

Forum: [Hack Forums](#)

Thread: [How to securely generate vanity addresses for others](#)

XMR - August 27, 2016, 6:31 pm

Someone recently wrote a tutorial on how to generate addresses with custom prefixes, but that isn't secure for generating addresses for other people as both parties will have access to the private key. This tutorial will cover how to generate vanity addresses for other people in a manner such that only the recipient will have access to the final private key. For this, both parties will need the vanitygen tools.

I'm using an Arch Linux system for this tutorial, though you can download v0.22 pre-built Windows binaries from here: <https://github.com/downloads/samr7/vanitygen/vanitygen-0.22-win.zip>

Lines starting with '#' symbols are comments and can be ignored. If you type them into your shell, nothing will happen.

Download the vanitygen tools and navigate to the new directory:

```
Code arccot@arccotangent-arch:~$ git clone https://github.com/samr7/vanitygen.git && cd vanitygen
```

You could also download the repository (source code) as a zip file from here:

<https://github.com/samr7/vanitygen/archive/master.zip>

If compiling from source, you will need to install the compiler and dependencies:

```
Code Debian/Ubuntu: sudo apt-get install build-essential git libssl-dev libpcre3-dev
```

```
Arch Linux: sudo pacman -S openssl pcre gcc git
```

Arch doesn't have separate development packages, everything is packaged as one. There is no openssl-dev or pcre-dev.

Compile the tools:

```
Code arccot@arccotangent-arch:~/vanitygen$ make
```

Optionally, you can install the tools to /usr/local/bin:

```
Code arccot@arccotangent-arch:~/vanitygen$ sudo install vanitygen /usr/local/bin && sudo install keyconv /usr/local/bin
```

To generate a partial private key, you'll need the recipient's public key. They can generate a new address by performing this command:

```
Code user@recipient:~$ keyconv -G
```

Pubkey (hex):

```
0434c524119cc9cc2cec1a056ab6270c49be404360fe359b6ad820456ce30b6b93f01e961f287d6fee  
c38dd7e589c2284f9b126c3b0a09fc8440765e944535fa29
```

Privkey (hex):

```
D17113CB2FE050EA8402A0A31994D846C15E16D2225E1CA55CF8CECFBB377CEC
```

Address: 1LJ2vQX8rdUHK982VHoBxRxNYpceinJ3yH

Privkey: 5KQXVJjWmKHGLoXWTjeSZ2TzU1o7tBr5giVCruCWRrvKMdLkfyACode # Generate a new, random address

```
keyconv -G
```

Then they will need to send you the hexadecimal public key that starts with "04" and you can use this public key with the -P option to generate a partial private key:

```
Code arccot@arccotangent-arch:~/vanitygen$ vanitygen -P
```

```
0434c524119cc9cc2cec1a056ab6270c49be404360fe359b6ad820456ce30b6b93f01e961f287d6fee  
c38dd7e589c2284f9b126c3b0a09fc8440765e944535fa29 1Tut
```

Difficulty: 4553521

Pattern: 1Tut

Address: 1Tut5yHDrbtbHJWqG1NmHXLBpm2WtRTMk

PrivkeyPart: 5K2a9aDFyE3AHAWq1rpAWPUIVgr5KPxDHb2p9UZMD6ymiQsuBfWCode #

Generate a partial private key, given someone's public key

vanitygen -P <public key> <address prefix>

Finally, the recipient of the address can combine the private key from the address they generated, and the partial privkey (shown here as "PrivkeyPart") that the other person generated and get the private key for the vanity address:

Code user@recipient:~\$ keyconv -c

5KQXVJjWmKHGLoXWTjeSZ2TzU1o7tBr5giVCruCWRrvKMdLkfyA

5K2a9aDFyE3AHAWq1rpAWPUIVgr5KPxDHb2p9UZMD6ymiQsuBfW

Address: 1Tut5yHDrbtbHJWqG1NmHXLBpm2WtRTMk

Privkey: 5Jg4sy3eof5DE9WMXpPi23btMrZTsmM7vZe7nujY6j4SB7Qr4UACode # Combine private key parts to get the vanity address private key

keyconv -c <private key 1> <private key 2>

The order in which the partial private keys are specified doesn't matter. The person that generated the partial private key has no access to your vanity address as the partial private key corresponds to a completely different address:

Code arccot@arccotangent-arch:~/vanitygen\$ keyconv

5K2a9aDFyE3AHAWq1rpAWPUIVgr5KPxDHb2p9UZMD6ymiQsuBfW

Address: 1PWetY6RK6UkPuUipKrZ7dESrWtRSey3Mn

Privkey: 5K2a9aDFyE3AHAWq1rpAWPUIVgr5KPxDHb2p9UZMD6ymiQsuBfWCode # Get the address corresponding to a certain private key

keyconv <private key>

The person that generated the partial private key has no access to the vanity address, only the person that requested it has access to the final key. Therefore, it is possible to generate vanity addresses for others as a paid service. ?

J?son - August 27, 2016, 6:32 pm

ayy lmao ok show off ?

B?ker - August 27, 2016, 6:33 pm

Does this work with only BTC?

minimalist - August 27, 2016, 6:37 pm

Nice write-up. I'm happy here with my non-vanity BTC address but this will probably come in handy to those who want to open up vanity gen services...because those are a thing.

XMR - August 27, 2016, 6:42 pm

(Aug 27, 2016 06:32 PM)BI?ck Wrote: [\[CITE\]](#) ayy lmao ok show off ?

It legitimately took me like 10 minutes to realize you wrote the other tut ?

(Aug 27, 2016 06:33 PM)?MrDark Wrote: [\[CITE\]](#) Does this work with only BTC?

Technically, no. It can work with any cryptocurrency, but you'll need to modify the code accordingly for the different address/private key versions.

This article can be of help: https://en.bitcoin.it/wiki/List_of_address_prefixes