STRATIS ICO PLATFORM

Overview

Table of Contents

Introduction

The Stratis ICO (Initial Coin Offering) Platform allows you to run a secure and flexible web-based application that enables buyers to purchase your tokens before the point of initial allocation. The platform has been built from the ground up using C# and the .NET Core.

The ICO Platform enables buyers to earn a proportion of your tokens by making contributions in either BTC or STRAT. However, Changelly is also integrated into the ICO Platform, and buyers can use this service to convert fiat currencies and other cryptocurrencies into STRAT seamlessly.

In addition, the platform is KYC (Know Your Customer) ready and currently supports Onfido integration.

> **Note**
> From now on in this document, the ICO Platform Web Application is simply referred to as the *Web Application*.

Timeline

The timeline for the ICO can be broken down into 3 phases: the deployment phase, the live phase, and the token distribution phase. The following sections introduce each phase:

Deployment phase

During this phase the Web Application is set up. At the end of this phase, you will:

- Have an instance of the Web Application branded and integrated with your site.
- Have obtained a new HD (hierarchal deterministic) wallet in which to collect BTC and STRAT contributed during the ICO.
- Notified potential buyers of when your ICO is due to start.

Live phase

The live phase runs for a specific period, which is configured using the administrator page of the Web Application.

At the end of this period:

- You will have an HD wallet containing all the BTC and STRAT which were contributed during the ICO.
- The ICO Platform will have recorded how many of your tokens are due to each buyer. *Each* individual contribution transaction is stored including the currency used, the amount contributed, and how many tokens were earnt.

Token distribution phase

At the end of this phase, buyers of your tokens will have supplied an address taken from a wallet which supports your tokens. The ICO Platform stores this address with the record of the buyer’s contribution towards earning your tokens.

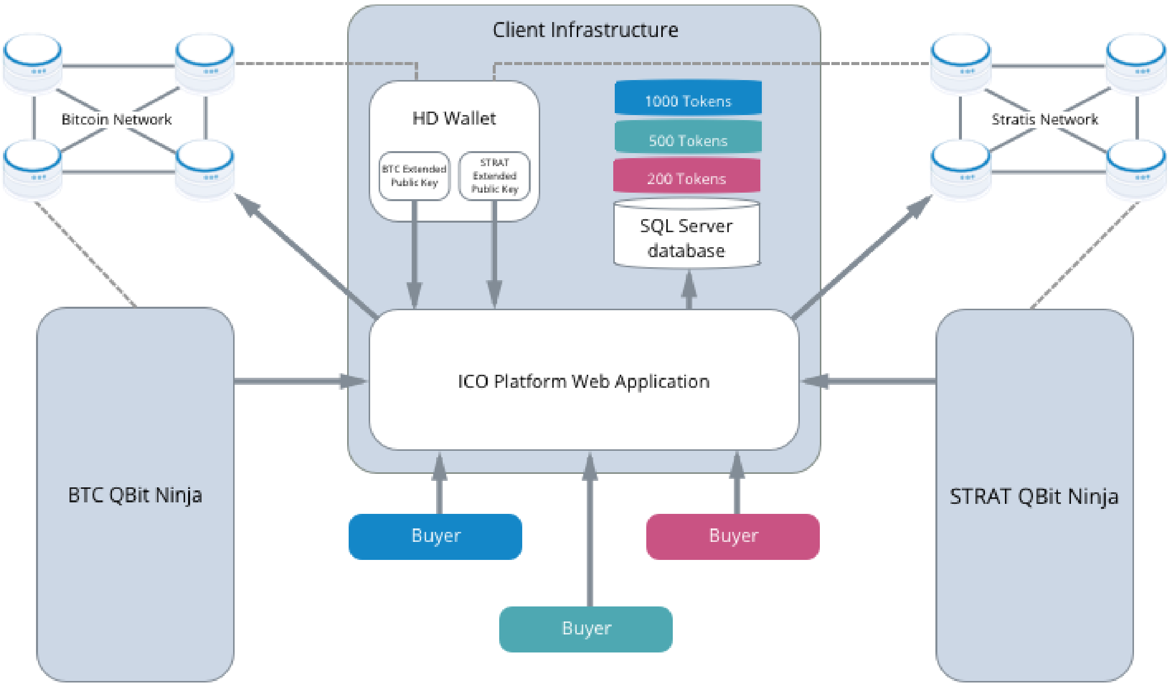
The following chapter covers the Deployment and Live phases in more detail.

Deployment and live phase

The following table lists the components, which you must deploy:

Component	Deployment notes
Web Application	Uses a web-based interface to an ARM template supplied by Stratis Group Ltd.
SQL Server Database	Uses a web-based interface to an ARM template supplied by Stratis Group Ltd.
HD wallet	Use a new wallet.

The following figure shows how these components interact during the live phase:



The following sections give a brief description of the components shown in the previous figure.

Web Application

The Web Application provides a user interface that allows buyers to purchase tokens using either BTC or STRAT. You define the price of your tokens in BTC or STRAT before the ICO begins. Therefore, the application can calculate the number of tokens purchased by a buyer at the point the buyer contributes. The application writes the amount in tokens to the SQL Server database.

The Web Application utilizes the QBit Ninja Web Application(s) hosted by Stratis Group Ltd. to provide buyers with up-to-date totals of the amount of BTC and STRAT they have contributed. This is achieved by querying the unique contribution address that is assigned to each registered user.

SQL Server database

When the ICO finishes, the SQL server database contains information on the purchases made. This includes the following:

1. The buyer’s email address.
2. The amount contributed in STRAT.
3. The amount contributed in BTC.
4. The time of the contribution.
5. The withdrawal address, which your tokens are paid to during the Token Distribution Phase.

Hierarchical deterministic wallet

You are responsible for running an HD wallet to which contributed BTC and STRAT are sent. You use the HD wallet to generate two extended public keys for the Web Application: one for BTC transactions and one for STRAT transactions. To enable the buyers to pay for your tokens using either BTC or STRAT, the Web Application uses the supplied extended public keys to generate public keys for the buyer’s contributions.

To clarify, each transaction representing a buyer contribution is assigned a public key generated from one of the two extended public keys which the HD wallet supplies. Even though it is the responsibility of the Web Application to generate the public keys for the transactions, these public keys map to private keys in the HD wallet. The HD wallet generates the private keys from either the BTC or STRAT private key used to create the original two extended public keys. The buyer contributions are UTXOs, and UTXOs linked to the private keys in a wallet contribute to the balance of the wallet.

Using a single HD wallet ensures that all contributions reside in a single wallet, which is not held on the server running the Web Application. This setup protects the contributed funds if the server is attacked. In addition, the extended public keys cannot be changed after they have been set, which also adds a layer of protection if the platform ever became compromised.

For more information regarding HD wallets, please refer to this Bitcoin Improvement Proposal.

QBit Ninja

QBit Ninja is a web service API for querying blockchains and tracking wallets, which was created by Metaco. The Web Application gets all its transaction data from the BTC instance of QBit Ninja and the STRAT instance of QBit Ninja. Confirmed BTC and STRAT transactions are tied to individual buyers at the point the relevant QBit Ninja instance relays the transactions back to the Web Application. It is at this point the Web Application calculates how much of your token a buyer has earned. You can find more information on QBit Ninja here.