SEE TEXT ONLY

svn UPDATE

by Steven Kreuzer

Very recently something exciting happened to me. One night I ran svn up in /usr/src, kicked off a build, and went to bed. When I woke up in the morning, I found my laptop sitting at a prompt asking me to log in to my FreeBSD 13.0 workstation. That can only mean one thing. We are currently in the middle of the FreeBSD 12.0 release cycle! For a while there was not much action in HEAD, as the release engineering team had frozen the tree to create a branch of what will become 12-STABLE. That freeze has been lifted and we once again are seeing lots of new features and exciting changes. Can you think of a better way to end the year?

Add macros for reading performance counter CSRs on RISC-V— https://svnweb.freebsd.org/changeset/base/340399

The RISC-V spec defines several performance counter CSRs such as: cycle, time, instret, hpmcounter(3...31). They are defined to be 64-bits wide on all RISC-V architectures. On RV64 and RV128 they can be read from a single CSR. On RV32, additional CSRs (given the suffix "h") are present that contain the upper 32 bits of these counters and must be read as well. (See section 2.8 in the User ISA Spec for full details.)

This change adds macros for reading these values safely on any RISC-V ISA length. Obviously, we aren't supporting anything other than RV64 at the moment, but this ensures we won't need to change how we read these values if we ever do.

Implement get_cyclecount(9) for RISC-V— https://svnweb.freebsd.org/changeset/base/340400

Add the missing implementation for get_cyclecount(9) on RISC-V by reading the cycle CSR.

Enable non-executable stacks by default for RISC-V— https://svnweb.freebsd.org/changeset/base/340231

Enable use of a global shared page for RISC-V— https://svnweb.freebsd.org/changeset/base/340228

machine/vmparam.h already defines the SHAREDPAGE constant. This change just enables it for ELF executables. The only use of the shared page currently is to hold the signal trampoline.

Add new rc keywords: enable, disable, delete— https://svnweb.freebsd.org/changeset/base/339971

This adds new keywords to rc/service to enable/disable a service's rc.conf(5) variable and "delete" to remove the variable.

When the "service_delete_empty" variable in rc.conf(5) is set to "YES" (default is "NO"), an rc.conf.d file (in /etc/ or /usr/local/etc) is deleted if empty after modification using "service $foo delete".

libcasper: introduce cap_fileargs service— https://svnweb.freebsd.org/changeset/base/340373

cap_fileargs is a Casper service that helps to sandbox applications that need access to the filesystem namespace. The main purpose of the service is to makeeasy to capsicumize applications that works on multiple files passed in argv.

Sandbox head using capsicum— https://svnweb.freebsd.org/changeset/base/340376

Sandbox wc using capsicum— https://svnweb.freebsd.org/changeset/base/340374

Add load balancer program for netmap— https://svnweb.freebsd.org/changeset/base/340279

Add the lb program, which is able to load-balance input traffic received from a netmap port over M groups, with N netmap pipes in each group. Each received packet is forwarded to one of the pipes chosen from each group (using an L3/L4 connection-consistent hash function).

Allow configuration of several ipsec interfaces with the same tunnel endpoints — https://svnweb.freebsd.org/changeset/base/340477

This can be used to configure several IPsec tunnels between two hosts with different security associations.

Add support for non-ACPI battery method batteries— https://svnweb.freebsd.org/changeset/base/340832

Remove the requirement that a device be an ACPI method battery to be supported as a battery. Require now that the device be in the battery devclass and implement the get_status and get_info functions. This allows batteries that are not ACPI method batteries to be supported.

Permit local kernel modules to be built as part of a kernel build— https://svnweb.freebsd.org/changeset/base/339901

Add support for "local" modules. By default, these modules are located in LOCALBASE/sys/modules (where LOCALBASE defaults to /usr/local). Individual modules can be built along with a kernel by defining LOCAL_MODULES to the list of modules. Each is assumed to be a subdirectory containing a valid Makefile. If LOCAL_MODULES is not specified, all of the modules present in LOCALBASE/sys/modules are built and installed along with the kernel.

This means that a port that installs a kernel module can choose to install its source along with a suitable Makefile to /usr/local/sys/modules/<foo>. Future kernel builds will then include that kernel module using the kernel configuration's opt_*.h headers and install it into /boot/kernel along with other kernel-specific modules.

Add minimal support for active AUX port multiplexers— https://svnweb.freebsd.org/changeset/base/340913

Active PS/2 multiplexing is a method for attaching up to four PS/2 pointing devices to a computer. Enabling of multiplexed mode allows commands to be directed to individual devices using routing prefixes.

Multiplexed mode reports input with each byte tagged to identify its source. This method differs from one currently supported by psm(4) where so-called guest device (trackpoint) is attached to special interface located on the host device (touchpad) and later performs guest protocol conversion to special encapsulation packet format.

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.