Subject: Re: Checkpoint/restart (was Re: [PATCH 0/4] - v2 - Object creation with a specified id)
Posted by Oren Laadan on Thu, 17 Jul 2008 23:19:51 GMT
View Forum Message <> Reply to Message

Dave Hansen wrote:
> On Thu, 2008-07-10 at 12:21 -0700, Eric W. Biederman wrote:
>>> Are we talking about the VMA itself, or the memory backing the VMA?
>> The memory backing the VMA. We need to store the page protections
>> that the memory was mapped with as well now that you point it out. A
>> VMA is not user space visible, which is why we can arbitrarily split
>> and merge VMAs.
>
> It is visible with /proc/$pid/{maps,smaps,numamaps?}. That's the only
> efficient way I know of from userspace to figure out where userspace's
> memory is and if it *is* file backed, and what the permissions are.
>
> We also can't truly split them arbitrarily because the memory cost of
> the VMAs themselves might become too heavy (the remap_file_pages()
> problem).
>
> It gets trickier when things are also private mappings in addition to
> being in a file-backed VMA. We *do* need to checkpoint those, but only
> the pages to which there was a write.
>
> There's also the problem of restoring things read-only, but VM_MAYWRITE.
> If there's a high level of (non-COW'd) sharing of these anonymous areas,
> we may not be able to even restore the set of processes unless we can
> replicate the sharing. We might run out of memory if the sharing isn't
> replicated.
>
> Memory is fun! :)

Heh .. and it can get much worse. By all means, the functions that analyze and save the VMAs during checkpoint and later reconstruct them during restart are the most complicated logic. The good news, however, is that it works :)

Oren.

>
> -- Dave
>

_____