



Cyberscope

Audit Report

Oxpad

January 2023

| | |
|------------|--------------------------------------------|
| Type | BEP20 |
| Network | BSC |
| Address | 0x15153e3Cfa74fd82e5f0cE6D16ce3f250D89d978 |
| Audited by | © cyberscope |

Table of Contents

| | |
|---------------------------------------------------------|-----------|
| Table of Contents | 1 |
| Review | 2 |
| Audit Updates | 2 |
| Source Files | 3 |
| Analysis | 4 |
| Diagnostics | 5 |
| L04 - Conformance to Solidity Naming Conventions | 6 |
| Description | 6 |
| Recommendation | 6 |
| L18 - Multiple Pragma Directives | 7 |
| Description | 7 |
| Recommendation | 7 |
| L19 - Stable Compiler Version | 8 |
| Description | 8 |
| Recommendation | 8 |
| Functions Analysis | 9 |
| Inheritance Graph | 14 |
| Flow Graph | 15 |
| Summary | 16 |
| Disclaimer | 17 |
| About Cyberscope | 18 |

Review

| | |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Contract Name | ZeroXPad |
| Compiler Version | v0.8.9+commit.e5eed63a |
| Optimization | 200 runs |
| Explorer | https://bscscan.com/address/0x15153e3cfa74fd82e5f0ce6d16ce3f250d89d978 |
| Address | 0x15153e3cfa74fd82e5f0ce6d16ce3f250d89d978 |
| Network | BSC |
| Symbol | ZXP |
| Decimals | 9 |
| Total Supply | 10,000,000 |

Audit Updates

| | |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Initial Audit | 19 Jan 2023 https://github.com/cyberscope-io/audits/blob/main/zxp/v1/audit.pdf |
| Corrected Phase 2 | 23 Jan 2023 https://github.com/cyberscope-io/audits/blob/main/zxp/v2/audit.pdf |
| Corrected Phase 3 | 25 Jan 2023 |

Source Files

| Filename | SHA256 |
|--------------|----------------------------------------------------------------------|
| ZeroXPad.sol | e1a669d0c60ee90240d7dc5070f632464 21cba36fd452fbcadafb32818adeb88 |

Analysis

● Critical ● Medium ● Minor / Informative ● Pass

| Severity | Code | Description | Status |
|----------|------|------------------------------------|--------|
| ● | ST | Stops Transactions | Passed |
| ● | OCTD | Transfers Contract's Tokens | Passed |
| ● | OTUT | Transfers User's Tokens | Passed |
| ● | ELFM | Exceeds Fees Limit | Passed |
| ● | ULTW | Transfers Liquidity to Team Wallet | Passed |
| ● | MT | Mints Tokens | Passed |
| ● | BT | Burns Tokens | Passed |
| ● | BC | Blacklists Addresses | Passed |

Diagnostics

● Critical ● Medium ● Minor / Informative

| Severity | Code | Description | Status |
|----------|------|--------------------------------------------|------------|
| ● | L04 | Conformance to Solidity Naming Conventions | Unresolved |
| ● | L18 | Multiple Pragma Directives | Unresolved |
| ● | L19 | Stable Compiler Version | Unresolved |

L04 - Conformance to Solidity Naming Conventions

| | |
|--------------------|---------------------|
| Criticality | Minor / Informative |
| Location | ZeroXPad.sol#L8 |
| Status | Unresolved |

Description

The Solidity style guide is a set of guidelines for writing clean and consistent Solidity code. Adhering to a style guide can help improve the readability and maintainability of the Solidity code, making it easier for others to understand and work with.

