# Should companies favour open source software?

Should companies always favour using open source software over closed source software? Is it worth developing open source software over closed source? Are there some circumstances where open source is a good idea, and some where it is a bad idea? I want to answer all of these questions, and find out a bit more about open source in this project.

One point that is made is that open source software isn’t user friendly. The main example that they use is Linux. Linux is one of the biggest open source operating systems on the market, it has had a huge impact on server systems, attracting companies as big as NASA to go open source[[1]](#footnote-1), for larger scale, and a lower price tag – money they can put into other things. A point of the past was that the same cannot be said for the desktop market. Over the past ten years, distributions of Linux have got even easier to install, some being even easier for the computer novice than Windows![[2]](#footnote-2) For instance, let us look at Ubuntu, a desktop Linux distribution by Canonical. To install it, all you need to do is pick up a blank CD, download the Ubuntu installer on to it, put the disk into your computer and turn it on. First it will ask you to connect to the internet, and then ask you whether you want to keep any data. From there you tell it your name and a password, and click go. When it is done, your computer reboots into your newly installed Ubuntu system, and starts you off with a simple getting started guide. In contrast, Windows gets you to configure your partitions and gets you to set up an administrator account. It also has the added difficulty of typing in the licence key, which naturally isn’t needed with Ubuntu, since it is free.

Another point that is used is that open source software isn’t reliable, which in some cases can be true. If we go back to our example of CraftBukkit, we can see that it went down in flames[[3]](#footnote-3), and many servers depended on it[[4]](#footnote-4). What a computer novice might ask is “Why can’t they just carry on using it?” The problem with this is that any newly discovered security flaws wouldn’t be fixed. However, CraftBukkit is just an unlucky example, this does not happen often. In fact, a fork of CraftBukkit called Spigot carried on with the legacy of CraftBukkit[[5]](#footnote-5), which isn’t only good for the people who were already setup with CraftBukkit, it is also more performance friendly, and allows previous users of CraftBukkit to get more out of their hardware. The simple fact is, that open source is not always reliable, but if it does go, for whatever reason, there is *always* a way out of it.

Another way to look at ‘reliability’ is whether or not the piece of software is suitable for production use – if a big security breach similar to Heartbleed was discovered, would the software be updated quickly, or would you be at risk? The theory that they use is that because it is open source, no one is responsible for it. In reality, the fact that hundreds of eyes are on the code, means that for open source software, bugs are fixed *as quickly* as closed source software, if not, *quicker*. Most bugs are only from single lines of code, and can be fixed quickly by someone who maybe does not have much to do that afternoon. After 3 commands, that fix can then be pushed to other users.[[6]](#footnote-6)

One thing that open source software has a flaw with is security. The problem is, anyone can look at the source code, and they can not only identify how it works, they can identify flaws in the security, or because they know exactly what it does and what it expects, they can create ‘invisible’ hacks. Unfortunately, other than fighting back with patches to the code, in the case of a simple security flaw, there is nothing that can be done. This time round, closed source simply *wins*, as open source does not have anything to battle this with, other than simple trust**.** On this point, there simply isn’t anything more to say.

Another very important point to look at, which the end user could not actually care less about, is ‘But someone could steal my idea!’ The fact that your code is out there, for anyone to access could mean that someone mean could download it, heavily refractor it and publish it as their own. Although with licences, this is not technically legal, there is a problem. If no one can find out, how does anyone know it is illegal?

There is one big caveat of open source. Due to the fact that the source is open, and anyone can download, copy and modify it, means that anyone can get a copy of it for free. This is a problem, as it means it is not a good way of making money. This sums up into one simple point: *Open source should not be used when developing an application commercially.*

Another one of those arguments which simply isn’t true anymore is supposedly, because you are paying for the software, you are going to get support, but because you do not pay for open source software, you will not get support. However, most companies that provide open source software provide a package of their software that comes with support, at a cost. The benefit of this is, because you do not pay for the software, you can put more money into paying for the support, thus (hopefully) better support.

On a side note, open source software is more of a community, so with most well-known pieces of open source software, other people in the community are generally quite happy to help.

1. https://web.archive.org/web/20151106100157/http://jaboutboul.blogspot.co.uk/2008/02/fedora-on-final-frontier.html (Last visited 09/11/2015, Archived) [↑](#footnote-ref-1)
2. https://web.archive.org/web/20151109095051/http://cohaesus.co.uk/dont-be-put-off-by-the-open-source-image-problem/ (Last visited 09/11/2015, Archived) [↑](#footnote-ref-2)
3. https://web.archive.org/web/20150206000441/http://dl.bukkit.org/dmca/notification.txt (Last visited 18/03/2016, Archived) [↑](#footnote-ref-3)
4. https://mineplex.com/forums/m/11929946/viewthread/23624830-bukkit-or-spigot (DEAD as of 08/01/2016) [↑](#footnote-ref-4)
5. http://web.archive.org/web/20160108100002/https://www.spigotmc.org/ (Last visited 18/03/2016, Archived) [↑](#footnote-ref-5)
6. https://web.archive.org/web/20151109095051/http://cohaesus.co.uk/dont-be-put-off-by-the-open-source-image-problem/ (Last visited 09/11/2015, Archived) [↑](#footnote-ref-6)