HackSafe

Smart Contract Security Audit Report

Quant

June 2022

Security Status

★★★★½

www.hacksafe.io

Audit Details

Audited project
Quant

Deployer address
0xF5E38bBEDC78EfeA055e0C56035AdB320E64C4Bc

Client contacts
Quant team

Blockchain
Ethereum

Website
https://quant.network/

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the 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 below 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 TechRate and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (HackSafe) owe no duty of care towards you or any other person, nor does HackSafe 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 HackSafe 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, HackSafe hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against HackSafe, 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 alone. No applications or operations were reviewed for security. No product code has been reviewed.

Procedure

Step 1 - In-Depth Manual Review

Manual line-by-line code reviews to ensure the logic behind each function is sound and safe from various attack vectors. This is the most important and lengthy portion of the audit process (as automated tools often cannot find the nuances that lead to exploits such as flash loan attacks).

Step 2 - Automated Testing

Simulation of a variety of interactions with your Smart Contract on a test blockchain leveraging a combination of automated test tools and manual testing to determine if any security vulnerabilities exist.

Step 3 – Leadership Review

The engineers assigned to the audit will schedule meetings with our leadership team to review the contracts, any comments or findings, and ask questions to further apply adversarial thinking to discuss less common attack vectors.

Step 4 - Resolution of Issues

Consulting with the team to provide our recommendations to ensure the code's security and optimize its gas efficiency, if possible. We assist project team's in resolving any outstanding issues or implementing our recommendations.

Step 5 - Published Audit Report

Boiling down results and findings into an easy-to-read report tailored to the project. Our audit reports highlight resolved issues and any risks that exist to the project or its users, along with any remaining suggested remediation measures. Diagrams are included at the end of each report to help users understand the interactions which occur within the project.

Background

HackSafe was commissioned by Quant to perform an audit of smart contracts:

- https://etherscan.io/address/0x4a220e6096b25eadb88358cb44068a3248254675#code

The purpose of the audit was to achieve the

- Ensutre that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

The information in this report should be understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Contract Details

Token contract details for 17.06.2022

Token Type	: ERC20
Contract name	: StandardToken
Contract address	: 0x4a220E6096B25EADb88358cb44068A3248254675
Compiler version	: v0.4.21+commit.dfe3193c
Total supply	: 45,467,000
Token Ticker	: QNT
Decimals	: 18
Token Holders	: 52,258
Top 100 token holder's dominance	: 38.84%
Transactions count	: 737,141
Contract deployer address	: 0xF5E38bBEDC78EfeA055e0C56035AdB320E64C4Bc
Crowdsale address	: 0x398e41ac3d5972b4bac2320cd130c7a25ca446f7

Social profiles

Twitter Profile	: https://twitter.com/quant_network
Github Profile	: https://github.com/quantnetwork
Facebook Profile	: https://www.facebook.com/quantnetwork
Whitepaper link	http://files.quant.network/files.quant.network/Quant_Overledger_Whitepaper_v0.1.pdf
Telegram Profile	: https://t.me/quantnetworkannouncements
Medium Profile	: https://medium.com/@quant_network
Coinmarketcap profile	: https://coinmarketcap.com/currencies/quant/
Coingecko profile	: https://www.coingecko.com/en/coins/quant
Reddit profile	: https://www.reddit.com/r/QuantNetwork/

Claimed Smart Contract Features

Claimed Feature Detail	Our Observation
Tokenomics : • **Name** : Quant • **Symbol** : QNT • **Decimals** : 18 • **Protocol** : ERC20 • **Max Total supply** : 45,467,000	Yes, This is valid.

Audit Summary

According to the standard audit assessment, Customer`s solidity smart contracts are “Secure”. This token contract does not contain owner control, which do make it fully decentralized as owner does not have control over smart contract.