The followings are a few key points from the Solidity style guide:

1. Use camelCase for function and variable names, with the first letter in lowercase (e.g., myVariable, updateCounter).
2. Use PascalCase for contract, struct, and enum names, with the first letter in uppercase (e.g., MyContract, UserStruct, ErrorEnum).
3. Use uppercase for constant variables and enums (e.g., MAX_VALUE, ERROR_CODE).
4. Use indentation to improve readability and structure.
5. Use spaces between operators and after commas.
6. Use comments to explain the purpose and behavior of the code.
7. Keep lines short (around 120 characters) to improve readability.

```
function WETH() external pure returns (address);
```

Recommendation

By following the Solidity naming convention guidelines, the codebase increased the readability, maintainability, and makes it easier to work with.

Find more information on the Solidity documentation

<https://docs.soliditylang.org/en/v0.8.17/style-guide.html#naming-convention>.

L18 - Multiple Pragma Directives

| | |
|--------------------|----------------------------------------------------------------|
| Criticality | Minor / Informative |
| Location | ZeroXPad.sol#L4,102,148,171,401,428,513,598,628,1019,1060,1097 |
| Status | Unresolved |

Description

If the contract includes multiple conflicting pragma directives, it may produce unexpected errors. To avoid this, it's important to include the correct pragma directive at the top of the contract and to ensure that it is the only pragma directive included in the contract.

```
pragma solidity >=0.6.2;  
pragma solidity >=0.5.0;  
pragma solidity ^0.8.0;  
pragma solidity 0.8.9;
```

Recommendation

It is important to include only one pragma directive at the top of the contract and to ensure that it accurately reflects the version of Solidity that the contract is written in.

By including all required compiler options and flags in a single pragma directive, the potential conflicts could be avoided and ensure that the contract can be compiled correctly.

L19 - Stable Compiler Version

| | |
|--------------------|-------------------------------------------------|
| Criticality | Minor / Informative |
| Location | ZeroXPad.sol#L171,401,428,513,598,628,1019,1060 |
| Status | Unresolved |

Description

The `^` symbol indicates that any version of Solidity that is compatible with the specified version (i.e., any version that is a higher minor or patch version) can be used to compile the contract. The version lock is a mechanism that allows the author to specify a minimum version of the Solidity compiler that must be used to compile the contract code. This is useful because it ensures that the contract will be compiled using a version of the compiler that is known to be compatible with the code.

```
pragma solidity ^0.8.0;
```

Recommendation

The team is advised to lock the pragma to ensure the stability of the codebase. The locked pragma version ensures that the contract will not be deployed with an unexpected version. An unexpected version may produce vulnerabilities and undiscovered bugs. The compiler should be configured to the lowest version that provides all the required functionality for the codebase. As a result, the project will be compiled in a well-tested LTS (Long Term Support) environment.

Functions Analysis

| Contract | Type | Bases | | |
|---------------------------|-------------------------------------------------|--------------------|------------|-----------|
| | Function Name | Visibility | Mutability | Modifiers |
| | | | | |
| IUniswapV2Router01 | Interface | | | |
| | factory | External | | - |
| | WETH | External | | - |
| | addLiquidity | External | ✓ | - |
| | addLiquidityETH | External | Payable | - |
| | removeLiquidity | External | ✓ | - |
| | removeLiquidityETH | External | ✓ | - |
| | removeLiquidityWithPermit | External | ✓ | - |
| | removeLiquidityETHWithPermit | External | ✓ | - |
| | swapExactTokensForTokens | External | ✓ | - |
| | swapTokensForExactTokens | External | ✓ | - |
| | swapExactETHForTokens | External | Payable | - |
| | swapTokensForExactETH | External | ✓ | - |
| | swapExactTokensForETH | External | ✓ | - |
| | swapETHForExactTokens | External | Payable | - |
| | quote | External | | - |
| | getAmountOut | External | | - |
| | getAmountIn | External | | - |
| | getAmountsOut | External | | - |
| | getAmountsIn | External | | - |
| | | | | |
| IUniswapV2Router02 | Interface | IUniswapV2Router01 | | |
| | removeLiquidityETHSupportingFeeOnTransferTokens | External | ✓ | - |

