Userspace verbs access

The ib_uverbs module, built by enabling CONFIG_INFINIBAND_USER_VERBS, enables direct userspace access to IB hardware via "verbs," as described in chapter 11 of the InfiniBand Architecture Specification.

To use the verbs, the libibverbs library, available from https://github.com/linux-rdma/rdma-core, is required. libibverbs contains a device-independent API for using the ib_uverbs interface. libibverbs also requires appropriate device-dependent kernel and userspace driver for your InfiniBand hardware. For example, to use a Mellanox HCA, you will need the ib_mthca kernel module and the libmthca userspace driver be installed.

User-kernel communication

Userspace communicates with the kernel for slow path, resource management operations via the /dev/infiniband/uverbsN character devices. Fast path operations are typically performed by writing directly to hardware registers mmap()ed into userspace, with no system call or context switch into the kernel.

Commands are sent to the kernel via write()s on these device files. The ABI is defined in drivers/infiniband/include/ib_user_verbs.h. The structs for commands that require a response from the kernel contain a 64-bit field used to pass a pointer to an output buffer. Status is returned to userspace as the return value of the write() system call.

Resource management

Since creation and destruction of all IB resources is done by commands passed through a file descriptor, the kernel can keep track of which resources are attached to a given userspace context. The ib_uverbs module maintains idr tables that are used to translate between kernel pointers and opaque userspace handles, so that kernel pointers are never exposed to userspace and userspace cannot trick the kernel into following a bogus pointer.

This also allows the kernel to clean up when a process exits and prevent one process from touching another process's resources.

Memory pinning

Direct userspace I/O requires that memory regions that are potential I/O targets be kept resident at the same physical address. The ib_uverbs module manages pinning and unpinning memory regions via get_user_pages() and put_page() calls. It also accounts for the amount of memory pinned in the process's pinned_vm, and checks that unprivileged processes do not exceed their RLIMIT_MEMLOCK limit.

Pages that are pinned multiple times are counted each time they are pinned, so the value of pinned_vm may be an overestimate of the number of pages pinned by a process.

/dev files

To create the appropriate character device files automatically with udev, a rule like:

```
KERNEL=="uverbs*", NAME="infiniband/%k"
```

can be used. This will create device nodes named:

```
/dev/infiniband/uverbs0
```

and so on. Since the InfiniBand userspace verbs should be safe for use by non-privileged processes, it may be useful to add an appropriate MODE or GROUP to the udev rule.