Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014

ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-42

Bitcoin mining acceleration and performance quantification

Jega Anish Dev

***Abstract*— Since its introduction in 2009, Bitcoin, an open source, peer to peer, digital crypto currency has been growing in popularity and wide spread use. Growing attention, recognition by major financial institutions and high valued currency units (BTC) ascertains Bitcoin to a sturdy and ever increasing choice of currency. A public transaction log called the "Blockchain" keeps records of all committed transactions and Bitcoin ownership details, that is, addresses derived by cryptographic keys. Bitcoin mining, a process which results in the generation of new Bitcoins, is performed by miner operators for reception of incentives in the form of Bitcoins. This mining process is essentially operations of SHA-256 hashing of values in search of a hash digest smaller than a specific value. Once this winning hash has been discovered, a new block to Blockchain is added and BTC incentives are furnished by the Bitcoin network to the miner. This paper discusses methods of performing Bitcoin mining on non-custom hardware which results in contextually faster mining by combined usage of computing elements within machines in mining networks, both illegal and legal.**

Keywords*— Bitcoin mining; Botnet based Bitcoin mining; combined usage of CPU and GPU for Bitcoin mining.*

I. Introduction

Cryptocurrency, unlike regular currency, is a digital medium exchange which involves a decentralized network of mutually distrustful parties to ensure integrity and general balance of all ledgers. As opposed to fiat money, monetary units of cryptocurrency require certain amounts of work, called Proof-Of-Work, expended to be produced and cannot be reproduced to a materialized representation by contemporary fashion. Proof-of-work is a measure which features the use of asymmetric work to deter abuse to a system, in this case, problems pertaining to integrity of the economics of the digital currency, Bitcoin. The asymmetric work involves operation of a feasible but computationally intensive work on the requester side which can be verified by an operation relevantly simple compared producing the work in the first place. This procedure of working toward producing a proof-of-work for generating monetary units of Bitcoin in called Bitcoin mining.

Bitcoin mining involves scanning for a value which when hashed with SHA-256, is lesser than a specific value. The average work required is exponential to the number of zero bits required and can be verified by executing a single hash [1].

Jega Anish Dev
Department of Computer Science and Engineering,
College of Engineering, Guindy, Anna University
India

The number of initial zeros and upper limit of value specified for the computation of a new block required to head the publicly accessible Blockchain, whose function is to essentially prevent double spending by maintenance of a public transaction ledger, is determined by the Difficulty Factor. The Difficulty Factor is adjusted in such a way that the production of a new Block, that is, the discovery of a hash lesser than the specific value, arises on an average of one in 10 minutes. As self-evident, the speed of discovery of the next required hash relates directly with the total hash rate of all the participating miners combined, which in turn results in the direct variance of the Difficulty factor of the next work.

The work expended in generating a Proof-of-Work is in the form of computation cycles involved in repeated SHA-256 computations trialled at discovering the required hash seed. This procedure when performed on a single machine with average home computer specifications with the Difficulty Factor at the time this paper was written would require an impractically large amount of time, even to a tune of years. To prevent discouragement and withdrawal of participation of average or low powered miners, mining pools have been formed which collectively utilize computation powers hundreds or thousands of miners and eventually splitting the incentive in proportions relevant to their computing contribution to solution of that particular block. Since each block generated results in an incentive of 25 BTC (each BTC having a value of $863 at the time this paper was written), this has given rise to a number of competing pools and has triggered a race to possess high hash rate capabilities with usage of GPU in contrast to contemporary CPUs. In addition to usage of high performance GPUs, dedicated hardware like Field Programmable Gate Array and Application Specific integrated circuits, for Bitcoin mining has also been in use and development.

Recent times have seen botnets, which consist of thousands, hundreds of thousands or sometimes millions of computers compromised by attackers to perform Bitcoin mining by usage of their graphics cards [2]. Usage of CPUs for Bitcoin mining in both solo and pooled mining has been partially considered to be a relatively minor fraction of the total hash rates contributed by GPUs. This paper aims to propose methods of achieving contextually higher speeds of Bitcoin mining involving simultaneous usage of CPUs and GPUs in individual machines in mining pools. The advantage served is quantified by observing ratios of hash generation rates on test conditions comprising of varying device configurations. Furthermore, adverse effects of "unnatural" boosts in overall hash rates of Bitcoin miners by illegal botnets which included use of weak CPUs and simultaneous use of CPUs and GPUs has been discussed on the bases of results obtained from the hash rate ratios observed. Though there are different approaches of pool based mining, the method

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014

ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-42

described to accelerate Bitcoin mining is independent of the approach used as it details procedures that involve only the mining bots and not the work load distribution of the mining pool.

