“Crowd” Computing – Volunteer Computing, Cryptojacking and their Policy Needs

A Research Paper submitted to the Department of Engineering and Society

Presented to the Faculty of the School of Engineering and Applied Science

University of Virginia • Charlottesville, Virginia

In Partial Fulfillment of the Requirements for the Degree

Bachelor of Science, School of Engineering

Ryan Pope

Spring 2022

On my honor as a University Student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments

Advisor

Sean M. Ferguson, Department of Engineering and Society

"Crowd" Computing – Volunteer Computing, Cryptojacking and their Policy Needs

Since the advent of the desktop PC in the mid-20th century, personal computing devices have become increasingly commonplace in society. In the modern day, practically every member of a developed society owns at least one computing device, and thus the total computational power of a combined society is often far larger than that of any one node. Following the increase in computing power is the potential of volunteer computing – a model which, instead of computing difficult problems on one powerful computer, solves the same problem on many less powerful 'volunteer' computers. This model aims to provide an alternative to centralized investment, such as in supercomputers, by distributing the computational load over several nodes on a network. Implementations of this model have been around for over 50 years in many different applications, including cloud servers (such as Amazon AWS), telecommunication networks (such as telephone and cellular data networks) and even the World Wide Web (through DNS servers, which then route to independent content servers).

More recently, the potential for distributed computing to generate monetary value has become evident through the rise of cryptocurrencies. By investing in computing hardware, participants on the cryptocurrency blockchain are rewarded with currency for solving complex mathematical problems. This has led to a significant increase in computing hardware investment, but has also caused increased danger for non-consensual use, or 'poaching', of computing resources belonging to unsuspecting actors (Saad et al., 2019). Existing legislation, such as the Computer Fraud and Abuse Act (CFAA), is far too broad and does not effectively target poachers of volunteer resources, and as such there is a large potential for volunteers to be taken advantage of as distributed methods enter the mainstream.

This paper will examine the existing state of volunteer computing, including its technological advantages, its environmental outcomes, as well as the policy that governs it. Additionally, as the concept of leveraging volunteer computing for revenue generation is rather novel, throughout the discussion in this paper I will present and examine existing solutions of distributed computing and crypto-mining. Using these case studies, I will synthesize their respective best practices and suggest a new cohesive model for volunteer crypto-mining. This model will include improvements to existing policy and implementation, which will mitigate the risk of malicious outcomes while simultaneously maximizing the potential for the emergence of volunteer computing as a sociotechnical system – possibly solving some of modern society's most difficult problems.

Distributed & Volunteer Computing

Technological Advantages

The main advantages of distributed computing originate from the decentralized nature of the computing hardware. Because each computing element ('node') computes separately from the other nodes in the network, they will tend to fail independently of one another, and thus system-wide crashes are far less likely to occur. This is a huge plus in continuity-intensive processing efforts, such as the World Wide Web or frequently polled cloud servers – even a small lapse in computing uptime for these efforts could lead to huge issues for the users. Centralized computing, on the other hand, often has what is known as 'single point of failure', i.e., if a single integral component of the central machine fails, the entire computer, server, etc. is at risk of going down entirely until a technician can repair the issue. Even with redundancy

(as is present in many centralized systems) it is impossible to affordably replicate the level of reliability that a multi-node distributed system can provide.

Additionally, the decentralization of the hardware allows for great strides to be made in terms of system scalability (Jogalekar & Woodside, 2000) – simply adding another identical node to the network has the potential to greatly increase computing capacity (depending, obviously, on the initial size of the network). To scale a centralized computer, generally the only options of adding more computing power are to either increase the size or speed of existing components (processors, memory, storage) which eventually hits a serious bottleneck as the cost of better components greatly outweighs the revenues (whether monetary or otherwise) from serving those additional clients. Cases do exist where distributed computing hardware is less scalable than centralized computing (Aguilera et al., 2007), but in the general case it is fair to assume that scalability favors distributed methods.

Environmental Impacts

One of the major downsides of existing methods of both distributed and centralized computing is their reliance on the construction, maintenance, and use of large-scale "data centers", which have begun to spring up nationwide as a response to the increasing need for computing power. These data centers not only incur a significant cost to build, but also have been found to use a significant amount of energy and contribute to around 0.3% of global carbon emissions (Gelenbe & Caseau, 2015). Especially when climate change is of such great concern, using volunteer computing to mitigate this environmental impact could be a huge benefit of the method.

Mengistu and Che (2019) found that many of the amortized costs of running a data center, such as power redundancies, cooling systems, and battery backups could easily be mitigated by employing decentralized volunteer computing, and that therefore "it is a greener solution that can reduce the carbon footprint of the Information and Communication Technology industry" (pp. 2).

Many of the distributed systems which already exist draw a significant number of parallels to the volunteer computing method which I propose in this paper – the proprietary hardware which are typically engaged in these parallel distributed computing efforts may have a significant amount more computing power than is present in, say, a typical desktop computer, but consumer hardware greatly outnumbers this proprietary hardware. Mengistu and Che (2019) found that there are over 2 billion underutilized (meaning having available computation resources) desktop computers worldwide, and that PCs belonging to institutions and organizations can be idle "up to 97% of the time" (pp. 2). Additionally, Mengistu and Che found that "the current computing landscape is also comprised of more than 50 billion portable devices" (pp. 2), many of which are also underutilized and potentially possess significant computing hardware. Some examples are gaming consoles, which possess robust graphics, processing, and memory hardware, and thus when not in use, this hardware could easily contribute towards distributed computing efforts.

Folding@home: A Case Study in Volunteer Computing

Currently based at Washington University St. Louis, the Folding@home project has used a number of hardware components of various devices including desktop computers, laptops, multi-core processing computers, gaming consoles, and some smartphones, to run

protein dynamics simulations in an effort to better understand a variety of diseases, and to allow research labs to more effectively develop therapeutics for them. To do this, each computer is given what is referred to as a 'work unit', which is a portion of a larger simulation. Once the work unit is processed, the result is sent back to the Folding@home servers, and the servers then award the volunteer 'credit points' which are tracked on a leaderboard on the Folding@home website (Larson et al., 2009). This sort of competitive nature not only gives some incentive (although this incentive holds no monetary value) to the volunteers, but also allows the volunteers to essentially compete against one another to see who contribute the most to the project.

Especially in current times, where the COVID-19 pandemic is at the forefront of everyone's lives, Folding@home has seen a huge uptick in computing participation – Mengistu and Che (2019) found that by April 2020, the effort had reached 2.43 exaflops of computing power (a flop is a floating point operation per second – so Folding@home was able to compute $2.43*10^{18}$ floating point operations every second). As a result, Folding@home is now one of the world's fastest computing systems (Patrizio, 2020) and has set a precedent for the potential power of future volunteer computing efforts.

Coinhive: A Case Study on the Shortcomings of Existing Policy

In 2017, a crypto-mining service known as Coinhive was launched, allowing website owners to inject a JavaScript library onto their sites, harnessing the computing power of users who visit the site while the page is open, thus allowing for monetization of web traffic outside of the traditional advertising-based monetization. Users would have a wallet key, as would

Coinhive, and thus 70% of the cryptocurrency revenues from the miner went to the user, and 30% went to Coinhive (Krebs, 2018).

The library was used to mine Monero coin, which unlike Bitcoin, is largely untraceable, and thus the Coinhive library became a very attractive weapon for cybercrime. Rather than being loaded by the website owners themselves, in most cases Coinhive was maliciously loaded onto hacked sites, including the Los Angeles Times, Blackberry, Politifact and more. Additionally, some hackers found ways to inject the Coinhive library through routers, thus granting them the ability to infect not just specific websites, but even every device connected to a certain wireless network.

