April 2022

QuickSwap Security Audit Report

Security Assessment

Z INSTITUTE

Summary

This report has been prepared for QuickSwap to discover issues and vulnerabilities in the source code of the staking contract as well as any contract dependencies that were not part of an officially recognized library. A comprehensive examination has been performed, utilizing Static Analysis, Manual Review, and Testnet Deployment 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 produced by industry leaders
- Thorough line-by-line manual review of the entire codebase by industry experts

The security assessment resulted in findings that ranged from critical 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
- Provide more comments per each function for readability, especially contracts that are verified in public
- Provide more transparency on privileged activities once the protocol is live

Overview

Project Summary

Project Name	QuickSwap Token Split
Description	The redenominated QUICK token will become xQUICK with an increase in total supply from 1 million to 1 billion.
Platform	Polygon
Language	Solidity
Codebase	`https://github.com/QuickSwap/token-swap.git`
Commits	`fdeb9f4560708cfba73d117e2ae1ed2d05cb61b1`

Audit Summary

Delivery Date	April. 19, 2022
Audit Methodology	Static Analysis, Manual Review
Key Components	Quick, QuickToken

Audit Scope

Name	**File**	**SHA Checksum**
QuickToken	`contracts/QuickEthereum.sol`	`4ccdfefc125eee160a333360bc6ced80e7858badeafe63796464af0b701bd48b`
Quick	`contracts/QuickPolygon.sol`	`12527776bc18061fab2e0d58edadd1619fd7a4125a9e4a715f2c82705bd6a4d6`

Contents

1 Understandings

1.1 Overview

To help facilitate further integrations and attract new buyers who are currently put off by QUICK's high price per unit, the governance proposal[1] on the QUICK token split has passed with an increase in total supply of QUICK from 1 million to 1 billion. This means that a QUICK holder who owns 1 QUICK now would own 1000 xQUICK after the split. A TokenSwap contract will be deployed on Polygon to perform the exchange of QUICK and xQUICK tokens.

1.2 Privileged functions

1.2.1 Quick (on Polygon)

- `deposit()`: As declared in the documentation[2], this function is called by the `ChildChainManagerProxy` contract whenever a deposit is initiated from the root chain (Ethereum). This deposit function internally mints the token on the child chain (Polygon). The `ChildChainManagerProxy`'s address is denoted as the `gateway` variable, specified on contract deployment.

1.2.2 QuickToken (on Ethereum)

- `mint()`: Only the minter address can mint new tokens.
- `setMinter()`: Only the current minter can set a new minter. Initially, the minter address is specified on contract deployment.

[1] https://quickswap-layer2.medium.com/should-quickswap-do-a-token-split-to-make-quick-more-appealing-to-investors-e739cf2cac98

[2] https://docs.polygon.technology/docs/develop/ethereum-polygon/pos/mapping-assets/#custom-child-token

1.3 Consistency with specifications

We remark that the token contracts meet specification requirements and detail them below.

- The total supply of the QUICK token will be increased to 1 billion.
- A QUICK holder who owns 1 QUICK can swap for 1000 xQUICK after the split.

1.4 Event logging

After our examination, we remark that all of the main functions listed below have excellent event logs:

- `deposit()`
- `withdraw()`
- `mint()`
- `setMinter()`
- `transfer()`
- `transferFrom()`
- `approve()`

We refer to the link[3] for the importance of the event logs.

1.5 Difference Check

Since the contracts are a modification of the already deployed contracts in production, we highlight the lines of differences between the new and the deployed contracts below.

[3] https://consensys.github.io/smart-contract-best-practices/recommendations/#use-events-to-monitor-contract-activity

1.5.1 QuickEthereum

DEPLOYED CONTRACT:
https://etherscan.io/address/0x6c28aef8977c9b773996d0e8376d2ee379446f2f#code

LINES CHANGED:

```
- string public constant name = "Quickswap";
+ string public constant name = "QuickSwap";
...
- uint public totalSupply = 1000000000000000000000000;
+ uint public totalSupply = 1_000_000_000e18;
```

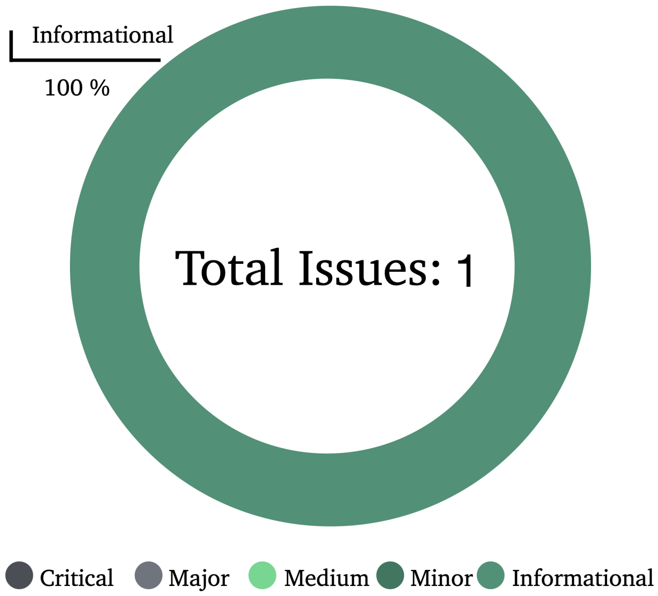
1.5.2 QuickPolygon

DEPLOYED CONTRACT:
https://polygonscan.com/address/0x831753dd7087cac61ab5644b308642cc1c33dc13#code

