RugFreeCoins Audit

Santa Staking Contract
Smart Contract Security Audit

December 9th, 2022

Contents

Audit details

Audited project
Santa Staking Contract Audit

Contract Address

0xB0C0E490c52C4F82fE8Aded69f591295F04bc904

Client contact

Santa Team

Blockchain

Binance Smart chain

Project website

https://santaclub.xyz

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 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 Rugfreecoins and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Rugfreecoins) owe no duty of care towards you or any other person, nor does Rugfreecoins 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 Rugfreecoins 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, Rugfreecoins hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Rugfreecoins, 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

Rugfreecoins was commissioned by the Santa Team to perform an audit of the smart contract.

https://bscscan.com/token/0xB0C0E490c52C4F82fE8Aded69f591295F04bc904

The focus of this audit is to verify that the smart contract is secure, resilient, and working according to the specifications.

The information in this report should be used to understand the risk exposure of the smart contract, project feasibility, and long-term sustainability, and as a guide to improving the smart contract's security posture by remediating the identified issues.

Contract details

Token contract details for 9th of December 2022

Contract name	SantaClubStaking
Contract address	0xB0C0E490c52C4F82fE8Aded69f591295F04bc904
Contract deployer address	0xB9C2A343Df040F38cff43af1Fa826B76b1F6F43E
Contract's current owner address	0xb9c2a343df040f38cff43af1fa826b76b1f6f43e

Contract code function details

No	Category	Item	Result
1	Coding conventions	BRC20 Token standards	pass
		compile errors	pass
		Compiler version security	pass
		visibility specifiers	pass
		Gas consumption	pass
		SafeMath features	pass
		Fallback usage	pass
		tx.origin usage	pass
		deprecated items	pass
		Redundant code	pass
		Overriding variables	pass
2	Function call audit	Authorization of function call	pass
		Low level function (call/delegate call) security	pass
		Returned value security	pass
		Self-destruct function security	pass
3	Business security	Access control of owners	pass
		Business logics	pass
		Business implementations	pass
4	Integer overflow/underflow		pass
5	Reentrancy		pass
6	Exceptional reachable state		pass
7	Transaction ordering dependence		pass
8	Block properties dependence		pass
9	Pseudo random number generator (PRNG)		pass
10	DoS (Denial of Service)		pass
11	Token vesting implementation		pass
12	Fake deposit		pass

13	**Event security**		**pass**

Contract description table

The below table represents the summary of the contracts and methods in the token contract. We scanned the whole contract and listed down all the Interfaces, functions, and implementations with their visibility and mutability.

Contract	Type	Bases		
L	Function Name	Visibility	Mutability	Modifiers
ISantaClubStaking	**Interface**			
L	getStakerTotalStakedNFTs	External ❗		NO❗
L	nftIDsOfOwner	External ❗		NO❗
SantaClubStaking	**Implementation**	**Ownable, IERC721 Receiver, Reentrancy Guard**		
L		External ❗	💵	NO❗
L		Public ❗	🛑	NO❗
L	getClaimableRewards	Public ❗		NO❗
L	nftIDsOfOwner	External ❗		NO❗
L	getStakerTotalStakedNFTs	External ❗		NO❗
L	getStakedAt	External ❗		NO❗
L	getStakerTotalClaimedRewards	External ❗		NO❗
L	getTotalUnclaimedRewards	External ❗		NO❗
L	getTotalClaimableRewards	External ❗		NO❗

L	getClaimableRewardsByIDs	External ❗		NO❗
L	getStakedAtByIDs	External ❗		NO❗
L	updateDailyReward	External ❗	🛑	onlyOwner
L	stakeNFTs	External ❗	🛑	nonReentrant
L	claimRewards	External ❗	🛑	nonReentrant
L	unstakeNFTs	External ❗	🛑	nonReentrant
L	getDailyRewardRate	External ❗		NO❗
L	_nftIDsOfOwner	Internal 🔒		
L	_unstake	Internal 🔒	🛑	
L	calculateDailyRewardRate	Internal 🔒		
L	_claimRewards	Internal 🔒	🛑	
L	onERC721Received	External ❗		NO❗
Ownable	**Implementation**	**Context**		
L		Public ❗	🛑	NO❗
L	owner	Public ❗		NO❗
L	_checkOwner	Internal 🔒		
L	renounceOwnership	Public ❗	🛑	onlyOwner
L	transferOwnership	Public ❗	🛑	onlyOwner
L	_transferOwnership	Internal 🔒	🛑	
IERC20	**Interface**			