Coinhive opened the door to a multitude of security concerns, and in fact inspired the coining of the term 'cryptojacking' – meaning any software application which poaches users' resources without their consent, with the intent of mining cryptocurrency. In response to the malicious use of Coinhive by cybercriminals, many anti-malware services such as Malwarebytes began blocking connections to cryptojacking servers (including those used by Coinhive), thus reducing the effectiveness of their efforts (Tahir et al., 2019). Coinhive was not at all motivated to protect sites and users against cryptojacking, either, as there was little to no legislation surrounding the unauthorized use of computing services. Additionally, Krebs (2018) found that even in cases where sites had requested the miner to be removed from their site, Coinhive would simply invalidate the user key, but maintain their own, thus giving them 100% of the revenue from the miner and doing absolutely nothing about the cryptojacking scripts.

In response to the sanctioning of their software by antivirus softwares, Coinhive developed a version of their mining software called AuthedMine, which required that the user authorized the use of the mining application while the site was active. Nonetheless, not only did Malwarebytes still block this version of Coinhive but found via its statistics that nearly 99% of the uses of Coinhive's code used the unauthorized version of the library (Segura, 2018).

While unauthorized computing such as that performed in cryptojacking is prohibited under the Computer Fraud and Abuse Act (CFAA), the lack of specific and targeted legislation leaves something to be desired. The only provisions of this act which deal with cryptojacking are those which deal with hacking in the general sense, addressing items such as "unauthorized access of protected computers" or "transmitting data to or from a protected computer and causing damage". While perhaps a judiciary body may read between the lines here and rule that a cryptojacking effort violates this Act, aside from the access of the computer, cryptojacking does not entail real 'damage' and thus cryptojacking seems to be an edge case that currently doesn't have much legislation addressing it (Dysart, 2019).

Discussion

Volunteering and Revenue Generation

The Coinhive fiasco alerted the world of the danger of malicious volunteer computing solutions. However, it also had a secondary outcome – unearthing the potential for a more meticulously crafted solution to bridge the gap between volunteer computing and revenue generation. Mengistu and Che (2019) write that "in order to attract volunteers […] it is vital to have a business model that includes economic incentives to encourage the volunteers not only

to donate resources but also continue donating" (pp. 8-9). If a method can be developed to ensure that volunteer resources are provisioned appropriately, and that each volunteer clearly understands their role as a part of the computing network, there emerges a tremendous computing potential in the volunteer network. Additionally, if a monetary value can be placed on a node's computational power, the owner of the node gains the ability to market this power. This could revolutionize the Internet's revenue streams, which are currently dominated by advertisement- and subscription-based models. Given an appropriately governed volunteer model, users could potentially surrender a small amount of their computing power rather than being subject to advertisements or subscription fees. How exactly to create a model which appropriately governs this network remains to be seen, however there are major policy and implementation improvements which are obvious from the current state of volunteer computing.

Policy Improvements

It is evident that the downfall of Coinhive was not necessarily because of its initial purpose, but rather was a result of the shortcomings of its implementation. Additionally, legislation in the field of volunteer computing can be very sparse when it comes to developing the software both ethically and effectively (Lavoie & Hendren, 2019). For volunteer computing resources to be used ethically and effectively, a new policy must first be drafted to meet the ever-accelerating world of computing.

As Eskandari et al. (2018) write, "cryptojacking […] might grow into a regulatory issue where governmental bodies could use legislative approaches to obtain consent." (pp. 6). This suggests that there must be distinct legislation to address the meaning of a user's consent

for use of their computing resources. One major issue with the unlegislated consent-based model, such as with AuthedMine, is that "cryptojacking after gaining user consent is controversial primarily because it is unclear is users understand what they are consenting to, what they receive in return […] and whether it is a fair exchange" (Eskandari et al., 2018, pp. 7). Many everyday users are not technically knowledgeable, and likely have a hard time understanding what exactly the surrender of their resources entails. Often there are unseen costs associated with this surrender, including faster degradation of components, and slower computation speeds when volunteering. Therefore, the responsibility for understanding these costs falls on the legislators who create the policy and the engineers who develop the system for revenue generation.

