MOTECH AUDIT

SMART CONTRACT SECURITY AUDIT

SECURITY ASSESSMENT

2021

OMG NETWORK PROGRESS REPORT

12 OCT 2021

SECURITY ASSESMENT

TABLE OF CONTENTS

Finding

Conclusion

SUMMARY

This report has been prepared for OMG Network smart contracts to discovering issues and vulnerabilitiesin the source code of their Smart Contract and any contract dependencies that were not part of anofficially recognized library. A comprehensive examination has been performed, utilizing Static Analysis andManual Review techniques.

The auditing process pays special attention to the following considerations:

- Testing the smart contracts against both common and uncommon attack vectors.
- Assessing the codebase to ensure compliance with current best practices and industry standards.
- Ensuring contract logic meets the specifications and intentions of the client.
- Cross-referencing contract structure and implementation against similar smart contracts producedby industry leaders.
- Thorough line-by-line manual review of the entire codebase by industry experts.

The security assessment resulted in findings that ranged from minor to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices. We suggest recommendations that could better serve the project from the security perspective:

- Enhance general coding practices for better structures of source codes;
- Add enough unit tests to cover the possible use cases given they are currently missing in therepository;
- Provide more comments per each function for readability, especially contracts that are verified inpublic;
- Provide more transparency on privileged activities once the protocol is live.

BACKGROUND

MotechAudit was commissioned by OMG Network to perform an audit of smart contracts:
https://etherscan.io/address/0xd26114cd6EE289AccF82350c8d8487fedB8A0C07
The purpose of the audit was to achieve the following:

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

The information in this report should be used to 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.

SECURITY ASSESMENT

AUDIT DETAILS

AUDITED PROJECT

OMG Network

DEPLOYER ADDRESS

0x140427a7D27144A4cDa83bD6b9052a63b0c5B589

CLIENT CONTACTS:

OMG Network team

BLOCKCHAIN

ETHEREUM Project

WEBSITE:

https://omg.network/

MOTECH AUDIT

SMART CONTRACT SECURITY AUDIT

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 MotechAudit and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (MotechAudit) owe no duty of care towards you or any other person, nor does MotechAudit 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 MotechAudit 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, MotechAudit hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against MotechAudit, 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.

CONTRACT DETAILS

Token contract details for Aug-08-2021

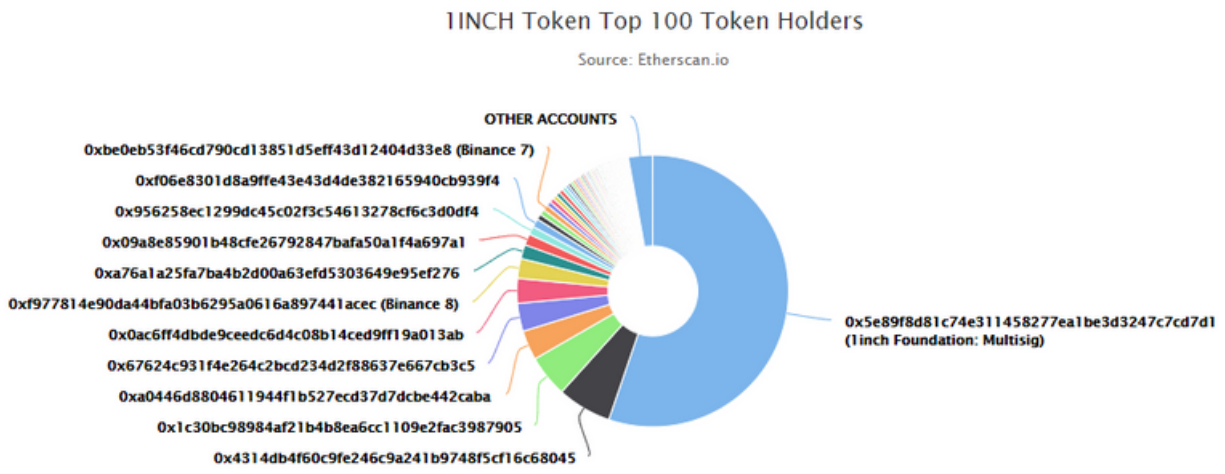
Contract name	**OMG Network**
Contract address	**0xd26114cd6EE289AccF82350c8d8487fedB8A0C07**
Total supply	**140,245,398.245132780789239631**
Token ticker	**OMG Network (OMG)**
Decimals	**18**
Token holders	**695,447**
Transactions count	**3,417,444**
Top 100 holders dominance	**6.426%**
Contract deployer address	**0x140427a7D27144A4cDa83bD6b9052a63b0c5B589**
Contract's current owner address	**0x140427a7D27144A4cDa83bD6b9052a63b0c5B589**

