soken

SMART CONTRACT SECURITY AUDIT

Base Reward Token

Scan and check this report
was posted at Soken Github

April, 2022

Website: soken.io

Table of Contents

Disclaimer

This is a comprehensive report based on our automated and manual examination of cybersecurity vulnerabilities and framework flaws. We took into consideration smart contract based algorithms, as well. Reading the full analysis report is essential to build your understanding of project's security level. It is crucial to take note, though we have done our best to perform this analysis and report, that you should not rely on the our research and cannot claim what it states or how we created it. Before making any judgments, you have to conduct your own independent research. We will discuss this in more depth in the following disclaimer - please read it fully.

DISCLAIMER: You agree to the terms of this disclaimer by reading this report or any portion thereof. Please stop reading this report and remove and delete any copies of this report that you download and/or print if you do not agree to these conditions. This report is for non-reliability information only and does not represent investment advice. No one shall be entitled to depend on the report or its contents, and Soken and its affiliates shall not be held responsible to you or anyone else, nor shall Soken provide any guarantee or representation to any person with regard to the accuracy or integrity of the report. Without any terms, warranties or other conditions other than as set forth in that exclusion and Soken excludes hereby all representations, warrants, conditions and other terms (including, without limitation, guarantees implied by the law of satisfactory quality, fitness for purposes and the use of reasonable care and skills). The report is provided as "as is" and does not contain any terms and conditions. Except as legally banned, Soken disclaims all responsibility and responsibilities and no claim against Soken is made to any amount or type of loss or damages (without limitation, direct, indirect, special, punitive, consequential or pure economic loses or losses) that may be caused by you or any other person, or any damages or damages, including without limitations (whether innocent or negligent).

Security analysis is based only on the smart contracts. No applications or operations were reviewed for security. No product code has been reviewed.

Procedure

Our analysis contains following steps:

1. Project Analysis;

2. Manual analysis of smart contracts:
- Deploying smart contracts on any of the network(Ropsten/Rinkeby) using Remix IDE
- Hashes of all transaction will be recorded
- Behaviour of functions and gas consumption is noted, as well.

3. Unit Testing:
- Smart contract functions will be unit tested on multiple parameters and under multiple conditions to ensure that all paths of functions are functioning as intended.
- In this phase intended behaviour of smart contract is verified.
- In this phase, we would also ensure that smart contract functions are not consuming unnecessary gas.
- Gas limits of functions will be verified in this stage.

4. Automated Testing:
- Mythril
- Oyente
- Manticore
- Solgraph

Terminology

We categorize the finding into 4 categories based on their vulnerability:

- Low-severity issue — less important, must be analyzed
- Medium-severity issue — important, needs to be analyzed and fixed
- High-severity issue —important, might cause vulnerabilities, must be analyzed and fixed
- Critical-severity issue —serious bug causes, must be analyzed and fixed.

Limitations

The security audit of Smart Contract cannot cover all vulnerabilities. Even if no vulnerabilities are detected in the audit, there is no guarantee that future smart contracts are safe. Smart contracts are in most cases safeguarded against specific sorts of attacks. In order to find as many flaws as possible, we carried out a comprehensive smart contract audit. Audit is a document that is not legally binding and guarantees nothing.

Token Contract Details for 08.04.2022

Contract Name: **BRW**

Deployed address: **0x82FabF676c7876694EDB013226f2a341dECa52Fd**

Total Supply: **12,000,000**

Token Tracker: **BRW**

Decimals: **8**

Token holders: **1**

Transactions count: **1**

Top 100 holders dominance: **100.00%**

Audit Details

Project Name: **Base Reward Token**

Language: **Solidity**

Compiler Version: **v0.8.13**

Blockchain: **BSC**

KYC Passed

CEO and CFO of Base Reward have passed KYC verification on behalf of Soken team. All personal data received from audited company will remain private until any fraudulent activity will happen.

