Quantstamp: A decentralized security platform for smart contracts

Steven Stewart and Richard Ma

Quantstamp Technologies Inc.
info@quantstamp.com

1 Overview

The Quantstamp (**QSP**) platform is an automated service, based on blockchain technology, for performing security audits of smart contracts. QSP maintains a history of security reports that are generated whenever an audit of a smart contract is requested. The Quantstamp network (**QN**) stores all reports in a decentralized database, which is updated by decentralized consensus. The QSP token is the key to accessing the software services offered by Quantstamp.

Our platform for verification rewards participants who provide compute resources for performing security checks on smart contracts. These checks make up the *security library* and are run by decentralized verifiers in our network. The Quantstamp protocol ensures the verification and certification of smart contracts is part of the mathematical problem that a verifier needs to solve. Quantstamp tokens are the API keys that are used to access the platform. When submitting smart contracts for checking by the network, the dynamic token transaction fee that is attached to the submission accounts for supply and demand on the QN.

The utility of QSP is best illustrated by example. Suppose a developer plans to deploy a smart contract written in Solidity on Ethereum. There is substantial risk when writing code that accesses a monetary system, and the developer must be careful to ensure that no funds are lost due to vulnerabilities. In order to minimize risk, the developer submits his code for a security audit, and calls the auditing function directly from his wallet by sending a small amount of QSP token to the QN network. Then, the QN broadcasts the audit request and verifiers immediately perform a set of security checks. Upon consensus, the network publishes a security report that summarizes the results. The report classifies issues based on a severity system from 1–10; a 1 is a minor warning, a 10 is a major vulnerability.

When requesting an audit, the developer can choose either a public or private security report. Private reports are encrypted using the public key of the smart contract and can be decrypted by the owner/developer. By using seamless cryptographic hashing, security reports viewed by the public exactly match the audited source code to prevent manipulation of report results.

A developer can perform security audits on a local machine prior to issuing a public audit, but may find that the computational overhead is too high. Verifier nodes in the QN are likely to have greater computational capacity in terms of memory and processing cores than the average developer's machine. Once the code is ready for deployment, the developer is ultimately motivated to produce a public security report in order to give users the reassurance that a decentralized security audit was performed.

When a security report identifies issues found within a smart contract, the developer has the option of publicly annotating the report with feedback. This gives developers the power to indicate false-positives in the report, and the community can validate the annotations.

Although it is not possible to 100% guarantee that source code is flawless, the Quantstamp team will continuously engage in research and development, making regular improvements to the security library. When there are new releases, developers can re-audit their smart contracts, demonstrating their commitment to securing code and increasing public confidence.