BNB Game Box

Smart Contract Audit Report

SYSFIXED

12 Jan 2023

TABLE OF CONTENTS

AUDITED DETAILS

Audited Project

Project name	Token ticker	Blockchain
BNB Game Box	BNBGB	Binance Smart Chain

Addresses

Contract address	0x63696D3D0e0C426AE83161b847d59ea3Ec83008C
Contract deployer address	0x343b48C12Ea98f360766e1B5Ad5b086fC751Aaa3

Project Website

https://bnbgamebox.com/

Codebase

https://bscscan.com/address/0x63696D3D0e0C426AE83161b847d59ea3Ec83008C#code

SUMMARY

Looking for the ultimate gaming gift this holiday season? Look no further than BNB Game Box! Our revolutionary new platform allows earn PC games as rewards while holding the BNBGB tokens!

Contract Summary

Documentation Quality

BNB Game Box provides a very good documentation with standard of solidity base code.

- The technical description is provided clearly and structured and also dont have any high risk issue.

Code Quality

The Overall quality of the basecode is standard.

- Standard solidity basecode and rules are already followed by BNB Game Box with the discovery of several low issues.

Test Coverage

Test coverage of the project is 100% (Through Codebase)

Audit Findings Summary

- SWC-101 | It is recommended to use vetted safe math libraries for arithmetic operations consistently on lines 510, 519, 531, 552, 555, 571, 572, 593, 676, 679, 691, 695, 799, 806, 807, 809, 813 and 822.

CONCLUSION

We have audited the BNB Game Box project released on January 2023 to discover issues and identify potential security vulnerabilities in BNB Game Box Project. This process is used to find technical issues and security loopholes which might be found in the smart contract.

The security audit report provides a satisfactory result with some low-risk issues.

The issues found in the BNB Game Box smart contract code do not pose a considerable risk. The writing of the contract is close to the standard of writing contracts in general. The low-risk issues found are some arithmetic operation issues. It is recommended to use vetted safe math libraries for arithmetic operations consistently.

AUDIT RESULT

Article	Category	Description	Result
Default Visibility	SWC-100 SWC-108	Functions and state variables visibility should be set explicitly. Visibility levels should be specified consciously.	PASS
Integer Overflow and Underflow	SWC-101	If unchecked math is used, all math operations should be safe from overflows and underflows.	ISSUE FOUND
Outdated Compiler Version	SWC-102	It is recommended to use a recent version of the Solidity compiler.	PASS
Floating Pragma	SWC-103	Contracts should be deployed with the same compiler version and flags that they have been tested thoroughly.	PASS
Unchecked Call Return Value	SWC-104	The return value of a message call should be checked.	PASS
SELFDESTRUCT Instruction	SWC-106	The contract should not be self-destructible while it has funds belonging to users.	PASS
Reentrancy	SWC-107	Check effect interaction pattern should be followed if the code performs recursive call.	PASS
Assert Violation	SWC-110	Properly functioning code should never reach a failing assert statement.	PASS
Deprecated Solidity Functions	SWC-111	Deprecated built-in functions should never be used.	PASS
Delegate call to Untrusted Callee	SWC-112	Delegate calls should only be allowed to trusted addresses.	PASS
DoS (Denial of Service)	SWC-113 SWC-128	Execution of the code should never be blocked by a specific contract state unless required.	PASS
Race Conditions	SWC-114	Race Conditions and Transactions Order Dependency should not be possible.	PASS

Authorization through tx.origin	SWC-115	tx.origin should not be used for authorization.	PASS
Block values as a proxy for time	SWC-116	Block numbers should not be used for time calculations.	PASS
Signature Unique ID	SWC-117 SWC-121 SWC-122	Signed messages should always have a unique id. A transaction hash should not be used as a unique id.	PASS
Shadowing State Variable	SWC-119	State variables should not be shadowed.	PASS
Weak Sources of Randomness	SWC-120	Random values should never be generated from Chain Attributes or be predictable.	PASS
Incorrect Inheritance Order	SWC-125	When inheriting multiple contracts, especially if they have identical functions, a developer should carefully specify inheritance in the correct order. The rule of thumb is to inherit contracts from more /general/ to more /specific/.	PASS

SMART CONTRACT ANALYSIS

Started	Wednesday Jan 11 2023 05:01:12 GMT+0000 (Coordinated Universal Time)
Finished	Thursday Jan 12 2023 18:55:01 GMT+0000 (Coordinated Universal Time)
Mode	Standard
Main Source File	BNBGB.sol

Detected Issues

ID	Title	Severity	Status
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+=" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+=" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+=" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "-" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "*" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "/" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged

SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "+" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "/" DISCOVERED	low	acknowledged
SWC-101	ARITHMETIC OPERATION "/" DISCOVERED	low	acknowledged

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 510

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
509
510   _beforeTokenTransfer(address(0), account, amount);
511
512   _totalSupply += amount;
513   _balances[account] += amount;
514
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 519

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
518
519   function _burn(address account, uint256 amount) internal virtual {
520   require(account != address(0), "ERC20: burn from the zero address");
521
522   _beforeTokenTransfer(account, address(0), amount);
523
```

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 531

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
530
531   emit Transfer(account, address(0), amount);
532
533   _afterTokenTransfer(account, address(0), amount);
534   }
535
```

