Cyberscope

Audit Report

Lava Oracle

April 2022

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

Table of Contents

Contract Review

Github	LavaFinance
commit	d59617e3ac107eea6d7601aac6e73e7f45ee00eb
File	Oracle.sol

Audit Updates

Initial Audit	9th April 2022
Corrected	

Source Files

Filename	SHA256
@openzeppelin/contracts/access/Ownable.sol	75e3c97011e75627ffb36f4a2799a4e887e1a3e27ed427490e82d7b6f51cc5c9
@openzeppelin/contracts/token/ERC20/IERC20.sol	c2b06bb4572bb4f84bfc5477dadc0fcc497cb66c3a1bd53480e68bedc2e154a6
@openzeppelin/contracts/utils/Context.sol	1458c260d010a08e4c20a4a517882259a23a4baa0b5bd9add9fb6d6a1549814a
contracts/interfaces/Pair.sol	c976359741ad850af98ccec9bce5fd6d6a2ede9a3d366f5889a245a8c90aab28
contracts/lib/Babylonian.sol	074426507d1e75fe16687010fef4b66c00cab4c0d73774947a6ee797c6dbbf29
contracts/lib/FixedPoint.sol	599039d717b75eea4fe19fbac01638380cfdfece43b372e125c24d75a7b6bfd8
contracts/lib/UniswapV2OracleLibrary.sol	61e600bbdbad3588779839933a2347819e40277ea04d1d723d0cc59c0e5cc67d
contracts/Oracle.sol	0351c0b9364df4bf12f4f48b23b1a98cc8e7ca2e0f54894d099076575aa1fdda

Contract Analysis

The Lava Oracle is trying to mirror the oracle mechanism that was initially introduced in the pairing contract of the Unswap v2.

You can read more about the fundamental Oracle principals in the Uniswap v2 Core whitepaper, section 2.2 Price oracle

According to the whitepaper, the estimated price is a result of the price difference in a specific time frame. This logic is implemented in the pair contracts of all the decentralised exchanges. You can check the core-v2 implementation of the Uniswap repository.

Since the pair already contains the price oracle mechanism, then the Lava price oracle could reuse this functionality instead of replicating it. That way, potential synchronization issues with the original price will be eliminated.

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	FSA	Fixed Swap Address
	L01	Public Function could be Declared External
	L02	State Variables could be Declared Constant
	L04	Conformance to Solidity Naming Conventions
	L07	Missing Events Arithmetic

FSA - Fixed Swap Address

Criticality	minor
Location	contract.sol#L41

Description

The swap address is assigned once in the constructor and it can not be changed. The decentralized swaps sometimes create a new swap version, abandon the current or move to a different pair. A contract that cannot change the swap address may not be able to catch-up the upgrade.

```
pair = _pair;
token0 = pair.token0();
token1 = pair.token1();
price0CumulativeLast = pair.price0CumulativeLast(); // fetch the current
accumulated price value (1 / 0)
price1CumulativeLast = pair.price1CumulativeLast(); // fetch the current
accumulated price value (0 / 1)
```

Recommendation

It could be better to allow the swap address mutation in case of future swap updates.

L01 - Public Function could be Declared External

Criticality	minor
Location	contracts/Oracle.sol#L79

Description

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

```
consult
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contracts/Oracle.sol#L22,23

Description

Constant state variables should be declared constant to save gas.

```
token1
token0
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contracts/Oracle.sol#L54,79,90

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.

```
_amount
_token
_amountIn
_period
```

Recommendation

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

L07 - Missing Events Arithmetic

Criticality	minor
Location	contracts/Oracle.sol#L54

Description

Detected missing events for critical arithmetic parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.

```
period = _period
```

