The 11.2-RELEASE code slush is currently in effect, and by the time you read this, there should be a releng/11.2. Expect to see a release announcement around June 27, 2018 and start preparing to upgrade, because FreeBSD 11.1 will reach end of life on September 30, 2018.

Introduce dwatch(1) as a tool for making DTrace more useful—
https://svnweb.freebsd.org/changeset/base/330559

Dwatch(1) is a tool for making dtrace more useful. Dwatch provides a fun and painless way to do everything from watching the system process scheduler in realtime, to filtering out filesystem events and everything in between.

Add support for zstd-compressed user and kernel core dumps—
https://svnweb.freebsd.org/changeset/base/329240

This works similarly to the existing gzip compression support, but zstd is typically faster and gives better compression ratios. Support for this functionality must be configured by adding ZSTDIO to one’s kernel configuration file. dumpon(8)’s new -Z option is used to configure zstd compression for kernel dumps. savecore(8) now recognizes and saves zstd-compressed kernel dumps with a .zst extension.

Read-behind / read-ahead support for zfs_getpages()—
https://svnweb.freebsd.org/changeset/base/329363

ZFS caches blocks it reads in its ARC, so in general, the optional pages are not as useful as with filesystems that read the data directly into the target pages. But the optional pages are still useful to reduce the number of page faults and associated VM / VFS / ZFS calls. Another case that gets optimized (as a side effect) is paging in from a hole. ZFS DMU does not currently provide a convenient API to check for a hole. Instead, it creates a temporary zero-filled block and allows accessing it as if it were a normal data block. Getting multiple pages one by one from a hole results in repeated creation and destruction of the temporary block (and an associated ARC header).

Reduce ARC fragmentation threshold—
https://svnweb.freebsd.org/changeset/base/315449

As ZFS can request up to SPA_MAXBLOCKSIZE memory block e.g. during zfs recv update, the threshold at which we start aggressive reclamation to use SPA_MAXBLOCKSIZE (16M) instead of the lower zfs_max_recordsize which defaults to 1M.

Fix OpenDowngrade for NFSv4.1 if a client sets the OPEN_SHARE_ACCESS_WANT* bits—
https://svnweb.freebsd.org/changeset/base/332790

The NFSv4.1 RFC specifies that the OPEN_SHARE_ACCESS_WANT bits can be set in the OpenDowngrade share_access argument and are basically ignored. It also changes the error from NFSERR_BADXDR to NFSERR_INVAL since the NFSv4.1 RFC specifies this as the error to be returned if bogus bits are set. (The NFSv4.0 RFC didn’t specify any error for this, so the error reply can be changed for NFSv4.0 as well.)

Make lagg creation more fault tolerant—
https://svnweb.freebsd.org/changeset/base/332645

Warn, don’t exit, when SIOCSLAGGPORT returns an error. When we exit with an error during lagg creation, a single, failed NIC (which no longer attaches) can prevent lagg creation and other configuration, such as adding an IPv4 address, and thus leave a machine unreachable. Preserve non-EEXISTS errors for exit status from SIOCSLAGGPORT in case scripts are looking for it. Hopefully, this can be extended if other parts of ifconfig can allow a "soft" failure. Improve the warning message to mention what lagg and what member are problematic.

Add support for TCP high precision timer system (tcp_hpts)—
https://svnweb.freebsd.org/changeset/base/332770

It is the forerunner/foundational work of bringing in both Rack and BBR which use hpts for pacing out packets. The feature is optional and requires the TCHPTS option to be enabled before the feature.
will be active. TCP modules that use it must assure that the base component is compiled in the kernel in which they are loaded.

**Remove caching from getlogin(2)—**
https://svnweb.freebsd.org/changeset/base/332119

This caching has existed since the CSRG import but serves no obvious purpose. Sure, setlogin() is called rarely, but calls to getlogin() should also be infrequent. The required invalidation was not implemented on aarch64, arm, mips, and riscv so updates would never occur if getlogin() was called before setlogin().

**Remove support for the Arcnet protocol—**
https://svnweb.freebsd.org/changeset/base/332490

While Arcnet has some continued deployment in industrial controls, the lack of drivers for any of the PCI, USB, or PCIe NICs on the market suggests such users aren’t running FreeBSD. Evidence in the PR database suggests that the cm(4) driver (our sole Arcnet NIC) was broken in 5.0 and has not worked since.

**Add RFC 5424 syslog message output to syslogd—**
https://svnweb.freebsd.org/changeset/base/332510

Add fprintlog_rfc5424() to emit RFC 5424 formatted log entries. Add a "-O" command line option to enable RFC 5424 formatting. It would have been nicer if we supported "-o rfc5424", just like on NetBSD. Unfortunately, the "-o" flag is already used for a different purpose on FreeBSD. For people interested in using this, the feature can be enabled by adding the following line to /etc/rc.conf:syslogd_flags="-s -O rfc5424"

**Add sortbench—**
https://svnweb.freebsd.org/changeset/base/332796

This is a set of benchmarks of qsort, mergesort, heapsort, and optionally wikisort and a script to run them.

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.