LINES CHANGED:

```
- string public constant name = "Quickswap";
+ string public constant name = "QuickSwap";
```

2 Findings



ID	Title	Category	Severity	Status
SSL-01	Potential Increase on Total Supply	Logical Issue	Informational	Resolved

2.1 SSL-01 | Potential Increase on Total Supply

Category	Severity	Location	Status
Logical Issue	Informational	`contracts/QuickEthereum.sol: line 261`	Resolved

2.1.1 Description

On contract deployment, the 1 billion total supply of the tokens will be minted and sent to the initial address specified. The timestamp after which minting may occur will also be determined on contract construction and cannot be further modified.

Although the current token supply cap is fixed at 1 billion, the mint() function allows the minter to mint new tokens after the specified timestamp. Thus, the total supply could potentially increase once the minting happens.

```
function mint(address dst, uint rawAmount) external {
    require(msg.sender == minter, "Quick::mint: only the minter can mint");
    require(block.timestamp >= mintingAllowedAfter, "Quick::mint: minting not
    allowed yet");
    ....
}
```

2.1.2 Response

The mint function is taken from our previous token contract which is already deployed. As per our tokenomics, 2% of supply will be minted every year to tackle inflation starting from 4th year of token launch.

Appendices

Appendix A

Finding Categories

Gas Optimization Gas Optimization findings do not affect the functionality of the code but generate different, more optimal compiled code resulting in a reduction on the total gas cost of a transaction.

Mathematical Operations Mathematical Operation findings relate to mishandling of math formulas, such as overflows, incorrect operations etc. Logical Issue Logical Issue findings detail a fault in the logic of the linked code, such as an incorrect notion on how block.timestamp works.

Control Flow Control Flow findings concern the access control imposed on functions, such as owner-only functions being invoke-able by anyone under certain circumstances.

Volatile Code Volatile Code findings refer to segments of code that behave unexpectedly on certain edge cases that may result in a vulnerability.

Data Flow Data Flow findings describe faults in the way data is handled at rest and in memory, such as the result of a struct assignment operation affecting an in-memory struct rather than an in-storage one.

Language Specific Language Specific findings are issues that would only arise within Solidity, i.e. incorrect usage of private or delete.

Centralization / Privilege Centralization / Privilege findings refer to the logic or implementation of the code exposing to concerns or scenarios that would go against decentralized manners.

Coding Style Coding Style findings usually do not affect the generated byte-code but rather comment on how to make the codebase more legible and, as a result, easily maintainable.

Inconsistency Inconsistency findings refer to functions that should seemingly behave similarly yet contain different code,such as a constructor assignment imposing different require statements on the input variables than a setter function.

Magic Numbers Magic Number findings refer to numeric literals that are expressed in the codebase in their raw format and should otherwise be specified as constant contract variables aiding in their legibility and maintainability.

Compiler Error Compiler Error findings refer to an error in the structure of the code that renders it impossible to compile using the specified version of the project.

Disclaimer

This report is subject to the terms and conditions (including without limitation, description of services,confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to you ("Customer" or the "Company") in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report 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 in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intended 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. The Z Institute's position is that each company and individual are responsible for their own due diligence and continuous security. The Z Institute'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 The Z Institute are subject to dependencies and 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. ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS,OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, THE Z INSTITUTE

HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, THE Z INSTITUTE SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANT ABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM THE COURSE OF DEALING, USAGE, OR TRADE PRACTICE. WITHOUT LIMITING THE FOREGOING, THE Z INSTITUTE MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT REPORT,WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF,WILL MEET THE CUSTOMER'S OR ANY OTHER PERSON'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE. WITHOUT LIMITATION TO THE FOREGOING, THE Z INSTITUTE PROVIDES NO WARRANTY OR UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET CUSTOMER'S REQUIREMENTS,ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE,APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.WITHOUT LIMITING THE FOREGOING, NEITHER THE Z INSTITUTE NOR ANY OF THE Z INSTITUTE'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY,RELIABILITY, OR CURRENCY OF ANY INFORMATION OR CONTENT PROVIDED THROUGH THE SERVICE. THE Z INSTITUTE WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES,OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS.ALL THIRD-PARTY MATERIALS ARE PROVIDED "AS IS" AND ANY REPRESENTATION OR WARRANTY OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN CUSTOMER AND THE THIRD-PARTY OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS.THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO,ANY OTHER PERSON WITHOUT THE Z INSTITUTE'S PRIOR WRITTEN CONSENT IN EACH INSTANCE.NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPA-

NYING MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST THE Z INSTITUTE WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS.THE REPRESENTATIONS AND WARRANTIES OF THE Z INSTITUTE CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST THE Z INSTITUTE WITH RESPECT TO SUCH REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE.FOR AVOIDANCE OF DOUBT, THE SERVICES, INCLUDING ANY ASSOCIATED ASSESSMENT REPORT FOR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

About

The Z Institute is an online blockchain talent incubator and accelerator with a focus on bringing more developers to the space and a blockchain transformer that empowers businesses in technology and research. Since 2017, The Z Institute's founding team has been providing consulting services in customized smart contracts, DApp development, and security audit.

consulting@zinstitute.net