SPYWOLF

Security Audit Report

Audit prepared for

Gelato

Completed on

December 5, 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

Gelato

PROJECT DESCRIPTION:

Gelato is an Innovative Utility Fee Token on PulseChain that combines Hyper Deflation, Deep Liquidity Webs and User Rewards.

- Gelato is the very first token to buy and burn SolidX
- Gelato rewards users with SolidX and pHex
- Gelato is Hyper Deflationary
- Gelato has a 3 phase liquidity strategy with deep, adaptive, intelegent webs designed for volume, scarcity, & continous price performance.

Release Date: Dex 5, 2024

Category: Dividend token

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 (2 blocks)
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

CONTRACT INFO

Token Name
Gelato

Symbol
GEL

Contract Address
0x616cb6a245Ed4c11216Ec58D10B6A2E87271845d

Network
PLS

Language
Solidity

Deployment Date
Dec 04, 2024

Contract Type
Dividend token

Total Supply
8,888,888,888

Decimals
18

TAXES

Buy Tax
4%

Sell Tax
4%

*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

SMART CONTRACT STATS

Calls Count	unavailable
External calls	unavailable
Internal calls	unavailable
Transactions count	unavailable
Last transaction time	unavailable
Deployment Date	unavailable
Create TX	unavailable
Owner	unavailable
Deployer	unavailable

TOKEN TRANSFERS STATS

Transfer Count	unavailable
Total Amount	unavailable
Median Transfer Amount	unavailable
Average Transfer Amount	unavailable
First transfer date	unavailable
Last transfer date	unavailable
Days token transferred	unavailable

FEATURED WALLETS

Owner address	0x02495ffa016f97d1463286fc570eb410b8295c63
Marketing fee receiver	0x8a320295d010b95663366e49dfe122fce6718ae0
LP address	0xf1dD0467DAd70fB47366eA31F69D3FA1e64BeC52 Gelato/WPLS

TOP 3 UNLOCKED WALLETS

11%	0xC07a42308C4C0E14566B93F3Efb3b10a374C1c00
5%	0x25825aBA6210Ba629848E0f3D4A7bD00b8E3c630
4%	0x90f95A7BdBDd10A4dB2725BbacbD9Dd20FB5BbdA

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: 0

No high risk-level threats found in this contract.

⚠ Medium Risk: 0

No medium risk-level threats found in this contract.

⚠ Low Risk: 0

No low risk-level threats found in this contract.

FOUND THREATS

Informational

Owner can change contract's auto swap settings.
When auto swap is enabled and swapThreshold is set to 0, swap will fail.
This will not revert the entire transaction as the autoswap occurs in try catch block. It is recommended that swapThreshold’s value be always above 1 token.

```
function setSwapBackSettings(bool _enabled, uint256 _amount) external onlyOwner {
    swapEnabled = _enabled;
    swapThreshold = _amount;

    emit ParameterUpdated();
}

function swapBack() internal swapping {
.............................
 uint256 amountGelSwap = swapThreshold.sub(amountGelLiquidity).sub(
        amountGelBurn
    );

     uint256 amountGelSwap = swapThreshold.sub(amountGelLiquidity).sub(
        amountGelBurn
    );

    uint256 balanceBefore = address(this).balance;

    address[] memory path = new address[](2);
    path[0] = address(this);
    path[1] = address(WPLS);

    try
        pulseRouterV2.swapExactTokensForETHSupportingFeeOnTransferTokens(
            amountGelSwap,
            0,
            path,
            address(this),
            block.timestamp
        )
    {
.............................
}
```

FOUND THREATS

Informational

Owner can change dividend's distributor distribution criteria like periods between each dividend distribution and amounts of each dividend distribution.

```
function setDistributionCriteria(
    uint256 _minSolidXPeriod,
    uint256 _minSolidXDistribution,
    uint256 _minHexPeriod,
    uint256 _minHexDistribution
) external onlyOwner {
    distributor.setDistributionCriteria(
        _minSolidXPeriod,
        _minSolidXDistribution,
        _minHexPeriod,
        _minHexDistribution
    );

    emit ParameterUpdated();
}

function setDistributionCriteria(
    uint256 _minSolidXPeriod,
    uint256 _minSolidXDistribution,
    uint256 _minHexPeriod,
    uint256 _minHexDistribution
) external override onlyToken {
    minSolidXPeriod = _minSolidXPeriod;
    minSolidXDistribution = _minSolidXDistribution;
    minHexPeriod = _minHexPeriod;
    minHexDistribution = _minHexDistribution;
}
```