As afore mentioned, Bitcoin mining can be performed by both a solo miner and by a pooled effort. In addition to this, botnets can also perform Bitcoin mining by either joining a single or set of public mining pools or their own dedicated mining pool (figure 2). They are briefly described below.

A. *Mining by legal means*

This involves either Pool based mining or mining using a solo mining rig consisting of hardware powerful enough to be capable of solving a block in practical amounts of time and hence probability, alongside serving to be profitable when in consideration of both initial investment and running costs in terms of power consumed (Figure 1).

Pool based mining involves a Pool server which attempts to generate blocks by providing a broken up load of work distributed among participating miner bots. Rewards for each participating bot depend on the type of mining pool. [3] and [4] discusses various approaches in detail.

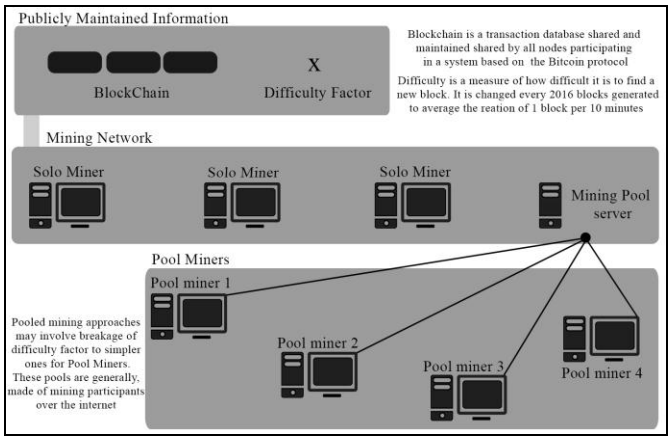

Figure 1. Pictorial represenation of Solo mining and Pooled mining with requirements for mining.

B. *Mining by illegal means*

Unauthorized Bitcoing mining was first detected in 2011 [5]. Since then, other malware associated with Bitcoin mining payloads have been discovered, like the Miner Bot [6]. These botnets have focused on almost exclusive usage of either ATI or Nvidea graphics cards of compromised machines to contribute to the enormous amount of hash rates of the botnet as a whole.

These botnets could either go on to perform Bitcoin mining by attemping to singlely produce blocks entirely by itself or by fractioning out various parts of the botnet as miner bots to one or several of the many public mining pools on the inernet (Figure 2). A botnet's mining setup depends on the degrees of ease of implemention and simulataneous consideration of required levels of reliance of botnet take down by law enforcement agencies.

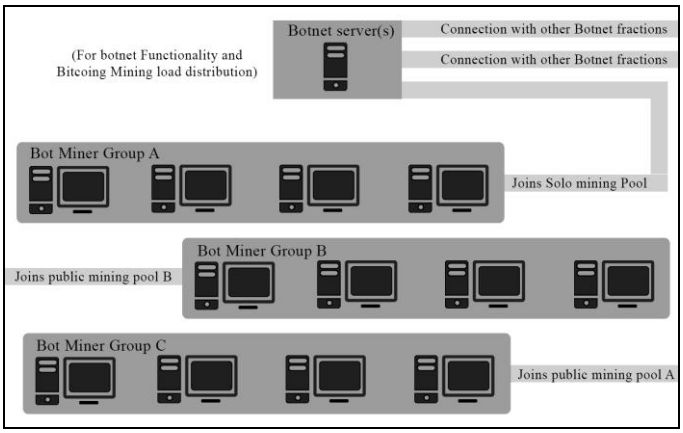

Figure 2. Pictorial represenation of illegal botnet mining by participation with public and solo mining pools.

II. Related works

With Bitcoin values having spiked manifold since its initial rise to popularity, a number of pursuits, both homebrew and professional, by both software and hardware means have been seen. A classified enumeration and description of some of these works are given below:

A. *Mining with non-custom, standard hardware*

This refers to use of commercially available hardware that does not exclusively pertain to Bitcoin mining. Combination of various computing devices within the same hardware framework to perform concurrent mining, networking of mining bots by means of public pools of illegal botnets, use of extensive arrays of GPUs, all of which are used to perform mining by mining software freely or commercially available.

