Techrate1 Techrate

TechRate

AUDIT COMPANY

Smart Contract Security Audit

TechRate

June, 2021

Audit Details

Audited project

American Shiba

Deployer address

0xC6c2aac7996d6670C25Ea821816D67AaDB6e89e3

Client contacts:

American Shiba team

Blockchain

Ethereum

Project website:

Not provided by American Shiba team

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the below disclaimer below – please make sure to read it in full.

DISCLAIMER: By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and TechRate and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (TechRate) owe no duty of care towards you or any other person, nor does TechRate make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and TechRate hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, TechRate hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against TechRate, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.

The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

Background

TechRate was commissioned by American Shiba to perform an audit of smart contracts:

https://etherscan.io/address/0xb893a8049f250b57efa8c62d51527a22404d7c9a#code

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

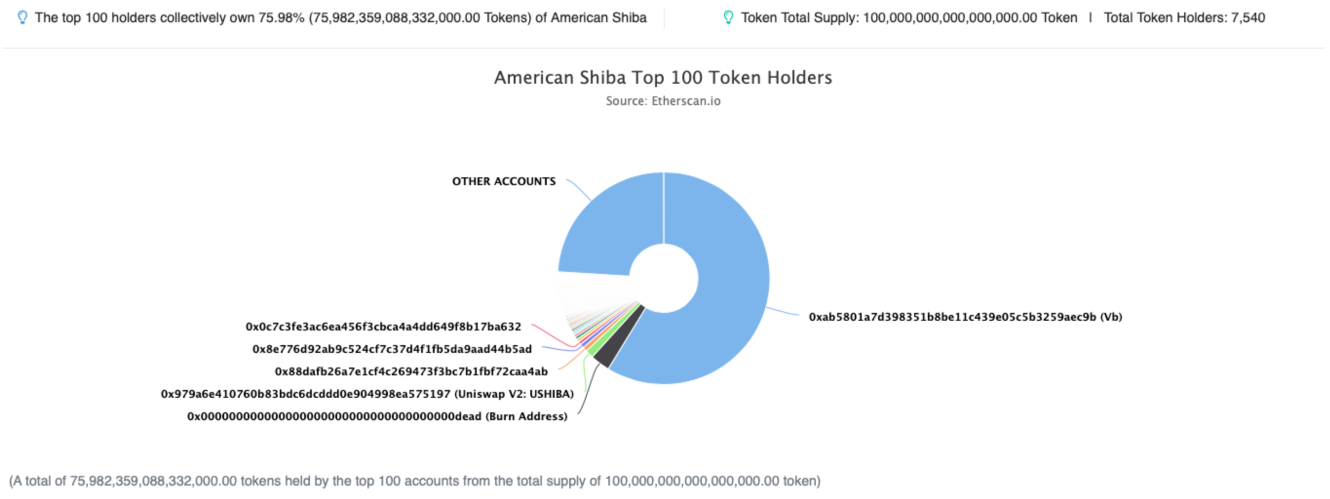
The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Contracts Details

Token contract details for 05.06.2021

Contract name	American Shiba
Contract address	0xB893A8049f250b57eFA8C62D51527a22404D7c9A
Total supply	100,000,000,000,000,000
Token ticker	USHIBA
Decimals	9
Token holders	7,540
Transactions count	19,269
Top 100 holders dominance	75.98%
Tax fee	2
Total fees	8719894940072234710526792
Contract deployer address	0xC6c2aac7996d6670C25Ea821816D67AaDB6e89e3
Contract’s current owner address	0x0000000000000000000000000000000000000000

American Shiba Token Distribution



The top 100 holders collectively own 75.98% (75,982,359,088,332,000.00 Tokens) of American Shiba

Token Total Supply: 100,000,000,000,000,000.00 Token | Total Token Holders: 7,540

