



Cyberscope

Audit Report

BullBuck\$

July 2023

Network ETH

Address 0xe419f3745388015ad0725792893F33e0F0343ca7

Audited by © cyberscope

Analysis

● Critical ● Medium ● Minor / Informative ● Pass

Severity	Code	Description	Status
●	ST	Stops Transactions	Passed
●	OTUT	Transfers User's Tokens	Passed
●	ELFM	Exceeds Fees Limit	Passed
●	MT	Mints Tokens	Passed
●	BT	Burns Tokens	Passed
●	BC	Blacklists Addresses	Passed

Diagnostics

● Critical ● Medium ● Minor / Informative

Severity	Code	Description	Status
●	L18	Multiple Pragma Directives	Unresolved
●	L19	Stable Compiler Version	Unresolved

Table of Contents

Analysis	1
Diagnostics	2
Table of Contents	3
Review	4
Audit Updates	4
Source Files	4
Findings Breakdown	5
L18 - Multiple Pragma Directives	6
Description	6
Recommendation	6
L19 - Stable Compiler Version	7
Description	7
Recommendation	7
Functions Analysis	8
Inheritance Graph	11
Flow Graph	12
Summary	13
Disclaimer	14
About Cyberscope	15

Review

Contract Name	BUCK
Compiler Version	v0.8.18+commit.87f61d96
Optimization	100 runs
Explorer	https://etherscan.io/address/0xe419f3745388015ad0725792893f33e0f0343ca7
Address	0xe419f3745388015ad0725792893f33e0f0343ca7
Network	ETH
Symbol	BUCK
Decimals	18
Total Supply	100,000,000,000,000

Audit Updates

Initial Audit	03 Jul 2023 https://github.com/cyberscope-io/audits/blob/main/buck/v1/audit.pdf
Corrected Phase 2	08 Jul 2023

Source Files

Filename	SHA256
BUCK.sol	5a3685603991709afb165ed5452308f06537685011f3c5aebcff1d0ed72fdd8

Findings Breakdown



● Critical	0
● Medium	0
● Minor / Informative	2

Severity	Unresolved	Acknowledged	Resolved	Other
● Critical	0	0	0	0
● Medium	0	0	0	0
● Minor / Informative	2	0	0	0

L18 - Multiple Pragma Directives

Criticality	Minor / Informative
Location	BUCK.sol#L6,33,114,144,511,551
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 ^0.8.0;  
pragma solidity ^0.8.12;
```

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	BUCK.sol#L6,33,114,144,511,551
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;  
pragma solidity ^0.8.12;
```

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.