BFGMiner, a novel miner written in C, possesses a number extensive feature, some of which are support for modular ASIC, FPGA, GPU and CPU, specifically, device drivers for Butterflylabs [8] ASIC product line and Avalon's mining rig [9]. It also supports the widely used method of OpenCL capable GPUs and a number of fine-tuned features for greater efficiency and support. Excluding features relevant to computing SHA-256 operations for Bitcoin, it also has a number of Bitcoin protocol and mining pool specific features which both boosts ease of work and efficiency. BFGMiner's source code is freely available [9]. As mentioned, this supports both standard hardware and custom hardware.

DiabloMiner, an open source software, performs mining exclusively using the OpenCL framework for use with supported NVidia and ATI graphics cards. This includes support for single pooled, multi pooled and solo mining [10].

Phoenix2, an open source software programmed in Python, supports OpenCL, CPU and CUDA ready NVidia graphics cards. In addition to standard features it also supports selective device usage for mining and performance aggression level modification [11].

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014

ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-42

Poclbm supports only OpenCL ready GPUs. It is coded in python with usage of pyOpenCL, a freely available OpenCL library for Python [12].

Ufasoft Bitcoin Miner solely supports SSE2 optimized CPU usage, ATI and NVidia GPU based mining. In addition to Bitcoin, it also supports a number of other Bitcoin cryptocurrency variants. Its source code is freely available[14].

B. *Mining with custom hardware*

CGMiner is a combined ASIC & FPGA Bitcoin miner written in c, cross platform for windows, Linux and OSX, with monitoring, fanspeed control and remote interface capabilities [15]. This miner does not support CPU or GPU for mining following the fact of the increasing minority of contribution of CPUs and GPUs to the total hash rates of all the miners combined. In addition to this, CGMiner is open source.

BFGMiner, mentioned and described in the previous section in lieu of its additional support for standard hardware, is also an open source miner with custom hardware usage support.

III. Approach

This section presents the proposed methodology to obtain faster hash rates when Bitcoin mining is performed on standard commercially available machines having GPUs along with their CPUs. The methodology does not describe technical details involved with the Bitcoin protocol, that is, network details of how the P2P system collectively performs or checks transactions and other details which do not ultimately influence the most elemental operation of Bitcoin mining: SHA256 operations.

Mining is most commonly done by publicly pooled Bitcoin mining with independent users having machine setups with concurrent used multiple GPUs, with software using these GPUs either by CUDA, for NVidia GPUs, STREAM for ATI GPUs or OpenCL for a common access to any GPU which is OpenCL supported. Although CPU mining is generally avoided because of the comparatively low hash rates in contrast to hash rates when using GPUs or arrays of GPUs, some users still run CPU miners using standard or superior processors.

The approach discussed here is to build a system capable of simultaneous usage of both CPU and GPU(s) in a system for Bitcoin mining. This, however, is not expected to be comparable with custom hardware based mining, but is expected to provide a certain and relevant boost to hash rates for non-custom equipment users as they contribute to considerable fractions of the total number of mining units in operation. Furthermore, this boost when considered applicable to a vast number of miner bots in operation would prove to be a significant boost in the overall hash rate of a mining pool.

The overall steps performed during Bitcoin mining, independent of the computing device used in the process, are enumerated as follows:

i. Reception of the hash of the last block, valid transactions and current Difficulty factor, which is measure of difficulty of the required correct nonce to be discovered for block generation.

ii. Interpretation of the Difficulty factor to define required success test condition for generated hashes

iii. Performance of trial and error of various nonce values until the double SHA256 hash digest of the block header is less than the target hash.

iv. Submission of the winning nonce value to the Bitcoin network or restart from step I if other miners find a winning nonce first

These overall steps are implemented to perform Bitcoin mining on both the CPU and the GPU(s). Ordered steps of execution for both a CUDA NVidia ready device and a multi core CPU are separately given below along with details of the hash computation algorithm common to both:

