*Ramsey Alsheikh*

# THE FREE SOFTWARE REVOLUTION

Richard Stallman once said, "Value your freedom or you will lose it" (Stallman, "Linux, GNU"). This comment was not alluding to any political or social situation, rather, it refers to free software.  Free software, or "open source" software, is software that has been published under permissive licensing that allows its source code, the human readable outline of the program, to be made accessible. It enforces all projects deriving from it to provide their source code as well, and is popular in today's coding scene.  However, this model needed to reassure doubts about its practicality amongst many in the field before it could see widespread adoption. Because of the rise of the GNU project and the Linux kernel's open sourcing, GNU/Linux was born, which influenced the masses' adoption of open software as a legitimate coding methodology.

Prior to all this, software was largely proprietary.  Standardized microprocessors had just been released, making the selling and distribution of identical programs to computer users easier than ever (Landley).  The Third Circuit Appeals court had ruled on Apple v. Franklin, which effectively allowed software companies to keep their source code from users without any legal consequences (Landley).  These developments made the software market a lucrative business, one in which you made a near 100% margin on every sale.  As such, commercial interests kept source code a heavily protected company secret.  If anyone were ever to get a hand on it, they would easily be able to replicate it, steal customers, and diminish the wild profit they were making. ("Proprietary Software Definition").  Such was the nature of life back then: code was the company's, and any attempt to decipher it would entail countless lawsuits and legal headaches.  This did not sit well with many of the so-called "hackers" of generations prior.  They viewed this ultra capitalism mode of development as a hindrance to innovation.  In their eyes,

progress was limited by those you had at the company.  Even though there were intelligent and passionate programmers who would love to see the inner workings of a program and to improve it, they were prohibited from ever coming near the source code, lest they should spread the product around for free.  Subsequently, many people came to resent proprietary software.  To these people, keeping source code from people was and remains "antisocial", "unethical", and "simply wrong" (Stallman, "Free Software"). Thus, the divide between the businessmen and the hackers came to be.

Amidst the hackers, one stood out.  His name was Richard Stallman.  From his early years as a Harvard undergraduate student, he had become a well-known figure amongst the hackers at a nearby laboratory at MIT, working on their Incompatible Timesharing System (Bretthauer).  This particular lab had one of the most liberal policies when it came to sharing their software: free for anyone who wanted it for any purpose (Stallman, "Free Software").  However, this utopia did not last. Eventually, all the hackers were hired off by commercial interests, and, perhaps worse, a new, closed-source operating system was adopted (Stallman, "Free Software").  Left with no other choice, Stallman started a movement.  On a web forum on September 27, 1983, Stallman announced the beginnings of his GNU (GNU's Not Unix) project, an ambitious undertaking to create a Unix like, usable, standalone operating that was above all else, free (Stallman, "Initial Announcement").  Throughout the following years, the project grew and developed most of the workings of an operating system.  They were able to build a text editor, programming language, terminal shell, and nearly everything else except for one thing: the kernel at the core of the operating system (Bretthauer).  Without one, nothing could be done. The

next logical step was to begin working on a kernel, but progress was slow. The project had stalled.

At the same time, a thousand miles away, across the ocean, University of Helsinki student Linus Torvalds had been dedicating significant time to his personal hobby of programming (Torvalds and Diamond 62).  Unhappy with the operating system Minix  that he had recently installed on his new computer, Torvalds had been working on reimplementing some of the features he didn't like (Torvalds and Diamond 62).  It grew to encompass many features, to the point where it could almost stand by itself, independent of Minix, as a kernel, the core of the operating system responsible for essential functions (Torvalds and Diamond 77).  But as an operating system, it was very far away from being independent - it used a lot of GNU software (Bretthauer).  Without the large mass of GNU code for Torvalds to use, his kernel would have never had enough features to be usable. After posting about it on an online newsgroup, a couple of fellow hackers had reached out to Torvalds to use his new kernel (Torvalds and Diamond 82). Some wanted to help with the development (Torvalds and Diamond 90).  Slowly, his new operating system was gaAt this point the fruits of both Torvald's and Stallman's labor came to be; Torvalds finally released his new operating system project, GNU/Linux, to the world (Torvalds and Diamond 88).  Running on purely Torvald's and the GNU project's code, it was completely open source and modifiable (Torvalds and Diamond 96).  Amidst the other proprietary or only semi-open systems, this was quite remarkable.  It was also an interesting proof of concept, with many intriguing features the developer community liked.  Unfortunately, this wasn't enough to make GNU/Linux popular by itself.  The fact was, at this stage, GNU/Linux was only usable for a highly technical audience, and many computers were unable

to support it as it stood.  It was prone to crashes, data corruption, and inefficient practices (Torvalds and Diamond 88).  Undeterred, Linus continued his work, slowly chasing down bugs plaguing his users and GNU/Linux's usability.  His project would continue to snowball in popularity as bug fixes and patches rolled out, which increased its influence and community. ining traction.

