Cyberscope

Audit Report

Lava IDO

April 2022

File	LavaIDO.sol
Commit	d59617e3ac107eea6d7601aac6e73e7f45ee00eb
Github	https://github.com/lavafinancial/LavaContracts
Audited by	© cyberscope

Table of Contents

Contract Review

Github	LavaIDO
commit	d59617e3ac107eea6d7601aac6e73e7f45ee00eb
File	LavaFinance.sol

Audit Updates

Initial Audit	9th April 2022
Corrected	

Source Files

Filename	SHA256
@openzeppelin/contracts/access/Ownable.sol	75e3c97011e75627ffb36f4a2799a4e887e1a3e27ed427490e82d7b6f51cc5c9
@openzeppelin/contracts/token/ERC20/extensions/IERC20Metadata.sol	af5c8a77965cc82c33b7ff844deb9826166689e55dc037a7f2f790d057811990
@openzeppelin/contracts/token/ERC20/IERC20.sol	c2b06bb4572bb4f84bfc5477dadc0fcc497cb66c3a1bd53480e68bedc2e154a6
@openzeppelin/contracts/token/ERC20/utils/SafeERC20.sol	b5a1340c5232f387b15592574f27eef78f6017bdc66542a1cea512ad4f78a0d2
@openzeppelin/contracts/utils/Address.sol	aafa8f3e41700a8353aabcdf020e06735753e6bc4b615279b43de53cfbb4f2cd
@openzeppelin/contracts/utils/Context.sol	1458c260d010a08e4c20a4a517882259a23a4baa0b5bd9add9fb6d6a1549814a
contracts/interfaces/ALAVA.sol	c4e418e0713a28c28f9f2d6793532d9bbe29735573ff53ca10d96ad3eae4b533
contracts/LavaIDO.sol	8c27d935b5a03cea9645d543cd5a6f76e5ca818759a42b10df20605c9c274d8e

Contract Analysis

IDO

- Users have the ability to buy plava and alava tokens by providing the usdc tokens.
- The price of the IDO phase is 1 usdc for 2 tokens.
- The users have the ability to choose the ratio of plava and alava that they will receive.
- There is a minimum and a maximum amount of tokens that the user can submit.
- Only whitelisted users can participate in the IDO process.

Convert

- Users have the ability to convert their aLava to Lava tokens according to a conversion rate.
- During the conversion process the aLava tokens are burned and the equivalent Lava tokens are moved from the IDO contract to the user.

Admin Privileges

- The aLava to Lava conversion rate is configured by the contract owner.
- The contract owner has the ability to change the deadline that the conversion method will be available.
- The contract owner can manipulate the whitelist.
- The contract owner has the ability to withdraw all the funds of the contract.

Notes

- The IDO contract should have the sufficient funds of pLava and aLava tokens in order to support the IDO process.
- The IDO contract should have sufficient Lava tokens in order to support the conversions.

Contract Diagnostics

● Critical ● Medium ● Minor

Severity	Code	Description
●	RE	Reentrant
●	MC	Missing Check
●	L01	Public Function could be Declared External
●	L04	Conformance to Solidity Naming Conventions
●	L13	Divide before Multiply Operation

RE - Reentrance

Criticality	minor
Location	contract.sol#L52,70

Description

Both buy and convertToLava methods are based on the fact that the user's balance is sufficient in order to proceed with the transaction. After the transfer the bought balances are updated. This is a potential re-entrance pattern.

```
function buy(uint256 amount, uint256 pLavaRatio) external {
```

```
function convertToLava() external {
```

Recommendation

The contract could use a reentrance guard in order to ensure that the functions are called once every time.

MC - Missing Check

Criticality	minor
Location	contract.sol#L1

Description

The safe transfer technique checks if the result of the transfer is successful. This is usually the caller's responsibility to call the pure transfer and check the result. The caller should not relly in the callee’s wrapped functions like safeTransfer. This may break the IDO business logic since it may assume that the transfer has been accomplished even if it did not.

```
pLavaToken.safeTransfer(msg.sender, pLavaAmount);
if (aLavaAmount > 0) {
    aLavaToken.safeTransfer(msg.sender, aLavaAmount);
}
```

```
lavaToken.transfer(msg.sender, lavaAmount);
```

Recommendation

The contract should validate if the result of the transfer functions are successfully.