Social Profiles

Project Website: **https://basereward.online/**

Project Dashboard: **https://app.cuex.capital/**

Project Twitter: **https://twitter.com/baserewardtoken**

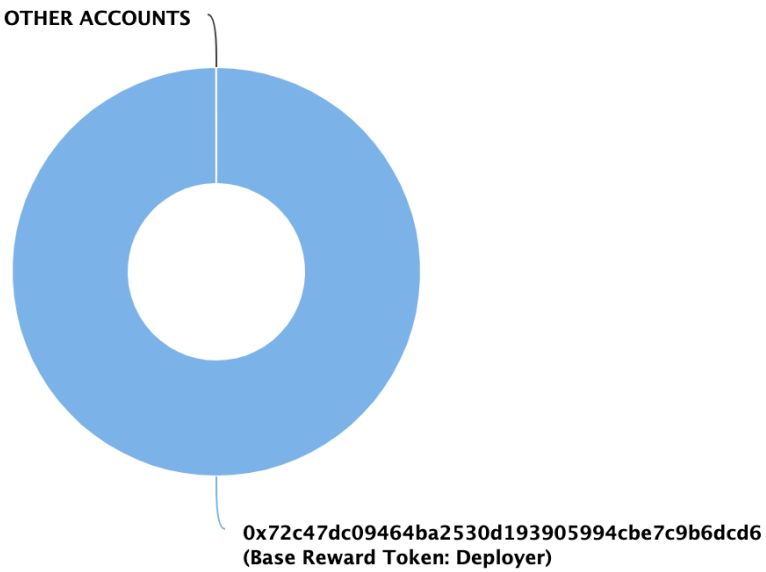
Project Telegram: **https://telegram.me/BASEREWARDOFFICIAL**

