Subject: Re: controlling mmap()'d vs read/write() pages
Posted by Balbir Singh on Wed, 28 Mar 2007 09:18:21 GMT
View Forum Message <> Reply to Message

Herbert Poetzl wrote:

```
>> To me, one of the keys of Linux's "global optimizations" is being able
>> to use any memory globally for its most effective purpose, globally
>> (please ignore highmem :).  Let's say I have a 1GB container on a
>> machine that is at least 100% committed.  I mmap() a 1GB file and touch
>> the entire thing (I never touch it again).  I then go open another 1GB
>> file and r/w to it until the end of time.  I'm at or below my RSS limit,
>> but that 1GB of RAM could surely be better used for the second file.
>> How do we do this if we only account for a user's RSS?  Does this fit
>> into Alan's unfair bucket? ;)
>
> what's the difference to a normal Linux system here?
> when low on memory, the system will reclaim pages, and
> guess what pages will be reclaimed first ...
>
```

But would it not bias application writers towards using read()/write() calls over mmap()? They know that their calls are likely to be faster when the application is run in a container. Without page cache control we'll end up creating an asymmetrical container, where certain usage is charged and some usage is not.

Also, please note that when a page is unmapped and moved to swap cache; the swap cache uses the page cache. Without page cache control, we could end up with too many pages moving over to the swap cache and still occupying memory, while the original intension was to avoid this scenario.

--
Warm Regards,
Balbir Singh
Linux Technology Center
IBM, ISTL

____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers