Cyberscope

Audit Report

DCASK

June 2024

Repository	https://github.com/DigiCask/dcask-blast-token
commit	1c86005a7adf71db8b4595fa1844afabcb1d8101
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
●	PLNT	Potential Locked Native Tokens	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**
PLNT - Potential Locked Native Tokens 6
Description 6
Recommendation 6
L19 - Stable Compiler Version 7
Description 7
Recommendation 7
Functions Analysis **8**
Inheritance Graph **10**
Flow Graph **11**
Summary **12**
Initial Audit, 18 June 2024 12
Disclaimer **13**
About Cyberscope **14**

Review

Contract Name	DCASK
Repository	https://github.com/DigiCask/dcask-blast-token/tree/main
Commit	1c86005a7adf71db8b4595fa1844afabcb1d8101
Testing Deploy	https://testnet.bscscan.com/address/0xab55461c6aee76b3f2f089741a32d3844d225478
Decimals	18

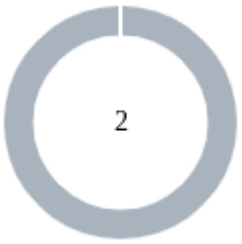
Audit Updates

Initial Audit	18 Jun 2024

Source Files

Filename	SHA256
contracts/DCASK.sol	ab1afea00f3d1669201f2ec837e23936248bb12feb6c5c1410cb851f1103e6ff
contracts/interface/IBlastPoints.sol	719786c4c8498081086c6c6e4f767c8e438274c2f5f3774bb543596ec4354dfc
contracts/interface/IBlast.sol	67d29e7c87fbe8ed70c03702f7008b02c82bc07c1d5e603fcf91022ebafed7d9

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

PLNT - Potential Locked Native Tokens

Criticality	Minor / Informative
Location	contracts/DCASK.sol#L57
Status	Unresolved

Description

The contract includes a `receive` callback function, which allows the contract to receive native tokens. However, there are no functions implemented within the contract that enable the retrieval or management of these accumulated native tokens. As a result, any native tokens sent to the contract via the `receive` function are effectively locked and cannot be accessed or utilized by the contract owner or any other entity.

```
receive() external payable {}
```

Recommendation

To address this issue, the team is advised to either remove the `receive` callback or implement a function that allows the contract owner to retrieve the native tokens accumulated within the contract.

L19 - Stable Compiler Version

Criticality	Minor / Informative
Location	contracts/DCASK.sol#L2
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 ^0.8.24;
```

Recommendation

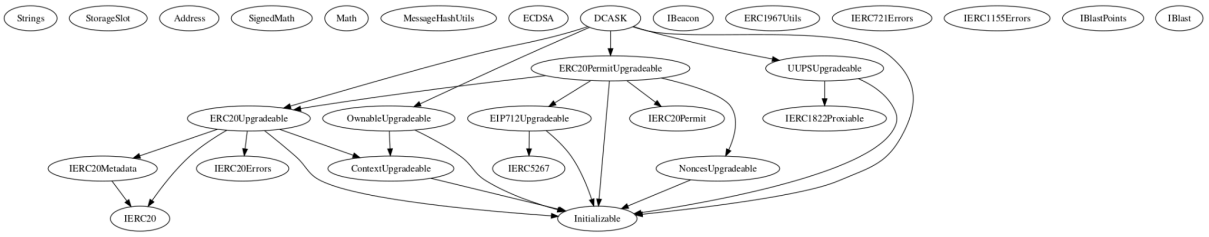
The team is advised to lock the pragma to ensure the stability of the codebase. The locked pragma version ensures that the contract will not be deployed with an unexpected version. An unexpected version may produce vulnerabilities and undiscovered bugs. The compiler should be configured to the lowest 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

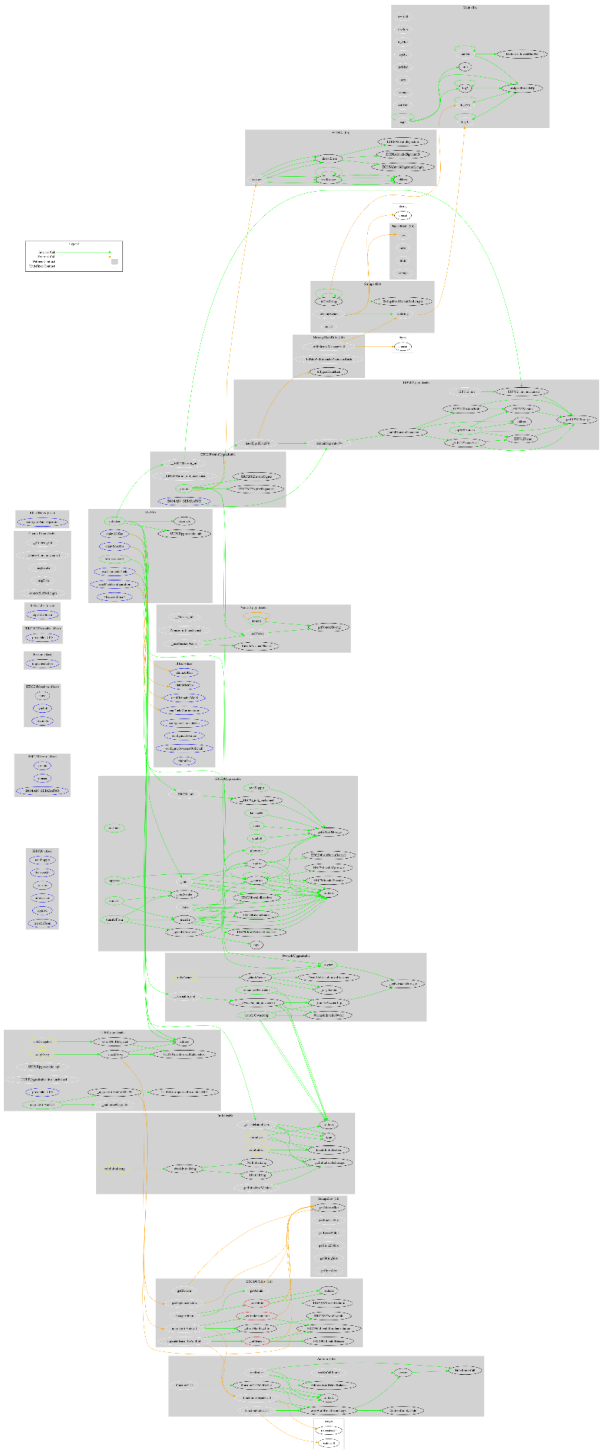
Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
DCASK	Implementation	Initializable, ERC20Upgradeable, ERC20PermitUpgradeable, OwnableUpgradeable, UUPSUpgradeable		
		Public	✓	-
	initialize	Public	✓	initializer
	_authorizeUpgrade	Internal	✓	onlyOwner
	claimAllGas	External	✓	onlyOwner
	claimMaxGas	External	✓	onlyOwner
	readClaimableYield	External		-
	readYieldConfiguration	External		-
		External	Payable	-
IBlastPoints	Interface			
	configurePointsOperator	External	✓	-
IBlast	Interface			
	configureClaimableGas	External	✓	-
	configureGovernor	External	✓	-

	configureGovernorOnBehalf	External	✓	-
	claimAllGas	External	✓	-
	claimMaxGas	External	✓	-
	claimGas	External	✓	-
	readClaimableYield	External		-
	readYieldConfiguration	External		-

Inheritance Graph



Flow Graph



Summary

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

Initial Audit, 18 June 2024

The contract is an upgrade proxy contract.

At the time of the audit report, the contract has not been deployed to the blast mainnet.

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