

Audit Report

AnimeHatchery

January 2022

Type	BEP20
Network	BSC
Address	0x51c64E6D749fd42bE28600CB771E4dCC91A488b4
Audited by	© coinscope

Table of Contents

Table of Contents 1
Contract Review 3
Audit Updates 3
Contract Analysis 4
ELFM - Exceed Limit Fees Manipulation 5
Description 5
Recommendation 5
BC - Blacklisted Contracts 6
Description 6
Recommendation 6
Contract Diagnostics 7
L01 - Public Function could be Declared External 8
Description 8
Recommendation 8
L02 - State Variables could be Declared Constant 9
Description 9
Recommendation 9
L05 - Unused State Variable 10
Description 10
Recommendation 10
L04 - Conformance to Solidity Naming Conventions 11
Description 11
Recommendation 11
L09 - Dead Code Elimination 12
Description 12
Recommendation 12

Contract Functions 13
Contract Flow 17
Domain Info 18
Summary 19
Disclaimer 20
About Coinscope 21

Contract Review

Contract Name	AnimeHatchery
Compiler Version	v0.8.0+commit.c7dfd78e
Optimization	200 runs
Licence	None
Explorer	https://bscscan.com/token/0x51c64E6D749fd42bE28600CB771E4dCC91A488b4
Symbol	AHY
Decimals	18
Total Supply	5,000,000
Source	contract.sol
Domain	animehatchery.io

Audit Updates

Initial Audit	8th January 2022
Corrected	

Contract Analysis

● Critical ● Medium ● Minor ● Pass

Severity	Code	Description
●	ST	Contract Owner is not able to stop or pause transactions
●	OCTD	Contract Owner is not able to transfer tokens from specific address
●	OTUT	Owner Transfer User's Tokens
●	ELFM	Contract Owner is not able to increase fees more than a reasonable percent (25%)
●	ULTW	Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent
●	MT	Contract Owner is not able to mint new tokens
●	BT	Contract Owner is not able to burn tokens from specific wallet
●	BC	Contract Owner is not able to blacklist wallets from selling

ELFM - Exceed Limit Fees Manipulation

Criticality	critical
Location	AnimeHatchery.sol#L154

Description

The contract owner has the authority to increase over the allowed limit of 25%. The owner may take advantage of it by calling the `setFees` function with a high percentage value.

```
function setFees(uint256 bnbFee, uint256 marketingFee) external onlyOwner() {
    require(_BNBFee != bnbFee || _marketingFee != marketingFee);
    _BNBFee = bnbFee;
    _marketingFee = marketingFee;
    emit UpdateFees(bnbFee, marketingFee);
}
```

Recommendation

The contract could embody a check for the maximum acceptable value.

The team should carefully manage the private keys of the owner's account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. That risk can be prevented by temporarily locking the contract or renouncing ownership.

BC - Blacklisted Contracts

Criticality	medium
Location	AnimeHatchery.sol#L217

Description

The contract owner has the authority to stop contracts from transactions. The owner may take advantage of it by calling the `blacklistAddress` function.

```
require(!_isBlacklisted[from] && !_isBlacklisted[to], 'Blacklisted address');
```

Recommendation

The team should carefully manage the private keys of the owner's account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. That risk can be prevented by temporarily locking the contract or renouncing ownership.

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	L01	Public Function could be Declared External
	L02	State Variables could be Declared Constant
	L05	Unused State Variable
	L04	Conformance to Solidity Naming Conventions
	L09	Dead Code Elimination

L01 - Public Function could be Declared External

Criticality	minor
Location	AnimeHatchery.sol#L49,L42,L38 and 8 more

Description

Public functions that are never called by the contract should be declared external to save gas.

```
unlock
lock
geUnlockTime
...
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	AnimeHatchery.sol#L44,L23,L26 and 2 more

Description

Constant state variables should be declared constant to save gas.

```
snipeBlockAmount
_tTotal
_symbol
...
```

Recommendation

Add the constant attribute to state variables that never change.

L05 - Unused State Variable

Criticality	minor
Location	AnimeHatchery.sol#L45,L24

Description

There are segments that contains unused state variable.

```
lastTxTimes
MAX
```

Recommendation

Remove unused state variables.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	AnimeHatchery.sol#L7,L32,L29 and 2 more

Description

Solidity defines a naming convention that should be followed. Rule exceptions:

- Allow constant variable name/symbol/decimals to be lowercase.
- Allow _ at the beginning of the mixed_case match for private variables and unused parameters.

