HD Wallets - BIPs and Terminology

Author	josh
Date	2019-10-26 12:05:20

Overview

The advent of HD wallets has made key management a far easier task for cryptocurrency users. These "Hierarchical Deterministic" wallets can generate an infinite amount of private keys and addresses from a single seed, eliminating the need for the periodic backups required by the old-style, nondeterministic wallets.

Despite the ease of backup, there are a few more moving parts inside an HD system than there are with traditional keypairs, so understanding how this works is a little more complex. Let's get a high level overview of how HD wallets work and the associated terminology.

An HD Wallet Glossary

BIP, Boop, Bop - What's With All the Proposals?

First, let's discuss the BIPs involved with the most common HD wallet technologies. BIP stands for "Bitcoin Improvement Proposal" - this is a standard for proposing new additions to the Bitcoin protocol and software that is community driven.

BIP32 - Titled "Hierarchical Deterministic Wallets" , this BIP defines the basic specification for a protocol that generates addresses deterministically from a single seed, rather than randomly. In this BIP, Pieter Wiulle describes algorithms for taking a cryptographic seed and generating a tree of keys and addresses that can be reproduced again from that same seed.

BIP39 - Titled "Mnemonic code for generating deterministic keys", this BIP describes a scheme for encoding the cryptographic seed as English words, making backups far easier and safer for end users. This is where those mnemonic seed phrases you may be used to seeing originated! So instead of having to backup a string of hexadecimal or base58 data, we can write down 12-24 words to backup an entire wallet!

BIP44 - Titled "Multi-Account Hierarchy for Deterministic Wallets", this BIP builds on top of the work done in BIP32 to make multi-currency, multi-account wallets possible. BIP44 makes implementing technologies like multi-currency wallets like Coinomi far easier, at it makes the handling of various cryptocurrencies *standard*, rather than up to the wallet developer to figure out.

Extended Keys (xprv, xpub) and Root vs. Other Keys

One of my YouTube viewers asked the question: "What's the difference between a BIP32 *root key* and BIP32 *extended private key* when they both use `xprv` as the prefix?

Well, let's remember that HD wallets generate keypairs in a *deterministic*, or predictable manner. It turns out that this structure takes the form of a *tree* of keys - meaning there is a tree *root* and many *branches* generated from that root.

The **Root Key** is generated directly from the wallet seed. This is the key that is generated after taking the user's mnemonic seed phrase, returning it to a binary format, and running it through a hashing algorithm called HMAC-SHA512. The root key is the very top of the tree of addresses, and other keys are derived from it.

The **Extended Private Key** is one of the many *branches* on the tree of keypairs generated from the root. It's relationship to the root is that of child to parent - the root is the ultimate origin of all the child keys in the tree.

Now what does the *extended* mean? As part of the BIP32 specification, it turns out there's actually two pieces of data used to generate the actual Bitcoin private key and address. There's 256 bits of information that serve as the private key, and 256 bits of information called the *chain code.* The chain code makes it impossible to find any "siblings" on the tree without it, so that the keypairs in the wallet *appear* to be random when they're actually not. This enhances security and privacy.

Together, the key and the chain code form a 512 bit piece of data called the *extended key*. This format applies to both private and public keys (xprv and xpub).

BIPs, Extended Keys, and HD Goodness

HD wallets are an interesting and complex topic, and their existence makes securing and backing up wallets much easier for users. This concept involves several incremental improvement proposals (BIPs) and the introduction of some new cryptography concepts - not just private and public keys, but chain codes and the combined format of extended keys.

It becomes a bit easier to understand this concept at a high level when understanding the terminology behind it, so I hope this glossary helps clarify some of the mystery.