A. *SHA256 Computation*

Each SHA256 computation, in both GPU and CPU mining, involves internal steps shown in Figure 3 and makes use of the following functions:

$$Ch(X,Y,Z) = (X \wedge Y) \oplus (\overline{X} \wedge Z)$$
$$Maj(X,Y,Z) = (X \wedge Y) \oplus (X \wedge Z) \oplus (Y \wedge Z)$$
$$\Sigma_0(X) = RotR(X,2) \oplus RotR(X,13) \oplus RotR(X,22)$$
$$\Sigma_1(X) = RotR(X,6) \oplus RotR(X,11) \oplus RotR(X,25)$$
$$\sigma_0(X) = RotR(X,7) \oplus RotR(X,18) \oplus ShR(X,3)$$
$$\sigma_1(X) = RotR(X,17) \oplus RotR(X,19) \oplus ShR(X,10)$$

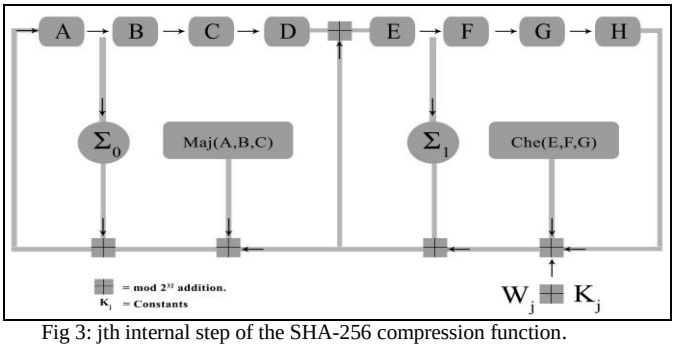

Fig 3: jth internal step of the SHA-256 compression function.

Detailed information about SHA256 like padding, block decomposition and hash computation can be found at [15].

B. *Mining on the CUDA device*

The following steps are performed on any CUDA compatible NVidia graphics cards.

i. Allocation of resources on the device, namely, a single instance of the structure holding block header details required in the hashing, and a copy of the best nonce for each thread executing in the device. The latter is given by the product of the number of blocks in use and the threads running in each block.

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014

ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-42

i. Initialization of variables in the kernel function for concurrent SHA256 operations, notably:

```
t_Id = blockIdx.x * blockDim.x + threadIdx.x
where,
    t_Id is an ID used for appropriate nonce initalization per thread
    blockIdx.x is the block index of the currently running block
    blockDim.x is number of threads along a dimension of blocks
    threadIdx.x is the thread index of the current thread
```

ii. Performance of double SHA256 computation on the trial block structure filled with static data as received using the Bitcoin protocol and with an incremented trial.

iii. Test if hash obtained from step iii is smaller than the target hash as given by the difficulty. This is efficiently performed by first testing for a certain number of required zeros as present in the beginning of the target hash, followed by the test of whether the remainder of the hash is smaller than the remainder of the target.

iv. On success, report the winning nonce. On failure, increment the nonce and perform step iii

C. *Mining on the CPU*

Mining on the CPU using all available cores have steps similar to Mining with the GPU. They are mentioned briefly sans redundancy as follows:

i. Allocation of resources, as mentioned in GPU mining, instead here, in the RAM.

ii. Initialization of variables to concurrently provide an id for starting values of the nonce to each parallel instance of the SHA256 operation.

iii. Performance of double SHA256 computation of the trial block structure the same way as mentioned in GPU mining

iv. Test if hash obtained from step iii is smaller than the target hash as given by the difficulty

v. On success, report the winning nonce. On failure, increment the nonce and perform step iii

An integrated representation of the entire process is shown in Figure 3.

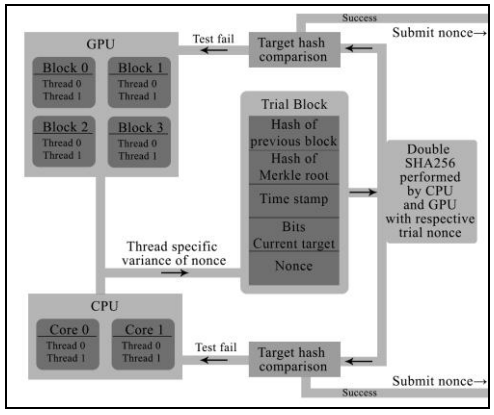

Figure 3. Diagrammatic represenation of concurrent CPU and GPU usage for mining.

