

Faster, smolBSD! Boot! Boot!

Pierre Pronchery

IT-Security Consultant, Defora Networks GmbH
< pierre@defora.net >

Developer & Board of Directors, The NetBSD Foundation
< khorben@NetBSD.org >

Developer, The FreeBSD Project
< khorben@FreeBSD.org >

Paper Summary

Virtualisation technology has gone a long way. Virtualisation began as early as the 1960s with IBM CP/CMS, and was already distributed as Open Source back then. On modern commodity hardware however, the technology really hit us in the mid-2000s with Intel's VT-x and AMD-V. Many different implementations compete on these platforms alone, and thankfully this is also true for our beloved BSD systems.

While the technology kept improving over time, both on the hardware and on the software side, raw performance has long been an issue and important criteria for comparing solutions. Different approaches have been released, like the emulation of existing components, the development of dedicated drivers with paravirtualization, or hybrid systems. All in all, these different approaches have been fairly complex to develop or operate.

But then, "containerization" started gaining real prominence in 2014. What we already knew as jails not only exploded in popularity, but also changed the way software was written, scaled, and distributed. Entire infrastructures can be deployed and adjusted within seconds, with new services starting in milliseconds in complete isolation from their environment.

This last property has notably been made possible with the introduction of lightweight virtual machines, or "microVMs", as implemented by Firecracker or in QEMU with the "microvm" virtual platform. Far from the complexity of prior approaches but building on its innovations, it offers a minimalist machine type without PCI nor ACPI support, allowing more barriers to be breached in performance.

Indeed, in the 2020s, we have witnessed a drag race for boot time between FreeBSD and NetBSD! This paper focuses on the progress made on the side of NetBSD, which saw the introduction of a number of different drivers, fixes, improvements and new features, each responsible for their respective gains. Overall, this has allowed reducing boot time from over 300 to just under 10 milliseconds!

In no particular order, this included: boot-time benchmarking with tslog(4), generic Xen PVH support, removing calibration loops and DELAY(), PCI-less operation with boot-time MMIO parameters, system console over VirtIO, or reducing the kernel's size with the MICROVM kernel configuration. We are happy to report that all of these improvements have made their way into the coming NetBSD release, in its version 11!

The latest addition to this pile is the possibility to start the NetBSD kernel with an in-memory root filesystem, when booting in the generic Xen PVH mode introduced here. While unfortunately, this will likely not be adopted in time for the coming release, it finally allowed us to break the 10 milliseconds barrier on the reference hardware.

Last but not least, and in line with the traditions from NetBSD, the whole solution is portable and can easily be deployed on other systems as well, like Linux or macOS. The smolBSD project, hosted on GitHub, offers a web interface and convenience scripts and tools, managing such NetBSD Virtual Machines from a web browser.

This presentation will summarise the different improvements relevant for the faster boot times, especially since Émile Heitor's presentation at BSDCan 2024. It will also explain the patch bringing support for RAM disks in the generic Xen PVH mode. Finally, the smolBSD project will be presented, with demos exploring different possibilities of the system.