SEE TEXT ONLY

svn UPDATE

by Steven Kreuzer

At the time of this writing, the code slush for 11.0-RELEASE is in effect, and by the time you are reading this, head will be frozen in preparation for a September 2 target release date. Even as we rapidly approach that date, we are still seeing heavy development, and new and exciting features being added daily. Below is a small sample of some improvements made over the past two months, but I encourage you to take a stroll through svn log and take a look at the continued work on areas such as high-performance storage subsystems, low-latency networking, security and improved support for exotic architectures such as RISV-V, ARM, and PowerPC.

Native PCI Express HotPlug Support

As of r299142 (https://svnweb.freebsd.org/base?view=revision&revision=299142), it is possible to insert a new PCI hot plug adapter into an available PCI slot and make the device accessible to the operating system and applications without having to restart the machine. PCI-express HotPlug support is implemented via bits in the slot registers of the PCI-express capability of the downstream port along with an interrupt that triggers when bits in the slot status register change. This is implemented for FreeBSD by adding HotPlug support to the PCI-PCI bridge driver that attaches to the virtual PCI-PCI bridges representing downstream ports on HotPlug slots. The PCI-PCI bridge driver registers an interrupt handler to receive HotPlug events. It also uses the slot registers to determine the current HotPlug state and drive an internal HotPlug state machine. For simplicity of implementation, the PCI-PCI bridge device detaches and deletes the child PCI device when a card is removed from a slot and creates and attaches a PCI child device when a card is inserted into the slot. PCI-express HotPlug support is conditional on the PCI_HP option, which is enabled by default on arm64, x86, and powerpc.

Native Graphics Support in bhyve

When the bhyve hypervisor was first introduced in FreeBSD 10.0 the excitement in the FreeBSD community was palpable. For a very long time, the virtualization options available for FreeBSD left quite a bit to be desired, but that soon quickly changed when a BSD-licensed hypervisor became part of the base system. At first only the serial console was supported and it lacked the ability to emulate graphical consoles, but, in r300829 (https://svnweb.freebsd.org/base?view=revision&revision=300829), support for native graphics was added. This adds emulations for a raw framebuffer device, PS2 keyboard/mouse, XHCI USB controller and a USB tablet. A simple VNC server is provided for keyboard/mouse input and graphics output. With the appropriate UEFI image, FreeBSD, Windows, and Linux guests can be installed and run in graphics mode using the UEFI/GOP framebuffer.

Fault Management Daemon for ZFS

I am happy to say that the wait is finally over! Ever since the project was first announced several years ago, anyone who is familiar with the zpool command has been waiting for r300906 (https://svnweb.freebsd.org/base?view=revision&revision=300906); zfsd(8), a daemon that deals with hard drive faults in ZFS pools has been imported into head. zfsd(8) allows FreeBSD to be able to catch and handle disk events as they happen as well as automatically manage hotspares and replacements in drive slots that publish physical paths. Storage administrators managing a few terabytes to multiple petabytes should now be able to sleep a little easier at night.

bsdinstall/zfsboot GPT+BIOS+GELI Installs Now Make Use of GELIBOOT

FreeBSD introduced support for full-disk encryption using GELI over a decade ago, but you

were still required to keep the loader and kernel unencrypted so that the GEOM module could be loaded to handle decryption. As of r300436 (https://svnweb.freebsd.org/base?view=revision&revision=300436), ZFS boot environments can now be used in combination with full disk encryption without requiring the kernel and bootloader live outside of the boot environment.

Skein Hashing Algorithm

Support for Skein as a ZFS checksum algorithm was introduced in r289422 (https://svnweb.freebsd.org/base?view=revision&revision=289422), but is disconnected because FreeBSD lacked a Skein implementation. The Skein hashing algorithm was added in r300921 (https://svnweb.freebsd.org/base?view=revision&revision=30092) and has been connected to both userland (libmd, libcrypt, sbin/md5) and kernel (crypto.ko). A future commit will enable this hashing function in ZFS, as well.

Updates to head/contrib

The base FreeBSD userland is made up of quite a few utilities, some of which are developed outside the project. In the past few months, we've seen a few updates to the 3rd-party software that help create a great user experience.

- file has been upgraded to version 5.26. (r298192) (https://svnweb.freebsd.org/base?view=revision&revision=298192)
- libucl has been upgraded to version 0.8.0 (r298166) (https://svnweb.freebsd.org/base?view=revision&revision=298166)
- sqlite3 has been upgraded to version 3.12.1 (r298161) (https://svnweb.freebsd.org/base?view=revision&revision=298161)
- clang, llvm, lldb, and compiler-rt have been upgraded to version 3.8.0 (r296417) (https://svnweb.freebsd.org/base?view=revision&revision=296417)
- libc++ has been upgraded to version 3.8.0 (r300770) (https://svnweb.freebsd.org/base?view=revision&revision=300770)
- ACPICA has been upgraded to version 20160527 (r300879) (https://svnweb.freebsd.org/base?view=revision&revision=300879)
- libarchive has been upgraded to version 3.2.0 (r299529) (https://svnweb.freebsd.org/base?view=revision&revision=299529)

STEVEN KREUZER is a FreeBSD Developer and Unix Systems Administrator with an interest in retro-computing and air-cooled Volkswagens. He lives in Queens, New York, with his wife, daughter, and dog.