
Subject: Re: [RFC][PATCH 3/7] Data structures changes for RSS accounting
Posted by [ebiederm](#) on Sun, 11 Mar 2007 19:13:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

Pavel Emelianov <xemul@sw.ru> writes:

- > Adds needed pointers to mm_struct and page struct,
- > places hooks to core code for mm_struct initialization
- > and hooks in container_init_early() to preinitialize
- > RSS accounting subsystem.

An extra pointer in struct page is unlikely to fly.
Both because it increases the size of a size critical structure,
and because conceptually it is ridiculous.

If you are limiting the RSS size you are counting the number of pages in
the page tables. You don't care about the page itself.

With the rmap code it is relatively straight forward to see if this is
the first time a page has been added to a page table in your rss
group, or if this is the last reference to a particular page in your
rss group. The counters should only increment the first time a
particular page is added to your rss group. The counters should only
decrement when it is the last reference in your rss subsystem.

This allow important little cases like glibc to be properly accounted
for. One of the key features of a rss limit is that the kernel can
still keep pages that you need in-core, that are accessible with just
a minor fault. Directly owning pages works directly against that
principle.

```
> diff -upr linux-2.6.20.orig/include/linux/mm_types.h
> linux-2.6.20-0/include/linux/mm_types.h
> --- linux-2.6.20.orig/include/linux/mm_types.h 2007-02-04 21:44:54.000000000
> +0300
> +++ linux-2.6.20-0/include/linux/mm_types.h 2007-03-06 13:33:28.000000000 +0300
> @@ -62,6 +62,9 @@ struct page {
>   void *virtual; /* Kernel virtual address (NULL if
>    not kmapped, ie. highmem) */
> #endif /* WANT_PAGE_VIRTUAL */
> +#ifdef CONFIG_RSS_CONTAINER
> + struct page_container *rss_container;
> +#endif
> };
>
> #endif /* _LINUX_MM_TYPES_H */
```

Eric

Containers mailing list

Containers@lists.osdl.org

<https://lists.osdl.org/mailman/listinfo/containers>
