Over the past few years we have started to see an increased interest in ARM, a family of reduced instruction set computing (RISC) processors. The explosion in popularity can be partly attributed to inexpensive, single-board computers such as the BeagleBone and Raspberry Pi, which are providing folks with a simple yet extremely powerful platform to experiment on. In production environments, IT professionals are also looking toward ARM to help scale their infrastructure while simultaneously attempting to reduce power and cooling costs.

We are starting to see ARM powering everything from phones and tablets to high-performance computing environments, and I expect its footprint to grow as the Internet enables our couches to tweet and refrigerators to friend your blender on Facebook.

While ARM is still not a Tier 1 platform, every day we are seeing massive amounts of work being done to improve ARM64 support from both hobbyist developers and companies with commercial interests in building ARM-based appliances with FreeBSD at the core. In addition to all that, a brand new IO scheduler has been committed along with several changes that increase ZFS performance and give you more control over how hard a user or process can hit your storage pools. If that wasn’t enough support for Haswell, GPUs were recently introduced that now enable FreeBSD to provide improved graphics and better battery life on a wide range of laptop configurations.

I/O Limits on ZFS—
https://svnweb.freebsd.org/changeset/base/297633

Four new resources—readbps, readiops, writebps, and writeiops—have been added to rctl that now allow you to set limits on disk I/O for a process, user, login class, or jail. This should be a welcome addition to any system administrator who has to share a machine with IOP-intensive applications or users.

Improved ZFS speculative prefetch of indirect blocks—
https://svnweb.freebsd.org/changeset/base/297832

Scalability of many operations on a wide ZFS pool can be limited by the requirement to prefetch indirect blocks first. Recently added asynchronous, indirect block read partially helped, but did not solve the problem completely. This change extends existing prefetcher functionality to explicitly work with indirect blocks. Before this change, prefetcher issued reads for up to 8MB of data in advance. With this change, it also issues indirect block reads for up to 64MB of data in advance, so that when it is time to actually read those data, it can be done immediately.

ARM64 copyinout improvements—
https://svnweb.freebsd.org/changeset/base/297209

Making use of wider load/stores when aligned buffers are being copied has introduced a massive performance boost on FreeBSD/arm64. Performing a simple test of using dd to copy 1G from /dev/zero to /dev/null shows an increase from 410MB/s to 3.6GB/s.
THE INTERNET NEEDS YOU
GET CERTIFIED AND GET IN THERE!
Go to the next level with BSD Certification.
Getting the most out of BSD operating systems requires a serious level of knowledge and expertise.

NEED AN EDGE?
BSD Certification can make all the difference. Today’s Internet is complex. Companies need individuals with proven skills to work on some of the most advanced systems on the Net. With BSD Certification, YOU’LL HAVE WHAT IT TAKES!

SHOW YOUR STUFF!
Your commitment and dedication to achieving the BSD ASSOCIATE CERTIFICATION can bring you to the attention of companies that need your skills.

BSDCERTIFICATION.ORG
Providing psychometrically valid, globally affordable exams in BSD Systems Administration
Implement GELI in gptboot and gptzfsboot— https://svnweb.freebsd.org/changeset/bas(heading) e/296963

You may have noticed that the boot loader got a little fatter recently. Support for booting off GELI encrypted UFS and ZFS partitions was recently added. Additional details on how this was implemented can be found in a paper Allen Jude presented at AsiaBSDCon 2016.

Support for boot-time DTrace— https://svnweb.freebsd.org/changeset/base/297773

It is now possible to enable DTrace probes relatively early during boot before `dtrace(1) can be invoked on i386 and amd64`. The desired enabling is created using `dtrace -A`, which writes a `/boot/dtrace.dof` file and uses `next-boot(8)` to ensure that DTrace kernel modules are loaded and that the DOF file describing the enabling is loaded by `loader(8)` during the subsequent boot. The trace output can then be fetched with `dtrace -a`.

Support tracing of userspace applications on ARM64— https://svnweb.freebsd.org/changeset/base/297611

The `dtrace_getupcstack`, which allows the function call stack to be captured, has been ported over to ARM64. This allows you to use DTrace to probe userspace applications.

Updated i915 GPU Driver— https://svnweb.freebsd.org/changeset/base/296548

Recently the FreeBSD graphics team undertook a massive initiative to update the i915 GPU driver to match Linux 3.8.13. The most exciting feature of this update is that it now brings support for Haswell GPUs to FreeBSD.

ARM64 kernel address space increased to 512GB— https://svnweb.freebsd.org/changeset/base/297914

This change, which also increases the DMAP region, has also been increased to 2TiB.

Support for 4 level pagetables on ARM64— https://svnweb.freebsd.org/changeset/base/297446

The userland address space has been increased to 256TiB.

Add kern.features flags for linux and linux64 modules— https://svnweb.freebsd.org/changeset/base/297597

To help third-party applications determine if the host supports linux emulation, a new kern.features flag has been introduced for linux and linux64 modules. If kern.features.linux is equal to 1, then 32-bit Linux binaries are supported, and if kern.features.linux64 is set to 1, 64-bit binaries are supported.

Updates to head/contrib

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

- bmake has been upgraded to version 20160307— https://svnweb.freebsd.org/changeset/base/297357
- byacc has been upgraded to version 20160324— https://svnweb.freebsd.org/changeset/base/297276
- libxo has been upgraded to version 0.6.1— https://svnweb.freebsd.org/changeset/base/298083

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.