What: /sys/kernel/debug/kmemtrace/
Date: July 2008
Contact: Eduard - Gabriel Munteanu <eduard.munteanu@linux360.ro>
Description:

In kmemtrace-enabled kernels, the following files are created:

/sys/kernel/debug/kmemtrace/

cpu<n>	(0400)	Per-CPU tracing data, see below. (binary)
total_overruns	(0400)	Total number of bytes which were dropped from cpu<n> files because of full buffer condition, non-binary. (text)
abi_version	(0400)	Kernel's kmemtrace ABI version. (text)

Each per-CPU file should be read according to the relay interface. That is, the reader should set affinity to that specific CPU and, as currently done by the userspace application (though there are other methods), use poll() with an infinite timeout before every read(). Otherwise, erroneous data may be read. The binary data has the following _core_ format:

Event ID	(1 byte)	Unsigned integer, one of: 0 - represents an allocation (KMEMTRACE_EVENT_ALLOC) 1 - represents a freeing of previously allocated memory (KMEMTRACE_EVENT_FREE)
Type ID	(1 byte)	Unsigned integer, one of: 0 - this is a kmalloc() / kfree() 1 - this is a kmem_cache_alloc() / kmem_cache_free() 2 - this is a __get_free_pages() et al.
Event size	(2 bytes)	Unsigned integer representing the size of this event. Used to extend kmemtrace. Discard the bytes you don't know about.
Sequence number	(4 bytes)	Signed integer used to reorder data logged on SMP machines. Wraparound must be taken into account, although it is unlikely.
Caller address	(8 bytes)	Return address to the caller.
Pointer to mem	(8 bytes)	Pointer to target memory area. Can be NULL, but not all such calls might be recorded.

In case of KMEMTRACE_EVENT_ALLOC events, the next fields follow:

Requested bytes	(8 bytes)	Total number of requested bytes, unsigned, must not be zero.
Allocated bytes	(8 bytes)	Total number of actually allocated bytes, unsigned, must not be lower than requested bytes.
Requested flags	(4 bytes)	GFP flags supplied by the caller.
Target CPU	(4 bytes)	Signed integer, valid for event id 1. If equal to -1, target CPU is the same as origin CPU, but the reverse might not be true.

The data is made available in the same endianness the machine has.

Other event ids and type ids may be defined and added. Other fields may be added by increasing event size, but see below for details.
Every modification to the ABI, including new id definitions, are followed by bumping the ABI version by one.

Adding new data to the packet (features) is done at the end of the mandatory data:

Feature size	(2 byte)
Feature ID	(1 byte)
Feature data	(Feature size - 3 bytes)

Users:

kmemtrace-user - git://repo.or.cz/kmemtrace-user.git