IV. Results

The proposed approach was tested on 2 machines dubbed C1 and C2. C1 had CPUs Intel I7-2600K @ 3.3 GHz, 4 Cores, 8 logical processors and GPUs NVIDIA GTX 550 TI. C2 had CPUs Intel I5-3210M @ 2.5 GHz, 2 cores, 4 logical processors. These machines were chosen to analyse ratio differences in hash rates obtained in a high end computer with an intermediately powerful graphics cards and a normal home computer with a standard graphics card and processor. This ratio can then be used to approximately calculate the total hash rate contributed by CPU inclusion in mining pool or illegal botnet.

The results were analysed in terms of increase in hash generation rate upon combined usage of CPU and GPU as opposed to sole usage of the GPU. Hash rates were observed along the first few seconds of mining till the rates stabilized.

The experiment to measure hash rates was performed during trial participation in the public mining pool "Deepbit" [16]. The Bitcoin protocol provides all necessary block details required for Bitcoin mining.

A. *Mining tests performed on C1*

Results measured on C1 are shown in Table 1. C1 showed an average hash generation rate of 22.3 Million hashes per second, the CPU generating 1.9 Million hashes per second and GPU generating 20.4 Million hashed per second.

TABLE I: Detailed C1 PERFORMANCE

Time (seconds)	Computing device	Hashes Generated (in Million hashes/Second)
1	CPU	1.5
2		1.8
3		1.9
1	GPU	19.6
2		20.1
3		20.4
Combined stable rate:		22.3

B. *Mining tests performed on C2*

Results measured on C2 are shown in Table 2. C2 showed an average hash generation rate of 64.4 Million hashes per second, the CPU generating 18.3 Million hashes per second and GPU generating 46.1 Million hashed per second.

TABLE II: Detailed C1 PERFORMANCE

Time (seconds)	Computing device	Hashes Generated (in Million hashes/Second)
1	CPU	17.5
2		17.8
3		18.3
1	GPU	45.1
2		45.9
3		46.1
Combined stable rate:		64.4

Proc. of the Intl. Conf. on Advances in Computer Science and Electronics Engineering -- CSEE 2014

ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-42

C. *Comparison of hash rate ratios of CPU to GPU of C1, C2*

Thus, the overall hash rate boost when concurrently using the CPU in C1, an additional 1.9 Million hashes, is 9.3%. C2 showed an additional generation of 39.62% of the hash generation rate of the GPU, an additional 18.3 Million hashes. These results are tabulated in Table III. Furthermore, a combined ratio comparison of hash rates from C1 and C2 has also been performed for inference of hash rate boosts that could be possible in mining pools or illegal botnets.

TABLE III: Total hash rate comparison

Machine	GPU	CPU	CPU+GPU	CPU boost %
C1	20.4	1.9	22.3	9.3
C2	46.1	18.3	64.4	39.69
C1+C2	66.5	20.2	153	30.37

v. Conclusions and future work

The 9.3%, 39.68% and 30.37% hash generation rate boost in machines C1, C2,C1 and C2 considered simultaneously, are numerically substantial figures. The actual advantage, that is, the quickness of solution to generating a new block, implied by these boosts in an isolated test as this are, however, very minimal. To appreciate how these boosts could serve to be actual advantages can be understood by its application in real world mining scenarios. To comprehend this, one must know that block generation, or Bitcoin mining, the success or failure to find a block depends more on tackling probability of finding the solution, which is not similar to a race of solving a problem with a definite size where at the end which lays a certain winning hash. Therefore, the greater than hash rate, the greater than chance of a miner or mining pool to find the winning hash. This has caused a recent trend for a number of users to construct expensive mining rigs and developers to pursue construction custom mining equipment capable of performing hundreds of millions or billions of hashes per second. These relatively few number of high end users, as compared to the number of users having commercially easily available equipment, possess a greater degree of control over the Difficulty factor. The greater than overall hash rate, the faster the blocks tend to be solved, and hence the greater the difficulty factor to maintain the average of generation of 10 blocks per hour.

In a soon and expected up slide of great difficulty factors in the near future, these high end miners could prove to be sole or greater profiteers of hash rates than the greater majority or non-custom hardware miners. Currently, a large number of pool miners use their GPUs, however powerful, solely for mining without necessarily including their CPUs, which are generally powerful, since high end GPUs are marketed along with high end CPUs for gamers and high end users. When considering an entire mining pool, for example, a totally of 27,682 miners, online on a public miner in December 2013 [17], constituting a total hash rate of 2,172 Trillion hashes per second, it is inferred that that each miner averages a little more than 4 billion hashes per second. It is unlikely that each and every one of the miners had used an FPGA, say, the “DeepBit Reclaimer One”, capable of 4 Billion hashes per second and costing $320 [18], to generated to such a massive hash rate. The pool, therefore, consists of a highly imbalanced and irregular distribution of miners with a minority of them contributing to a large fraction of the total hash rate.

