
Subject: [RFC][PATCH 3/7] Data structures changes for RSS accounting
Posted by [xemul](#) on Tue, 06 Mar 2007 14:55:46 GMT

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

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.

```
diff -upr linux-2.6.20.orig/include/linux/mm.h linux-2.6.20-0/include/linux/mm.h
--- linux-2.6.20.orig/include/linux/mm.h 2007-02-04 21:44:54.000000000 +0300
+++ linux-2.6.20-0/include/linux/mm.h 2007-03-06 13:33:28.000000000 +0300
@@ -220,6 +220,12 @@ struct vm_operations_struct {
 struct mmu_gather;
 struct inode;
```

```
+#ifdef CONFIG_RSS_CONTAINER
+#define page_container(page) (page->rss_container)
+#else
+#define page_container(page) (NULL)
+#endif
+
+#define page_private(page) ((page)->private)
+#define set_page_private(page, v) ((page)->private = (v))
```

```
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 */
diff -upr linux-2.6.20.orig/include/linux/sched.h linux-2.6.20-0/include/linux/sched.h
--- linux-2.6.20.orig/include/linux/sched.h 2007-03-06 13:33:28.000000000 +0300
+++ linux-2.6.20-0/include/linux/sched.h 2007-03-06 13:33:28.000000000 +0300
@@ -373,6 +373,9 @@ struct mm_struct {
 /* aio bits */
 rwlock_t ioctx_list_lock;
 struct kiocx *ioctx_list;
+#ifdef CONFIG_RSS_CONTAINER
+ struct rss_container *rss_container;
+#endif
```

```

};

struct sighand_struct {
diff -upr linux-2.6.20.orig/kernel/fork.c linux-2.6.20-0/kernel/fork.c
--- linux-2.6.20.orig/kernel/fork.c 2007-03-06 13:33:28.000000000 +0300
+++ linux-2.6.20-0/kernel/fork.c 2007-03-06 13:33:28.000000000 +0300
@@ -57,6 +57,8 @@
#include <asm/cacheflush.h>
#include <asm/tlbflush.h>

+#include <linux/rss_container.h>
+
+/*
+ * Protected counters by write_lock_irq(&tasklist_lock)
+ */
@@ -325,7 +328,7 @@ static inline void mm_free_pgd(struct mm

#include <linux/init_task.h>

-static struct mm_struct * mm_init(struct mm_struct * mm)
+static struct mm_struct * mm_init(struct mm_struct *mm, struct task_struct *tsk)
{
atomic_set(&mm->mm_users, 1);
atomic_set(&mm->mm_count, 1);
@@ -341,10 +344,18 @@ static struct mm_struct * mm_init(struct
mm->free_area_cache = TASK_UNMAPPED_BASE;
mm->cached_hole_size = ~0UL;

- if (likely(!mm_alloc_pgd(mm))) {
- mm->def_flags = 0;
- return mm;
- }
+ if (unlikely(mm_init_container(mm, tsk)))
+ goto out_cont;
+
+ if (unlikely(mm_alloc_pgd(mm)))
+ goto out_pgd;
+
+ mm->def_flags = 0;
+ return mm;
+
+out_pgd:
+ mm_free_container(mm);
+out_cont:
free_mm(mm);
return NULL;
}
@@ -359,7 +370,7 @@ struct mm_struct * mm_alloc(void)

```

```

    mm = allocate_mm();
    if (mm) {
        memset(mm, 0, sizeof(*mm));
- mm = mm_init(mm);
+ mm = mm_init(mm, current);
    }
    return mm;
}
@@ -373,6 +384,7 @@ void fastcall __mmdrop(struct mm_struct
{
    BUG_ON(mm == &init_mm);
    mm_free_pgd(mm);
+ mm_free_container(mm);
    destroy_context(mm);
    free_mm(mm);
}
@@ -493,7 +505,7 @@ static struct mm_struct *dup_mm(struct t
    mm->token_priority = 0;
    mm->last_interval = 0;

- if (!mm_init(mm))
+ if (!mm_init(mm, tsk))
    goto fail_nomem;

    if (init_new_context(tsk, mm))
@@ -520,6 +532,7 @@ fail_nocontext:
    * because it calls destroy_context()
    */
    mm_free_pgd(mm);
+ mm_free_container(mm);
    free_mm(mm);
    return NULL;
}
diff -upr linux-2.6.20.orig/kernel/container.c linux-2.6.20-0/kernel/container.c
--- linux-2.6.20.orig/kernel/container.c 2007-03-06 13:33:28.000000000 +0300
+++ linux-2.6.20-0/kernel/container.c 2007-03-06 13:35:48.000000000 +0300
@@ -60,6 +60,8 @@
#include <asm/atomic.h>
#include <linux/mutex.h>

+#include <linux/rss_container.h>
+
#define CONTAINER_SUPER_MAGIC 0x27e0eb

static struct container_subsys *subsys[CONFIG_MAX_CONTAINER_SUBSYS];
@@ -1721,6 +1725,8 @@ int __init container_init_early(void)
}
init_task.containers = &init_container_group;

```

```
+ container_rss_init_early();  
+  
  return 0;  
}
```