MOTECH AUDIT
SMART CONTRACT SECURITY AUDIT

OMG NETWORK DISTRIBUTION

The top 100 holders collectively own 97.11% (1,456,603,962.91 Tokens) of 1INCH Token

Token Total Supply: 1,500,000,000.00 Token | Total Token Holders: 73,296

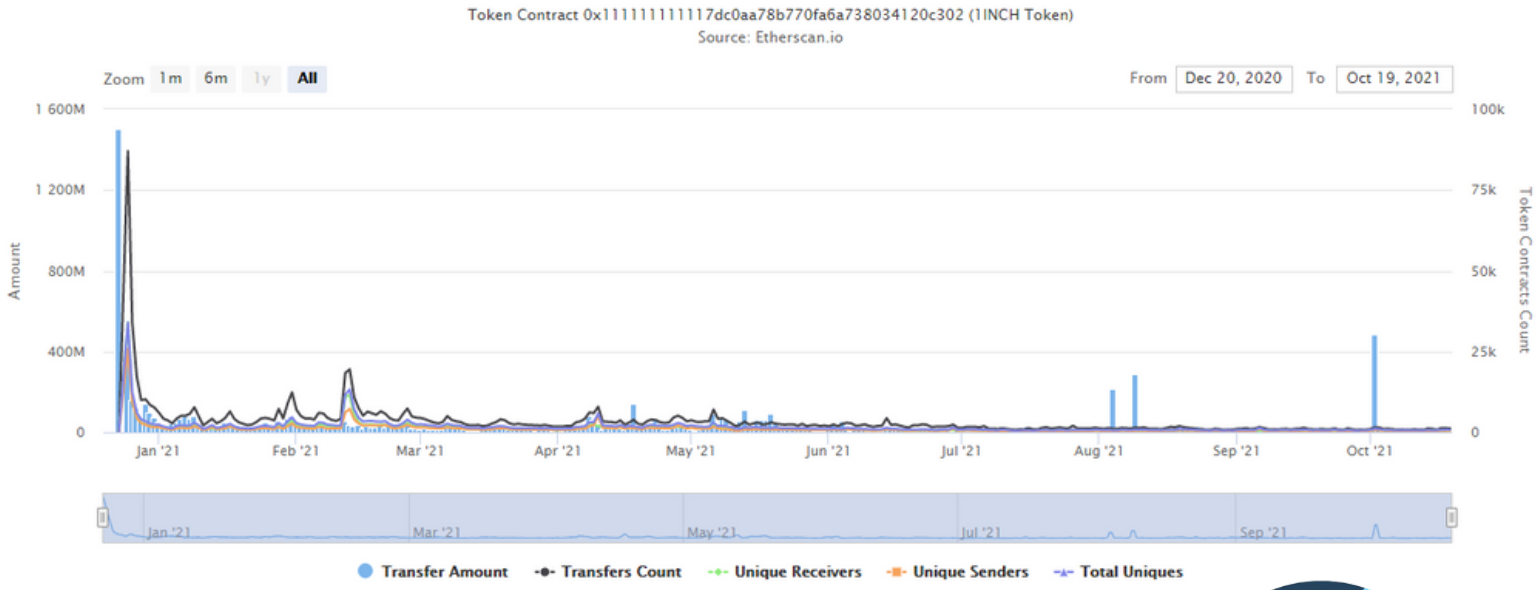


(A total of 1,456,603,962.91 tokens held by the top 100 accounts from the total supply of 1,500,000,000.00 token)