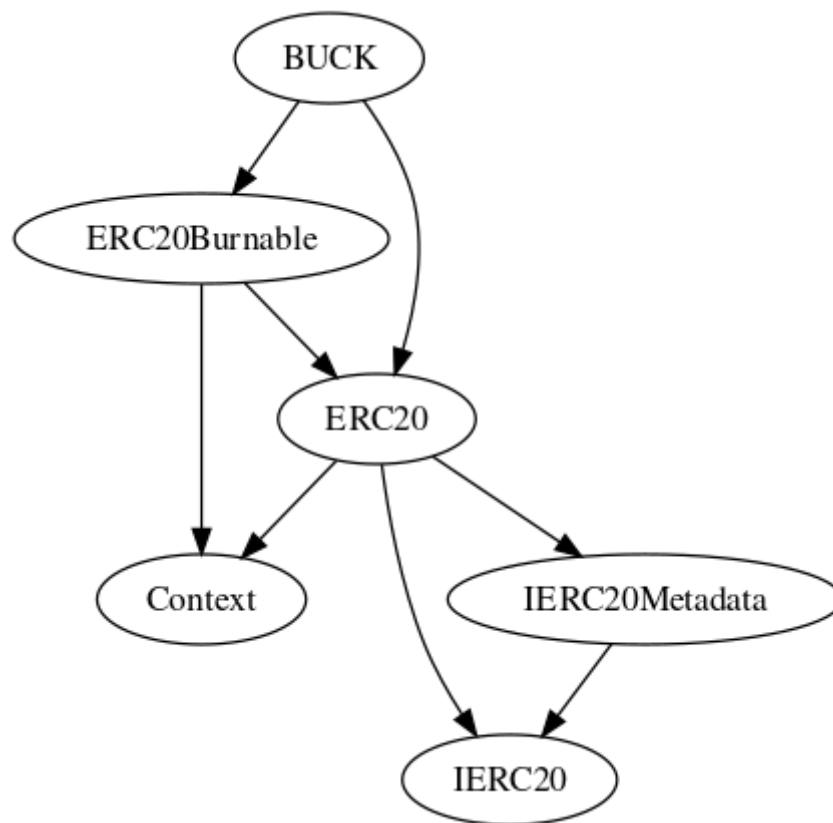
Functions Analysis

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-

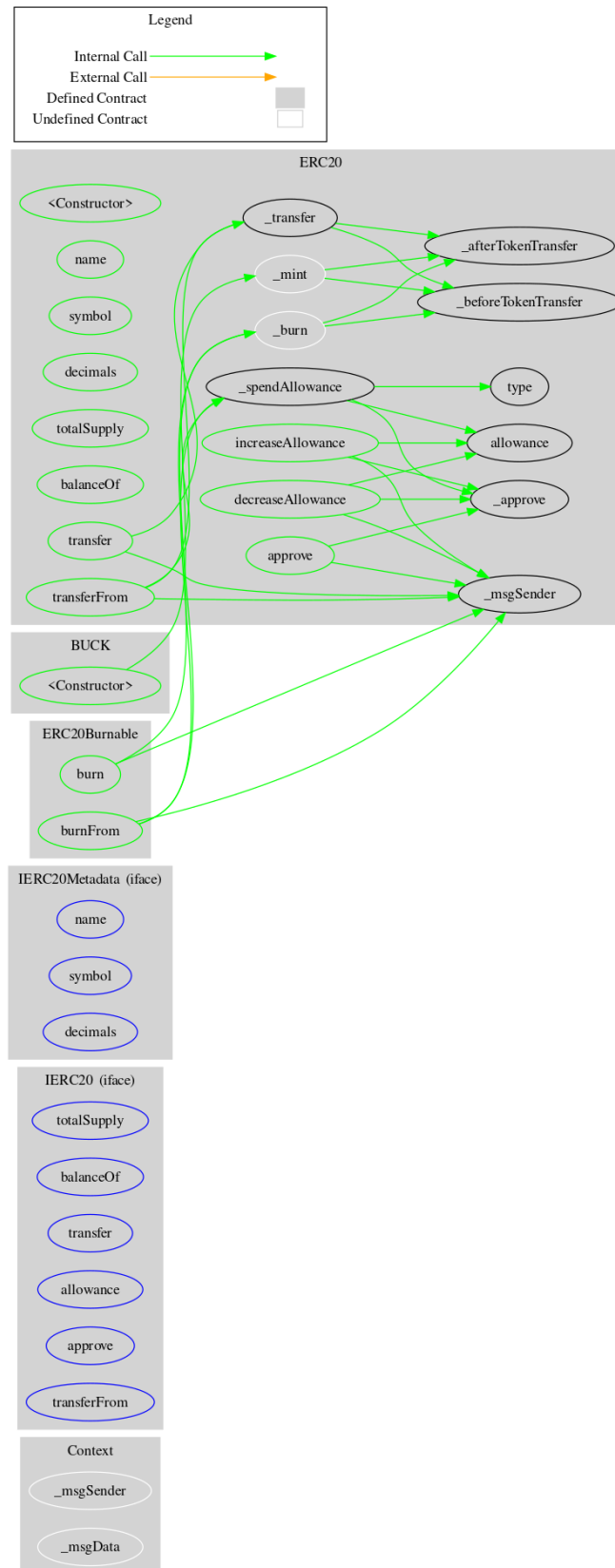
ERC20	Implementation	Context, IERC20, IERC20Meta data		
		Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	
	_approve	Internal	✓	
	_spendAllowance	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
ERC20Burnable	Implementation	Context, ERC20		

	burn	Public	✓	-
	burnFrom	Public	✓	-
BUCK	Implementation	ERC20, ERC20Burna ble		
		Public	✓	ERC20

