COINSCOPE

Audit Report

LilHeroes

January 2022

Type	ERC-721
SHA256	5d8e8a39b99976fd27f96be848263f5d189cd4223f0b91772cf8bb1ebf586358
Audited by	© coinscope

Table of Contents

Audit Updates

SHA256	5d8e8a39b99976fd27f96be848263f5d189cd4223f0b91772cf8bb1ebf586358
Initial Audit	12th January 2022
Corrected	

Contract Analysis

● Critical ● Medium ● Minor

Severity	Code	Description
●	CR	Code Repetition
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L04	Conformance to Solidity Naming Conventions
●	L11	Unnecessary Boolean equality

CR - Code Repetition

Criticality	minor
Location	contract.sol#L192,L216,L235,L251

Description

There are code segments that are repetitive in the contract. Those segments increase the code size of the contract unnecessarily.

```
        for(uint256 i=totalSupply; i < (totalSupply + numToMint); i++){
            _mint(msg.sender, i+1);
        }
```

Recommendation

Create an internal function that contains the code segment and remove it from all the sections.

L01 - Public Function could be Declared External

Criticality	minor
Location	contract/contract.sol#L164,L150,L136 and 7 more

Description

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

```
safeTransferFrom
transferFrom
setApprovalForAll
...
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract/contract.sol#L14

Description

Constant state variables should be declared constant to save gas.

```
maxItems
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract/contract.sol#L179,L13,L19 and 17 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.

```
_data
mintTypes
__extensionURI
...
```

Recommendation

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

L11 - Unnecessary Boolean equality

Criticality	minor
Location	contract/contract.sol#L228,L205,L181

Description

The comparison to boolean constants is redundant. Boolean constants can be used directly and do not need to be compared to true or false.

```
require(bool,string)(publicIsActive == true,minting not activated)
require(bool,string)(raffleIsActive == true,minting not activated)
require(bool,string)(whitelistIsActive == true,minting not activated)
```

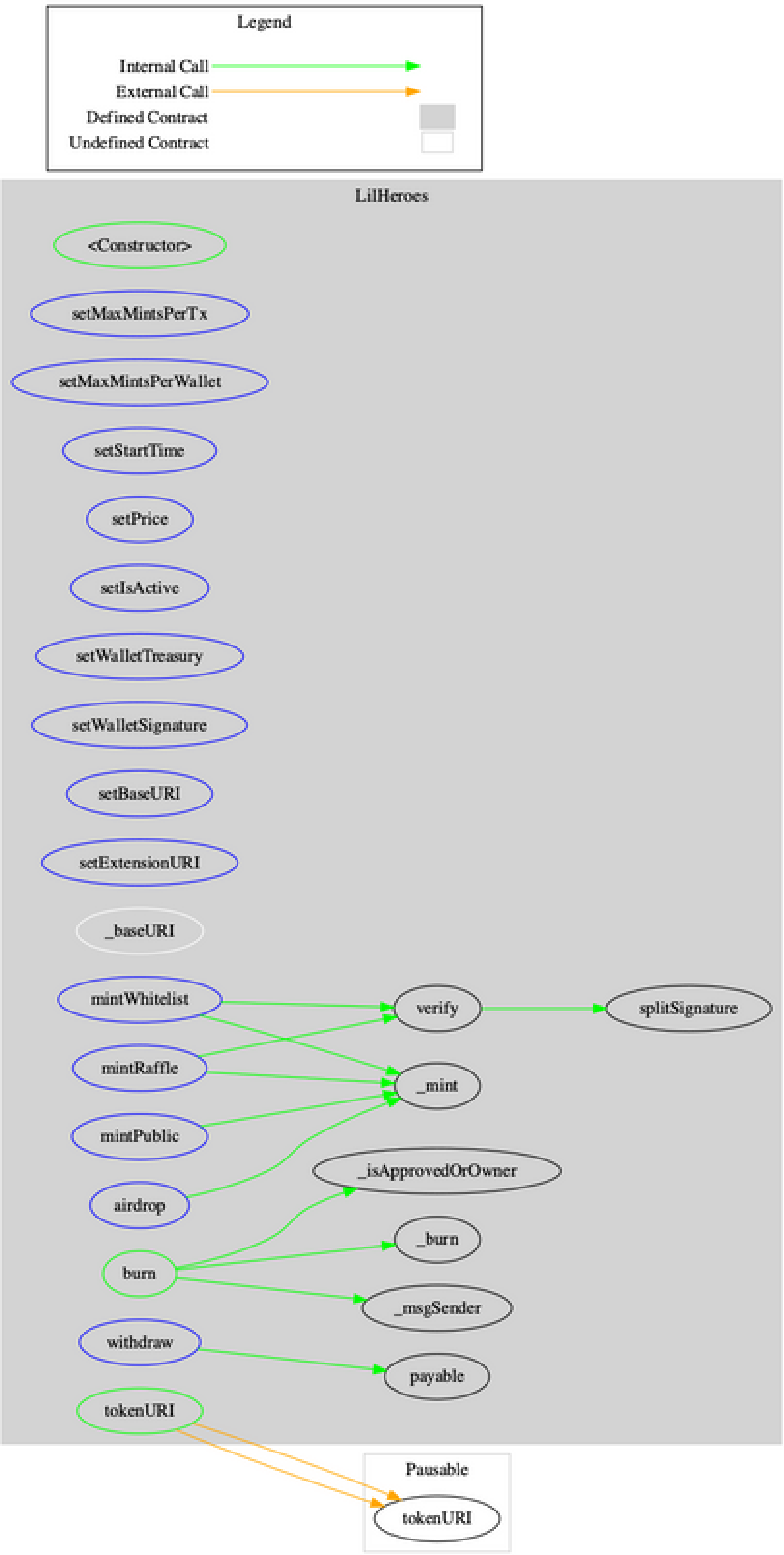
Recommendation

Remove the equality to the boolean constant.

Contract Functions

Contract	**Type**	**Bases**		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
LilHeroes	Implementation	ERC721URIStorage, Ownable, Pausable		
	<Constructor>	Public	✓	ERC721
	setMaxMintsPerTx	External	✓	onlyOwner
	setMaxMintsPerWallet	External	✓	onlyOwner
	setStartTime	External	✓	onlyOwner
	setPrice	External	✓	onlyOwner
	setIsActive	External	✓	onlyOwner
	setWalletTreasury	External	✓	onlyOwner
	setWalletSignature	External	✓	onlyOwner
	setBaseURI	External	✓	onlyOwner
	setExtensionURI	External	✓	onlyOwner
	_baseURI	Internal		
	tokenURI	Public		-
	mintWhitelist	External	Payable	-
	mintRaffle	External	Payable	-
	mintPublic	External	Payable	whenNotPaused
	airdrop	External	✓	onlyOwner
	verify	Public		-
	splitSignature	Public		-
	burn	Public	✓	-
	withdraw	External	Payable	onlyOwner

Contract Flow



Summary

LilHeroes is a project that implements the fundamental NFT functionality. The contract can mint new NFTs with four different ways:

1. Whitelist from specific wallets
2. Raffle from specific wallets
3. The standard public NFT mint
4. Airdrop that are minted from the contract owner

Whitelist and raffle share the same functionality. The Smart Contract analysis reported no compiler error or critical issues. There are some informative comments that do not affect the contract security.

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