While the general anti-hacking legislation which already exists, such as that in the CFAA, does cover a large portion of cybercrime, edge cases such as that uncovered by the Coinhive fiasco are left wholly unprotected. A user's computing resources should be treated as property, whether idle or otherwise, and therefore software which non-consensually poaches these resources should be treated as theft. Any time volunteer computing software such as Coinhive wishes to use resources, it is critically important that the user is fully aware of what the purpose of their participation is, what type of performance implications it has, and to be able to fully opt out of volunteering if desired.

As Dysart (2019) mentions, another of the main downsides of cryptojacking was that users would find that their devices would become incredibly slow while navigating infected sites, as the Coinhive software did not appropriately cap its usage as a proportion of the device's available resources. Users on more powerful devices may not have noticed that they were being cryptojacked, but users on less powerful devices experienced significant slowness,

and occasional crashes. Therefore, legislation must ensure all details of resource use be delineated in a document separate from the Terms of Service and Privacy Policy, which we will call the 'Terms of Resource Use'. It is critical for this document to be clearly understood by the public, as all users have the right to know why and how their computing resources are used. Additionally, it must be made certain that each document clearly explains to the user the unseen costs of their volunteering and requires that the user confirms after reading – this ensures that the disconnect between the host and the volunteer mentioned by Eskandari et al. (2018) is resolved. The Terms of Resource Use must also have a clause for the provisioning of user resources, and because every computing device has a finite amount of computing resources, this provision should be a modest proportion of overall resources.

Implementation Improvements

For a resource provisioner to respect the user's privacy and not give too much information to the volunteer computing software, it would likely be optimal for the browser to have an inbuilt virtual resource engine, as suggested by Eskandari et al. (2018). Anderson (2019) explains that BOINC, a common platform for volunteer computing applications, includes support for wrapped and virtual machine applications, and writes that "VM technology provides a strong security sandbox" (pp. 106). A sandboxed resource engine could be provisioned as a portion of the host device's overall resources and could be spun up each time resources were requested from the browser. Not only does separating out this engine allow for proper provisioning of resources, but also ensures that the volunteering user's data is protected.

Another issue that Coinhive suffered during its existence was that it could easily be injected into hacked sites to generate revenue for cybercriminals (Krebs, 2018). To protect

against this, legislation must require implementation of a digital signature-based system which is validated with the host at runtime (similarly to the current validation of SSL certificates), as mentioned by Mengistu and Che (2019, pp. 11). While this implementation description is very general, if implemented correctly it would ensure that site owners or webmasters are the only ones who can run volunteering code on their pages, and thus could eliminate the risk of nefarious entities being able to generate revenue from others' content.

While the methods used by the Coinhive team were obviously less than optimal, the precedent set by the project leaves several opportunities for positive volunteer computing. Provided that the CFAA (or a similar Act) is expanded to include requirement of consent for resource use, consider a case in which a commercial website partners with a project such as Folding@home. Instead of showing obstructive advertisements to users, or pay-walling content, the website offers an option to consume a small amount of the user's resources to perform small computations towards the project goal. As browser-based volunteer computing solutions are ever-increasing in efficiency, such as the model suggested by Matsuo et al. (2019), this small amount of volunteered resources could potentially perform a significant amount of computations. In most cases, this revenue model would leave both parties satisfied – the user would not be bothered by advertisements and would likely not even notice the impact to their device's performance, and the provider would receive revenues (in terms of work units rather than advertising revenue).

Conclusion

All in all, this model could open the door for some serious advancements in all computation-driven fields, from mathematics to genetic science, and more. If the policy can

appropriately accelerate to meet the state of the volunteer computing world, the potential for ethical progress in both industry and non-profit organizations via volunteer computing is immense. The fact that Folding@home became one of the world's most powerful supercomputers with its relatively small amount of participation leaves one to wonder: if even 50% of those 2 billion underutilized PCs started using their computing power for greater volunteer computing efforts, how much would the world of computing accomplish with little to no additional hardware required? It's likely that we will see the answer to this question within the next 10 to 15 years.