Inheritance Graph



Flow Graph



Summary

BullBuck\$ contract implements a token mechanism. This audit investigates security issues, business logic concerns and potential improvements. BullBuck\$ is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error 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.

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>



Cyberscope

Audit Report

BullBuck\$

July 2023

Network GOERLI ETH

Address 0x452ad37B61BE1C247871cF0932e62e9dBeFC7Bb5

Audited by © cyberscope

Analysis

● Critical ● Medium ● Minor / Informative ● Pass

Severity	Code	Description	Status
●	ST	Stops Transactions	Passed
●	OTUT	Transfers User's Tokens	Passed
●	ELFM	Exceeds Fees Limit	Passed
●	MT	Mints Tokens	Passed
●	BT	Burns Tokens	Passed
●	BC	Blacklists Addresses	Passed

Diagnostics

● Critical ● Medium ● Minor / Informative

Severity	Code	Description	Status
●	L18	Multiple Pragma Directives	Unresolved
●	L19	Stable Compiler Version	Unresolved

Table of Contents

Analysis	1
Diagnostics	2
Table of Contents	3
Review	4
Audit Updates	4
Source Files	4
Findings Breakdown	5
L18 - Multiple Pragma Directives	6
Description	6
Recommendation	6
L19 - Stable Compiler Version	7
Description	7
Recommendation	7
Functions Analysis	8
Inheritance Graph	11
Flow Graph	12
Summary	13
Disclaimer	14
About Cyberscope	15

Review

Contract Name	BUCK
Compiler Version	v0.8.20+commit.a1b79de6
Optimization	200 runs
Explorer	https://goerli.etherscan.io/address/0x452ad37b61be1c247871cf0932e62e9dbefc7bb5
Address	0x452ad37b61be1c247871cf0932e62e9dbefc7bb5
Network	GOERLI
Symbol	BUCK
Decimals	18
Total Supply	100,000,000,000,000

Audit Updates

Initial Audit	03 Jul 2023
---------------	-------------

Source Files

Filename	SHA256
BUCK.sol	024212d8dde3d5f7618bf1491c3e09e7aae7d38743abc8dd54180ef8abd2e5fd

Findings Breakdown



● Critical	0
● Medium	0
● Minor / Informative	2

Severity	Unresolved	Acknowledged	Resolved	Other
● Critical	0	0	0	0
● Medium	0	0	0	0
● Minor / Informative	2	0	0	0

L18 - Multiple Pragma Directives

Criticality	Minor / Informative
Location	BUCK.sol#L6,33,118,148,515,555
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 ^0.8.0;  
pragma solidity ^0.8.19;  
pragma solidity ^0.8.12;
```

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	BUCK.sol#L6,33,118,148,515,555
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;  
pragma solidity ^0.8.19;  
pragma solidity ^0.8.12;
```

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.

