



Cyberscope

# Audit Report

## **Vultor**

September 2023

Network    ETH

Address    0xbE1D499E1a63456bE574Effa32a904465C6Dc75b

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
●	IDI	Immutable Declaration Improvement	Unresolved
●	L04	Conformance to Solidity Naming Conventions	Unresolved
●	L19	Stable Compiler Version	Unresolved

# Table of Contents

<b>Analysis</b>	<b>1</b>
<b>Diagnostics</b>	<b>2</b>
<b>Table of Contents</b>	<b>3</b>
<b>Review</b>	<b>4</b>
Audit Updates	4
Source Files	5
<b>Findings Breakdown</b>	<b>6</b>
IDI - Immutable Declaration Improvement	7
Description	7
Recommendation	7
L04 - Conformance to Solidity Naming Conventions	8
Description	8
Recommendation	8
L19 - Stable Compiler Version	9
Description	9
Recommendation	9
<b>Functions Analysis</b>	<b>10</b>
<b>Inheritance Graph</b>	<b>11</b>
<b>Flow Graph</b>	<b>12</b>
<b>Summary</b>	<b>13</b>
<b>Disclaimer</b>	<b>14</b>
<b>About Cyberscope</b>	<b>15</b>

## Review

Contract Name	VultorToken
Compiler Version	v0.8.17+commit.8df45f5f
Optimization	200 runs
Explorer	<a href="https://etherscan.io/address/0xbe1d499e1a63456be574effa32a904465c6dc75b">https://etherscan.io/address/0xbe1d499e1a63456be574effa32a904465c6dc75b</a>
Address	0xbe1d499e1a63456be574effa32a904465c6dc75b
Network	ETH
Symbol	VLT
Decimals	18
Total Supply	1,000,000,000

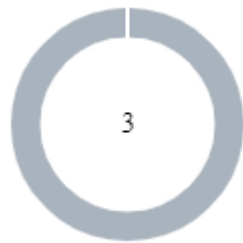
## Audit Updates

Initial Audit	05 Sep 2023
---------------	-------------

## Source Files

Filename	SHA256
<b>contracts/VultorToken.sol</b>	fbc52967925bcf5d6e24b93f580b136d282 1112ef192039f5191fe74ade33238
<b>@openzeppelin/contracts/utils/Context.sol</b>	1458c260d010a08e4c20a4a517882259a2 3a4baa0b5bd9add9fb6d6a1549814a
<b>@openzeppelin/contracts/token/ERC20/IERC20.sol</b>	7ebde70853ccafcf1876900dad458f46eb9 444d591d39bfc58e952e2582f5587
<b>@openzeppelin/contracts/token/ERC20/ERC20.sol</b>	d20d52b4be98738b8aa52b5bb0f88943f6 2128969b33d654fbca731539a7fe0a
<b>@openzeppelin/contracts/token/ERC20/extensions /IERC20Metadata.sol</b>	af5c8a77965cc82c33b7ff844deb9826166 689e55dc037a7f2f790d057811990
<b>@openzeppelin/contracts/access/Ownable.sol</b>	a8e4e1ae19d9bd3e8b0a6d46577eec098c 01fbaffd3ec1252fd20d799e73393b

## Findings Breakdown



● Critical	0
● Medium	0
● Minor / Informative	3

Severity	Unresolved	Acknowledged	Resolved	Other
● Critical	0	0	0	0
● Medium	0	0	0	0
● Minor / Informative	3	0	0	0

## IDI - Immutable Declaration Improvement

<b>Criticality</b>	Minor / Informative
<b>Location</b>	contracts/VultorToken.sol#L19
<b>Status</b>	Unresolved

### Description

The contract declares state variables that their value is initialized once in the constructor and are not modified afterwards. The `immutable` is a special declaration for this kind of state variables that saves gas when it is defined.

```
_decimals
```

### Recommendation

By declaring a variable as immutable, the Solidity compiler is able to make certain optimizations. This can reduce the amount of storage and computation required by the contract, and make it more gas-efficient.



## L04 - Conformance to Solidity Naming Conventions

<b>Criticality</b>	Minor / Informative
<b>Location</b>	contracts/VultorToken.sol#L11
<b>Status</b>	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.

```
uint8 _decimals
```

### 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>.

## L19 - Stable Compiler Version

<b>Criticality</b>	Minor / Informative
<b>Location</b>	contracts/VultorToken.sol#L2
<b>Status</b>	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 ^0.8.17;
```

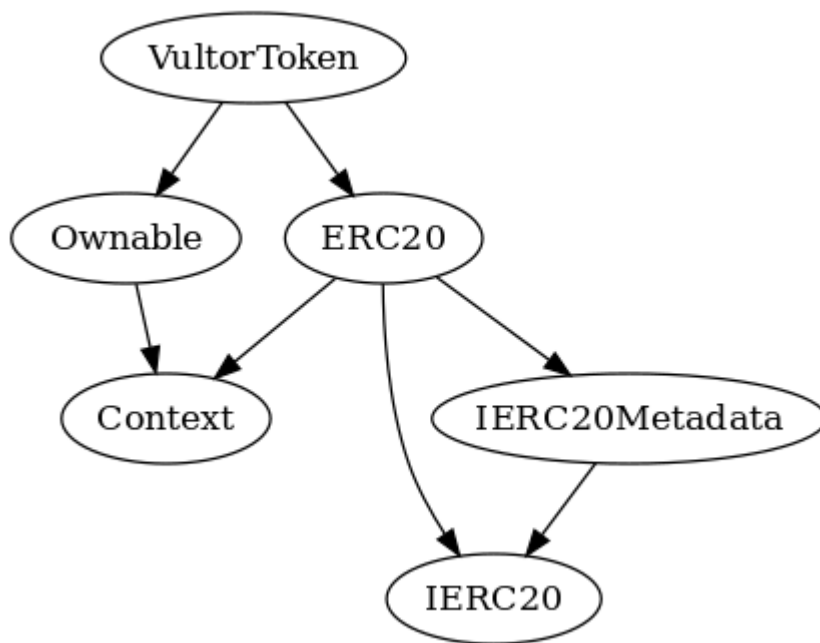
### Recommendation

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.

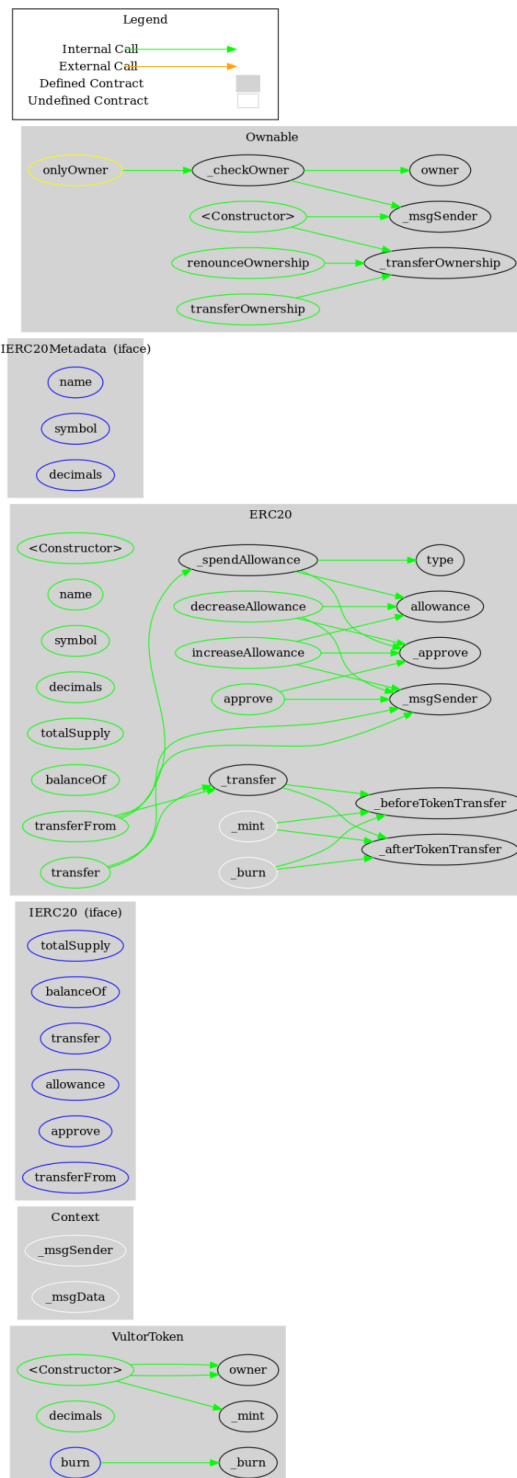
## Functions Analysis

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
VultorToken	Implementation	Ownable, ERC20		
		Public	✓	ERC20
	decimals	Public		-
	burn	External	✓	-

## Inheritance Graph



# Flow Graph



## Summary

Vultor contract implements a token mechanism. This audit investigates security issues, business logic concerns and potential improvements. Vultor 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>