FOUND THREATS

Informational

Owner can set fees buy/sell fees up to 10%.
When transfer fees are enabled, transfers will be taxed with the buy fees' value.
Combined buy+sell = 20%.

```
function setFees(
    uint256 _solidXBurnFee,
    uint256 _stackedBurnFee,
    uint256 _gelatoBurnFee,
    uint256 _genesisFee,
    uint256 _solidXReflectionFee,
    uint256 _hexReflectionFee,
    uint256 _liquidityFee,
    uint256 _totalBuyFee,
    uint256 _totalSellFee,
    bool _feesOnNormalTransfers
) external onlyOwner {
    solidXBurnFee = _solidXBurnFee;
    stackedBurnFee = _stackedBurnFee;
    gelatoBurnFee = _gelatoBurnFee;
    genesisFee = _genesisFee;
    solidXReflectionFee = _solidXReflectionFee;
    hexReflectionFee = _hexReflectionFee;
    liquidityFee = _liquidityFee;
    totalBuyFee = _totalBuyFee;
    totalSellFee = _totalSellFee;
    require(genesisFee <= 500, "5 percent max.");
    require(
        solidXBurnFee
            .add(stackedBurnFee)
            .add(gelatoBurnFee)
            .add(genesisFee)
            .add(solidXReflectionFee)
            .add(hexReflectionFee)
            .add(liquidityFee) <= feeDenominator,
        "The total of all combined fees must be 10000 for 100 percent."
    );
    require(totalBuyFee <= feeDenominator, "Buy fee too high");
    require(totalSellFee <= feeDenominator, "Sell fee too high");

    feesOnNormalTransfers = _feesOnNormalTransfers;
}
```

FOUND THREATS

Informational

Owner can exclude address from dividends.

```
function setIsDividendExempt(
    address holder,
    bool exempt
) external onlyOwner {
    require(holder != address(this) && holder != pulseV2Pair);
    isDividendExempt[holder] = exempt;
    if (exempt) {
        distributor.setShare(holder, 0);
    } else {
        distributor.setShare(holder, _balances[holder]);
    }
}
```

Owner can set max transaction limit but cannot lower it than 0.2% of total supply.

```
function setTxLimit(
    uint256 amount
) external onlyOwner {
    require(amount >= _totalSupply / 2000);
    _maxTxAmount = amount;
}

function checkTxLimit(address sender, address recipient, uint256 amount) internal view {
    require(
        amount <= _maxTxAmount || isTxLimitExempt[sender] || isTxLimitExempt[recipient],
        "TX Limit Exceeded"
    );
}
```

FOUND THREATS

Informational

Owner can add and remove address from liquidity pairs list.

```
function addPair(address pair) external onlyOwner {
    pairs.push(pair);
    emit ParameterUpdated();
}

function removeLastPair() external onlyOwner{
    pairs.pop();
    emit ParameterUpdated();
}
```

Owner can change autoswap's liquidity receiver and genesis receiver.

```
function setLiquidityFeeReceiver(
    address _autoLiquidityReceiver
) external onlyOwner {
    autoLiquidityReceiver = _autoLiquidityReceiver;

    emit ParameterUpdated();
}

function setGenesisReceiver(address _newGenesis) external onlyOwner {
    genesisReceiver = _newGenesis;

    emit ParameterUpdated();
}
```

Owner can exclude address from fees and max transaction limits.

```
function setIsFeeExempt(address holder, bool exempt) external onlyOwner {
    isFeeExempt[holder] = exempt;
}

function setIsTxLimitExempt(address holder, bool exempt) external onlyOwner {
    isTxLimitExempt[holder] = exempt;
}
```

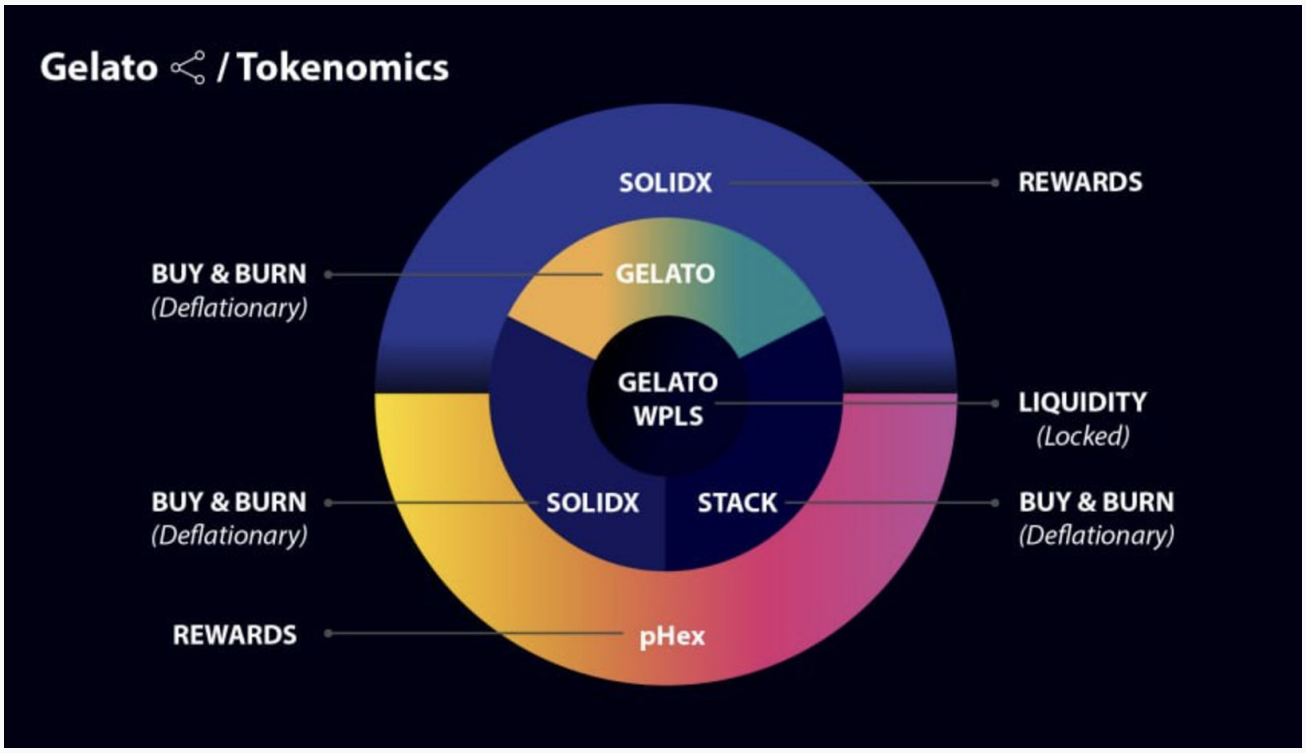
Gelato features a stabilized utility fee consisting of 4% Buy, 4% Transfer and 4% Sell. Fees are generated from volume running through the Token. Every time a user buys, sells or transfers Gelato, the utility fees are triggered capturing a small percentage and feeding it into the system.
The utility fees are directed to 4 distinct categories:

Rewards: 50%-70% of the fees from the 4% being captured go to rewarding holders with SolidX and pHex.

Deflation: 25%-45% of the fees being captured go to Buying and Buring SolidX, Gelato and Stacked Italian.

Liquidity Build: 5%-15% of the fees will be allocated to building liquidity in the WPLS pair which is locked for 1 year.

Genesis: 5% or less of the 4% fees will go to the Genesis.
The Main Pairing with Gelato will be WPLS and this pool will be locked for at minimum, 1 year. The tokens being bought and burnt will be sent to the zero address.



TOKENOMICS

WEBSITE

Website URL:
unavailable

Domain Registry
unavailable

Domain Expiration
unavailable

Technical SEO Test
unavailable

Security Test
unavailable

Design
unavailable

Content
unavailable

Whitepaper
unavailable

Roadmap
unavailable

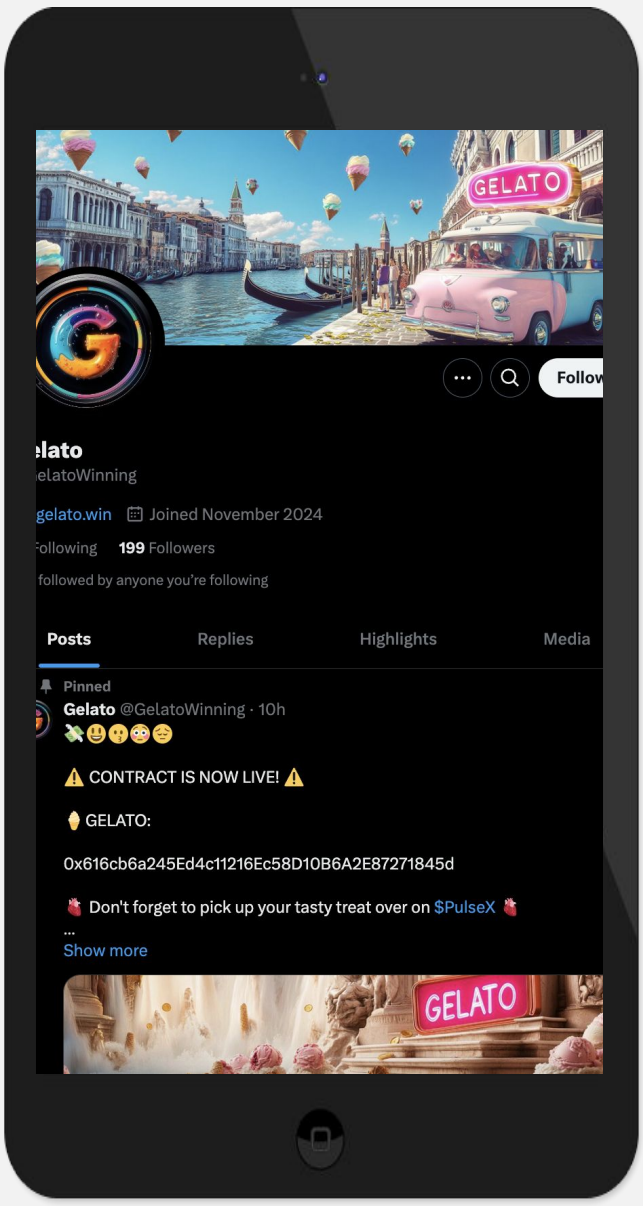
Mobile-friendly?
unavailable

gelato.win

SOCIAL MEDIA

ANALYSIS

Project's social media pages are new but active.



Twitter:

@GelatoWinning

- 199 Followers
- Daily posts

Discord

unavailable

Telegram:

@GelatoWin

- 464 members,
- Active mods and members

Medium

unavailable

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

@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.