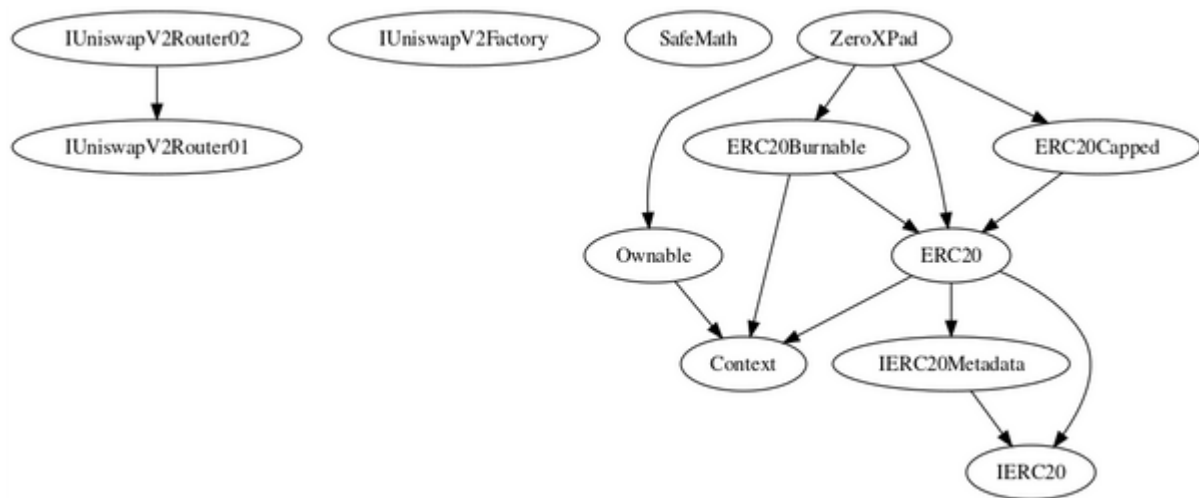
| | | | | |
|--------------------------|-----------------------------------------------------------|----------|---------|---|
| | removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | External | ✓ | - |
| | swapExactTokensForTokensSupportingFeeOnTransferTokens | External | ✓ | - |
| | swapExactETHForTokensSupportingFeeOnTransferTokens | External | Payable | - |
| | swapExactTokensForETHSupportingFeeOnTransferTokens | External | ✓ | - |
| | | | | |
| IUniswapV2Factory | Interface | | | |
| | feeTo | External | | - |
| | feeToSetter | External | | - |
| | getPair | External | | - |
| | allPairs | External | | - |
| | allPairsLength | External | | - |
| | createPair | External | ✓ | - |
| | setFeeTo | External | ✓ | - |
| | setFeeToSetter | External | ✓ | - |
| | | | | |
| SafeMath | Library | | | |
| | tryAdd | Internal | | |
| | trySub | Internal | | |
| | tryMul | Internal | | |
| | tryDiv | Internal | | |
| | tryMod | Internal | | |
| | add | Internal | | |
| | sub | Internal | | |
| | mul | Internal | | |
| | div | Internal | | |
| | mod | Internal | | |
| | sub | Internal | | |
| | div | Internal | | |

| | | | | |
|-----------------------|--------------------|---------------------------------------|---|-----------|
| | mod | Internal | | |
| | | | | |
| Context | Implementation | | | |
| | _msgSender | Internal | | |
| | _msgData | Internal | | |
| | | | | |
| Ownable | Implementation | Context | | |
| | | Public | ✓ | - |
| | owner | Public | | - |
| | _checkOwner | Internal | | |
| | renounceOwnership | Public | ✓ | onlyOwner |
| | transferOwnership | Public | ✓ | onlyOwner |
| | _transferOwnership | Internal | ✓ | |
| | | | | |
| IERC20 | Interface | | | |
| | totalSupply | External | | - |
| | balanceOf | External | | - |
| | transfer | External | ✓ | - |
| | allowance | External | | - |
| | approve | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | | | | |
| IERC20Metadata | Interface | IERC20 | | |
| | name | External | | - |
| | symbol | External | | - |
| | decimals | External | | - |
| | | | | |
| ERC20 | Implementation | Context, IERC20, IERC20Metadata | | |

| | | | | |
|----------------------|----------------------|-------------------|---|---|
| | | Public | ✓ | - |
| | name | Public | | - |
| | symbol | Public | | - |
| | decimals | Public | | - |
| | totalSupply | Public | | - |
| | balanceOf | Public | | - |
| | transfer | Public | ✓ | - |
| | allowance | Public | | - |
| | approve | Public | ✓ | - |
| | transferFrom | Public | ✓ | - |
| | increaseAllowance | Public | ✓ | - |
| | decreaseAllowance | Public | ✓ | - |
| | _transfer | Internal | ✓ | |
| | _mint | Internal | ✓ | |
| | _burn | Internal | ✓ | |
| | _approve | Internal | ✓ | |
| | _spendAllowance | Internal | ✓ | |
| | _beforeTokenTransfer | Internal | ✓ | |
| | _afterTokenTransfer | Internal | ✓ | |
| | | | | |
| ERC20Burnable | Implementation | Context, ERC20 | | |
| | burn | Public | ✓ | - |
| | burnFrom | Public | ✓ | - |
| | | | | |
| ERC20Capped | Implementation | ERC20 | | |
| | | Public | ✓ | - |
| | cap | Public | | - |
| | _mint | Internal | ✓ | |
| | | | | |

| | | | | |
|-----------------|-----------------------|-----------------------------------------------------|---|----------------------|
| ZeroXPad | Implementation | ERC20, ERC20Capped, ERC20Burnable, Ownable | | |
| | | Public | ✓ | ERC20Capped ERC20 |
| | decimals | Public | | - |
| | setBuyFee | Public | ✓ | onlyOwner |
| | setSellFee | Public | ✓ | onlyOwner |
| | setProtectionModifier | Public | ✓ | onlyOwner |
| | enableTrading | Public | ✓ | onlyOwner |
| | disableProtection | Public | ✓ | onlyOwner |
| | excludeFromFees | Public | ✓ | onlyOwner |
| | includeInFees | Public | ✓ | onlyOwner |
| | setPancakePair | Public | ✓ | onlyOwner |
| | _transfer | Internal | ✓ | |
| | _mint | Internal | ✓ | |

Inheritance Graph



Flow Graph

Summary

Oxpad is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler errors or critical issues. The Contract Owner can access some admin functions that can not be used in a malicious way to disturb the users' transactions. The contract includes a protection fee, causing the fees to be set at 24% until the owner disables the protection. There is also a limit of max 3% fee.

Disclaimer

The information provided in this report does not constitute investment, financial or trading advice and you should not treat any of the document's content as such. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes nor may copies be delivered to any other person other than the Company without Cyberscope's prior written consent. This report is not nor should be considered an "endorsement" or "disapproval" of any particular project or team. This report is not nor should be regarded as an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Cyberscope to perform a security assessment. This document does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors' business, business model or legal compliance. This report should not be used in any way to make decisions around investment or involvement with any particular project. This report represents an extensive assessment process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. Cyberscope's position is that each company and individual are responsible for their own due diligence and continuous security. Cyberscope's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies and in no way claims any guarantee of security or functionality of the technology we agree to analyze. The assessment services provided by Cyberscope are subject to dependencies and are under continuing development. You agree that your access and/or use including but not limited to any services reports and materials will be at your sole risk on an as-is where-is and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives and other unpredictable results. The services may access and depend upon multiple layers of third parties.

About Cyberscope

Cyberscope is a blockchain cybersecurity company that was founded with the vision to make web3.0 a safer place for investors and developers. Since its launch, it has worked with thousands of projects and is estimated to have secured tens of millions of investors' funds.

Cyberscope is one of the leading smart contract audit firms in the crypto space and has built a high-profile network of clients and partners.



The Cyberscope team

<https://www.cyberscope.io>