Bitzuma

1. Blog
2. Courses
3. Books
4. Newsletter
5. About

Six Things Bitcoin Users Should Know about Private Keys

By Rich Apodaca | Updated May 25th, 2017

Private keys have been part of Bitcoin from the beginning. Wallet software often tries to shield users from the need to directly handle and understand private keys. Even so, most users eventually come face to face with private keys, too often with unpleasant results.

A basic understanding of private keys can protect you from losing money and other mishaps, but it can also offer useful insights into how Bitcoin works. This guide outlines Bitcoin's the most important private key concepts.

Bitcoin: A Secure Messaging System

Bitcoin may be best known as an electronic cash system, but underneath it all runs a secure messaging system built on the Internet. Instead of relaying emails, texts, or web pages, the Bitcoin network processes value-transfer messages called *transactions*. Private keys help authenticate these messages and identify each other.

An example helps illustrate the problems that private keys solve. Imagine that Alice wants to pay Bob using an electronic coin with a face value of ฿1. To do so, she must create a transaction identifying Bob as the payee. Then Alice needs to publish the transaction to the Bitcoin network.

To use this system, Alice must solve two fundamental problems:

1. Alice needs a way to identify both herself and Bob in the transaction. She can't employ a trusted authority such as a government registry or email provider because that would create a central point of failure — the very thing Bitcoin was created to eliminate.
2. Alice needs a way to prevent others from changing her transaction and forging transactions in her name.

Bitcoin solves both problems through a system called public key cryptography. This system uses two pieces of information to authenticate messages. A *public key* identifies a sender or recipient, and can be distributed to others. A *private key* creates an unforgeable message signature. Unlike the public keys, the private key must be kept secret. Public and private keys are mathematically linked through a *signature algorithm*, a mathematical procedure for creating identities, signing messages, and validating signatures.

Public Key Cryptography. Alice (top) begins by choosing a private key. Using a signature algorithm, Alice obtains a public key from her private key (left). Alice then sends this public key to Bob (bottom) while keeping her private key secret (center-left). Alice signs a message by passing it to the signature algorithm together with her private key. The algorithm returns a signature in response (center). Alice attaches this signature to her message and sends both to Bob (center-right). Finally, Bob passes the message, signature, and public key he was given to the signature algorithm. If the message is authentic, the algorithm returns a confirmation (right).

With this overview in mind, here are six things about private keys to keep in mind as you use Bitcoin.

1. A Private Key is Just a Number

A Bitcoin private key is simply an integer between one and about 10^{77}. This may not seem like much of a selection, but for practical purposes it's essentially infinite.

If you could process *one trillion* private keys per second, it would take more than one million times the age of the universe to count them all. Even worse, just enumerating these keys would consume more than the total energy output of the sun for 32 years. Bitcoin's entire security model rests on the infeasibility of mapping this vast keyspace.

Because private keys contain many digits, an alternative called Wallet Import Format (WIF) has been devised. This format begins with the number "5" and contains a sequence of letters and numbers. For example, here's a private key represented in WIF format:

```
5KJvsngHeMpm884wtkJNzQGaCErckhHJBGFsvd3VyK5qMZXj3hS
```

Given the importance of keeping private keys secret, they are sometimes encrypted. A popular method produces strings of text that look like WIF encoding, but starting with the number "6." Decrypting a private key encoded in this way requires the password that was set when the private key was encrypted.

2. Transactions are Messages Signed with a Private Key

To prevent forgery, Bitcoin requires that each transaction bear a digital signature. This signature, like a private key, is just a number selected from a very large range. Wallet software generates a signature by mathematically processing a transaction together with the correct private key.

Anyone with a signature and public key can easily authenticate a message. However, the only way to produce a valid message signature is to use the private key matching the published public key. In other words, digital signatures are practically impossible to forge.


Message Tampering. The signature algorithm will notify Bob if a message signed by Alice was changed at all. He can likewise tell if the message was signed with a key different from the one Alice gave him.

Unlike a physical signature you might write on a check, a transaction signature changes if the transaction changes even slightly. The way the signature will change is unpredictable, ensuring that only a person in possession of a private key can provide the correct signature.

Notice that the internal format of a transaction is less important than the idea that transactions are digitally signed messages whose authenticity can be quickly and cheaply checked. For details on transactions and how they're used in Bitcoin, see A Visual Language for Bitcoin Transactions.

3. Anyone Who Knows Your Private Key Can Steal Your Funds

Any valid transaction bearing a valid signature will be accepted by the Bitcoin network. At the same time, any person in possession of a private key can sign a transaction. These two facts taken together mean that someone knowing only your private key can steal from you.

Many avenues are open to thieves who steal private keys. Two of the most popular are storage media and communications channels. For this reason, extreme caution must be taken whenever storing or transmitting private keys.

Software wallets usually store private keys in a "wallet file" on the main hard drive. Wallets often place this file in a standard, well-known directory, making it an ideal target bitcoin-specific malware.

To counter this threat, software wallets offer an option to encrypt the wallet file. Any attacker gaining access to your wallet file would then need to decrypt it. The difficulty of decryption depends on the quality of the encryption and strength of the password being used. Wallet files can be encrypted on many software wallets by adding a password.


Password Protection. Encrypting Electrum's wallet file by adding a password.

Although wallet backups are a good idea, they can potentially leak private keys. For example, it may be tempting to save a backup of your software wallet to a cloud storage service such as Dropbox. However, anyone capable of viewing this backup online (a potentially long list of people) would be in a position to steal some or all of your funds. A similar problem could arise through emailing backups to yourself or leaving a private key around the house. Encryption can reduce, but not eliminate the risk.

