

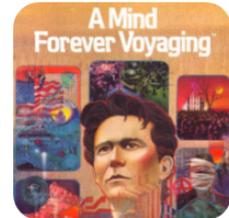
pooler/cpuminer: CPU miner for Litecoin and Bitcoin

github.com/pooler/cpuminer

pooler

pooler/cpuminer

CPU miner for Litecoin and Bitcoin



19
Contributors

17
Issues

3k
Stars

1k
Forks



Failed to load latest commit information.

[compat.h](#)

Oct 4, 2013

[configure.ac](#)

Jun 25, 2020

[cpu-miner.c](#)

Nov 10, 2020

[elist.h](#)

May 16, 2014

[example-cfg.json](#)

	Jan 16, 2012
<u>miner.h</u>	
	Apr 29, 2021
<u>minerd.1</u>	
	Jun 25, 2020
<u>nomacro.pl</u>	
	Feb 25, 2015
<u>scrypt-arm.S</u>	
	May 25, 2014
<u>scrypt-ppc.S</u>	
	Jun 21, 2017
<u>scrypt-x64.S</u>	
	May 25, 2014
<u>scrypt-x86.S</u>	
	May 25, 2014
<u>scrypt.c</u>	
	Feb 25, 2015
<u>sha2-arm.S</u>	
	May 16, 2014
<u>sha2-ppc.S</u>	
	Jun 21, 2017

sha2-x64.S

May 17, 2015

sha2-x86.S

May 16, 2014

sha2.c

May 17, 2015

util.c

Nov 10, 2020

README

This is a multi-threaded CPU miner for Litecoin and Bitcoin, fork of Jeff Garzik's reference cpuminer.

License: GPLv2. See COPYING for details.

Downloads: <https://sourceforge.net/projects/cpuminer/files/>

Git tree: <https://github.com/pooler/cpuminer>

Dependencies:

libcurl <http://curl.haxx.se/libcurl/>
jansson <http://www.digip.org/jansson/>
(jansson is included in-tree)

Basic *nix build instructions:

```
./autogen.sh # only needed if building from git repo
./nomacro.pl # in case the assembler doesn't support macros
./configure CFLAGS="-O3" # make sure -O3 is an O and not a zero!
make
```

Notes for AIX users:

- * To build a 64-bit binary, export OBJECT_MODE=64
- * GNU-style long options are not supported, but are accessible via configuration file

Basic Windows build instructions, using MinGW:

Install MinGW and the MSYS Developer Tool Kit (<http://www.mingw.org/>)

- * Make sure you have mstcpip.h in MinGW\include

If using MinGW-w64, install pthreads-w64

Install libcurl devel (<http://curl.haxx.se/download.html>)

- * Make sure you have libcurl.m4 in MinGW\share\aclocal

- * Make sure you have curl-config in MinGW\bin

In the MSYS shell, run:

```
./autogen.sh # only needed if building from git repo
LIBCURL="-lcurl.dll" ./configure CFLAGS="-O3"
make
```

Architecture-specific notes:

ARM: No runtime CPU detection. The miner can take advantage of some instructions specific to ARMv5E and later processors, but the decision whether to use them is made at compile time, based on compiler-defined macros.

To use NEON instructions, add "-mfpu=neon" to CFLAGS.

PowerPC: No runtime CPU detection.

To use AltiVec instructions, add "-maltivec" to CFLAGS.

x86: The miner checks for SSE2 instructions support at runtime, and uses them if they are available.

x86-64: The miner can take advantage of AVX, AVX2 and XOP instructions, but only if both the CPU and the operating system support them.

- * Linux supports AVX starting from kernel version 2.6.30.

- * FreeBSD supports AVX starting with 9.1-RELEASE.

- * Mac OS X added AVX support in the 10.6.8 update.

- * Windows supports AVX starting from Windows 7 SP1 and Windows Server 2008 R2 SP1.

The configure script outputs a warning if the assembler doesn't support some instruction sets. In that case, the miner

can still be built, but unavailable optimizations are left off.
The miner uses the VIA Padlock Hash Engine where available.

Usage instructions: Run "minerD --help" to see options.

Connecting through a proxy: Use the --proxy option.

To use a SOCKS proxy, add a socks4:// or socks5:// prefix to the proxy host.
Protocols socks4a and socks5h, allowing remote name resolving, are also available since libcurl 7.18.0.

If no protocol is specified, the proxy is assumed to be a HTTP proxy.
When the --proxy option is not used, the program honors the http_proxy and all_proxy environment variables.

Also many issues and FAQs are covered in the forum thread dedicated to this program,

<https://bitcointalk.org/index.php?topic=55038.0>