References

Aguilera, M. K., Merchant, A., Shah, M., Veitch, A., & Karamanolis, C. (2007). Sinfonia: A New Paradigm for Building Scalable Distributed Systems. *Proceedings of Twenty-first ACM SIGOPS Symposium on Operating Systems Principles - SOSP '07*. doi:10.1145/1294261.1294278

Anderson, D. P. (2019). BOINC: A platform for Volunteer Computing. Journal of Grid Computing, 18(1), 99–122. doi:10.1007/s10723-019-09497-9

Bijmans, H. L. J., Booij, T. M., & Doerr, C. (2019). Inadvertently Making Cyber Criminals Rich: A Comprehensive Study of Cryptojacking Campaigns at Internet Scale. *Proceedings of the 28th USENIX Security Symposium*.

Dysart, J. (2019, September). Computer Hackers' New Trick: Stealing your computing processing power. *Parks & Recreation*, 18-20.

Eskandari, S., Leoutsarakos, A., Mursch, T., & Clark, J. (2018). A first look at browser-based cryptojacking. *2018 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. doi:10.1109/eurospw.2018.00014

Gelenbe, E., & Caseau, Y. (2015). The impact of information technology on energy consumption and carbon emissions. *Ubiquity*, *2015*(June), 1–15. doi:10.1145/2755977

Hu, J. (2018). E-commerce big data computing platform system based on distributed computing logistics information. *Cluster Computing, 22*(S6), 13693-13702. doi:10.1007/s10586-018-2074-6

Jogalekar, P., & Woodside, M. (2000). Evaluating the scalability of distributed systems. *IEEE Transactions on Parallel and Distributed Systems, 11*(6). doi:10.1109/hicss.1998.649248

Koomey, J. G. (2008). Worldwide electricity used in data centers. Environmental Research Letters, 3(3), 034008. doi:10.1088/1748-9326/3/3/034008

Krebs, B. (2018, March 26). Who and What Is Coinhive? Retrieved October 25, 2020, from https://krebsonsecurity.com/2018/03/who-and-what-is-coinhive/

Larson, Stefan & Snow, Christopher & Shirts, Michael & Pande, Vijay. (2009). Folding@Home and Genome@Home: Using distributed computing to tackle previously intractable problems in computational biology. arXiv.

Lavoie, E., & Hendren, L. (2019). Personal volunteer computing. *Proceedings of the 16th ACM International Conference on Computing Frontiers*. doi:10.1145/3310273.3322819

Matsuo, H., Matsumoto, S., Higo, Y., & Kusumoto, S. (2019). Madoop: Improving browser-based volunteer computing based on modern web technologies. *2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER).* doi:10.1109/saner.2019.8668014

Mengistu, T. M., & Che, D. (2019). Survey and Taxonomy of Volunteer Computing. *ACM Computing Surveys, 52*(3), 1-35. doi:10.1145/3320073

Patrizio, A. (2020, April 14). The coronavirus pandemic turned Folding@Home into an exaFLOP supercomputer. Retrieved October 25, 2020, from https://arstechnica.com/science/2020/04/how- the-pandemic-revived-a-distributed-computing-project-and-made-history/

Saad, M., Khormali, A., & Mohaisen, A. (2019). Dine and Dash: Static, Dynamic, and Economic Analysis of In-Browser Cryptojacking. *2019 APWG Symposium on Electronic Crime Research (eCrime)*. doi:10.1109/ecrime47957.2019.9037576

Segura, J. (2018, September 04). The state of malicious cryptomining. Retrieved October 25, 2020, from https://blog.malwarebytes.com/cybercrime/2018/02/state-malicious-cryptomining/

Segura, J. (2019, November 18). Cryptojacking in the post-Coinhive era. Retrieved October 25, 2020, from https://blog.malwarebytes.com/cybercrime/2019/05/cryptojacking-in-the-post- coinhive-era/

Tahir, R., Durrani, S., Ahmed, F., Saeed, H., Zaffar, F., & Ilyas, S. (2019). The Browsers Strike Back: Countering Cryptojacking and Parasitic Miners on the Web. *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*. doi:10.1109/infocom.2019.8737360