
Subject: Re: [RFC PATCH] posix timers: allocate timer id per task

Posted by [Eric Dumazet](#) on Mon, 15 Oct 2012 16:34:43 GMT

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

On Mon, 2012-10-15 at 20:17 +0400, Stanislav Kinsbursky wrote:

- > This patch is required CRIU project (www.criu.org).
- > To migrate processes with posix timers we have to make sure, that we can
- > restore posix timer with proper id.
- > Currently, this is not true, because timer ids are allocated globally.
- > So, this is precursor patch and it's purpose is make posix timer id to be
- > allocated per task.
- >
- > Patch replaces global idr with global hash table for posix timers and
- > makes timer ids unique not globally, but per task. Next free timer id is type
- > of integer and stored on signal struct (posix_timer_id). If free timer id
- > reaches negative value on timer creation, it will be dropped to zero and
- > -EAGAIN will be returned to user.
- > Hash table is size of page (4KB).
- > Key is constructed as follows:
- > key = hash_ptr(current->signal) ^ hash_32(posix_timer_id);
- >
- > Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

Hmm, it seems you removed idr, rcu friendly, and reinstated a fixed size hash table, protected by a `_single_` spinlock ? Oh well.

Please take a look at commit 8af088710d1e, and make sure you fix your problem and keep performance as good as before.
