

Re: [openib-general] Re: [PATCH][RFC][0/4] InfiniBand userspace verbs implementation

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2005-04/6628.html>

From: Roland Dreier (roland_at_topspin.com)

Date: 04/27/05

To: Andrew Morton <akpm@osdl.org>

Date: Tue, 26 Apr 2005 19:13:24 -0700

Andrew> The kernel can simply register and unregister ranges for
Andrew> RDMA. So effectively a particular page is in either the
Andrew> registered or unregistered state. Kernel accounting
Andrew> counts the number of registered pages and compares this
Andrew> with rlimits.

Andrew> On top of all that, your userspace library needs to keep
Andrew> track of when pages should really be registered and
Andrew> unregistered with the kernel. Using overlap logic and
Andrew> per-page refcounting or whatever.

This is OK as long as userspace is trusted. However I don't see how this works when we don't trust userspace. The problem is that for an RDMA device (IB HCA or iWARP RNIC), a process can create many memory regions, each of which a separate virtual to physical translation map. For example, an app can do:

- a) register 0x0000 through 0xffff and get memory handle 1
- b) register 0x0000 through 0xffff and get memory handle 2
- c) use memory handle 1 for communication with remote app A
- d) use memory handle 2 for communication with remote app B

Even though memory handles 1 and 2 both refer to exactly the same memory, they may have different lifetimes, might be attached to different connections, and so on.

Clearly the memory at 0x0000 must stay pinned as long as the RDMA device thinks either memory handle 1 or memory handle 2 is valid. Furthermore, the kernel must be the one keeping track of how many regions refer to a given page because we can't allow userspace to be able to tell a device to go DMA to memory it doesn't own any more.

Creation and destruction of these memory handles will always go through the kernel driver, so this isn't so bad. And `get_user_pages()` is almost exactly what we need: it stacks perfectly, since it operates

Linux-Kernel: Re: [openib-general] Re: [PATCH][RFC][0/4] InfiniBand userspace verbs implementation

on the page_count rather than just setting a bit in vm_flags. The main problem is that it doesn't check against RLIMIT_MEMLOCK.

The most reasonable thing to do would seem to be having the IB kernel memory region code update current->mm->locked_vm and check it against RLIMIT_MEMLOCK. I guess it would be good to figure out an appropriate abstraction to export rather than monkeying with current->mm directly. We could also put this directly in get_user_pages(), but I'd be worried about messing with current users.

I just don't see a way to make VM_KERNEL_LOCKED work.

It would also be nice to have a way for apps to set VM_DONTCOPY appropriately. Christoph's suggestion of extending mmap() and mprotect() with PROT_DONTCOPY seems good to me, especially since it means we don't have to export do_mlock() functionality to modules.

- R.

-

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@vger.kernel.org

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>