Insecure	Poor secured	Secure	Well-secured

You are here 

We used various tools like Slither, Mythril and Remix IDE. At the same time this finding is based on critical analysis of the manual audit. All issues found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the Audit overview section. General overview is presented in AS-IS section and all identified issues can be found in the Audit overview section.

We found 0 critical, 0 high, 0 medium and 2 low and some very low-level issues. These issues are not critical ones.

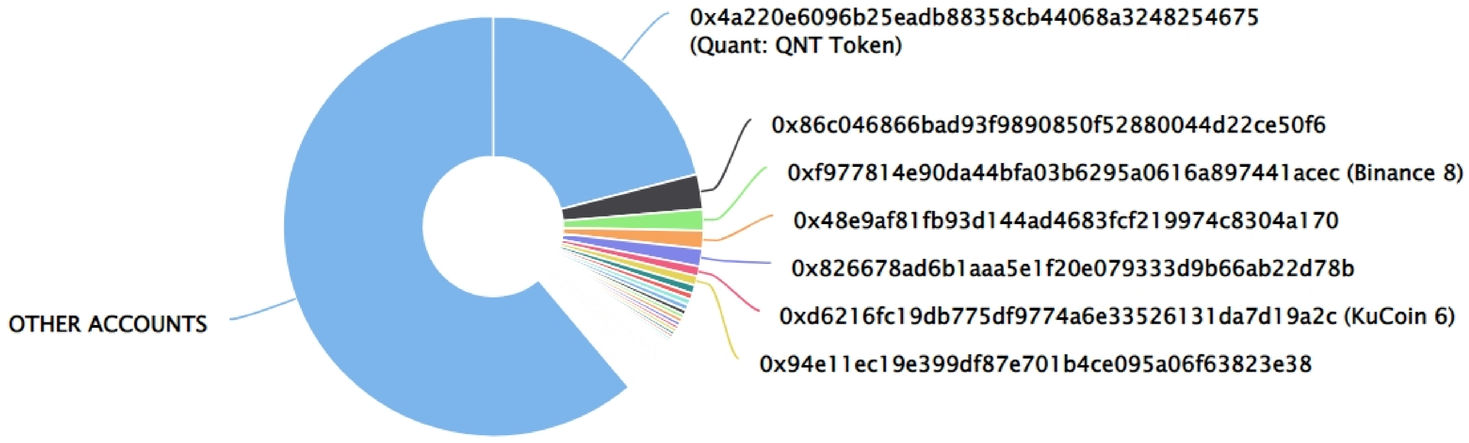
Quant Distribution

The top 100 holders collectively own 38.84% (17,661,092.43 Tokens) of Quant

Token Total Supply: 45,467,000.00 Token | Total Token Holders: 52,262

Quant Top 100 Token Holders

Source: Etherscan.io



Quant Top 20 Token Holders

(A total of 17,661,092.43 tokens held by the top 100 accounts from the total supply of 45,467,000.00 token)

Rank	Address	Quantity (Token)	Percentage
1	Quant: QNT Token	9,549,784.302692691442475987	21.0038%
2	0x86c046866bad93f9890850f52880044d22ce50f6	1,225,376.130196895	2.6951%
3	Binance 8	737,726	1.6226%
4	0x48e9af81fb93d144ad4683fcf219974c8304a170	625,776	1.3763%
5	0x826678ad6b1aaa5e1f20e079333d9b66ab22d78b	622,807	1.3698%
6	KuCoin 6	340,408	0.7487%
7	0x94e11ec19e399df87e701b4ce095a06f63823e38	326,138.426270244199400878	0.7173%
8	0xded9c9b6177fb8d6e5e1c476b80b9e67e9cc0eec	290,215.487662348663907536	0.6383%
9	0x9fa2dd9fab3a7698bb51615b14039f4bc4d61187	215,667	0.4743%
10	CoinMetro: MultiSig Cold Wallet	213,934.103	0.4705%
11	Bitrue	190,791.528288168432895011	0.4196%
12	0x5575425b4c1c6cbd772fc34a8f6d2005a0daba65	166,749.630501043005727284	0.3667%
13	IDEX	149,372.58747843296352473	0.3285%
14	Bittrex	146,604.753724990000743226	0.3224%
15	0x5a52e96bacdabb82fd05763e25335261b270efcb	138,610	0.3049%
16	0x19184ab45c40c2920b0e0e31413b9434abd243ed	127,593.766	0.2806%
17	Crypto.com	119,569.062200026300396402	0.2630%
18	0x4d031618cd3f6384a9fd95fc78d1e9e86a2b8f1e	117,593.02	0.2586%
19	0x744903c4e1b331a39848528e91c31234a6a33944	100,000.538679930547615478	0.2199%
20	0xec60a69dcd3ac1858929d1e5c75cc2995ccc3330	100,000.473192842247701798	0.2199%

