SEE TEXT ONLY

WELCOME To FreeBSD 11

The FreeBSD system is constantly changing. FreeBSD 11 brings new features and fixes from two and a half years of active development. Some of these changes have been merged to recent 10.x releases such as 10.2 and 10.3, but most of them are brand new in 11.

Desktop and Laptop

FreeBSD 11 offers a variety of improvements for desktop and laptop users. First, a new system console driver is enabled by default. This driver is less VGA- and x86-centric than previous drivers. Rather than depending on BIOS ROM support for VGA text modes, the console renders text in software on framebuffers. This supports VGA adapters via graphics modes as well as the UEFI framebuffer. It also supports graphics adapters that disable VGA compatibility when using higher resolution graphics modes such as modern Intel GPUs. Software text rendering allows the console driver to render any glyph, which in turn enables UTF-8 support. In addition, the in-kernel graphics drivers include native support for Intel graphics adapters on systems with fourth-generation Core ("Haswell") processors.

FreeBSD 11 includes broader support for wireless networks. The new iwm(4) driver supports Intel integrated wireless adapters using the 3160, 3165, 7260, 7265, and 8260 chipsets via 802.11a/b/g. These adapters are used on most laptops with a fourth generation or later Intel Core processor. The iwn(4) driver (used on laptops with earlier Core processors) now supports 5-Ghz channels as well as 802.11n. The ath(4) driver for Atheros adapters supports newer, 802.11n-capable adapters with full 802.11n support in both station and AP modes. The bwn(4) driver for Broadcom BCM43xx wireless adapters now supports devices with an N-PHY (BCM4312 and BCM4321 chipsets). These adapters support 802.11n on 5-Ghz channels. The rsu(4) and urtwn(4) drivers for Realtek USB adapters now fully support 802.11n on 2.4-Ghz channels.

Support for UEFI has been improved in 11 as well. The FreeBSD amd64 (also known as x86_64 or x64) install media will now boot from either UEFI or legacy mode. UEFI systems can now boot directly from a ZFS filesystem. Other booting improvements for both UEFI and legacy systems including support for ZFS boot environments and whole-disk encryption via GELI are covered in more detail in Allan Jude's article in this issue.

The bhyve hypervisor has several new features in FreeBSD 11. Guest machines can now be started with UEFI firmware rather than using a userspace boot loader. This permits guest operating systems that support UEFI to use a native boot process. In addition, bhyve supports a graphics framebuffer when using UEFI along with additional device emulations for keyboard and mouse input. These emulated devices are backed by a VNC server. This allows guest operating systems to use a graphical interface. Users interact with these guests via a VNC client. Together with other fixes, these changes permit Microsoft Windows to run as a guest in a bhyve virtual machine. In addition, bhyve in FreeBSD 11 includes a device emulation for the Intel 82545 network adapter. This permits the use of networking with operating systems that do not support VirtIO devices. In particular, Windows can be used out-of-the-box without requiring additional VirtIO device drivers during installation.

Finally, FreeBSD 11 includes support for PCI-express native HotPlug. This includes handling of runtime insertion and removal of ExpressCard adapters in laptops as well as runtime insertion and removal of PCI-express adapters in HotPlug-capable slots.

Enterprise

FreeBSD isn't purely a desktop OS, of course, and 11 includes several new features for the enterprise. Along with support for PCI-express HotPlug, 11 also includes support for PCI Single-Root I/O Virtualization (SR-IOV). This includes the ability to create virtual functions (VFs) on supported device drivers. These VFs can be passed to virtual machines executing under the bhyve hypervisor to enable direct access to hardware for I/O requests.

The iSCSI stack introduced in FreeBSD 10 has gained several improvements in 11. FreeBSD now supports iSCSI Extensions for RDMA (iSER) providing more efficient zero-copy I/O to SCSI data buffers. The cxgbei(4) driver supports hardware-accelerated offload of iSCSI connections on TOE-capable Chelsio adapters. Finally, the reroot utility provides a means for booting a system with an iSCSI root filesystem.

