

J. Random Hacker

Dr. Emma Carelton

PHIL 2990 Engineering Philosophy Seminar

23 November 2024

### On Open-Source Culture

In their brief essay on the current status quo of open-source culture, izder456 highlights certain issues that plague the community, mixing personal experience and commentary to explain a route to improvement. While I agree that Izzy was mostly correct, I would remark that they made some assumptions I would like to expound on with my own knowledge, experience, and understanding of computing history; Izzy has a solid grasp of one aspect of modern-day hacker culture, but lacks some perspectives. In this essay, I will provide commentary on every point they make.

At the start of the essay, Izzy notes that “I feel like Linux’s user base is stuck in this weird fanboyish space where nerds think they’re hotshots just because they stopped using Windows or OS X.” I would certainly agree with this, as I have encountered plenty of novice Linux users that have a very un-apt understanding of the GNU/Linux operating system – as someone that enjoys building totally custom Linux systems from scratch (even choosing to disregard things like LFS), I would like to stress that most Linux users I have ran across have not truly used their operating system to its fullest potential; novice users will tend to (at least, at the time of this essay’s composition) install Arch Linux without understanding what is going on under the hood when they copy and paste commands out of a tutorial, seeking to set up the cliché

Hyperland desktop so many wish for. While I do not consider this to be bad in its own right, I do consider copying-and-pasting commands to be counterproductive to understanding Linux system at a deep level. If one wishes to truly understand a software package, they must take it apart and rebuild it from scratch. Izzy knows this all too well from OpenBSD, and I believe this allows them to make the appeals they will make later regarding that operating system. I, on the other hand, will attempt to appeal to other operating systems – many “hotshots” (as both they and I call overtly-gutsy Linux users) believe the only way to gain “computing fame” is to abandon Windows and OS X, but there is plenty of deep understanding to be had with these systems as well.

Izzy then goes on to state “The BSDs -- and especially OpenBSD -- seem to attract people who love the systems for what they are, not for what they’re not.” This is certainly a statement I have noticed the ramifications of for years. OpenBSD, at least in the past, attracted true systems engineers. It was a difficult system, but there was merit to be had in the difficulty: those that trudged through the lack of YouTube videos and copy-and-paste configuration guides found themselves rewarded with a remarkably stable, albeit somewhat archaic in some regards, UNIX system. As someone that has used OpenBSD in countless routing appliances since 2017, I feel as if I am truly a part of OpenBSD’s “target market” – people in search of OpenBSD because of what it is, not because of what is not. Paradoxically, even though OpenBSD formerly was surprisingly short on documentation outside of the manual pages and FAQ, I often compose “copy-and-paste” reminder guides for all manner of software products; I do not believe that complex computing systems should be out-of-reach of anyone, and I also do not believe that, for example, someone that has deployed OpenAFS once should have to slog through the manuals every time when they could just have easily adapted a copy-and-paste guide to fit their

environment requirements. Sadly, the broader Linux community appears to lack exactly this: novice Linux hotshot users will seemingly accept some pre-configured turnkey environment (for example, Doom Emacs), and never attempt to replicate it from scratch. While there is nothing wrong with this mindset, as I do understand that people need to get work done in a productive fashion, I also concede that I come from a point of personal bias here – I am perfectly willing to accept the default/stock configuration of EMACS and use it for everything. Therefore, I do not look down upon users that do not wish to tailor their system.

Izzy similarly remarks with “How can we go about unifying our user bases when so many discussions start and stop with ‘Well, it’s not Windows?’” Ultimately, this is true for a lot of the open-source community. They are almost immediately dismissive of commercial software systems, without realizing that so much in open-source is just an (often times, arguably inferior) clone of commercial software that already exists. I know I am sure to get a lot of flame for saying this, but most famous open-source programs are either clones/rewrites of closed-source programs, or expansions on ideals that could not be expanded upon because of a lack of source code. Linux and the GNU operating system are great examples of this – in its early days, the GNU OS was bug-ridden, lacking in performance, and seen as a second-rate solution to what was then a rather expensive UNIX System V software product. In reality, what ended up happening was a strong drive by hackers to make that terrible GNU OS into a serious competitor against the existent (and bug-ridden) commercial UNIX offerings. As commercial UNIX caught up in quality in the late 1990s, GNU/Linux once again found itself neck-and-neck against other systems. Comparing Windows to UNIX is like simultaneously comparing plants to animals, and also comparing apples to oranges. Both are living beings, both are usually consumable, but they accomplish their goals differently. The Windows influence on Linux is pronounced, as is the