The results established in this paper show how standard hardware miners in large mining pools such as this, could quite significantly add to the overall hash rate. This can be roughly quantified by assuming a mere 1000 of these users to be possessing hardware capable of hash rates between the hash rates of test machines C1 and C2. Taking an average of 35 Million GPU hashes per second and 15 Million CPU hashes per second, 1000 miners can be expected to output an additional 15 Trillion hashes per second, which is a considerably large boost when seen in terms of actual hashes generated per second.

This boost could serve to be advantageous for botnet herders attempting to harness compromised machines to perform Bitcoin mining. As of present timer, a few botnets like ZeroAccess and the Miner Bot are known to actively perform Bitcoin mining by usage of the GPUs of controller machines. A modified malware updated to harness both the CPUs and the GPUs of machines in the botnet for Bitcoin could significantly boost their earnings.

Also, unable to compete with enormously powerful ASIC mining rigs and steadily rising hash rates, and therefore, rising difficulty rates in Bitcoin, average miners are now turning to mine for other increasingly popular cryptocurrencies like Litecoin [19]. These are derived from Bitcoin and share its fundamental principles. Owing to its comparatively lower value ($30, December 2013) most miners are tempted to contribute to Bitcoin mining instead of Litecoin mining, which uses scrypt hashing instead of Bitcoin’s SHA256. A botnet herder could take advantage of the comparatively smaller number of miners and hence, smaller total hash rates, and go on to be a relatively dominant Litecoin miner, which is very improbable in the case of Bitcoin mining. Furthermore, if a botnet comes to possess enough computing power to attribute to 51% or more of the entire hash rate, an attacker could even modify a past bock, undo and then redo “Proof-of-Works” of blocks and eventually surpass the work of honest miners. This is, however, not very possible in the Bitcoin network as achieving 51% of the entire mining hash rate is extremely impractical on any botnet, even with ones having millions of bots.

Future works would involve a similar study featuring Litecoin mining and other Bitcoin based cryptocurrency mining. Litecoin has most of its features similar to that of Bitcoin with the exceptions of the hashing algorithm used in mining and certain differences in its protocol of operation.

ISBN: 978-1-63248-000-2 doi: 10.15224/ 978-1-63248-000-2-42

References

[1] Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System", 2008, Online, http://bitcoin.org/bitcoin.pdf

[2] James Wyke, "The ZeroAccess Botnet – Mining and Fraud for Massive Financial Gain", 2012, Online, http://www.sophos.com/en-us/medialibrary/PDFs/technical%20papers/Sophos_ZeroAccess_Botnet.pdf

[3] https://en.bitcoin.it/wiki/Pooled_mining

[4] https://en.bitcoin.it/wiki/Comparison_of_mining_pools

[5] http://www.securelist.com/en/blog/208188132/Gold_rush

[6] Plohmann, D. ; Cyber Defense Res. Group, Fraunhofer FKIE, Wachtberg, Germany ; Gerhards-Padilla, E.Rwh, "Case study of the Miner Botnet", Cyber Conflict (CYCON), 2012 4th International Conference on , pp. 1 – 16, 2012

[7] http://www.butterflylabs.com

[8] https://www.avalon-asics.com

[9] https://github.com/luke-jr/bfgminer

[10] https://github.com/Diablo-D3/DiabloMiner

[11] https://github.com/phoenix2/phoenix

[12] https://github.com/m0mchil/poclbm

[13] http://ufasoft.com/open/bitcoin/

[14] https://github.com/ckolivas/cgminer

[15] http://csrc.nist.gov/groups/STM/cavp/documents/shs/sha256-384-512.pdf

[16] https://deepbit.net/

[17] https://www.btcguild.com/

[18] http://www.cryptocurrency.org/hardware/

[19] https://github.com/litecoin-project/litecoin

About Author:

I'm an outgoing student of Anna University, Dept. of Science and Engineering, CEG. I've been actively interested in researching and implementing novel ideas, tools and software since pre college days. I am interested in digital security, study and beating of reverse engineering, network programming and game programming