But at the time Torvalds made GNU/Linux available in 1991, its relevance was limited to his peers in the field of computer science. There was ambivalence among his fellow computer scientists. While they were supportive, they acknowledged that GNU/Linux had a long way to go before it could be used professionally in regards to code faults and features (Torvalds and Diamond 88).  Over time, they would get more and more involved with his project, contributing patches and suggestions (Torvalds and Diamond 91).  The early stages of GNU/Linux release mainly affected the size of the GNU/Linux developer community, essential for the project's development.  People from all over the world were contributing code to make Linus's operating system better, very soon after its release.  In this way GNU/Linux made a name for itself and got people working for it fairly quickly.

The most significant benefits came later.  As GNU/Linux matured and evolved, releasing officially in 1994, it would find many niches across many fields.  It powers a majority of the Internet's servers, making available websites for the masses and hosting information for millions, if not billions, of people across the world (Finley).  The Android mobile operating system adopted the GNU/Linux kernel as its own, thus, GNU/Linux runs on a majority of today's mobile phones (Finley).  But perhaps the biggest effect GNU/Linux has had on the world comes from the approval and attention it brought to open source software (Finley).  Developers today

are well aware of the huge potential of open source software. From the wild success of GNU/Linux, companies have become less afraid of making their source code available to users. They are acutely aware of the customizability and better code that comes from open sourcing. Many corporations and individuals alike have used external free software in their own coding projects and make open source software because of GNU/Linux, one of the first open source operating systems to strike it big in a world full of proprietary software. Without it, free software as a movement would not be near the level which it is today.

    In brief, GNU/Linux has changed the world. From the GNU project's humble beginnings as Stallman's personal hobby against the commercial world, to Linux Torvald's homebrew kernel, it birthed one of the world's first open source operating systems. The new GNU/Linux slowly gained influence, as it built up a community of developers contributing their code for free. Despite claims that open source was inefficient, GNU/Linux came to be a versatile, well-designed, and free operating system used for hosting the internet, running phones, among a myriad of other things. Most importantly, GNU/Linux showed the world that one does not have to keep his or her code a secret to make good technology. Companies have adopted the open source way, and code is now often available for anyone from anywhere to learn from and improve upon. GNU/Linux led a revolution - the free software revolution.

Works Cited

Bretthauer, David. "Open source software: A history." *Information Technology and Libraries*.

    *SIRS Researcher*,

    search.proquest.com/docview/215830273/B2F0503E22F74A3EPQ/1?accountid=66038.

    Accessed 10 Sept. 2019.

Finley, Klint. "Linux Took Over the Web. Now, It's Taking Over the World." *Wired*,

    www.wired.com/2016/08/linux-took-web-now-taking-world/. Accessed 18 Sept. 2019.

Landley, Rob. "May 23, 2009." *Rob Landley's blog thing for 2009.*, landley.net/notes-2009.html.

    Accessed 10 Sept. 2019.

"Proprietary Software Definition." *linfo.org*, www.linfo.org/proprietary.html. Accessed 10 Sept.

    2019.

Stallman, Richard. Free Software, Free Society: Selected Essays of Richard M. Stallman.

    www,gnu.org. Accessed 27 Sept. 2019.

---. "Initial Announcement." *GNU Operating System*,

    www.gnu.org/gnu/initial-announcement.en.html. Accessed 22 Sept. 2019.

---. "Linux, GNU, and Freedom." *gnu.org*, www.gnu.org/philosophy/linux-gnu-freedom.html.

    Accessed 21 Sept. 2019.

Torvalds, Linus, and David Diamond. *Just For Fun*. *archive.org*, archive.org/details/JustForFun.

    Accessed 12 Sept. 2019.