Recommendation

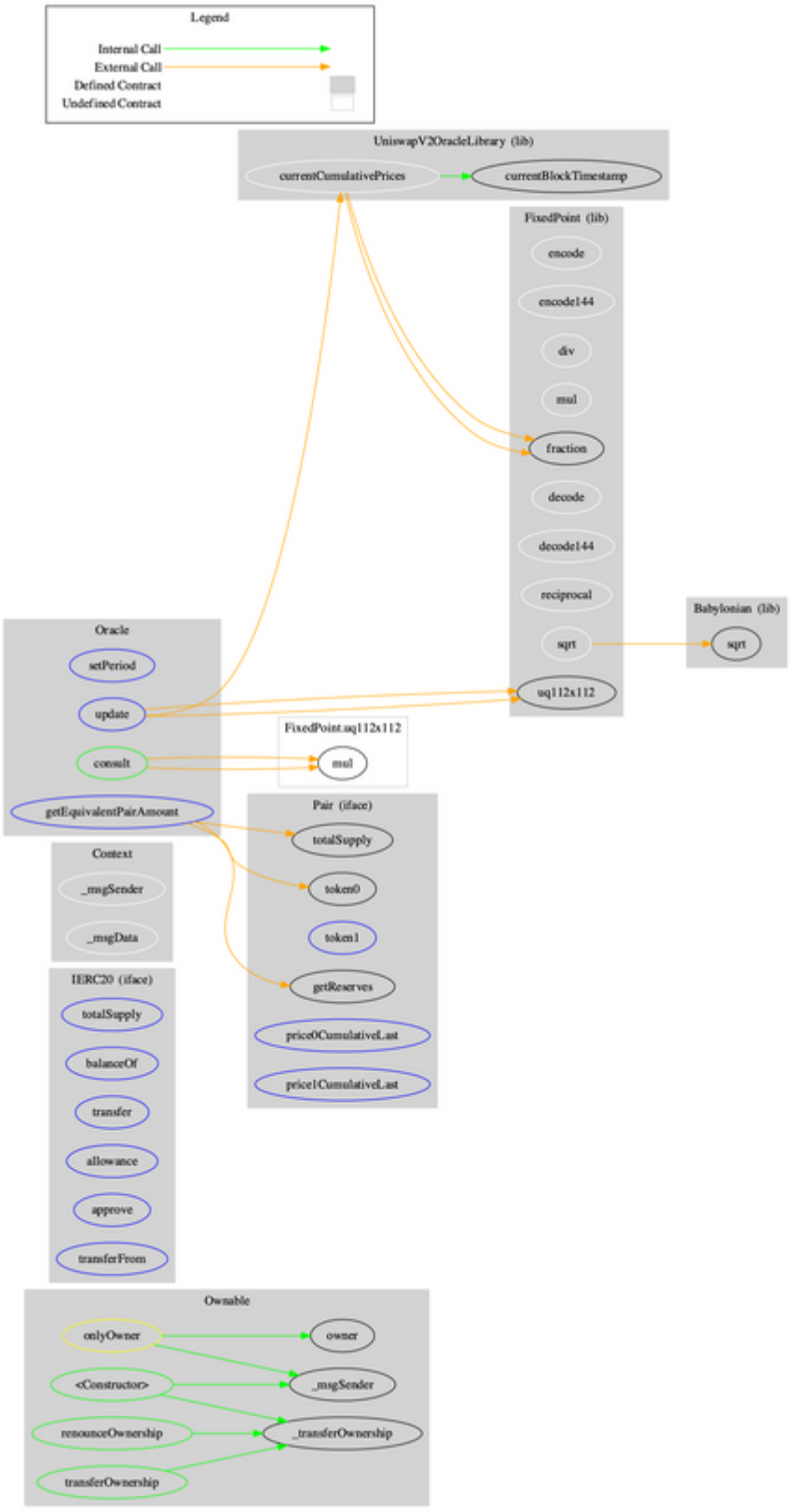
Emit an event for critical parameter changes.

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	✓	
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Pair	Interface			
	totalSupply	External		-
	token0	External		-
	token1	External		-
	getReserves	External		-
	price0CumulativeLast	External		-
	price1CumulativeLast	External		-
Babylonian	Library			
	sqrt	Internal		

FixedPoint	Library			
	encode	Internal		
	encode144	Internal		
	div	Internal		
	mul	Internal		
	fraction	Internal		
	decode	Internal		
	decode144	Internal		
	reciprocal	Internal		
	sqrt	Internal		
UniswapV2OracleLibrary	Library			
	currentBlockTimestamp	Internal		
	currentCumulativePrices	Internal		
Oracle	Implementation	Ownable		
	<Constructor>	Public	✓	-
	setPeriod	External	✓	onlyOwner
	update	External	✓	-
	consult	Public		-
	getEquivalentPairAmount	External		-

Contract Flow



Summary

The Lava Oracle is a mechanism that provides a price estimation by tracking the swap’s pair reserves. This audit mentions the initial price oracle fundamentals, discusses ways to reuse the core oracle functionality and suggested potential improvements.

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