Cyberscope

Audit Report

Locker

July 2022

SHA256 5cf6ae2c2c0360240e3245861ab154cd14724c605ee8494067e808d24cbafb86

Audited by © cyberscope

Table of Contents

Contract Review

Contract Name	Locker
Test Deploy	https://testnet.bscscan.com/address/0x4CcBA70f1e46cfa1D1E38d31D006a6a851527F50#code
Domain	https://hyfinance.net

Audit Updates

Initial Audit	15th July 2022
Corrected	21th July 2022

Source Files

Filename	SHA256
@openzeppelin/contracts/access/Ownable.sol	754825f501dd014526eee0c415687b0f6c600533adfc872f7d45edb4f8b3b053
@openzeppelin/contracts/math/SafeMath.sol	f6d6214aa03f8dd6d6d14b7c15ffa387b3f1ce38ba3a215177baa132a44636e2
@openzeppelin/contracts/token/ERC20/IERC20.sol	c4b741712b8dc93ab3945205554a3ba2f80953e64d684e752d5a0fd07fc93f22
@openzeppelin/contracts/token/ERC20/SafeERC20.sol	74e10f4538df92e1c89140f16654914be8d7e9a66b24d6272ff0f28f89f8728b
@openzeppelin/contracts/utils/Address.sol	a22903d00a93aa211164d90ad11f01ccc7d34648114be89ec38c859fdea0f8d4
@openzeppelin/contracts/utils/Context.sol	eafb62c654640a07832b56e00902b4bf249633346585331af311c738b1c23bc5
@openzeppelin/contracts/utils/Pausable.sol	e59e348bb0a6a4a7f5f88896f6a1b9f151b9857bf362bb2aa431b910ee579eea
@openzeppelin/contracts/utils/ReentrancyGuard.sol	a84a635e520d932183fc216c6f0ec109f8578149b15a91c728557a370430882a
contracts/interfaces/IERC20Meta.sol	6d83cc8a7eb156aec4ac633bfe9d8bcc330654dddbecc6601f78bfafe9abb064
contracts/interfaces/IInfinityPool.sol	deb9472d20dcc210ccf6e699f6c5bf8471b8bf4bd77bca6345a3155b2e09564b

contracts/Locker.sol	5cf6ae2c2c0360240e3245861ab154cd14724c605ee8494067e808d24cbafb86

Introduction

The Locker contract implements three core functionalities of the Hybrid Finance ecosystem. The main methods are lock, unlock, and increaseLockAmount of liquidity that can be held by Hybrid Finance.

- The user has the ability to lock Hybrid Finance version 2 tokens. They receive the exact amount of Hybrid Finance version 1 token.
- The user has the ability to unlock the token when the lock period elapses. In order to unlock the Hybrid Finance version 2 tokens, the user has to provide the corresponding Hybrid Finance version 1 token. The reward of Hybrid Finance version 2 tokens are returned to the user in relation to the ratio of the balance of the contract and the total Hybrid Finance version 1 tokens. If the contract owner withdraws tokens, then the user will take less tokens in relation with the initial locked amount.
- The user can unlock the Hybrid Finance version 2 tokens only if he has the corresponding Hybrid Finance version 1 tokens.
- The user can lock tokens once but he has the ability to increase the lock amount. If the user increases the lock amount, then the locked time will be increased proportionally.

Contract Diagnostics

🔴 Critical 🟠 Medium ⚪ Minor

Severity	Code	Description	Status
⚪	ST	Contract Owner is not able to stop or pause transactions	Multi-Sign
🟠	OCTD	Contract Owner is not able to transfer tokens from specific address	Multi-Sign
⚪	UAV	Unlock Amount Volatilisation	
⚪	L04	Conformance to Solidity Naming Conventions	Acknowledged

ST - Stop Transactions

Criticality	minor
Location	contract.sol#L127,L159
Status	Multi-Sign

Description

The contract owner has the authority to pause transactions for all users. The owner may take advantage of it by using the `pause` function.

```
function lock(uint256 amount, uint256 risk) external unlockPeriod whenNotPaused nonReentrant
function unlock() external unlockPeriod whenNotPaused nonReentrant
```

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.

Updated 20 July 2022

The team has acknowledged that thread and transferred the contract ownership to a multi-sign mechanism.

OCTD - Owner Contract Tokens Drain

Criticality	medium
Location	contract.sol#L196
Status	Multi-Sign

Description

The contract owner has the authority to claim all the balance of the contract. The owner may take advantage of it by calling the adminWithdraw function.

```
function adminWithdraw(address token, uint256 amount) external onlyOwner {
	require(IERC20Meta(token).balanceOf(address(this)) >= amount, "Amount too high");
	IERC20Meta(token).safeTransfer(msg.sender, amount);
}
```

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.

Updated 20 July 2022

The team has acknowledged that thread and transferred the contract ownership to a multi-sign mechanism.

