Checking Status
1	Compiler warnings.	Passed
2	Race conditions and Reentrancy. Cross-function race conditions.	Passed
3	Possible delays in data delivery.	Passed
4	Oracle calls.	Passed
5	Front running.	Passed
6	Timestamp dependence.	Passed
7	Integer Overflow and Underflow.	Passed
8	DoS with Revert.	Passed
9	DoS with block gas limit.	Passed
10	Methods execution permissions.	Passed
11	Design Logic.	Passed
12	Cross-function race conditions.	Passed
13	Safe Zeppelin module.	Passed
14	Malicious Event log.	Passed
15	Scoping and Declarations.	Passed
16	Fallback function security.	Passed
17	Arithmetic accuracy.	Passed

SWC ATTACK TEST

SWC ID	Description	Test Result
SWC-100	Function Visibility	Passed
SWC-101	Integer Overflow and Underflow	Passed
SWC-102	Outdated Compiler Version	Passed
SWC-103	Floating Pragma	Passed
SWC-104	Unchecked Call Return Value	Passed
SWC-105	Unprotected Ether Withdrawal	Passed
SWC-106	Unprotected SELFDESTRUCT Instruction	Passed
SWC-107	Re-entrancy	Passed
SWC-108	State Variable Default Visibility	Passed
SWC-109	Uninitialized Storage Pointer	Passed
SWC-110	Assert Violation	Passed
SWC-111	Use of Deprecated Solidity Functions	Passed
SWC-112	Delegate Call to Untrusted Callee	Passed
SWC-113	DoS with Failed Call	Passed
SWC-114	Transaction Order Dependence	Passed
SWC-115	Authorization through tx.origin	Passed
SWC-116	Block values as a proxy for time	Passed

SWC ID	Description	Test Result
SWC-117	Signature Malleability	Passed
SWC-118	Incorrect Constructor Name	Passed
SWC-119	Shadowing State Variables	Passed
SWC-120	Weak Sources of Randomness from Chain Attributes	Passed
SWC-121	Missing Protection against Signature Replay Attacks	Passed
SWC-122	Lack of Proper Signature Verification	Passed
SWC-123	Requirement Violation	Passed
SWC-124	Write to Arbitrary Storage Location	Passed
SWC-125	Incorrect Inheritance Order	Passed
SWC-126	Insufficient Gas Grieving	Passed
SWC-127	Arbitrary Jump with Function Type Variable	Passed
SWC-128	DoS With Block Gas Limit	Passed
SWC-129	Typographical Error	Passed
SWC-130	Right-To-Left-Override control character (U+202E)	Passed
SWC-131	Presence of unused variables	Passed
SWC-132	Unexpected Ether balance	Passed
SWC-133	Hash Collisions with Multiple Variable Length Arguments	Passed
SWC-134	Unencrypted Private Data On-Chain	Passed

Inheritance Tree



Summary

- Anti-Bot implementation : for up to 5 blocks, buyers and sellers get taxes by 99%
- Owner is able to change taxes, buy and sell up to 12% each.(24% total)
- Owner is not able to set max buy/sell/transferring amount
- Owner is not able to mint new tokens
- Owner is not able to pause trades
- Owner must enable trading in order for investors to be able to trade
- Owner is not able to blacklist an arbitrary address

Classification of Risks

Severity	Description
◆ High-Risk	A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way.
◆ Medium-Risk	A vulnerability that could affect the desired outcome of executing the contract in a specific scenario.
◆ Low-Risk	A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective.
◆ Gas Optimization /Suggestion	A vulnerability that has an informational character but is not affecting any of the code.

Findings

Severity	Found
◆ High-Risk	0
◆ Medium-Risk	2
◆ Low-Risk	2
◆ Gas Optimization / Suggestions	3

MANUAL AUDIT

Medium Risk Findings

Centralization -owner is able to update dividend tracker, changing dividend tracker to a malicious contract may effect trades (eg disable them)

```
  function updateDividendTracker(address newAddress) public
onlyOwner {
    iStableDividendTracker newDividendTracker =
iStableDividendTracker(
      payable(newAddress)
    );

newDividendTracker.excludeFromDividends(address(newDividendTr
acker), true);
newDividendTracker.excludeFromDividends(address(this), true);
newDividendTracker.excludeFromDividends(owner(), true);
    newDividendTracker.excludeFromDividends(address(router),
true);
    dividendTracker = newDividendTracker;
  }
```

MANUAL AUDIT

Logical -setting swapTokensAtAmount to 0 can disable sells if collected tokens from taxes (i.e contract's eldendoge token balance) is greater than swapTokensAtAmount.

```
  function setSwapTokensAtAmount(uint256 amount) external onlyOwner {
    require(amount < 1e6,"Swap Threshold should be less than 1% of total supply");
    swapTokensAtAmount = amount * 10**9;
  }
```

suggestion: make sure that you can not set swapTokensAtAmount to 0 at setSwapTokensAtAmount

MANUAL AUDIT

Low Risk Findings

Centralization - owner is able to change buy/sell taxes each one up to 12%.

```
  function setBuyTaxes(uint256 _rewards, uint256 _marketing,
uint256 _dev, uint256 _liquidity) external onlyOwner{
      buyTaxes = Taxes(_rewards, _marketing, _dev, _liquidity);
      require((_rewards + _marketing + _dev + _liquidity ) <= 12, "Must
keep fees at 12% or less");
}
  function setSellTaxes(uint256 _rewards, uint256 _marketing,
uint256 _dev, uint256 _liquidity) external onlyOwner{
      sellTaxes = Taxes(_rewards, _marketing, _dev, _liquidity);
      require((_rewards + _marketing + _dev + _liquidity ) <= 12, "Must
keep fees at 12% or less");
}
```

MANUAL AUDIT

Logical -setting rewardToken to a non-erc20 contract can revert the transaction if holder is eligible for claiming rewards (auto claim).

```
function setRewardToken(address newToken) external onlyOwner {
require(newToken!=address(0),"New token can't be zero address");
require(newToken!=deadWallet,"New token can't be dead address");
require(newToken!=address(this),"New token can't be contract
address itself");
dividendTracker.setRewardToken(newToken);
  }
```

suggestion: make sure that you can not set swapTokensAtAmount to 0 at setSwapTokensAtAmount

MANUAL AUDIT

Gas Optimizations

- do not use SafeMath library, overflow/underflows are handled internally by compiler if compiler's version is more than 0.8.0, using safemath only increases gas usage.
- create 2 variables for total buy taxes and total sell taxes, this will decrease gas usage in _transfer function which leads to lower gas usage for all buys/sells/transfers, currently SLOAD opcode costs 2100 gas

Suggestions

- do not use SafeMath library, overflow/underflows are handled internally by compiler if compiler's version is more than 0.8.0