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 552

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
551   uint256 amount
552   ) internal virtual {}
553
554   function _afterTokenTransfer(
555   address from,
556
```

SWC-101 | ARITHMETIC OPERATION "+=" DISCOVERED

LINE 555

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
554   function _afterTokenTransfer(
555   address from,
556   address to,
557   uint256 amount
558   ) internal virtual {}
559
```

SWC-101 | ARITHMETIC OPERATION "+=" DISCOVERED

LINE 571

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
570
571   uint256 public  liquidityFeeOnBuy;
572   uint256 public  liquidityFeeOnSell;
573
574   uint256 public  marketingFeeOnBuy;
575
```

SWC-101 | ARITHMETIC OPERATION "+=" DISCOVERED

LINE 572

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
571   uint256 public  liquidityFeeOnBuy;
572   uint256 public  liquidityFeeOnSell;
573
574   uint256 public  marketingFeeOnBuy;
575   uint256 public  marketingFeeOnSell;
576
```

SWC-101 | ARITHMETIC OPERATION "-" DISCOVERED

LINE 593

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
592   event ExcludeFromFees(address indexed account, bool isExcluded);
593   event MarketingWalletChanged(address marketingWallet);
594   event SwapAndLiquify(uint256 tokensSwapped,uint256 bnbReceived,uint256
tokensIntoLiqudity);
595   event SwapAndSendMarketing(uint256 tokensSwapped, uint256 bnbSend);
596   event SwapTokensAtAmountUpdated(uint256 swapTokensAtAmount);
597
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 676

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
675   function isExcludedFromFees(address account) public view returns(bool) {
676   return _isExcludedFromFees[account];
677   }
678
679   function changeMarketingWallet(address _marketingWallet) external onlyOwner{
680
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 679

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
678
679   function changeMarketingWallet(address _marketingWallet) external onlyOwner{
680   require(_marketingWallet != marketingWallet,"Marketing wallet is already that
address");
681   require(_marketingWallet != address(0),"Marketing wallet cannot be the zero
address");
682   marketingWallet = _marketingWallet;
683
```

SWC-101 | ARITHMETIC OPERATION "*" DISCOVERED

LINE 691

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
690   require(!tradingEnabled, "Trading already enabled.");
691   tradingEnabled = true;
692   swapEnabled = true;
693   }
694
695
```

SWC-101 | ARITHMETIC OPERATION "/" DISCOVERED

LINE 695

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
694
695   //
??????????????????????????????????????????????????????????????????????????????????????????????
??
696   //
??????????????????????????????????????????????????????????????????????????????????????????????
??
697   //
??????????????????????????????????????????????????????????????????????????????????????????????
??
698   //
??????????????????????????????????????????????????????????????????????????????????????????????
??
699
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 799

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
798   0,
799   address(0xdead),
800   block.timestamp
801   );
802
803
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 806

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
805
806   function swapAndSendMarketing(uint256 tokenAmount) private {
807   uint256 initialBalance = address(this).balance;
808
809   address[] memory path = new address[](2);
810
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 807

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
806   function swapAndSendMarketing(uint256 tokenAmount) private {
807   uint256 initialBalance = address(this).balance;
808
809   address[] memory path = new address[](2);
810   path[0] = address(this);
811
```

SWC-101 | ARITHMETIC OPERATION "+" DISCOVERED

LINE 809

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
808
809   address[] memory path = new address[](2);
810   path[0] = address(this);
811   path[1] = uniswapV2Router.WETH();
812
813
```

SWC-101 | ARITHMETIC OPERATION "/" DISCOVERED

LINE 813

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
812
813   uniswapV2Router.swapExactTokensForETHSupportingFeeOnTransferTokens(
814   tokenAmount,
815   0,
816   path,
817
```

SWC-101 | ARITHMETIC OPERATION "/" DISCOVERED

LINE 822

low SEVERITY

This plugin produces issues to support false positive discovery within mythril.

Source File

- BNBGB.sol

Locations

```
821
822   payable(marketingWallet).sendValue(newBalance);
823
824   emit SwapAndSendMarketing(tokenAmount, newBalance);
825   }
826
```

DISCLAIMER

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you ("Customer" or the "Company") in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to, or relied upon by any person for any purposes, nor may copies be delivered to any other person other than the Company, without Sysfixed's prior written consent in each instance.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Sysfixed to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model, or legal compliance.

This is a limited report on our findings based on our analysis, in accordance with good industry practice as of 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.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing 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.

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 Sysfixed and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers, and other representatives) (Sysfixed) owe no duty of care.

ABOUT US

Sysfixed is a blockchain security certification organization established in 2021 with the objective to provide smart contract security services and verify their correctness in blockchain-based protocols. Sysfixed automatically scans for security vulnerabilities in Ethereum and other EVM-based blockchain smart contracts. Sysfixed a comprehensive range of analysis techniques—including static analysis, dynamic analysis, and symbolic execution—can accurately detect security vulnerabilities to provide an in-depth analysis report. With a vibrant ecosystem of world-class integration partners that amplify developer productivity, Sysfixed can be utilized in all phases of your project's lifecycle. Our team of security experts is dedicated to the research and improvement of our tools and techniques used to fortify your code.