OneChain Wallet Technology

# Introduction

This document is a work in progress, but its ultimate goal is describe the various OneChain ideas and strategies for integrating blockchain wallet technology with the existing OneChain infrastructure.

Throughout this document, we will alternatively discuss wallet technology (general) and Bitcoin wallet technology (specific), but we will not always highlight this shift. It should be understood that these represent the general versus the specific and that the one that provides the best explanation of each topic will be used.

# Security

If the news-grabbing hacks like those of Mt. Gox and The DAO have taught us anything, it’s that security must be a top priority in any cryptocurrency development project. However, this document is much too high-level to delve into the intricacies of this topic and the resources required to implement such security in an appropriate way.

Keep in mind that any technologies outlined here will require an additional layer of development that involves proper security of the entire project. The additional complexity required for such security can represent a significant portion of the overall project requirements and should not be underestimated.

Any mentions of security-related measures in this document will be high-level and general, and they should be treated as such.

# Wallet Overview

In this section, we’ll look at some of the fundamental aspects of a blockchain wallet.

### Communicating With Bitcoin Network

There are three ways to communicate with the Bitcoin network:

1. As a full node
2. As an SPV node
3. Through an HTTP API (i.e. QBitNinja’s HTTP API)

Programming the blockchain is not the same as using a blockchain API (such as the RPC API provided with Bitcoin Core). Blockchain programming can require significantly more effort, but quoting from one of the blockchain books:

“While programming to an API can assist in getting an application up quickly, the developer is limited to innovations that can take place against the API. By fully understanding the blockchain, the developer is empowered to unleash its full potential.”

We must therefore always weigh the additional potential of direct blockchain programming against its commensurate increase in complexity and developer hours required—a trade-off that is not easy to quantify.

If we follow the typical OneChain development pattern, we would opt for a hybrid approach: Start wallet development by using existing APIs, and subsequently (or in parallel) start work on a homegrown, pure blockchain programming solution. The advantages we’ve seen with this approach is that it allows for quick development—and the ability to brainstorm features by actually implementing them—while also building a custom solution. The disadvantages include the use of extra development cycles and the possibility that we could bump up against a hard limit in the functionality of a chosen API, thus negating the quick-rollout advantage.

### Wallet Form Factors

The resource requirements can vary greatly between a full-node wallet and an SPV/HTTP-API wallet. These resource requirements can determine what types of wallet “form factors” are optimal:

1. Desktop App (i.e. NEON)
2. Mobile App (i.e. Bitpay)
3. Browser Extension (i.e. ???)
4. Hardware (i.e. Ledger Nano)

When designing our own wallet technology, we could use C# to develop both a desktop app and mobile app (using latest versions of Visual Studio with integrated Xamarin technology). We could use Javascript to develop a browser extension. And we could use Raspberry Pi to develop a hardware solution (or at least the prototype for such a solution).

# OneChain Wallet Integration

One idea is to integrate wallet technology into the OneChain API (“OCAPI”) by creating a consistent interface to various crypto wallets.

A consistent OCAPI interface to crypto wallets would look something like this:

* GenerateWallet
* RecoverWallet
* ShowBalances
* ShowHistory
* Receive
* Send

That’s it (for v1.0). The interface is quite simple. The difficulties will arise in the actual engineering and the (sometimes significant) differences among the various blockchain technologies.

# Topics for Further Research

NBitcoin (C# Bitcoin library)