Quant Distribution

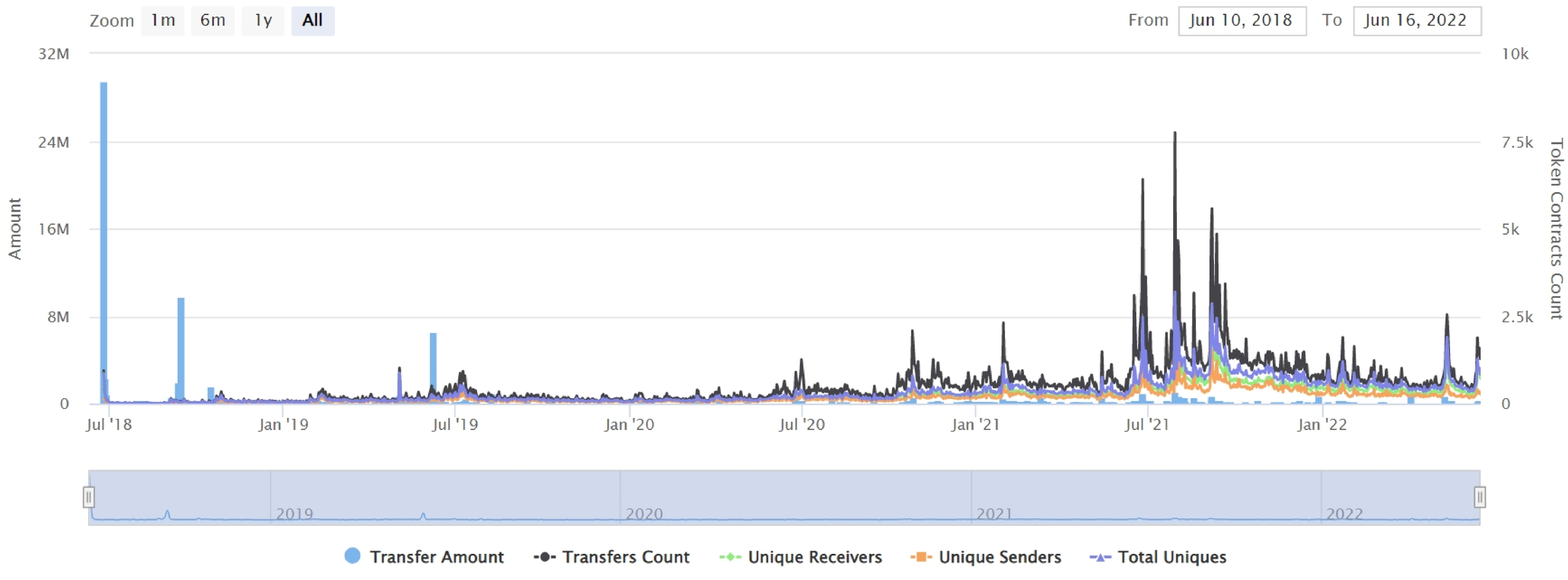
Quant Contract Overview

Time Series: Token Contract Overview

Mon 25, Jun 2018 - Thu 16, Jun 2022

Token Contract 0x4a220e6096b25eadb88358cb44068a3248254675 (Quant)

