

Forum:[Hack Forums](#)

Thread: [Custom script to generate vanity btc addresses](#)

Voxcon - October 11, 2020, 8:28 am

Hey,

i've written a small custom script to generate vanity btc addresses from random bytearrays. Initially i wanted to use vanitygen to generate a custom vanity address, but i didn't like the fact that they seem to use seeds to generate and check a bunch of addresses at once. Of course this is more time and cost efficient than using random byte arrays for every check, but especially if you don't compile vanitygen from source there is a risk that the binary uses precoded seeds to derive addresses from. And i didn't want to take that risk, so i wrote a simple python script to generate vanity addresses myself.

Currently checking for an address starting with "1Vox". Checked more than 1.5 mio keypairs already.

Best regards,
Vox

Edit: Apparently it seems like addresses starting with a letter from 1A - 1P (1/22) are way more likely to be generated than addresses starting with 1R - 1Z (1/1353). Interestingly 1Q falls in neither of those categories (1/65).

T0Bi - October 11, 2020, 10:50 am

Where do you get your randomness from?

Because you can't tell what is more likely or not if you use a pseudo random source.

Voxcon - October 11, 2020, 11:09 am

(Oct 11, 2020 10:50 AM)T0Bi Wrote: [\[CITE\]](#) Where do you get your randomness from?

Good question, I'm using pseudo-random random.SystemRandom() as well as the floating point representation of the cpu temperature of a "randomly" picked cpu core to determine bytearray-values.

Both in combination should grant quite random results.

Best regards,
Vox

T0Bi - October 11, 2020, 1:25 pm

(Oct 11, 2020 11:09 AM)Voxcon Wrote: [\[CITE\]](#) Good question, I'm using pseudo-random random.SystemRandom() as well as the floating point representation of the cpu temperature of a "randomly" picked cpu core to determine bytearray-values.

Both in combination should grant quite random results.

Best regards,

Vox

Yeah that should probably be random enough but tbh with crypto I'd be extra careful.
I quite like `os.urandom()` myself, it's probably as random as you can be on a PC/server.

Here's some interesting stuff about randomness in python: <https://realpython.com/python-random/>

Voxcon - *October 11, 2020, 1:38 pm*

Pretty sure `os.urandom()` uses `random.SystemRandom()` behind the scenes, but i might be wrong.
However with using cpu core temperatures as well, the randomness should be enough.

Best regards,
Vox

PS: also thanks for the article