Preventing the accidental release of private keys is the main purpose of "cold storage." For more information, see A Gentle Introduction to Bitcoin Cold Storage.

4. A Private Key Generates a Public Key Which Generates an Address

A public key is obtained by subjecting a private key to a set of mathematical operations defined in a set of standards known as Elliptic Curve Cryptography (ECC). Whereas a private key is an integer, a public key is a 2D coordinate composed of two integers. To make a public key easier to process, it can be transformed into a single value. One approach appends the y-coordinate to the x-coordinate. This technique produces an "uncompressed" public key. A "compressed" public key uses only the x-coordinate with a symmetry flag.

Private Key to Address. A private key, which is just a number such as 42, can be transformed mathematically into a public key. A public key is then transformed into an address. Each step is irreversible.

Each of these steps is irreversible. An address can't generate a public key, nor can a public key generate a private key. This relationship is known as a mathematical trapdoor — a function that's easy to perform in one direction, but practically impossible to perform in the opposite direction. This unidirectionality underpins Bitcoin's security model.

Just as private keys can be shortened to make them more usable with displays and keyboards, so too can public keys. An address results from applying a multi-step transformation to a public key. This results in a string of text and digits, usually starting with the number "1".

Notice that no network is needed at any point in the generation of a private key or the corresponding address. Every computer on the Bitcoin network knows about the mathematical relationship between public and private keys. This enables each participant to select private keys and sign transactions independently of the Bitcoin network. The vast private keyspace ensures that any properly-selected key will be unique.

5. Security Depends on Choosing a Good Private Key

Knowledge of a private key is the only verification needed to spend an electronic coin. Private keys should therefore be kept secret. However, careless selection of a private key can lead to theft just as easily as its accidental release.

For example, imagine that we want to use a private key that's easy to remember. The number `1` is both easy to remember and a valid Bitcoin private key. But how secure would it be?

The private key `1` generates this address:

`1EHNa6Q4Jz2uvNExL497mE43ikXhwF6kZm`

If you follow the link, you'll notice that the address has already been involved in over 1,000 transactions for a total of over 7 BTC within the last few years. If you wanted, you could easily spend any available funds at this address because the private key is known to you.

Now imagine you're a thief determined to steal bitcoin. One strategy might be to compile a list of easy-to-remember private keys. Next, generate the addresses for these keys and monitor the Bitcoin network for incoming payments to one of them. When one arrives, immediately sign a transaction moving the funds to another address you control.

Contrast the ease of this scheme with a situation in which a private key was chosen by a perfect random number generator. With no clue what the key might be, brute force iteration would be the only option. As we've already seen, carrying out this plan is physically impossible.

What would happen if the random number generator were not quite random? For example, what if all output private keys were clustered about a constant value within a narrow range?


Private Keyspace. Random private key distribution (left) versus one that is clustered (right). The clustered distribution limits the search space, favoring an attacker.

Any attacker aware of such a defect could drastically reduce the necessary search space. Under the right conditions, it would become practical to monitor all of the addresses based on the faulty random number generator and steal funds from any one of them at will.

The need to select a good private key becomes especially important with *brain wallets*. One method to create a brain wallet starts with a passphrase such as "to be or not to be", then applies a mathematical function to convert this text to a private key. Applying the most popular conversion algorithm (SHA-256) to this passphrase generates the address:

`1J3m4nneGFppRjx6qv92qyz7EsMVdLfr8R`

As you can see, this address was used as late as 2016 to store funds, which were immediately withdrawn.

Unfortunately, it's not always easy to tell what qualifies as an insecure brain wallet passphrase and what doesn't. Attackers can exploit this uncertainty and the inexperience of new users to steal funds. For example, a thief might compile an enormous database of common phrases and passwords. Such a database might number in the trillions of entries or more, but would still be searchable in its entirety with little computational effort.

Compare this situation to the one with website passwords. If you register for a web service using a password someone else happens to have chosen, you don't take over their account because your username must be unique. Bitcoin private keys are different in that they serve the dual role of user identification (via address generation) and authentication (via digital signatures).

Secure private keys are generated with a high degree of unpredictability so they can't be guessed before or after the fact.

6. Private Keys are (Somewhat) Portable

For the most part, wallet software hides the process of generating, using, and storing private keys. However, private keys can become visible from time to time. When this happens, understanding private keys and how they interact with your specific software becomes important.

Paper wallets present the most common route by which private keys show up outside of software wallets. Although they come in a multitude of formats, the essential feature of any paper wallet is a printed private key.


Paper Wallet. Example paper wallet. To the right is the private key, represented both as a QR code and a string of text beginning with the number "5" and written vertically.

Many software wallets support *sweeping*. A sweep creates a new transaction paying one of the software wallet's existing addresses. This procedure may or may not empty the address associated with the private key. For more information on the dangers of manipulating bare private keys, see Five Ways to Lose Money with Bitcoin Change Addresses.

Should your wallet application begin to malfunction, its private keys can often be imported into another application. This rescue procedure provides the second main route through which private keys become visible to end users. A closely-related procedure consists of restoring the state of a software wallet through a backup file.

Conclusions

Bitcoin can be thought of as an open messaging system secured by public key cryptography. In contrast to other systems protected by username and password logins, Bitcoin is secured through digital message signatures created with a unique private key. This single point of access places a very high value on the secure generation, use, and storage of private keys.

- Twitter