The local storage stack has also seen a raft of changes in 11. FreeBSD now includes a zfsd daemon to handle automatic activation of hot spare devices and other ZFS-related events not handled in the kernel. The sesutil(8) utility supports management of disk enclosures, and the mpsutil(8) utility permits management of LSI Fusion-MPT 2 (mps(4)) and Fusion-MPT 3 (mpr(4)) SAS/SATA controllers. FreeBSD 11 includes support for pluggable disk I/O scheduling in the CAM layer. A CAM front-end for NVMe disks is present in 11 that can be used instead of the nvd(4) driver to enable CAM-specific behavior with NVMe disks.

FreeBSD 11 also includes new drivers for various hardware. The OFED Infiniband stack has been updated to version 2.1 including support for RoCE. The ixl(4) driver supports Intel XL710 40Gb Ethernet adapters, and the mlx5en(4) driver supports Mellanox ConnectX-4 and ConnectX-4LX adapters.

ARM

FreeBSD 10.1 was the first release to ship release images for supported FreeBSD/arm systems. FreeBSD 11 has expanded support for ARM-based systems in several ways.

First, ARM kernels now include a global shared-page exported to all user processes. As a result, user processes on ARM systems now use a non-executable stack. In addition, user processes are able to fetch the current time of day without system call overhead.

Second, the default floating point ABI for 32-bit ARMv6+ has been changed from soft-float to hard-float. This enables increased floating-point performance on modern processors.

Third, the virtual memory system in FreeBSD's kernel supports transparent, 1-megabyte super pages on 32-bit ARMv6+ processors. As with support for super pages on x86, this reduces the overhead of TLB misses. ARM systems are even able to map the text segment of the C runtime library with a super page. (The text segment on x86 is too small to use a super page mapping.)

Fourth, FreeBSD 11 includes support for more systems. Support for several Allwinner SoCs has been added including support for the Banana Pi, Cubieboard 1, and Cubieboard 2. 11 also includes install images for the PandaBoard and Raspberry Pi 2 systems.

Finally, FreeBSD 11 adds support for 64-bit ARMv8 processors via the FreeBSD/arm64 platform. 11 boots out of the box on Cavium ThunderX systems and support for additional systems will be available in future releases. The arm64 platform includes a global shared-page enabling use of non-executable stacks and fast time-of-day queries as on the 32-bit ARM platform. Support for super pages is already present in HEAD and will be available in FreeBSD 11.1. The FreeBSD package system includes over 20,000 prebuilt packages for FreeBSD/arm64 that are updated on a regular basis.

RISC-V

RISC-V is a new, open-source instruction set architecture. Initially motivated by computer architecture research, it is freely available for all types of use including commercially produced CPUs. FreeBSD 11 includes a new FreeBSD/riscv platform that supports the 64-bit RISC-V architecture. The RISC-V ISA for kernel mode is still in flux, but FreeBSD 11 supports version 1.9 of the draft privileged ISA specification. FreeBSD/riscv boots to multiuser in both the Spike simulator and QEMU emulator.

Developer Friendly

FreeBSD 11 is more developer friendly than ever. The llvm toolchain in the base system (including clang and lldb) has been updated to release 3.8.0. lldb is now included as part of the base system on FreeBSD/amd64 and FreeBSD/arm64. The C++ runtime library (libc++) has also been updated, which includes support for C++14.

The entire base system is now built with debug symbols. These can be installed either at install time or after install. These symbols permit developers to inspect state and single-step through base system libraries as well as application code.

Threading support has also received several improvements. The POSIX threads library now supports process-shared primitives such as mutexes and condition variables. An implementation of robust mutexes has also been added to the thread library. In addition, internal improvements to the debugging subsystem permit more robust debugging of multithreaded processes.

Finally, the DTrace tracing utility is now supported on more platforms in 11, including ARM, MIPS, and RISC-V.

Conclusion

FreeBSD's community has put a ton of effort into FreeBSD 11 over the past two and a half years, and it shows. Thank you to everyone who has contributed, whether by testing snapshots, reporting bugs, submitting patches, maintaining patches, working with users on social media, organizing conferences, etc. We hope you enjoy FreeBSD 11, and we look forward to your contributions to FreeBSD 12! ●

JOHN BALDWIN joined the FreeBSD Project as a committer in 1999. He has worked in several areas of the system including SMP infrastructure, the network stack, virtual memory, and device driver support. John has served on the Core and Release Engineering teams and organizes an annual FreeBSD developer summit each spring.