UNIX influence on Windows (the SUA/POSIX subsystem (a complete port of System V release 5), the MKS Toolkit, UWIN, Cygwin, Mingw, the UNIX commands in Apple WebObjects, etc). The open-source community is filled to the brim with people that vehemently refuse the notion that a commercial software product could have any redeeming qualities, but I've been running file servers on AIX for years... and they have served me quite well. Izzy says "However, we need to steer clear of oversimplifying Linux into just a Windows alternative. Linux is so much more than that, and false advertising helps no one" – a very, very true statement. There is no Windows alternative: those that wish to make a Windows alternative will find themselves, often times, right back at Windows. Those wishing to switch to Linux for the first time will find themselves making software substitutions – GIMP for Photoshop, LibreOffice for Office, so on and so forth. Some users are content with these substitutions, but I truly began to appreciate UNIX systems when I stopped comparing them to Windows. I learned to speak the system's language; freeing my mind of its expectations was key to me. Instead of a second-rate Office clone to compose documents, I used vi and troff to compose papers. I found that UNIX utilities could be used to spellcheck my document, add tables, format it into a PDF, and print it if I so desired. I produced many term papers on an old DEC VT102 dumb terminal connected to my PC's serial port – I used UNIX because of what it was, not because of what it wasn't. I stopped comparing UNIX to Windows, and just learnt to use it (instead of crutching on familiarity with existing systems I knew).

Izzy then goes on to state a rather curious point: "I feel like desktop Linux is following the same trajectory the internet once did. At first, the web was the domain of comp sci majors and hardcore nerds. Then AOL came along, and suddenly anyone could get online." There are some issues, and one that I will jokingly remark on: in 1992, you would probably be getting a

degree in electrical engineering, instead of computer science. All joking aside, Izzy is talking about something called the “eternal September” or “the September that never ended” – this is a slang term that refers to the rise of “new users” on Usenet (that did not know netiquette or how to behave online) that would wave in every fall college semester start, but one that was pushed on forever as home users could get Internet access. What Izzy is describing, what the classical hackers felt, is normal in the human experience. Tribalism dominates all aspects of society, both in-person and online. I am as guilty as the next person at being tribalistic – I help maintain some very insular online communities, just like hacker culture used to be before the birth of the Internet for consumers. When the Internet became available to people, they found themselves at an interesting crossroads: they could either “fit in” to the existing buckets of hacker culture (of course, non-hacker-culture buckets didn’t really exist), or get flamed by hackers (oldheads, as Izzy calls them). Of course, the hacker culture of the late 80s and early 90s was extremely toxic to outsiders and very insular/tribalistic; even though I purvey the “ways of the old,” I do resent the almost hostile amount of tribalism that plagued (and still plagues) the hacker culture then.

Izzy builds on this with the following statement: “Oldheads don’t like it. Just look at how some Linux distro forums treat newbies -- it’s awful. Elitism is stupid. Just shut up, RTFM, hack, and share your knowledge with others. No one cares how l33t you are, y’know?” Ultimately, I think that Izzy is both right and wrong here. While I am in no position to defend elitism or would I prefer to do it, I also have to understand that these l33t people are also the ones solving the hard problems of the day. I also avoid telling users to RTFM – if I have to tell a user to RTFM, I clearly do not have a good grasp of a subject and cannot explain it on the spot (possibly helping a novice user in a particular subject with things that the so-called manual omits, especially prevalent in the GNU world). On the contrary, it is perfectly reasonable to tell a user to RTFM if

they have ignored your advice, or do not understand that you have an incomplete understanding of a subject and do not wish to mislead them with possibly wrong information. Izzy states that “At the end of the day, people should use whatever OS they want,” and this is ultimately where I lie. I use Windows, System V UNIX, the BSDs, GNU/Linux, OpenVMS, and VM/CMS in a day for various things, because I use the best tool for the job. Izzy then states that “...they should also understand the implications of that choice,” something I know all too well. I understand the implications of, for example, using OpenBSD – what I gain in security and networking performance, I loose in disk I/O, filesystem performance, and graphics support. Therefore, I use OpenBSD for what it’s good at – a wonderful network operating system, a pretty good desktop workstation system, but a terrible CAD workstation. For that, I use Windows. For hammering out programming tasks in various programming languages, I use VM/CMS – a one-keypress virtual-machine restart can occur in 0.025 seconds should a program crash, allowing me to test programs that have full control over a (virtual) computer and iterate quickly. So on and so forth, users should use what works best for them. The excessive evangelism of open-source users has gotten somewhat extreme in recent memory – these suggestors must understand that they are subjecting their suggestees to what is fundamentally a radically different world. Windows and UNIX both do the same things, but, ultimately, sometimes it is the journey that matters – not the destination. Izzy states this too when they round off the paragraph: “[These implications are] especially important for systems like Linux and BSDs, which are fundamentally different in philosophy and purpose from consumer-focused OSes.”