OMG NETWORK CONTRACT INTERACTION DETAILS

Time Series: Token Contract Overview

Wed 23, Dec 2020 - Tue 19, Oct 2021



MOTECH AUDIT
SMART CONTRACT SECURITY AUDIT

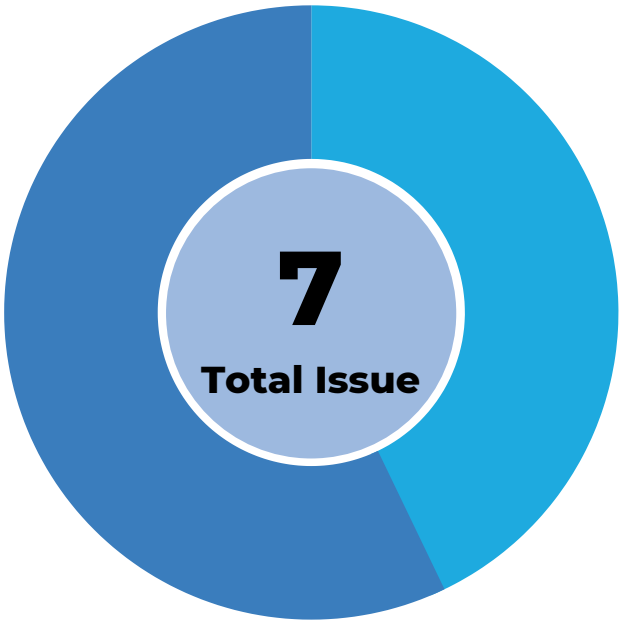
TOP 10 TOKEN HOLDERS

Rank	Address	Quantity	Percentage	Value	Analytics
1	1inch Foundation: Multisig	828,932,718.494516890874324592	55.2622%	$3,058,761,731.24	
2	0x4314db4f60c9fe246c9a241b9748f5cf16c68045	96,500,000	6.4333%	$356,085,000.00	
3	0x1c30bc98984af21b4b8ea6cc1109e2fac3987905	75,000,000	5.0000%	$276,750,000.00	
4	0xa0446d8804611944f1b527ecd37d7dcbe442caba	52,902,688.081805187737046637	3.5268%	$195,210,919.02	
5	0x67624c931f4e264c2bcd234d2f88637e667cb3c5	50,000,000	3.3333%	$184,500,000.00	
6	0x0ac6ff4dbde9ceedc6d4c08b14ced9ff19a013ab	44,295,000	2.9530%	$163,448,550.00	
7	Binance 8	35,332,879.1537354251502	2.3555%	$130,378,324.08	
8	0xa76a1a25fa7ba4b2d00a63efd5303649e95ef276	25,000,000	1.6667%	$92,250,000.00	
9	0x09a8e85901b48cfe26792847bafa50a1f4a697a1	20,000,000	1.3333%	$73,800,000.00	
10	0x956258ec1299dc45c02f3c54613278cf6c3d0df4	15,000,000	1.0000%	$55,350,000.00	

source:etherscan.io

SECURITY ASSESMENT

FINDINGS



- Medium 0(0.00%)
- Major 0 (0.00%)
- Minor 3 (42.86%)
- Critical 0 (0.00%)
- Informational 4 (57.14%)
- Discussion 0 (0.00)

ID	Tittle	Category	severity	Satuts
QCK-01	Redundant Member bondValue in StructTicket	Gas Optimization, CodingStyle	Informational	Resolved
QCK-02	Function Should be Declared External	Gas Optimization	Informational	Resolved
QCK-03	Centralization Risks	Centralization / Privilege	Minor	Resolved
QCK-04	Logic Related to IFE Claims and OwedAmount	Logical Issue	Minor	Resolved
QPC-01	Function Should be Declared External	Gas Optimization	Informational	Resolved
QPC-02	Lack of Checks for Reentrancy	Logical Issue	Minor	Resolved
QTC-01	Deployment Risks	Centralization / Privilege	Informational	Resolved