Project Facebook: **https://www.facebook.com/basereward/**

Project Linkedin: **https://www.linkedin.com/company/base-reward-token/**

Project YouTube: **https://www.youtube.com/channel/UCwGhxoP3URH---QSnfF7XoQ**

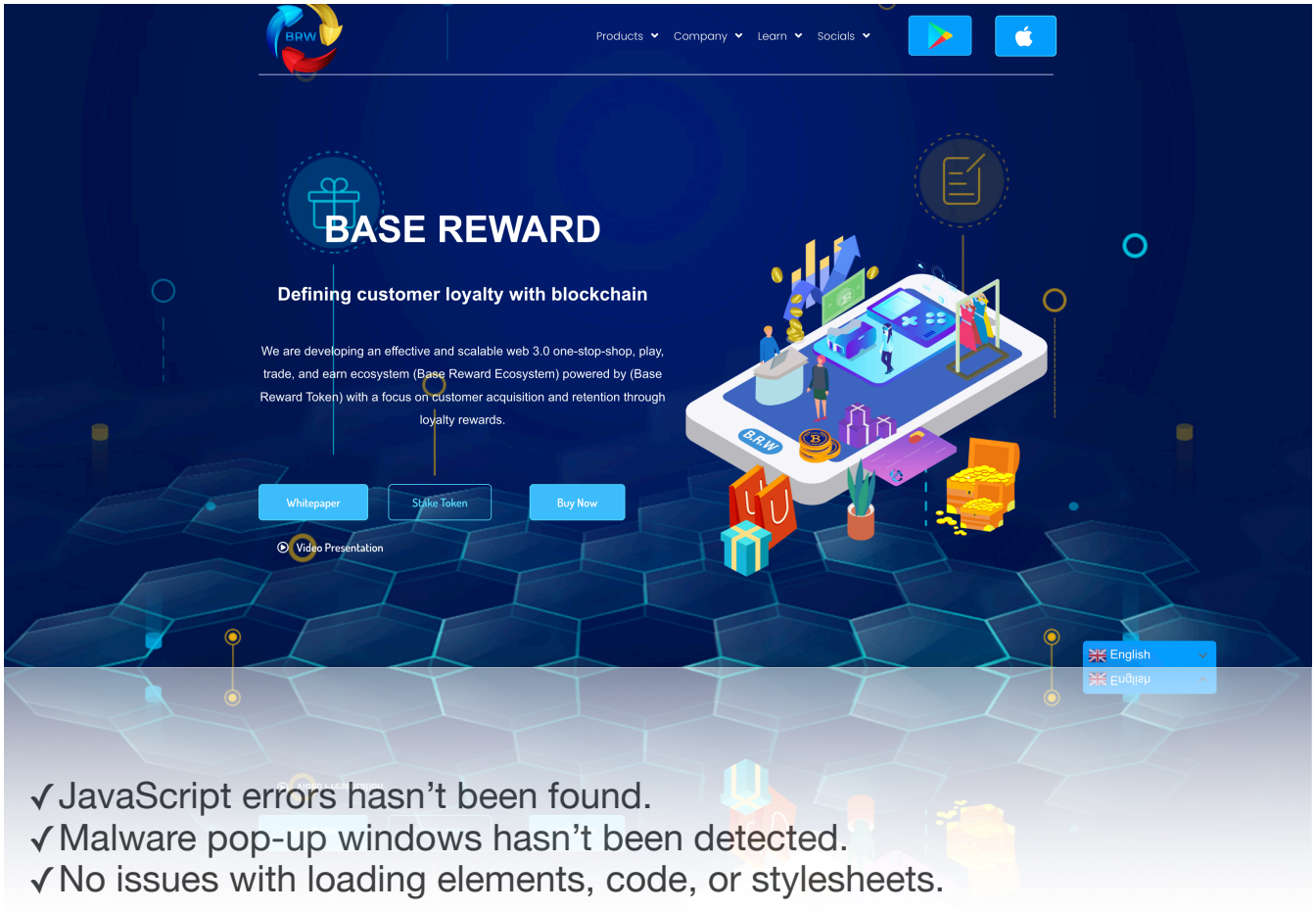
BRW Token Distribution



BRW Top Holders

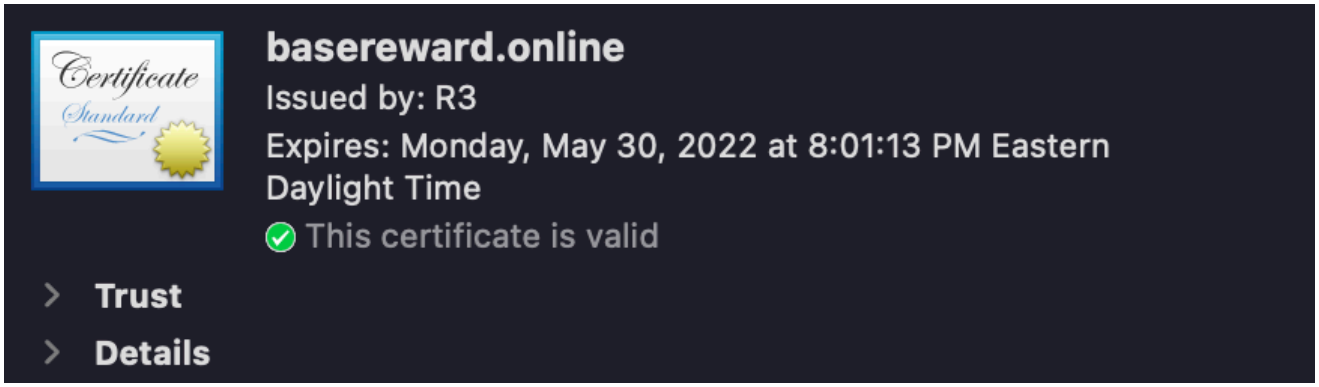
Rank	Address	Quantity (Token)	Percentage
1	Base Reward Token: Deployer	12,000,000	100.0000%

Project Website Overview



✓JavaScript errors hasn't been found.
✓Malware pop-up windows hasn't been detected.
✓No issues with loading elements, code, or stylesheets.