L01 - Public Function could be Declared External

Criticality	minor
Location	contracts/LavaIDO.sol#L96

Description

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

```
adminWithdrawAvax
```

Recommendation

Use the external attribute for functions never called from the contract

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contracts/LavaIDO.sol#L38,42,47,81,85,25,26

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.

```
maxAmountLimit
minAmountLimit
_status
_users
_user
_factor
_lavaToken
_convertDeadline
```

Recommendation

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

L13 - Divide before Multiply Operation

Criticality	minor
Location	contracts/LavaIDO.sol#L52

Description

Performing divisions before multiplications may cause lose of prediction.

```
totalLavaAmount = (amount * 2 * (10 ** aLavaToken.decimals())) / (10 **
usdce.decimals())
```

Recommendation

The multiplications should be prior to the divisions.

Unit Test

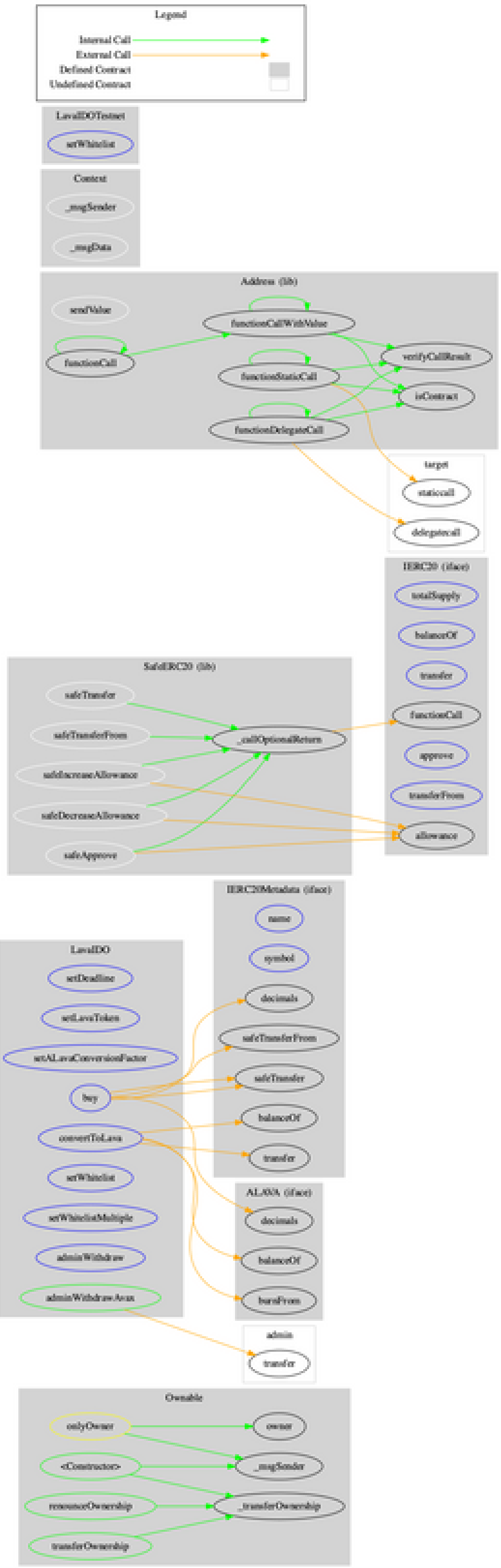
- ✓ Test lava non-whitelist
- ✓ Test lava buy limit (62ms)
- ✓ Test lava buy 50/50 (46ms)
- ✓ Test full plava buy (48ms)
- ✓ Test alava convert (118ms)
- ✓ Test alava convert 50% (131ms)
- ✓ Test withdraw (85ms)

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
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	Internal		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
ALAVA	Interface	IERC20Metadata		
	burnFrom	External	✓	-
LavaIDO	Implementation	Ownable		
	<Constructor>	Public	✓	-
	setDeadline	External	✓	onlyOwner
	setLavaToken	External	✓	onlyOwner
	setALavaConversionFactor	External	✓	onlyOwner
	buy	External	✓	-
	convertToLava	External	✓	-
	setWhitelist	External	✓	onlyOwner
	setWhitelistMultiple	External	✓	onlyOwner
	adminWithdraw	External	✓	onlyOwner
	adminWithdrawAvax	Public	✓	onlyOwner
LavaIDOTestnet	Implementation	LavaIDO		
	<Constructor>	Public	✓	LavaIDO
	setWhitelist	External	✓	-

Contract Flow



Summary

The Lava IDO contract gives the ability to the users to buy aLava and pLava tokens by providing USDC. After the IDO phase, the users can convert their aLava for Lava tokens. This audit focuses on the business logic, performance improvements, security concerns and potential optimizations.

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 provides all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io