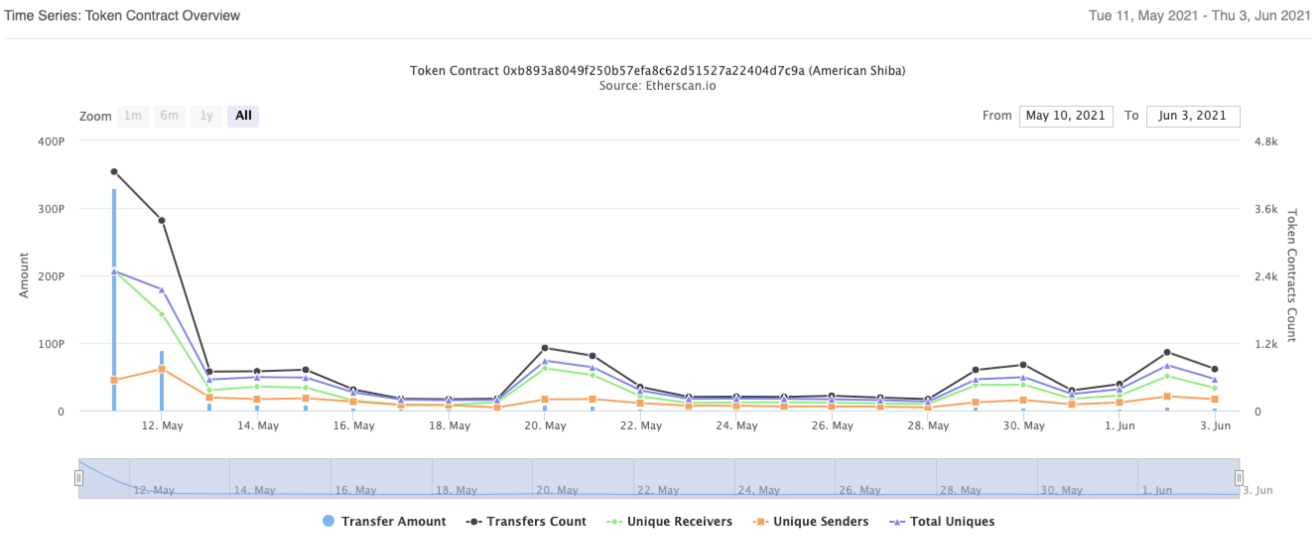
American Shiba Top 100 Token Holders

Source: Etherscan.io

OTHER ACCOUNTS
0xab5801a7d398351b8be11c439e05c5b3259aec9b (Vb)
0x0c7c3fe3ac6ea456f3cbca4a4dd649f8b17ba632
0x8e776d92ab9c524cf7c37d4f1fb5da9aad44b5ad
0x88dafb26a7e1cf4c269473f3bc7b1fbf72caa4ab
0x979a6e410760b83bdc6dcddd0e904998ea575197 (Uniswap V2: USHIBA)
0x000000000000000000000000000000000000dead (Burn Address)

(A total of 75,982,359,088,332,000.00 tokens held by the top 100 accounts from the total supply of 100,000,000,000,000,000.00 token)

