SEE TEXT ONLY

Choosing FreeBSD AS A PRODUCTION ENVIRONMENT

BY ASHIK SALAHUDEEN

I lead a small team of developers who write and operate software that serves a small (but growing) clientele at Accordium APS. The following is an account of how we ended up choosing FreeBSD as a production environment and some of the pros and cons of that choice.

HOW DID WE END UP WITH FreeBSD IN THE FIRST PLACE?

I have been a Free and Open Source Software advocate since 2001 and have been using Linux machines as primary drivers ever since. Until 2017, my only close encounter with FreeBSD was when I set up an LDAP instance sometime in 2007. I thought it was sane then, but never gave it much thought. 2017 was also the year when Accordium came into being, and the first time I got to be fully in charge of a production environment right from the get go.

Doubting Linux-based Operating Systems–systemd

Around 2014, the Linux ecosystem started undergoing a change that I didn't understand at first—when the systemd project started making news. I didn't give it much thought then, but later on, when Arch Linux (my preferred OS then) started pushing me towards it, I had to start using it. Immediately, I encountered a bunch of warts that I would, in hindsight, describe as growing pains.

My operating systems pre-systemd were a kernel and a bunch of programs, very loosely coupled, and while I hated the lack of uniformity, I at least knew what my computers were doing most of the time. At this point, I was getting exasperated with the general state of things, especially when most mainstream distributions decided that they'd just force systemd on everyone (notable exception: Gentoo).

I tried to live with it for a while. But it constantly broke my expectation of how my computer would run. The changes were pervasive and no amount of grappling with it would give me an idea of just what it set out to accomplish. Later on, I realized that systemd was trying to present a coherent interface--unify all Linux based operating systems—which is a laudable goal.

My computers would constantly refuse to shut down, with cryptic messages like "A stop job is running for <something>" and then hanging in there. They would start-up with cryptic error messages and troubleshooting them made me an expert in trawling github and the Unix stackoverflow. Coupled with the constant stream of vulnerabilities and the project's ridiculous responses to it, I really wanted to switch from using operating systems that used systemd.

Evaluating BSDs

In November 2016, I started to consider BSDs as a potential refuge, at least for personal use. I have been friends with Cherry G. Mathew, a NetBSD developer, for a long time, and was toying with the idea of using a BSD, at least for personal use. He, of course, stood up for NetBSD and tried to get me to use it, but also very helpfully told me what it was not capable of doing—like running VirtualBox. I decided to look for a systemd-free Linux and stick with it on my desktop.

Around the same time, Philip Paeps (trouble) spent a few days with me, and in one of our chats, he explained the coherence of FreeBSD—and the fact that it was a single operating system developed along with applications. He also told me about FreeBSD's user base on the server side, and I was surprised to learn that it was a very popular choice.

Of course, I set up a VM and tried it out to see how a Linux user would fare in a FreeBSD environment. I was productive from day one, and for most things I couldn't figure out, I just had to read the handbook.

FreeBSD AT ACCORDIUM

Accordium's development officially started in April 2017 with just two developers (including me). I found myself in charge of selecting the server environment and with nobody to whom I had to justify it. This was the first time I had to make choices regarding the entire software stack, and I was keen to at least test out FreeBSD. To my relief, there was a FreeBSD image available for deployment on Amazon Web Services.

Our requirements were very minimal, just a few API end points written in Spring Boot framework, running on JVM, a Postgres database server and a rabbitmq instance—all of which were available as binary packages. At the time, there was no production environment and the only users were us.

