
Subject: [PATCH 12/16] Reference counting of pid namespaces by pids

Posted by [Pavel Emelianov](#) on Fri, 06 Jul 2007 08:10:06 GMT

[View Forum Message](#) <> [Reply to Message](#)

Getting and putting the pid namespace in `alloc_pid()` and `free_pid()` is too slow. Instead this I get/put the namespace by the pidmaps. When the pidmap allocates its first pid the namespace is get, when the pidmap becomes empty - the namespace is put.

Although pids may live longer than their "fingerprints" in the pidmaps, this is ok to release the namespace with not yet freed struct pids, as this pid is not alive and no routines will use it.

Signed-off-by: Pavel Emelianov <xemul@openvz.org>

pid.c | 19 ++++++++-----
1 files changed, 15 insertions(+), 4 deletions(-)

diff -upr linux-2.6.22-rc4-mm2.orig/kernel/pid.c linux-2.6.22-rc4-mm2-2/kernel/pid.c
--- linux-2.6.22-rc4-mm2.orig/kernel/pid.c 2007-06-14 12:14:29.000000000 +0400
+++ linux-2.6.22-rc4-mm2-2/kernel/pid.c 2007-07-04 19:00:38.000000000 +0400
@@ -89,16 +95,23 @@ static __cacheline_aligned_in_smp DEFIN

```
static fastcall void free_pidmap(struct pid_namespace *pid_ns, int pid)
{
- struct pidmap *map = pid_ns->pidmap + pid / BITS_PER_PAGE;
  int offset = pid & BITS_PER_PAGE_MASK;
+ int map_id = pid / BITS_PER_PAGE;
+ struct pidmap *map = pid_ns->pidmap + map_id;
+ int free_pids;

  clear_bit(offset, map->page);
- atomic_inc(&map->nr_free);
+ free_pids = atomic_inc_return(&map->nr_free);
+
+ if (map_id == 0)
+   free_pids++;
+ if (free_pids == BITS_PER_PAGE)
+   put_pid_ns(pid_ns);
}
```

```
static int alloc_pidmap(struct pid_namespace *pid_ns)
{
- int i, offset, max_scan, pid, last = pid_ns->last_pid;
```

```

+ int i, offset, max_scan, pid, last = pid_ns->last_pid, free_pids;
  struct pidmap *map;

  pid = last + 1;
@@ -126,7 +139,11 @@ static int alloc_pidmap(struct pid_names
  if (likely(atomic_read(&map->nr_free))) {
    do {
      if (!test_and_set_bit(offset, map->page)) {
-       atomic_dec(&map->nr_free);
+       free_pids = atomic_dec_return(
+         &map->nr_free);
+       if (free_pids == BITS_PER_PAGE - 1)
+         get_pid_ns(pid_ns);
+
        pid_ns->last_pid = pid;
        return pid;
      }

```

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>