MOTECH AUDIT
SMART CONTRACT SECURITY AUDIT

SECURITY ASSESMENT

QCK-01 | REDUNDANT MEMBER BONDVALUE IN STRUCT TICKET

Category	Severity	Location	Status
Gas Optimization, Coding Style	Informational	contracts/quasar/Quasar.sol: 57, 174	Resolved

Description

The state bondValue of the contract is immutable after contract initialization. According to the code implementation in L174 and L205, the field bondValue within any Ticket STRUCT instance would be initialized as the same value as the state bondValue of this CONTRACT. Afterwards, the state bondValue within any struct instance will not be mutated after struct initialization. Therefore, the member bondValuein struct Ticket is unnecessary. It can be replaced with the state bondValue of the contract whenever used.

Recommendation

It is highly recommended to remove the member bondValue from the struct Ticket and use state variable bondValue of the contract to replace ticket.bondValue.

Alleviation

The OMG Network team heeded our advice and resolved this issue in the commit e0a304c29cf878f54e2bea98bdd99c4b0df0b685.

QCK-02 | FUNCTION SHOULD BE DECLARED EXTERNAL

Category	Severity	Location	Status
Gas Optimization,	Informational	contracts/quasar/Quasar.sol: 119, 128, 143, 150, 157, 173, 222, 247, 279, 320	Resolved

Description

The functions which are never called internally within the contract should have external visibility. Forexample：

- Quasar.setSafeBlockMargin()
- Quasar.flushExpiredTicket()
- Quasar.pauseQuasar()
- Quasar.resumeQuasar()
- Quasar.withdrawUnclaimedBonds()
- Quasar.obtainTicket()
- Quasar.claim()
- Quasar.ifeClaim()
- Quasar.challengeIfeClaim()
- Quasar.processIfeClaim()

Recommendation

It is highly recommended to change the visibility of the aforementioned functions from public to external for gas optimization.

Alleviation

The Quasar contract's bytecode size is very close to the EIP-170 limit. Using an external function withcalldata parameters increases the bytecode size. The OMG Network team changed public to externalwhere it is possible in the commit e0a304c29cf878f54e2bea98bdd99c4b0df0b685.

QCK-03 | CENTRALIZATION RISKS

Category	Severity	Location	Status
Centralization / Privilege	Minor	contracts/quasar/Quasar.sol: 119, 143, 150, 157	Resolved

Description

The role quasarMaintainer has authority to:

- modify safe block margin by calling Quasar.setSafeBlockMargin();
- pause the contract by calling Quasar.pauseQuasar();
- resume the contract by calling Quasar.resumeQuasar();
- withdraw unclaimed bonds by calling Quasar.withdrawUnclaimedBonds().

Recommendation

We advise the client to handle the quasar Maintainer account carefully to avoid any potential hack. We also advise the client to consider the following solutions:

1. Apply an associated Timelock contract to implement above functions, with reasonable latency for community awareness on privileged operations;
2. Apply Multisig with community-voted 3rd-party independent co-signers;
3. Apply DAO or Governance module to increase transparency and community involvement.

Alleviation

The OMG Network team implemented the library TimelockedValue to update the safe block margin withlatency in the commit e0a304c29cf878f54e2bea98bdd99c4b0df0b685.

[**OMG Network Team**]: While we agree that the quasarMaintainer account should be carefully managed, the effects of it being compromised are minimal and most of the maintainer methods would not affect the users.

- pauseQuasar() only prevents new withdrawals from being started. Existing withdrawals have hadtheir funds reserved and can continue as normal without fear of losing funds.
- withdrawUnclaimedBonds() can only withdraw funds that are destined for the quasarMaintaineranyway and so should not be considered a Centralization risk as no user funds are in dangerset
- SafeBlockNumber() does protect the liquidity pool in the event that the plasma chain goesbyzantine and the Plasma operator continues publishing blocks. We have added a timelock to this method to warn liquidity providers when safeBlockNum is changed and allow them time to withdrawtheir funds if they don't agree with it.

QCK-04 | LOGIC RELATED TO IFE CLAIMS AND OWED AMOUNT

Category	Severity	Location	Status
Logical Issue	Minor	contracts/quasar/Quasar.sol: 1	Resolved

Description

According to the code implementation, if a bad IFE claim is not challenged within the eight-day limitation, itwould finally get processed. In this case, the attacker could withdraw the tokens that do not belong to himfrom the contract. This might lead to the contract not having enough balance to pay other users' claimslater.