Project Website SSL Certification



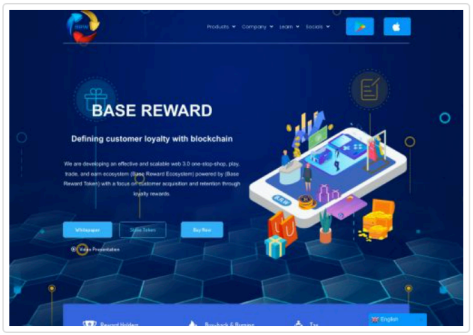
Project Website Performance Audit

73

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 ■ 50–89 ● 90–100



METRICS Expand view

■ First Contentful Paint 0.9 s	■ Time to Interactive 3.9 s
▲ Speed Index 2.6 s	● Total Blocking Time 20 ms
▲ Largest Contentful Paint	● Cumulative Layout Shift

Project Website Optimization for Mobile

33

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 ■ 50–89 ● 90–100

METRICS Expand view

▲ First Contentful Paint 4.8 s	▲ Time to Interactive 18.0 s
▲ Speed Index 9.7 s	■ Total Blocking Time 570 ms
▲ Largest Contentful Paint	● Cumulative Layout Shift

Contract Function Details

+ Contract Source Code

- [Ext] totalSupply
- [Ext] balanceOf
- [Ext] transfer
- [Ext] allowance
- [Ext] approve
- [Ext] transferFrom
- Int] add
- [Int] sub
- [Int] sub
- [Int] mul
- [Int] div
- [Int] div
- [Int] _msgSender
- [Pub] owner
- [Pub] renounceOwnership
- [Pub] transferOwnership
- [Pub] name
- [Pub] symbol
- [Pub] decimals
- [Int] isContract
- [Int] safeTransfer
- [Int] safeTransferFrom
- [Int] safeApprove
- [Prv] callOptionalReturn
- [Ext] feeTo
- [Ext] feeToSetter
- [Ext] getPair
- [Ext] allPairs
- [Ext] allPairsLength
- [Ext] createPair
- [Ext] setFeeTo
- [Ext] setFeeToSetter
- [Ext] name
- [Ext] symbol
- [Ext] decimals
- [Ext] totalSupply
- [Ext] balanceOf
- [Ext] allowance
- [Ext] approve
- [Ext] transfer
- [Ext] transferFrom
- [Ext] DOMAIN_SEPARATOR

- [Ext] PERMIT_TYPEHASH
- [Ext] nonces
- [Ext] permit
- [Ext] MINIMUM_LIQUIDITY
- [Ext] factory
- [Ext] token0
- [Ext] token1
- [Ext] getReserves
- [Ext] price0CumulativeLast
- [Ext] price1CumulativeLast
- [Ext] kLast
- [Ext] mint
- [Ext] burn
- [Ext] swap
- [Ext] skim
- [Ext] sync
- [Ext] initialize
- [Ext] factory
- [Ext] WETH
- [Ext] addLiquidity
- [Ext] addLiquidityETH
- [Ext] removeLiquidity
- [Ext] removeLiquidityETH
- [Ext] removeLiquidityWithPermit
- [Ext] removeLiquidityETHWithPermit
- [Ext] swapExactTokensForTokens
- [Ext] swapTokensForExactTokens
- [Ext] swapExactETHForTokens
- [Ext] swapTokensForExactETH
- [Ext] swapExactTokensForETH
- [Ext] swapETHForExactTokens
- [Ext] quote
- [Ext] getAmountOut
- [Ext] getAmountIn
- [Ext] getAmountsOut
- [Ext] getAmountsIn
- [Ext] removeLiquidityETHSupportingFeeOnTransferTokens
- [Ext] removeLiquidityETHWithPermitSupportingFeeOnTransferTokens
- [Ext] swapExactTokensForTokensSupportingFeeOnTransferTokens
- [Ext] swapExactETHForTokensSupportingFeeOnTransferTokens
- [Ext] swapExactTokensForETHSupportingFeeOnTransferTokens
- [Pub] totalSupply
- [Pub] balanceOf
- [Pub] transfer
- [Pub] allowance