American Shiba Contract Interaction Details



Time Series: Token Contract Overview

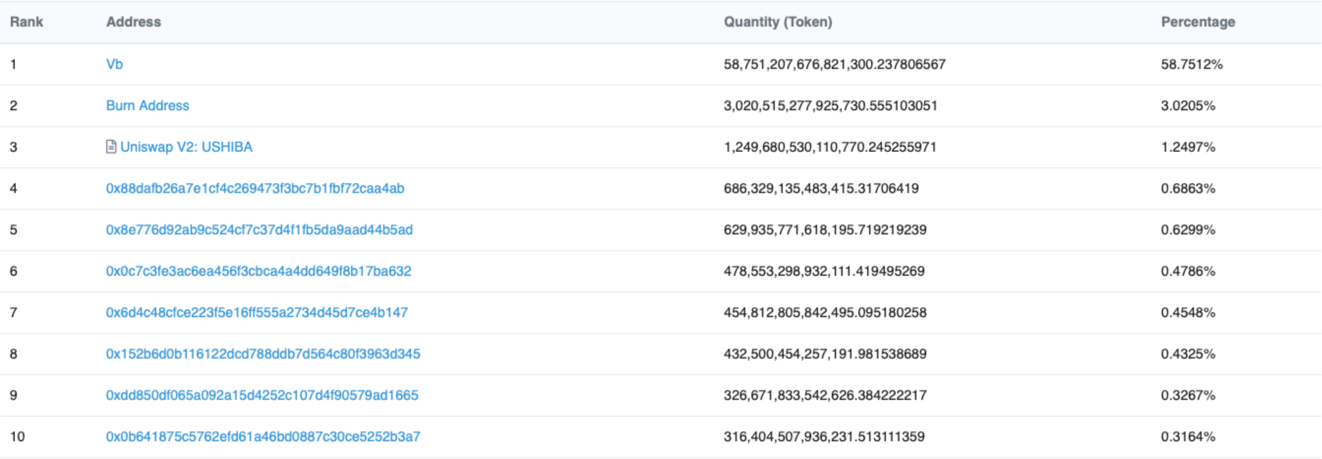
Tue 11, May 2021 - Thu 3, Jun 2021

Token Contract 0xb893a8049f250b57efa8c62d51527a22404d7c9a (American Shiba)

Source: Etherscan.io

From May 10, 2021 To Jun 3, 2021

American Shiba Top 10 Token Holders



Rank	Address	Quantity (Token)	Percentage
1	Vb	58,751,207,676,821,300.237806567	58.7512%
2	Burn Address	3,020,515,277,925,730.555103051	3.0205%
3	🗎 Uniswap V2: USHIBA	1,249,680,530,110,770.245255971	1.2497%
4	0x88dafb26a7e1cf4c269473f3bc7b1fbf72caa4ab	686,329,135,483,415.31706419	0.6863%
5	0x8e776d92ab9c524cf7c37d4f1fb5da9aad44b5ad	629,935,771,618,195.719219239	0.6299%
6	0x0c7c3fe3ac6ea456f3cbca4a4dd649f8b17ba632	478,553,298,932,111.419495269	0.4786%
7	0x6d4c48cfce223f5e16ff555a2734d45d7ce4b147	454,812,805,842,495.095180258	0.4548%
8	0x152b6d0b116122dcd788ddb7d564c80f3963d345	432,500,454,257,191.981538689	0.4325%
9	0xdd850df065a092a15d4252c107d4f90579ad1665	326,671,833,542,626.384222217	0.3267%
10	0x0b641875c5762efd61a46bd0887c30ce5252b3a7	316,404,507,936,231.513111359	0.3164%

Contract functions details

```
 +  Context
  - [Int] _msgSender
  - [Int] _msgData

+ [Int] IERC20
  - [Ext] totalSupply
  - [Ext] balanceOf
  - [Ext] transfer #
  - [Ext] allowance
  - [Ext] approve #
  - [Ext] transferFrom #

+ [Lib] SafeMath
  - [Int] add
  - [Int] sub
  - [Int] sub
  - [Int] mul
  - [Int] div
  - [Int] div
  - [Int] mod
  - [Int] mod

+ [Lib] Address
  - [Int] isContract
  - [Int] sendValue #
  - [Int] functionCall #
  - [Int] functionCall #
  - [Int] functionCallWithValue #
  - [Int] functionCallWithValue #
  - [Prv] _functionCallWithValue #

+  Ownable (Context)
  - [Int] <Constructor> #
  - [Pub] owner
  - [Pub] renounceOwnership #
    - modifiers: onlyOwner
  - [Pub] transferOwnership #
    - modifiers: onlyOwner

+  AmericanShiba (Context, IERC20, Ownable)
  - [Pub] <Constructor> #
  - [Pub] name
  - [Pub] symbol
  - [Pub] decimals
  - [Pub] totalSupply
  - [Pub] balanceOf
  - [Pub] transfer #
  - [Pub] allowance
  - [Pub] approve #
  - [Pub] transferFrom #
  - [Pub] increaseAllowance #
```

- [Pub] decreaseAllowance #
- [Pub] isExcluded
- [Pub] totalFees
- [Ext] setMaxTxPercent #
  - modifiers: onlyOwner
- [Ext] rescueFromContract #
  - modifiers: onlyOwner
- [Pub] reflect #
- [Pub] reflectionFromToken
- [Pub] tokenFromReflection
- [Ext] excludeAccount #
  - modifiers: onlyOwner
- [Ext] includeAccount #
  - modifiers: onlyOwner
- [Prv] _approve #
- [Prv] _transfer #
- [Prv] _transferStandard #
- [Prv] _transferToExcluded #
- [Prv] _transferFromExcluded #
- [Prv] _transferBothExcluded #
- [Prv] _reflectFee #
- [Prv] _getValues
- [Prv] _getTValues
- [Prv] _getRValues
- [Prv] _getRate
- [Prv] _getCurrentSupply