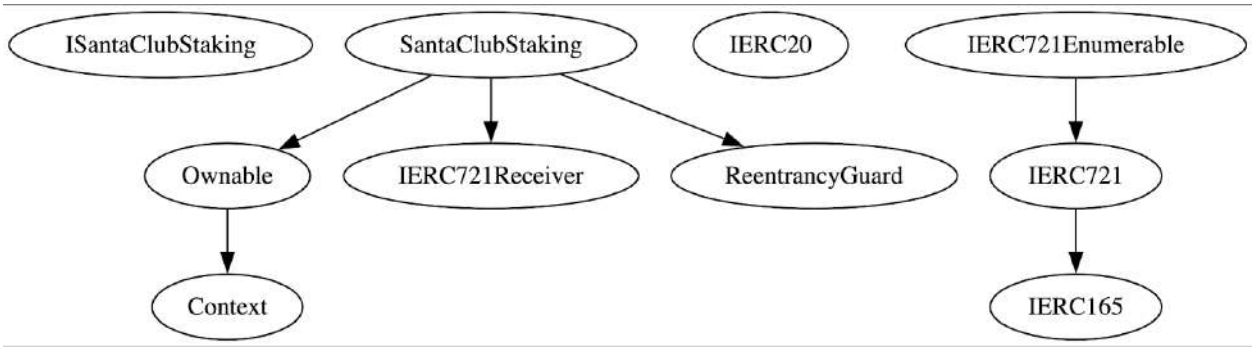
L	totalSupply	External ❗		NO❗
L	balanceOf	External ❗		NO❗
L	transfer	External ❗	🛑	NO❗
L	allowance	External ❗		NO❗
L	approve	External ❗	🛑	NO❗
L	transferFrom	External ❗	🛑	NO❗
IERC721Receiver	**Interface**			
L	onERC721Received	External ❗	🛑	NO❗
IERC721 Enumerable	**Interface**	**IERC721**		
L	totalSupply	External ❗		NO❗
L	tokenOfOwnerByIndex	External ❗		NO❗
L	tokenByIndex	External ❗		NO❗
ReentrancyGuard	**Implementation**			
L		Public ❗	🛑	NO❗
Context	**Implementation**			
L	_msgSender	Internal 🔒		
L	_msgData	Internal 🔒		

IERC721	Interface	IERC165		
L	balanceOf	External ❗		NO❗
L	ownerOf	External ❗		NO❗
L	safeTransferFrom	External ❗	🛑	NO❗
L	safeTransferFrom	External ❗	🛑	NO❗
L	transferFrom	External ❗	🛑	NO❗
L	approve	External ❗	🛑	NO❗
L	setApprovalForAll	External ❗	🛑	NO❗
L	getApproved	External ❗		NO❗
L	isApprovedForAll	External ❗		NO❗
IERC165	**Interface**			
L	supportsInterface	External ❗		NO❗

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

Inheritance Hierarchy



Security issue checking status

- **High severity issues**

(Informed and fixed)

Anyone can update the daily reward value: by updating the daily reward value anyone can drain the pool this function should only call by the owner

```
ftrace | funcSig
function updateDailyReward(uint256 amount) external {
    dailyReward = amount;
    emit UpdateDailyReward(amount);
}
```

- **Medium severity issues**
  No medium severity issues found

- **Low severity issues**

(Informed and fixed)

Out of gas error can trigger: in stakeNFTs function, users can submit any number of NFTs, if the user submits a large amount of NFTs it can trigger out of gas error. Hence, better to add validation to a number of NFTs that can submit at once.

```
ftrace | funcSig
function stakeNFTs(uint256[] calldata tokenIds) external nonReentrant {
    uint256 tokenId;
    uint256 stakedCount;
    for (uint256 i = 0; i < tokenIds.length; i++) {
        tokenId = tokenIds[i];
        require(
            nftStake[tokenId].owner == address(0),
            "Stake NFT: The NFT is already staked."
        );
        require(
            nftCollection.ownerOf(tokenId) == msg.sender,
            "Stake NFT: You do not own this NFT."
        );

        nftCollection.transferFrom(msg.sender, address(this), tokenId);

        nftStake[tokenId] = NFTStake({
            owner: msg.sender,
            stakedAt: block.timestamp
        });

        stakedCount += 1;

        emit NFTStaked(msg.sender, tokenId, block.timestamp);
    }

    totalStakedNFTs += stakedCount;
    staker[msg.sender].totalStakedNFTs += stakedCount;
}
```

Unwanted library imported
(Informed and fixed)

Contract imported ownable library. Still, it's not using any owner function, better to remove it and save the gas fees.

- **Centralization Risk**

(Informed and fixed)

Owner can change the daily reward without any minimum limitation (users will not get any rewards if this is set to 0)

```
ftrace | funcSig
function updateDailyReward(uint256 amount) external onlyOwner {
    dailyReward = amount;
    emit UpdateDailyReward(amount);
}
```

Owner privileges

- The owner can update the daily reward rate from 5 to 35

```
ftrace | funcSig
function updateDailyReward(uint256 amount) external onlyOwner {
    require(
        amount > 5,
        "Update Daily Reward: reward should be greater than 5 per day"
    );
    require(
        amount <= 35,
        "Update Daily Reward: reward should not be more than 35 per day"
    );
    dailyReward = amount;
    emit UpdateDailyReward(amount);
}
```

Audit conclusion

RugFreeCoins team has performed in-depth tests, line-by-line manual code review, and automated audit of the smart contract. The smart contract was analyzed mainly for common smart contract vulnerabilities, exploits, manipulations, and hacks. According to the smart contract audit.

Smart contract functional Status: **PASS**

Number of risk issues: **0**

Solidity code functional issue level: **PASS**

Number of owner privileges: **1**

Centralization risk correlated to the active owner: **NONE**

Smart contract active ownership: **ACTIVE**