Izzy states that BSD won’t behave like Linux, and, I only partially agree with this. If I know how to work UNIX 4.2 from 1993 (or, more commonly known as UnixWare 2), I can quickly get to work on both BSD and Linux. However, if I only know how to use UNIX, I am

going to have a hard time learning Windows's fundamentally different system usage model. However, I must also recognize that I am ultimately the one at fault here: it is through my own "skill issue" in this context that I do not know the details of Windows. Izzy also states that YouTubers market Linux as a Windows alternative, and that it isn't – this is, as they have been expounding on throughout their essay, just a feint to avoid explaining the real answer to end users: how to actually morph Windows into the tool for them. I believe fully that users should be motivated to adapt any system to their needs, not just Linux. I know this quite well too – I run my mail servers on Windows, simply because MS Exchange has superior database performance compared to a text file storing 5 gigabytes of emails in a slow-to-access sequential file (or a slow directory that requires many random seeks to read through). I morphed Windows into the solution for me, I did not run to Linux. Actually, this is false: I did run to Linux for a mail server, and found myself hotly disappointed in the performance of a standard Postfix/Dovecot solution (as I have 20 gigabytes of emails).

I would also like to address the feasibility of Linux as a desktop OS. I understand that I have been rather omission of discussing desktop Linux, and I have been putting this off in an effort to build up the context to what I wish to discuss. Izzy states that "Linux shines in DevOps, sysadmin tasks, or corporate development but often falls short as a plug-and-play desktop experience." This is very true, and the BSDs have a similar issue. Izzy states quite correctly that the Linux world is very fragmented, and I can concur myself. My Linux image that I replicate and run on both servers and workstations consists of a 25 gigabyte distribution, source included. This is a distribution that I have tailored to be made in the image of UNIX System V release 4.1, and is essentially my "response" to the rampant BSD usage that I find myself in (not that I'm complaining). I have to realize here that I was merely putting another gear in the machine that is

the fragmented Linux world, and that I have shown only but one color in a spectrum of Linux distributions. I tried to produce a coherent Linux system, and it feels like a drop-in-replacement for classical commercial UNIX, but I am unsure if it could be used as a desktop system by a normal computer user. Yes, it provides Xorg, CDE, WindowMaker and GNUstep, as well as mwm and twm, but I am not confident that end-users (as I call them) should ever be subjected to this rather technical side of working a desktop operating system. That image is not, and should not, meant to be marketed towards end users – I really constructed it as more of an exercise in systems engineering, rather than seeking to fill some need I had (that need was solved by other systems, I just wanted to give Linux a chance). I found myself right in the middle of the thesis of Linux – make it yourself. Izzy remarks this is the spirit of BSD, but I also have to realize that this is a rather anachronistic view. Classical BSD was marketed as a complete OS, and many sites ran “plain” 4.3 BSD (for example). Sure, the documentation was solid and carried down to what we have today, but the BSD community has become filled with “Linux refugees.” Believe it or not, these Linux refugees were originally BSD refugees, escaping the PC BSDs of the mid-1990s due to a lack of support for much of anything (and, of course, the infamous CSRG lawsuit that effectively made BSD irrelevant in the mid-1990s), but slowly found themselves back at home as Linux (in their eyes) went off-course; they fled back to BSD to get an operating system that was on the “straight and narrow way.” Ultimately, I do not think this effect of user mindsets cycling from BSD, to Linux, and back to BSD, but Izzy was right to say that “jaded Linux oldtimers often make the jump to the \*BSDs ‘cos of this.”

All in all, I think that Izzy brings up some very important points. The open-source community, as we know it today, seems excessively recalcitrant, tribalistic, hostile, and out-of-step with what actual end-users want. In a previous seminar, I gave a discussion on historic user-

interface design patterns as they related to the shift in user interfaces that we see today. I believe that a similar shift has happened with the open-source community at large, as we are barreling towards excessive “not-invented-here syndrome,” but that is a discussion for another day. In the next seminar, we will have a guest speaker from the game development club to discuss how game development has changed over time. Thank you for attending this seminar, the next one will be next week.