All of our development happened in Linux machines. Ahmed Uways Zulkurnain, the engineer who manages all our production machines now, was unfamiliar with FreeBSD at the time. He raised

SYSTEMS ADMINISTRATION

Uways is the first engineer to work at Accordium and has the following comments on FreeBSD:

“Prior to working at Accordium I was not familiar with any operating system other than Debian GNU/Linux. The only things I knew about BSDs were their legendary uptime and more permissive license. Despite my inexperience, I was able to learn how to install/configure applications, write startup scripts and configure firewalls without too much effort. I don't recall ever having to reference any documentation outside of the books and articles on the FreeBSD website. I was pleasantly surprised at how up-to-date the package versions were in the current production release. There aren't many things I have to do when it comes to managing our FreeBSD instances.

1. Keep up to date with security advisories.
2. Update packages when necessary.
3. Stay on the currently supported production release.
4. Periodically audit the logs.

One year later, FreeBSD has become my server operating system of choice. systemd logging has always annoyed me; I'm happy to once again have plain text log files. ”

Ahmad Uways Zulkurnain discovered open source software in 2005. He has since explored a wide range of topics related to computing, primarily in the realm of information security. He is currently a full stack software developer at Accordium. He enjoys twisty puzzles and chess.

a couple of very logical questions, the chief one being, "Why not Linux, we are using it ourselves after all." I asked him to bear with me and test it out first, promising to change the environments if he ever found FreeBSD lacking. It is April 2018, and he is yet to come back with an objection.

I switched to using TrueOS in June 2017, so that there was at least one person using FreeBSD to develop. My development pace was not affected in any way—if it was not available via pkg, I had it in ports.

FreeBSD IN PRODUCTION—ACCORDIUM GOES BETA

Early December 2017 saw us releasing a limited beta of our software—a bunch of APIs running on JVM, fronted by a client application written in javascript. We just went with FreeBSD in production, without stopping to think about it—why change it when it works perfectly well?

Here are the things that made me happy I selected FreeBSD.

- Our computers never crashed because of operating system problems.
- Everything works as expected—log files in `/var/log`, configurations in `/usr/local/etc` or `/etc` and so on. It is all organized well.
- The service scripts for our applications were easy to make.
- The operating system configuration is centralized. The base system configurations go in `/etc/` and the local installation specific configurations go in `/usr/local/etc`.
- Software availability—not once did we have to scratch our heads saying,"This is not available in FreeBSD."
- Updates are regular and easy to make.

We've been running for about five months, serving an increasing number of customer needs. None of our problems were due to FreeBSD. It has stayed firmly out of the way, a tireless and dependable work horse.

WHAT DO WE WANT FROM FreeBSD NOW?

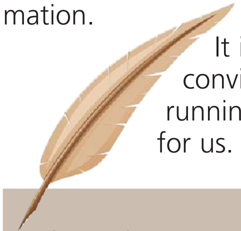
Accordium is a Java and Javascript shop. OpenJDK 8 works well on FreeBSD, but the next versions of Java (9/10/11) show no indication of supporting FreeBSD. The BSD port efforts seem to be minimal. When OpenJDK 11 rolls around, if there is no FreeBSD version, we'd be forced to move on, especially when interesting things like GraalVM are showing up.

WOULD WE RECOMMEND FreeBSD TO A START-UP?

If your applications are all one process per VM, FreeBSD is a safe, sane choice. I encourage everyone to give it a chance and save yourself a lot of pain.

If you need process/environment isolation (like Docker provides), FreeBSD has a stable, well tested feature called "jail" that would help you achieve it easily.

If you find yourself looking at OS-specific problems, the handbook will often have enough information.



It is mostly due to my personal convictions, but we found that running BSD machines made sense for us. ●

ASHIK SALAHUDEEN has been working computers for about 18 years, most of it with Unix-like operating systems and open source tech. He runs the engineering team at Accordium and works with other FOSS communities during spare time, primarily the indic language computing group, Swathanthra Malayalam Computing.

ARMv8
ENABLED AS TIER 1 ARCHITECTURE:
A COLLABORATIVE DEVELOPMENT PROJECT
FreeBSD Journal
Globally Accessible FreeBSD
Fast Filesystem
by Marshall Kirk McKusick