```
WETH
_marketingFee
_BNBFee
...
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions

L09 - Dead Code Elimination

Criticality	minor
Location	Context.sol#L33,L45,L74 and 6 more

Description

Functions that are not used in the contract, and make the code's size bigger.

```
trySub
tryMul
tryMod
...
```

Recommendation

Remove unused functions.

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
AnimeHatchery	Implementation	Context, IBEP20, Ownable		
	<Constructor>	Public	✓	-
	symbol	External		-
	name	External		-
	decimals	External		-
	totalSupply	External		-
	getOwner	External		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	setBNBPoolAddress	External	✓	onlyOwner
	setMarketingAddress	External	✓	onlyOwner
	setCoolDownTime	External	✓	onlyOwner
	excludeFromFee	External	✓	onlyOwner
	includeInFee	External	✓	onlyOwner
	setFees	External	✓	onlyOwner
	setNumTokensToSwap	External	✓	onlyOwner
	setSwapEnabled	External	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	_getFeeValues	Private		
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	isExcludedFromFee	External		-

	_approve	Private	✓	
	_transfer	Private	✓	
	swapAndCharge	Private	✓	lockTheSwap
	blacklistAddress	External	✓	onlyOwner
	swapTokensForEth	Private	✓	
	_tokenTransfer	Private	✓	
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
IBEP20	Interface			
	totalSupply	External		-
	decimals	External		-
	symbol	External		-
	name	External		-
	getOwner	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IPancakeswap V2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IPancakeswap V2Router01	Interface			

	factory	External		-
	WETH	External		-
	addLiquidity	External	✓	-
	addLiquidityETH	External	Payable	-
	removeLiquidity	External	✓	-
	removeLiquidityETH	External	✓	-
	removeLiquidityWithPermit	External	✓	-
	removeLiquidityETHWithPermit	External	✓	-
	swapExactTokensForTokens	External	✓	-
	swapTokensForExactTokens	External	✓	-
	swapExactETHForTokens	External	Payable	-
	swapTokensForExactETH	External	✓	-
	swapExactTokensForETH	External	✓	-
	swapETHForExactTokens	External	Payable	-
	quote	External		-
	getAmountOut	External		-
	getAmountIn	External		-
	getAmountsOut	External		-
	getAmountsIn	External		-
IPancakeswapV2Router02	Interface	IPancakeswapV2Router01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner

	transferOwnership	Public	✓	onlyOwner
	geUnlockTime	Public		-
	lock	Public	✓	onlyOwner
	unlock	Public	✓	-
SafeMath	Library			
	tryAdd	Internal		
	trySub	Internal		
	tryMul	Internal		
	tryDiv	Internal		
	tryMod	Internal		
	add	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	mod	Internal		
	sub	Internal		
	div	Internal		
	mod	Internal		

Contract Flow



Domain Info

Domain Name	animehatchery.io
Registry Domain ID	4eb0d59d3d37408fa1009d028f44b4dd-DONUTS
Creation Date	2021-12-02T10:48:08Z
Updated Date	2021-12-07T10:49:05Z
Registry Expiry Date	2022-12-02T10:48:08Z
Registrar WHOIS Server	whois.namecheap.com
Registrar URL	https://www.namecheap.com/
Registrar	NameCheap, Inc.
Registrar IANA ID	1068

The domain has been created about 1 month before the creation of the audit. It will expire in 11 months.

There is no public billing information, the creator is protected by the privacy settings.

Summary

Anime Hatchery is aiming to create a Decentralized Community-Driven NFT-based online video game. The token has a friendly and growing community. There are some functions that can be abused by the owner, like manipulating fees and blacklisting addresses. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

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.

Coinscope 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 Coinscope 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 Coinscope 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 Coinscope team disclaims any liability for the resulting losses.

About Coinscope

Coinscope is the leading early coin listing, voting and auditing authority firm. The audit process is analyzing and monitoring many aspects of the project. That way, it gives the community a good sense of security using an informative report and a generic score.

Coinscope is aiming to make crypto discoverable and efficient globally. It provides all the essential tools to assist users draw their own conclusions.



The Coinscope.co team

https://www.coinscope.co