
Subject: Re: [PATCH v5] posix timers: allocate timer id per process

Posted by [Eric Dumazet](#) on Tue, 23 Oct 2012 07:52:03 GMT

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

On Tue, 2012-10-23 at 11:40 +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 process.
>
> Patch replaces global idr with global hash table for posix timers and
> makes timer ids unique not globally, but per process. 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 has 512 slots.
> Key is constructed as follows:
> key = hash_32(hash_32(current->signal) ^ posix_timer_id));
>
> Note: with this patch, id, returned to user, is not the minimal free
> anymore. It means, that id, returned to user space in loop, listed below, will
> be increasing on each iteration till INT_MAX and then dropped to zero:
>
> while(1) {
>   id = timer_create(...);
>   timer_delete(id);
> }
>
> Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>
>
> ---
```

SGTM

Signed-off-by: Eric Dumazet <edumazet@google.com>

Thanks !