UAV - Unlock Amount Volatilisation

Criticality	minor
Location	contract.sol#L174

Description

The unlock amount is calculated based on a rate. The rate divides the contract hybridv2 balance in comparison with the minted tokens. The minted tokens counter and the contract’s hybridv2 balance are increased on every lock. The contract’s balance can also be withdrawn by the contract owner. If the contract’s hybridv2 balance decreases, then the rate will yield values less than one. As a result, the users may take less tokens in relation to the initial locked amount that they invest.

```
uint256 hybridAmount = veAmount.mul(currentRatio).div(1e18);
```

Recommendation

The unlocked amount could be independent from the contract’s balance.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contracts/Locker.sol#L80,91,74,69,86

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.

```
_lockDuration
_infinityPool
_unlockStart
_maxRatio
_unlockDuration
```

Recommendation

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

Updated 20 July 2022

The team has acknowledged that it is not a security issue.

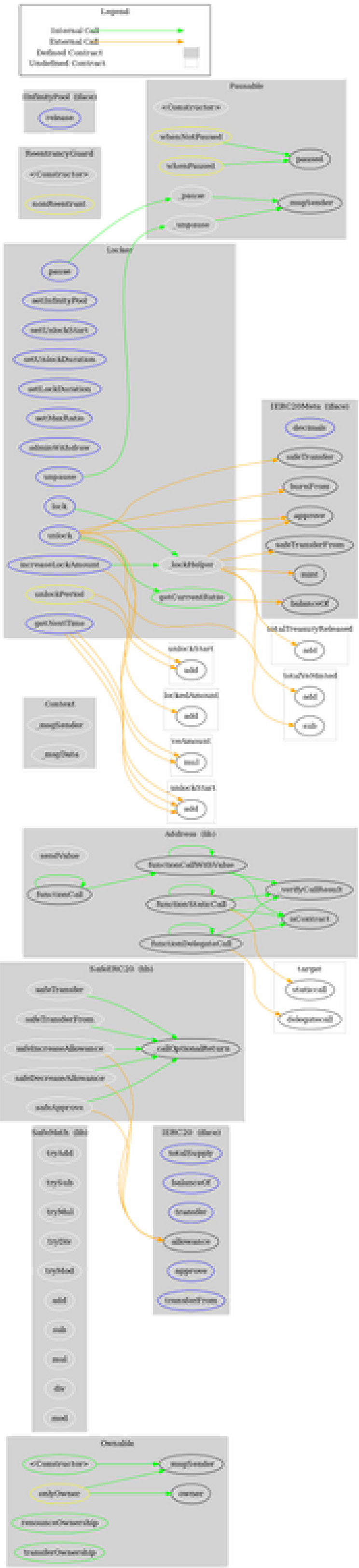
Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
Ownable	Implementation	Context		
	<Constructor>	Internal	✓	
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
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		
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-

SafeERC20	Library			
	safeTransfer	Internal	✓	
	safeTransferFrom	Internal	✓	
	safeApprove	Internal	✓	
	safeIncreaseAllowance	Internal	✓	
	safeDecreaseAllowance	Internal	✓	
	_callOptionalReturn	Private	✓	
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	_verifyCallResult	Private		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Pausable	Implementation	Context		
	<Constructor>	Internal	✓	
	paused	Public		-
	_pause	Internal	✓	whenNotPaused
	_unpause	Internal	✓	whenPaused
ReentrancyGuard	Implementation			

	<Constructor>	Internal	✓	
IERC20Meta	Interface	IERC20		
	decimals	External		-
	burnFrom	External	✓	-
	mint	External	✓	-
IInfinityPool	Interface			
	release	External	✓	-
Locker	Implementation	Ownable, Pausable, ReentrancyGuard		
	<Constructor>	Public	✓	-
	setInfinityPool	External	✓	onlyOwner
	setUnlockStart	External	✓	onlyOwner
	setUnlockDuration	External	✓	onlyOwner
	setLockDuration	External	✓	onlyOwner unlockPeriod
	setMaxRatio	External	✓	onlyOwner unlockPeriod
	adminWithdraw	External	✓	onlyOwner
	pause	External	✓	onlyOwner
	unpause	External	✓	onlyOwner
	getCurrentRatio	Public		-
	_lockHelper	Internal	✓	
	lock	External	✓	unlockPeriod whenNotPaused nonReentrant
	increaseLockAmount	External	✓	unlockPeriod whenNotPaused nonReentrant
	unlock	External	✓	unlockPeriod whenNotPaused nonReentrant
	getNextTime	External		-

Contract Flow



Domain Info

Domain Name	hyfinance.net
Registry Domain ID	2683607355_DOMAIN_NET-VRSN
Creation Date	2022-03-22T21:24:53.00Z
Updated Date	0001-01-01T00:00:00.00Z
Registry Expiry Date	2023-03-22T21:24:53.00Z
Registrar WHOIS Server	whois.namecheap.com
Registrar URL	http://www.namecheap.com
Registrar	NAMECHEAP INC
Registrar IANA ID	1068

The domain has been created in 8 months before the creation of the audit.

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

Summary

The locker implements a typical functionality of locking tokens for a period of time. This lockerer works like a fixed staking contract but it does not reward the users with extra tokens. There are some functions that can be abused by the owner like stopping transactions and transferring tokens to the team's wallet. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

Updated 20 July 2022

The team has transferred the contract ownership to a multi-sign mechanism.

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.

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

About Cyberscope

Coinscope audit and K.Y.C. service has been rebranded to Cyberscope.

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.

Cyberscope and Coinscope are aiming to make crypto discoverable and efficient globally. They provide all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io