($) = payable function
# = non-constant function

Issues Checking Status

	Issue description	Checking status
1.	Compiler errors.	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.	Low issues
10.	Methods execution permissions.	Passed
11.	Economy model of the contract.	Passed
12.	The impact of the exchange rate on the logic.	Passed
13.	Private user data leaks.	Passed
14.	Malicious Event log.	Passed
15.	Scoping and Declarations.	Passed
16.	Uninitialized storage pointers.	Passed
17.	Arithmetic accuracy.	Passed
18.	Design Logic.	Passed
19.	Cross-function race conditions.	Passed
20.	Safe Open Zeppelin contracts implementation and usage.	Passed
21.	Fallback function security.	Passed

Security Issues

High Severity Issues

No high severity issues found.

Medium Severity Issues

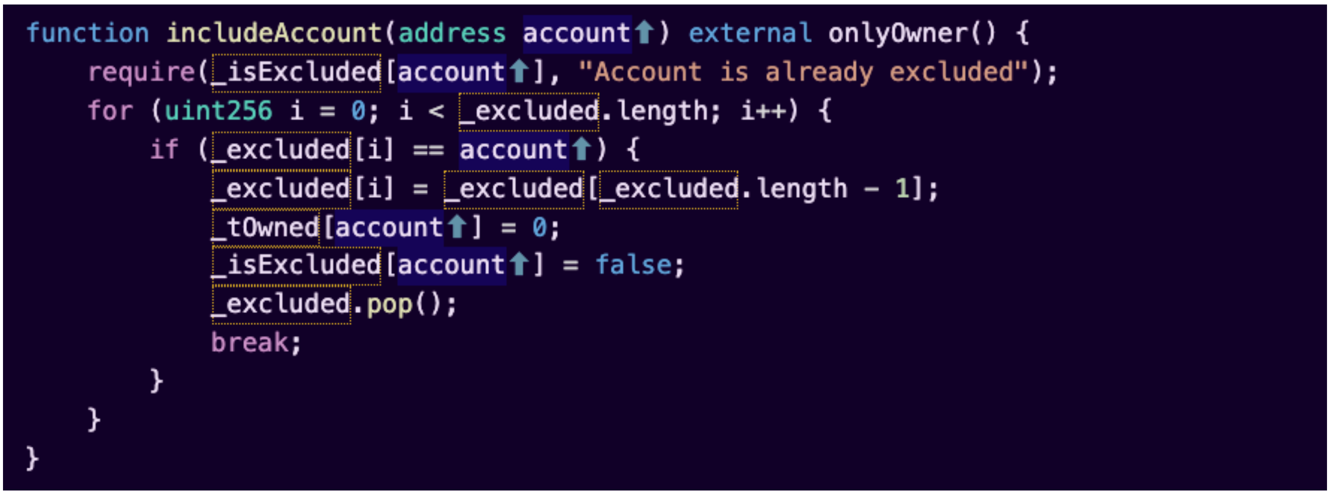
No medium severity issues found.

Low Severity Issues

1. Out of gas

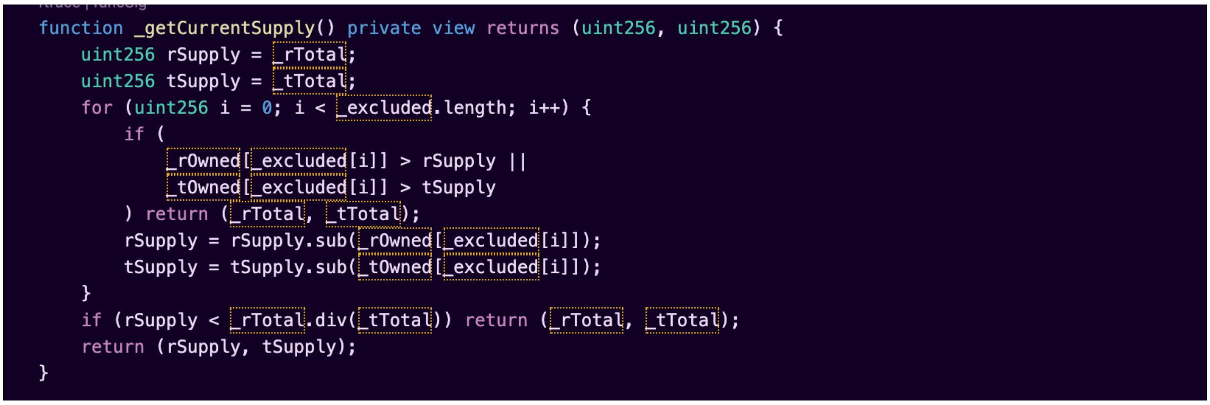
Issue:

- The function includeInAccount() uses the loop to find and remove addresses from the _excluded list. Function will be aborted with OUT_OF_GAS exception if there will be a long excluded addresses list.



```
function includeAccount(address account) external onlyOwner() {
    require(_isExcluded[account], "Account is already excluded");
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (_excluded[i] == account) {
            _excluded[i] = _excluded[_excluded.length - 1];
            _tOwned[account] = 0;
            _isExcluded[account] = false;
            _excluded.pop();
            break;
        }
    }
}
```

- The function _getCurrentSupply also uses the loop for evaluating total supply. It also could be aborted with OUT_OF_GAS exception if there will be a long excluded addresses list.



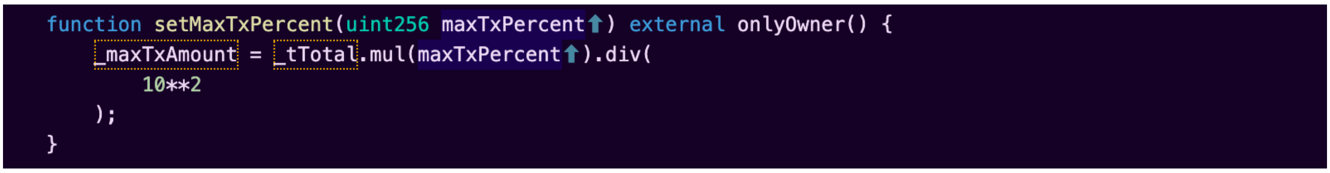
```
function _getCurrentSupply() private view returns (uint256, uint256) {
    uint256 rSupply = _rTotal;
    uint256 tSupply = _tTotal;
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (
            _rOwned[_excluded[i]] > rSupply ||
            _tOwned[_excluded[i]] > tSupply
        ) return (_rTotal, _tTotal);
        rSupply = rSupply.sub(_rOwned[_excluded[i]]);
        tSupply = tSupply.sub(_tOwned[_excluded[i]]);
    }
    if (rSupply < _rTotal.div(_tTotal)) return (_rTotal, _tTotal);
    return (rSupply, tSupply);
}
```

Recommendation:

Check that the excluded array length is not too big.

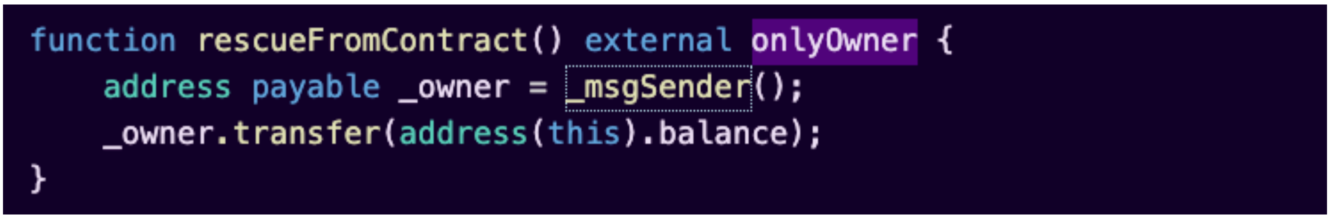
Owner privileges (In the period when the owner is not renounced)

- Owner can change the maximum transaction amount.



```
function setMaxTxPercent(uint256 maxTxPercent) external onlyOwner() {
    _maxTxAmount = _tTotal.mul(maxTxPercent).div(
        10**2
    );
}
```

- Owner can withdraw all contract balance.



```
function rescueFromContract() external onlyOwner {
    address payable _owner = _msgSender();
    _owner.transfer(address(this).balance);
}
```

Conclusion

Smart contracts contain low severity issues! Liquidity pair contract's security is not checked due to out of scope.

Liquidity locking details NOT provided by the team.

TechRate note:

Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by Owner.