We noticed that in the contract QuasarPool, users are allowed to send a certain amount of tokens (amountnot exceeding tokenData[token].owedAmount), to the contract account by calling the functionQuasarPool.repayOwedToken(). We hope to confirm with the team about the using scenarios of thefunction: if the function QuasarPool.repayOwedToken() and the variable tokenData[token].owedAmountare designed to handle the situation when a bad IFE claim is processed.

Alleviation

[OMG Network Team]: IFE claims are intended as a way of making sure that the user does not lose fundsin the event that the user initiated a withdrawal and sent funds to the quasarOwner, but the Plasmaoperator does not include the transaction in a block. This mirrors the Plasma MoreVP protocol. One of thesecurity assumptions of Plasma is that users are able to monitor invalid transactions or IFEs and challengethem. This holds true for Quasar users as well - they can either check for invalid IFEs once every 8 days, ortrust someone else to do that for them.However, in the unlikely event that an invalid IFE does get processed, then yes, the quasarOwner can makeup the funds by calling repayOwedToken(). Note that this means that users must trust the quasarOwner todo the right thing at cost to themself. The assumption is that users would prefer to monitor and challengeinvalid IFEs instead.

QPC-01 | FUNCTION SHOULD BE DECLARED EXTERNAL

Category	Severity	Location	Status
Gas Optimization	Informational	contracts/quasar/QuasarPool.sol: 40, 50, 79, 110, 120	Resolved

Description

The functions which are never called internally within the contract should have external visibility. For example：

- QuasarPool.addEthCapacity()
- QuasarPool.addTokenCapacity()
- QuasarPool.withdrawFunds()
- QuasarPool.registerQToken()
- QuasarPool.repayOwedToken()

Recommendation

It is highly recommended to change the visibility of the aforementioned functions from public toexternal for gas optimization

Alleviation

The OMG Network team heeded our advice and resolved this issue in the commite0a304c29cf878f54e2bea98bdd99c4b0df0b685.

QPC-02 | LACK OF CHECKS FOR REENTRANCY

Category	Severity	Location	Status
Logical Issue	Minor	contracts/quasar/QuasarPool.sol: 50, 79, 120	Resolved

Description

Functions that contain state updates or event emits after external calls are vulnerable to potentialreentrancy attacks. For example,

- QuasarPool.addTokenCapacity()
- QuasarPool.withdrawFunds()
- QuasarPool.repayOwedToken()

Recommendation

It is highly recommended to apply OpenZeppelin ReentrancyGuard library - nonReentrant modifier for theaforementioned functions to prevent any potential reentrancy attack.

Alleviation

The OMG Network team heeded our advice and resolved this issue in the commite0a304c29cf878f54e2bea98bdd99c4b0df0b685.

QPC-01 | FUNCTION SHOULD BE DECLARED EXTERNAL

Category	Severity	Location	Status
Centralization / Privilege	Informational	contracts/quasar/QToken.sol: 22~23, 31	Resolved

Description

According to the contract implementation, the owner account quasarContract is capable to mint anunlimited amount of tokens by calling the function mint. On the other hand, quasarContract is capable toburn all the amount of tokens of an account without any restriction. The concern is if quasarContract isnot set up properly, or it accidentally calls the aforementioned functions, it might cause some unexpectedloss, thus introducing centralization risks.

Recommendation

We advise the team to review the flow and confirm if it is an intended design. If the owner quasarContractis designed to be the contract QuasarPool, please ensure quasarContract is set up properly, and QTokenis always bundled with the contract QuasarPool to work together, since the contract QToken is vulnerablealone.

Alleviation

[OMG Network Team]: That's correct, the owner of the QToken contract (stored as quasarContract)should be set as the address of the QuasarPool contract. If this has been initialized incorrectly, then theQuasarPool for that ERC20 token won't work and it should not be used.

CONCLUSION

Smart contracts contain owner privileges!

TechRate 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.