Source: Etherscan.io



Contract functions details

+ ERC20Basic
    -[Pub] totalSupply
    -[Pub] balanceOf
    -[Pub] transfer

+[Lib] SafeMath
    -[Int] mul
    -[Int] sub
    -[Int] div
    -[Int] add

+ BasicToken (ERC20Basic)
    -[Pub] totalSupply
    -[Pub] transfer #
    -[Pub] balanceOf

+ ERC20 (ERC20Basic)
    -[Pub] allowance
    -[Pub] transferFrom #
    -[Pub] approve #

+ StandardToken (ERC20, BasicToken)
    -[Pub] StandardToken
    -[Pub] mint #
      -modifiers: onlyCrowdsale
    -[Pub] transferFrom #
    -[Pub] approve #
    -[Pub] allowance
    -[Pub] increaseApproval #
    -[Pub] decreaseApproval #

($) = payable function
= non-constant function

Issues Checking Status

No.	Title	Status
1.	Unlocked Compiler Version	Low issue
2.	Missing Input Validation	Passed
3.	Race conditions and Reentrancy. Cross-function race conditions.	Passed
4.	Possible delays in data delivery	Passed
5.	Oracle calls.	Passed
6.	Timestamp dependence.	Passed
7.	Integer Overflow and Underflow	Passed
8.	DoS with Revert.	Passed
9.	DoS with block gas limit.	Passed
10.	Methods execution permissions.	Passed
11.	Economy model of the contract.	Passed
12.	Private use data leaks.	Passed
13.	Malicious Event log.	Passed
14.	Scoping and Declarations.	Passed
15.	Uninitialized storage pointers.	Passed
16.	Arithmetic accuracy.	Passed
17.	Design Logic.	Low issue
18.	Safe Open Zeppelin contracts implementation and usage.	Passed
19.	Incorrect Naming State Variable	Passed

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to assets loss or data manipulations.
High	High-level vulnerabilities are difficult to exploit; however, they also have a significant impact on smart contract execution, e.g., public access to crucial functions
Medium	Medium-level vulnerabilities are important to fix; however, they can't lead to assets loss or data manipulations.
Low	Low-level vulnerabilities are mostly related to outdated, unused, etc. code snippets that can't have a significant impact on execution.

Security Issues

Critical Severity Issues

No critical severity issue found.

High Severity Issues

No high severity issue found.

Medium Severity Issues

No medium severity issues found.

Low Severity Issues

Two low severity issue found.

1. Unlocked Compiler Version.

- **Description**

  The contract utilizes an unlocked compiler version. An unlocked compiler version in the contract's source code permits the user to compile it at or above a particular version. This, in turn, leads to differences in the generated bytecode between compilations due to differing compiler version numbers. This can lead to ambiguity when debugging as compiler-specific bugs may occur in the codebase that would be difficult to identify over a span of multiple compiler versions rather than a specific one.

- **Recommendation**

  It is advisable that the compiler version is alternatively locked at the lowest version possible so that the contract can be compiled. For example, for version v0.4.21 the contract should contain the following line:

  pragma solidity 0.4.21;

2. Design logic

Description

Maximum total supply has been set 45,467,000 but onlyCrowdsale address can mint new tokens which is not ideal as minting amount is not even adding to total supply. It does not maintain the total supply equals to circulating supply.

Location: mint function

Recommendation

We advise you to either remove mint function as maximum total supply has been already decided in to code or add minting amount to total supply.

Centralization

Owner Privileges (in the period when the owner is not renounced) :

- Quant Contract:
    - Crowdsale address can mint new tokens

Conclusion

Smart contract contains two low severity issues! The further transfer and operations with the fund raised are not related to this particular contract.

HackSafe note: Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by Owner.