SPYWOLF

Security Audit Report

Audit prepared for

LiftOff

Completed on

December 10, 2024

OVERVIEW

This goal of this report is to review the main aspects of the project to help investors make an informative decision during their research process.

You will find a a summarized review of the following key points:

- ✔ Contract's source code
- ✔ Owners' wallets
- ✔ Tokenomics
- ✔ Team transparency and goals
- ✔ Website's age, code, security and UX
- ✔ Whitepaper and roadmap
- ✔ Social media & online presence

The results of this audit are purely based on the team’s evaluation and does not guarantee nor reflect the projects outcome and goal

- SPYWOLF Team -

TABLE OF CONTENTS

Project Description	01
LiquidityPool.sol Stats & Review	02 -06
PoolFactory.sol Stats & Review	07-08
Token.sol Stats & Review	09-10
Website Analysis & Score	11
Social Media Review & Score	12
About SPYWOLF	13
Disclaimer	14

KEY RESULTS

Cannot mint new tokens	PASSED
Cannot pause trading (honeypot)	PASSED
Cannot blacklist an address	PASSED
Cannot raise taxes over 25%?	PASSED
No proxy contract detected	PASSED
Not required to enable trading	PASSED
No hidden ownership	PASSED
Cannot change the router	PASSED
No cooldown feature found	PASSED
Bot protection delay is lower than 5 blocks	PASSED
Cannot set max tx amount below 0.05% of total supply	PASSED
The contract cannot be self-destructed by owner	PASSED

For a more detailed and thorough examination of the heightened risks, refer to the subsequent parts of the report.

N/A = Not applicable for this type of contract

*Only new deposits/reinvestments can be paused

PROJECT DESCRIPTION:

LiftOff.monster is a decentralized token launch platform designed to revolutionize the introduction and interaction of digital assets. It leverages blockchain technology to provide a fun, interactive, and innovative method for token creation and distribution, all within a transparent and secure environment. The platform caters to a broad range of users, from developers to community participants, offering opportunities for token discovery and engagement.

Release Date: TBD

Category: Token creator

LiquidityPool.sol INFO

Token Name
Unavailable

Symbol
Unavailable

Contract Address
Unavailable

Network
Unavailable

Language
Solidity

Deployment Date
Unavailable

Contract Type
Liquidity pool

Total Supply
Unavailable

Decimals
Unavailable

TAXES

Buy Tax
1%

Sell Tax
1%

*Taxes cannot be changed

Our Contract Review Process

The contract review process pays special attention to the following:

- Testing the smart contracts against both common and uncommon vulnerabilities
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

Blockchain security tools used:

- OpenZeppelin
- Mythril
- Solidity Compiler
- Hardhat

VULNERABILITY ANALYSIS

ID	Title	
SWC-100	Function Default Visibility	**Passed**
SWC-101	Integer Overflow and Underflow	**Passed**
SWC-102	Outdated Compiler Version	**Passed**
SWC-103	Floating Pragma	**Passed**
SWC-104	Unchecked Call Return Value	**Passed**
SWC-105	Unprotected Ether Withdrawal	**Passed**
SWC-106	Unprotected SELFDESTRUCT Instruction	**Passed**
SWC-107	Reentrancy	**Passed**
SWC-108	State Variable Default Visibility	**Passed**
SWC-109	Uninitialized Storage Pointer	**Passed**
SWC-110	Assert Violation	**Passed**
SWC-111	Use of Deprecated Solidity Functions	**Passed**
SWC-112	Delegatecall to Untrusted Callee	**Passed**
SWC-113	DoS with Failed Call	**Passed**
SWC-114	Transaction Order Dependence	**Passed**
SWC-115	Authorization through tx.origin	**Passed**
SWC-116	Block values as a proxy for time	**Passed**
SWC-117	Signature Malleability	**Passed**
SWC-118	Incorrect Constructor Name	**Passed**

VULNERABILITY ANALYSIS

ID	Title	
SWC-119	Shadowing State Variables	Passed
SWC-120	Weak Sources of Randomness from Chain Attributes	Passed
SWC-121	Missing Protection against Signature Replay Attacks	Passed
SWC-122	Lack of Proper Signature Verification	Passed
SWC-123	Requirement Violation	Passed
SWC-124	Write to Arbitrary Storage Location	Passed
SWC-125	Incorrect Inheritance Order	Passed
SWC-126	Insufficient Gas Griefing	Passed
SWC-127	Arbitrary Jump with Function Type Variable	Passed
SWC-128	DoS With Block Gas Limit	Passed
SWC-129	Typographical Error	Passed
SWC-130	Right-To-Left-Override control character (U+202E)	Passed
SWC-131	Presence of unused variables	Passed
SWC-132	Unexpected Ether balance	Passed
SWC-133	Hash Collisions With Multiple Variable Length Arguments	Passed
SWC-134	Message call with hardcoded gas amount	Passed
SWC-135	Code With No Effects	Passed
SWC-136	Unencrypted Private Data On-Chain	Passed

VULNERABILITY ANALYSIS

NO ERRORS FOUND

MANUAL
CODE REVIEW

When performing smart contract audits, our specialists look for known vulnerabilities as well as logical and access control issues within the code. The exploitation of these issues by malicious actors may cause serious financial damage to projects that failed to get an audit in time.

