



SPYWOLF

Security Audit Report

(TESTNET)



Audit prepared for
Chain Factory
(Vesting Contract)

Completed on
May 9, 2024

@SPYWOLFNETWORK



@SPYWOLFNETWORK



SPYWOLF.CO





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 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
Contract 1 Information	02
Current Stats	03-04
Vulnerability Check	05-06
Threat Levels	07
Found Threats	08-A/08-B
About SPYWOLF	09
Disclaimer	10



CHAIN FACTORY



PROJECT DESCRIPTION

"With ChainFactory, users can choose from a variety of customizable templates and features, making it simple to create contracts tailored to your specific needs. It is designed to be user-friendly and intuitive, guiding users through the entire process step-by-step, providing a centralized platform to create, deploy, and manage your Smart-Contracts with ease."

Release Date: TBD

Category: Ecosystem





VESTING CONTRACT INFO

Token Name

N/A

Symbol

N/A

Contract Address

0x254865d907D4e5152a638D533E4928dB1B3dedD0

Network

Ethereum Sepolia TESTNET

Language

Solidity

Deployment Date

May 07, 2024

Contract Type

Vesting

Total Supply

N/A

Status

Not launched

TAXES

Buy Tax
0.25%

Sell Tax
none

*Taxes can be changed in future



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



CURRENT STATS

(As of May 9, 2024)



Liquidity

Not added yet



Burn

No burnt tokens

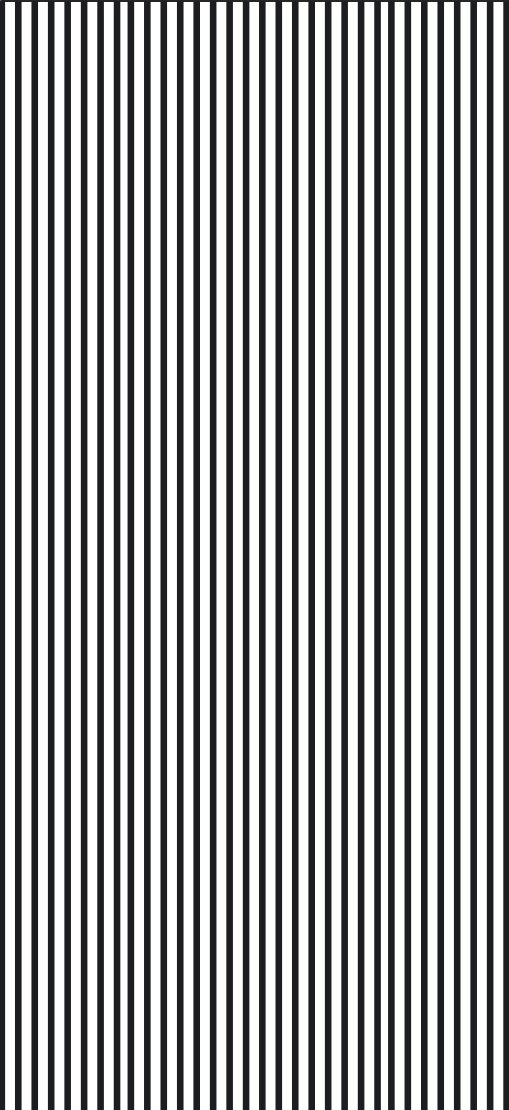
Status:
Not Launched!

MaxTxAmount
N/A

LP Address(es)



Liquidity not added yet





TOKEN TRANSFERS STATS

Transfer Count	N/A
Uniq Senders	N/A
Uniq Receivers	N/A
Total Amount	N/A
Median Transfer Amount	N/A
Average Transfer Amount	N/A
First transfer date	N/A
Last transfer date	N/A
Days token transferred	N/A

SMART CONTRACT STATS

Calls Count	N/A
External calls	N/A
Internal calls	N/A
Transactions count	N/A
Uniq Callers	N/A
Days contract called	N/A
Last transaction time	N/A
Created	N/A
Create TX	N/A
Creator	N/A



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



THREAT LEVELS

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 the following 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.



⚠ Medium Risk

Owner can change token deposit fees without limitations.
Owner can change token free lock duration.

Token fees are paid in vested tokens for each newly created vesting. Fees are locked up and cannot be retrieved by owner at least for the tokenFeeLockDuration period.

Current token fees lock duration period is 30 days.

Current deposit fees are 0.25%.

```
uint256 public tokenFee = 25;
uint256 public tokenFeeLockDuration = 30 days;

function changeTokenFeeLockDuration(uint256 newDuration) external onlyOwner {
    tokenFeeLockDuration = newDuration;
}

function changeTokenFee(uint256 newFee) external onlyOwner {
    tokenFee = newFee;
}

function _deposit(address token, uint256 amount, uint256 cliffDuration, uint256 duration,
    VestingType vestingType, uint256[] memory gradedPercentages) internal {
    uint256 vestingId = _generateVestingId();
    uint256 fee = (amount * tokenFee) / 10000;
    uint256 depositAmount = amount - fee;
    uint256 cliffStart = (vestingType == VestingType.CLIFF) ? block.timestamp + cliffDuration : block.timestamp;

    IERC20(token).transferFrom(msg.sender, address(this), amount);

    if (fee > 0) {
        _lockTokenFee(token, fee);
    }
    .....
}

function _lockTokenFee(address token, uint256 amount) internal {
    uint256 vestingId = _generateVestingId();

    VestingInfo memory newVesting = VestingInfo({
        vestingId: vestingId,
        user: feeWallet,
        token: token,
        amount: amount,
        start: block.timestamp,
        cliff: block.timestamp, // Correctly set the cliff start time
        duration: tokenFeeLockDuration, // Duration of the vesting
        claimed: 0,
        finished: false,
        vestingType: VestingType.BASIC_LOCK,
        gradedPercentages: new uint256[](0)
    });

    userVestings[feeWallet].push(newVesting);
    vestingById[vestingId] = newVesting;
}
```

- Recommendation:
 - Deposit fees should be limited to reasonable percent



Informational

Owner can withdraw ETH from the contract.

When this function is present, in cases ETH is sent to the contract by mistake or purposefully, contract's owner can retrieve it.

```
function withdrawEther(uint256 amount) external onlyOwner {  
    require(amount <= address(this).balance, "Insufficient balance");  
    payable(owner).transfer(amount);  
}
```

Owner can change deposit fee without limitations.

Deposit fees are paid in ether on each new deposit (vesting creation).

```
function changeDepositFee(uint256 newFee) external onlyOwner {  
    depositFee = newFee;  
}
```



SPYWOLF

CRYPTO SECURITY

Audits | KYCs | dApps
Contract Development

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

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

FIND US ONLINE



[SPYWOLF.CO](https://spywolf.co)



[@SPYWOLFNETWORK](https://t.me/SPYWOLFNETWORK)



[@SPYWOLFNETWORK](https://t.me/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.