- [Pub] approve
- [Pub] transferFrom
- [Pub] increaseAllowance
- [Pub] decreaseAllowance
- [Ext] setLiquidityFeePercent
- [Pub] setSwapAndLiquifyEnabled
- [Pub] excludeFromFee
- [Pub] includeInFee
- [Pub] excludeFromBotList
- [Pub] includeInBotList
- [Ext] isBot
- [Ext] changeNumTokensSellToAddToLiquidity
- [Int] _transfer
- [Prv] swapAndLiquify
- [Prv] swapTokensForEth
- [Prv] addLiquidity
- [Int] _approve
- [Ext] withdrawStuckBNB
- [Ext] removeStuckToken

Vulnerabilities checking

Issue Description	Checking Status
Compiler Errors	Completed
Delays in Data Delivery	Completed
Re-entrancy	Completed
Transaction-Ordering Dependence	Completed
Timestamp Dependence	Completed
Shadowing State Variables	Completed
DoS with Failed Call	Completed
DoS with Block Gas Limit	Completed
Outdated Complier Version	Completed
Assert Violation	Completed
Use of Deprecated Solidity Functions	Completed
Integer Overflow and Underflow	Completed
Function Default Visibility	Completed
Malicious Event Log	Completed
Math Accuracy	Completed
Design Logic	Completed
Fallback Function Security	Completed
Cross-function Race Conditions	Completed
Safe Zeppelin Module	Completed

Security Issues

1) Owner Privileges

The contract contains ownership functionality and ownership is not renounced which allows the creator or current owner to modify contract behaviour (for example, disable selling or mint new tokens).

2) Volatile Code:

The return values of functions *swapExactTokensForETHSupportingFeeOnTransferTokens* and *addLiquidityETH* are not properly handled.

Recommendation:

We recommend using variables to receive the return value of the functions mentioned above and handle both success and failure cases if needed by the business logic.

3) Unprotected Ether Withdrawal: L940 - 947;

```
940       function recover(address tokenAddress, uint256 tokenAmount) public onlyOwner {
941           if (tokenAddress == address(0)) {
942               (bool success, ) = payable(owner()).call{value: tokenAmount}('');
943               require(success, 'unable to send value');
944           } else {
945               IBEP20(tokenAddress).transfer(owner(), tokenAmount);
946           }
947       }
```

Ether and tokens are the basis of smart contracts on which the contract runs and executes transactions. Therefore, it is absolutely necessary to

have input and access control validations on the functions executing funds withdrawal within the contract. The following unprotected public and external functions were found which were accepting addresses controlled by external users.

Recommendation:

It is recommended to go through the functions and make sure that the ether withdrawal implements an access control, input validation, and/or that the funds of the user is depreciated after they withdraws the amount.

4) Third-Party Dependency:

The contract BaseReward.sol is serving as the underlying entity to interact with third parties antisnipe and liquidityRestrictor protocols. The scope of the audit treats 3rd party entities as black boxes and assumes their functional correctness. However, in the real world, 3rd parties can be compromised and this may lead to lost or stolen assets. Moreover, the addresses of these third parties can be updated through functions setAntisnipeAddress() and setLiquidityRestrictionAddress().

```
770     function setAntisnipeAddress(address addr) external onlyOwner {
771         antisnipe = IAntisnipe(addr);
772         emit AntisnipeAddressChanged(addr);
773     }
774
775     function setLiquidityRestrictionAddress(address addr) external onlyOwner {
776         liquidityRestrictor = ILiquidityRestrictor(addr);
777         emit LiquidityRestrictionAddressChanged(addr);
778     }
779
```

Recommendation:

We understand that the business logic of BaseReward.sol requires interaction with antisnipe and liquidityRestrictor. We encourage the team to constantly monitor the statuses of 3rd parties to mitigate the side effects when unexpected activities are observed

Conclusion

Low-severity issues exist within smart contracts. Smart contracts are free from any critical or high-severity issues.

NOTE: Please check the disclaimer above and note, that audit makes no statements or warranties on business model, investment attractiveness or code sustainability.

Soken Contact Info

Website: www.soken.io

Mob: (+1)416-875-4174

32 Britain Street, Toronto, Ontario, Canada

Telegram: @team_soken

GitHub: sokenteam

Twitter: @soken_team