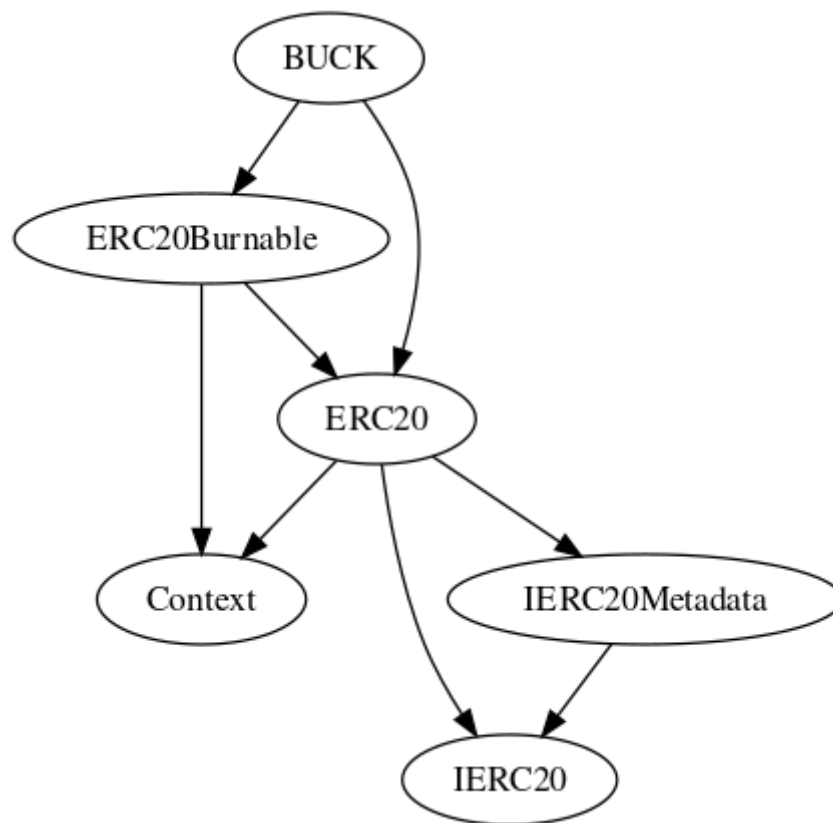
Functions Analysis

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
Context	Implementation			
	_msgSender	Internal		
	_msgData	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, IERC20Meta data		
		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	✓	-
BUCK	Implementation	ERC20, ERC20Burna ble		
		Public	✓	ERC20

Inheritance Graph



The diagram illustrates the mapping of ERC20 tokens to BUCK tokens using the ERC20Burnable interface. It shows various ERC20 functions (like `_transfer`, `_mint`, `_burn`) and their corresponding BUCK functions (like `burnFrom`, `burn`). It also includes a legend for internal/external calls and defined/undefined contracts.

Legend:

- Internal Call (Green arrow)
- External Call (Orange arrow)
- Defined Contract (Grey box)
- Undefined Contract (White box)

ERC20 (Defined Contract):

- `<Constructor>`
- `name`
- `symbol`
- `decimals`
- `totalSupply`
- `balanceOf`
- `transferFrom`
- `transfer`
- `_spendAllowance`
- `increaseAllowance`
- `decreaseAllowance`
- `approve`
- `_transfer`
- `_mint`
- `_burn`
- `type`
- `allowance`
- `_approve`
- `_msgSender`
- `_beforeTokenTransfer`
- `_afterTokenTransfer`

BUCK (Defined Contract):

- `<Constructor>`

ERC20Burnable (Defined Contract):

- `burnFrom`
- `burn`

IERC20Metadata (iface) (Undefined Contract):

- `name`
- `symbol`
- `decimals`

IERC20 (iface) (Undefined Contract):

- `totalSupply`
- `balanceOf`
- `transfer`
- `allowance`
- `approve`
- `transferFrom`

Context (Undefined Contract):

- `_msgSender`
- `_msgData`

The diagram shows the following mappings:

- `ERC20._transfer` maps to `BUCK.burnFrom` and `ERC20Burnable.burnFrom`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer` maps to `ERC20Burnable.burn`.
- `ERC20._mint` maps to `ERC20Burnable.burn`.
- `ERC20._burn` maps to `ERC20Burnable.burn`.
- `ERC20._spendAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.increaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.decreaseAllowance` maps to `ERC20Burnable.burn`.
- `ERC20.approve` maps to `ERC20Burnable.burn`.
- `ERC20._transfer`

Summary

BullBuck\$ contract implements a token mechanism. This audit investigates security issues, business logic concerns and potential improvements. BullBuck\$ is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error 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.

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>