We categorize these vulnerabilities by 4 different threat levels.

THREAT LEVELS

High Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Medium Risk

Issues on this level are critical to the smart contract's performance, functionality and should be fixed before moving to a live environment.

Low Risk

Issues on this level are minor details and warning that can remain unfixed.

Informational

Information level is to offer suggestions for improvement of efficacy or security for features with a risk free factor.

FOUND THREATS

⚠ High Risk

No high risk-level threats found in this contract.

⚠ Medium Risk

No medium risk-level threats found in this contract.

⚠ Low Risk

No low risk-level threats found in this contract.

FOUND THREATS

Informational

Function cannot be used as pool contract's owner is the factory contract and transferToNewPool() is not called anywhere in factory contract.

'deadline' parameter on IPancakeRouter01.addLiquidity() should be block.timestamp.
Current implementation allows waiting in line for 365 days for tx to be executed if not adequate gas is provided for the transaction.

Liquidity tokens receiver is the Liquidity presale contract and LP tokens cannot be retrieved once added to the main DEX.
If this is the desired behaviour, better send the LP tokens to address(0) burning them instead.

```
function transferToNewPool() external onlyOwner lock notMigrated {
    migrated = true;

    token.burn((realTokenBalance / 100) * 75);

    address router = factory.dexRouterAddress();
    uint tokenAmount = (realTokenBalance / 100) * 25;

    token.approve(router, tokenAmount);
    wvtru.approve(router, realCoinBalance);

    IPancakeRouter01(router).addLiquidity(address(wvtru), address(token),
    realCoinBalance, tokenAmount, 1, 1, address(this), block.timestamp + 365 days);

    _updateReserves();

    emit PoolTransfered();
}
```

PoolFactory.sol INFO

Token Name
Unavailable

Symbol
Unavailable

Contract Address
Unavailable

Network
Unavailable

Language
Solidity

Deployment Date
Unavailable

Contract Type
Liquidity pool

Total Supply
Unavailable

Decimals
Unavailable

TAXES

Buy Tax
1%

Sell Tax
1%

*Taxes cannot be changed

Our Contract Review Process

The contract review process pays special attention to the following:

- Testing the smart contracts against both common and uncommon vulnerabilities
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

Blockchain security tools used:

- OpenZeppelin
- Mythril
- Solidity Compiler
- Hardhat

FOUND THREATS

⚠ High Risk

No high risk-level threats found in this contract.

⚠ Medium Risk

No medium risk-level threats found in this contract.

⚠ Low Risk

No low risk-level threats found in this contract.

FOUND THREATS

Informational

'_value' should be always greater than 'contractPrice', otherwise overflow might occur, causing the transaction to revert.

```
function createPoolWithToken(
    string calldata _name,
    string calldata _ticker,
    string calldata _uri,
    uint256 _amount,
    uint256 _value
) public returns (address, address) {

uint256 allowance = wvtru.allowance(msg.sender, address(this));
require(allowance >= _value, "check the token allowance");
wvtru.transferFrom(msg.sender, address(this), _value);
wvtru.transfer(creationFeeAddress, contractPrice);
.....................
    userTokens[msg.sender].push(tokenAddress);
    tokens.push(tokenAddress);

    emit PoolCreated(msg.sender, poolAddress, tokenAddress);

    uint256 amountToBuyTokens = _value - contractPrice;
.....................
}
```

Owner can set price for new tokens/LP contract creation.

```
function setContractPrice(uint256 _price) public onlyOwner returns (bool) {
    require(_price > 0, "too low price");
    contractPrice = _price;
    return true;
}
```

Token.sol INFO

Token Name
Unavailable

Symbol
Unavailable

Contract Address
Unavailable

Network
Unavailable

Language
Solidity

Deployment Date
Unavailable

Contract Type
Liquidity pool

Total Supply
Unavailable

Decimals
Unavailable

TAXES

Buy Tax
1%

Sell Tax
1%

*Taxes cannot be changed

Our Contract Review Process

The contract review process pays special attention to the following:

- Testing the smart contracts against both common and uncommon vulnerabilities
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

Blockchain security tools used:

- OpenZeppelin
- Mythril
- Solidity Compiler
- Hardhat

FOUND THREATS

⚠ High Risk

No high risk-level threats found in this contract.

⚠ Medium Risk

No medium risk-level threats found in this contract.

⚠ Low Risk

No low risk-level threats found in this contract.

FOUND THREATS

Informational

Owner can mint new tokens.
Since owner is the liquidity pool contract, and the mint functionality is used during token's deployment from the liquidity contract, no further tokens can be issued once the token is deployed.

Suggestion: Renounce token's ownership (transfer it to address(0)) after deployment as all token analyzers will display that new tokens can be minted which can scare investors in future.

```
function mint(address _to, uint256 _amount) public onlyOwner {
    _mint(_to, _amount);
}
```

WEBSITE

Website URL:
Under construction

Domain Registry
Under construction

Domain Expiration
Under construction

Technical SEO Test
Under construction

Security Test
Under construction

Design
Under construction

Content
Under construction

Whitepaper
Under construction

Roadmap

Under construction

Mobile-friendly?

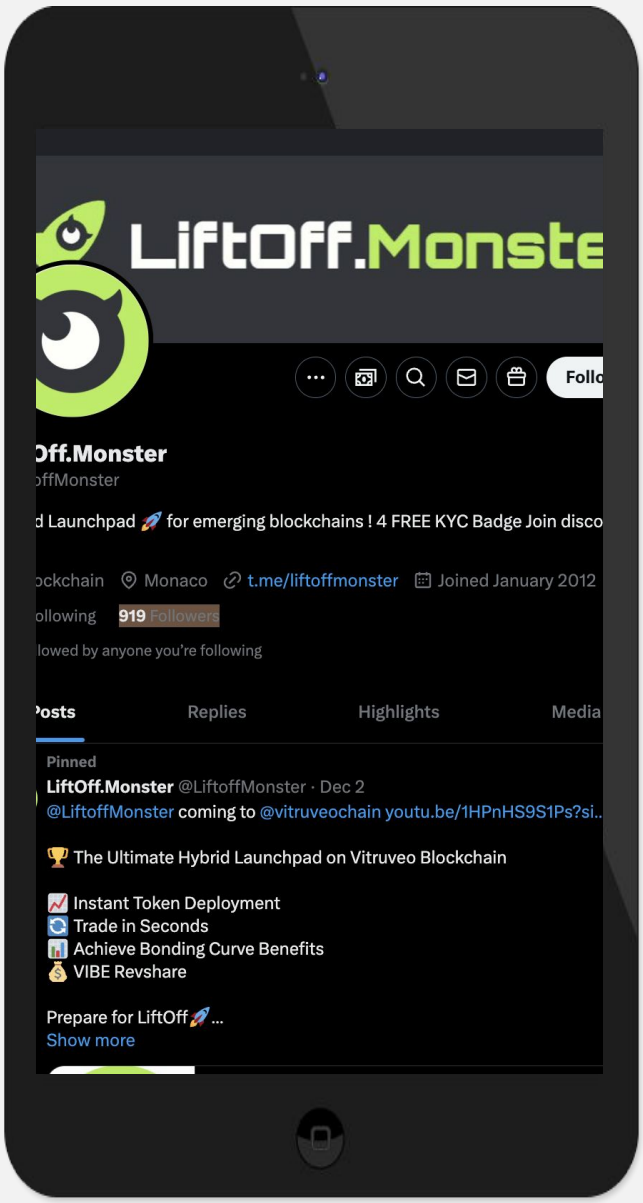
Under construction

Under Construction

SOCIAL MEDIA

ANALYSIS

The project's social media pages are active with daily posts.



Twitter:

@liftoffmonster

- 919 Followers
- Daily posts

Discord

@NaKW9tUTCf

- 134 members

Telegram:

@liftoffmonster

- 1 513 members
- Active members and mods

Medium

Unavailable

ABOUT US

We are a growing crypto security agency offering audits, KYCs and consulting services for some of the top names in the crypto industry.

- **OVER 700 SUCCESSFUL CLIENTS**
- **MORE THAN 1000 SCAMS EXPOSED**
- **MILLIONS SAVED IN POTENTIAL FRAUD**
- **PARTNERSHIPS WITH TOP LAUNCHPADS, INFLUENCERS AND CRYPTO PROJECTS**
- **CONSTANTLY BUILDING TOOLS TO HELP INVESTORS DO BETTER RESEARCH**

SPYWOLF

CRYPTO SECURITY

Audits | KYCs | dApps
Contract Development

To hire us, reach out to
contact@spywolf.co or
t.me/joe_SpyWolf

FIND US ONLINE

SPYWOLF.CO

@SPYWOLFNETWORK

@SPYWOLFNETWORK

Disclaimer

This report shows findings based on our limited project analysis, following good industry practice from the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, overall social media and website presence and team transparency details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report.
While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – please make sure to read it in full.

DISCLAIMER:
By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice.
No one shall have any right to rely on the report or its contents, and SpyWolf and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (SpyWolf) owe no duty of care towards you or any other person, nor does SpyWolf make any warranty or representation to any person on the accuracy or completeness of the report.
The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and SpyWolf hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, SpyWolf hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against SpyWolf, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts, website, social media and team.
No applications were reviewed for security. No product code has been reviewed.