Subject: + attach_pid-with-struct-pid-parameter.patch added to -mm tree
Posted by Andrew Morton on Thu, 11 Jan 2007 21:13:09 GMT
View Forum Message <> Reply to Message

The patch titled
     attach_pid() with struct pid parameter
has been added to the -mm tree. Its filename is
     attach_pid-with-struct-pid-parameter.patch

*** Remember to use Documentation/SubmitChecklist when testing your code ***

See http://www.zip.com.au/~akpm/linux/patches/stuff/added-to-mm.txt to find
out what to do about this

Subject: attach_pid() with struct pid parameter
From: Sukadev Bhattiprolu <sukadev@us.ibm.com>

Implement a new version of attach_pid() with a struct pid parameter and wrap
find_attach_pid() around it. attach_pid() would also be used in subsequent
container patches.

Signed-off-by: Sukadev Bhattiprolu <sukadev@us.ibm.com>
Cc: Cedric Le Goater <clg@fr.ibm.com>
Cc: Dave Hansen <haveblue@us.ibm.com>
Cc: Serge Hallyn <serue@us.ibm.com>
Cc: <containers@lists.osdl.org>
Signed-off-by: Andrew Morton <akpm@osdl.org>
---

 include/linux/pid.h | 28 ++++++++++++++++++----------
 kernel/pid.c | 7 +++----
 2 files changed, 20 insertions(+), 15 deletions(-)

diff -puN include/linux/pid.h~attach_pid-with-struct-pid-parameter include/linux/pid.h
--- a/include/linux/pid.h~attach_pid-with-struct-pid-parameter
+++ a/include/linux/pid.h
@@ -72,17 +72,6 @@ extern struct task_struct *FASTCALL(get_
 extern struct pid *get_task_pid(struct task_struct *task, enum pid_type type);

 /*
- * find_attach_pid() and detach_pid() must be called with the tasklist_lock
- * write-held.
- */
-extern int FASTCALL(find_attach_pid(struct task_struct *task,
-    enum pid_type type, int nr));
-
-extern void FASTCALL(detach_pid(struct task_struct *task, enum pid_type));

```
-extern void FASTCALL(transfer_pid(struct task_struct *old,
-      struct task_struct *new, enum pid_type));
-
-/*
  * look up a PID in the hash table. Must be called with the tasklist_lock
  * or rcu_read_lock() held.
  */
@@ -94,6 +83,23 @@ extern struct pid *FASTCALL(find_pid(int
 extern struct pid *find_get_pid(int nr);
 extern struct pid *find_ge_pid(int nr);

+/*
+ * attach_pid(), find_attach_pid() and detach_pid() must be called with the
+ * tasklist_lock write-held.
+ */
+extern int FASTCALL(attach_pid(struct task_struct *task, enum pid_type type,
+    struct pid *pid));
+
+static inline int find_attach_pid(struct task_struct *task, enum pid_type type,
+    int nr)
+{
+ return attach_pid(task, type, find_pid(nr));
+}
+
+extern void FASTCALL(detach_pid(struct task_struct *task, enum pid_type));
+extern void FASTCALL(transfer_pid(struct task_struct *old,
+      struct task_struct *new, enum pid_type));
+
 extern struct pid *alloc_pid(void);
 extern void FASTCALL(free_pid(struct pid *pid));

diff -puN kernel/pid.c~attach_pid-with-struct-pid-parameter kernel/pid.c
--- a/kernel/pid.c~attach_pid-with-struct-pid-parameter
+++ a/kernel/pid.c
@@ -247,14 +247,13 @@ struct pid * fastcall find_pid(int nr)
 }
 EXPORT_SYMBOL_GPL(find_pid);

-int fastcall find_attach_pid(struct task_struct *task, enum pid_type type,
-    int nr)
+int fastcall attach_pid(struct task_struct *task, enum pid_type type,
+    struct pid *pid)
 {
  struct pid_link *link;
- struct pid *pid;

  link = &task->pids[type];
- link->pid = pid = find_pid(nr);
```

```
+ link->pid = pid;
  hlist_add_head_rcu(&link->node, &pid->tasks[type]);

  return 0;
_
```

Patches currently in -mm which might be from sukadev@us.ibm.com are

rename-attach_pid-to-find_attach_pid.patch
attach_pid-with-struct-pid-parameter.patch

_____
Containers mailing list
Containers@lists.osdl.org
https://lists.osdl.org/mailman/listinfo/containers