Subject: [patch 40/68] statically initialize struct pid for swapper
Posted by akpm on Fri, 11 May 2007 05:23:00 GMT
View Forum Message <> Reply to Message

From: Sukadev Bhattiprolu <sukadev@us.ibm.com>

Statically initialize a struct pid for the swapper process (pid_t == 0) and
attach it to init_task. This is needed so task_pid(), task_pgrp() and
task_session() interfaces work on the swapper process also.

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: Eric Biederman <ebiederm@xmission.com>
Cc: Herbert Poetzl <herbert@13thfloor.at>
Cc: <containers@lists.osdl.org>
Acked-by: Eric W. Biederman <ebiederm@xmission.com>
Signed-off-by: Andrew Morton <akpm@linux-foundation.org>

```
 include/linux/init_task.h |   27 ++++++++++++++++++++++++++
 include/linux/pid.h       |    2 ++
 kernel/pid.c              |    2 ++
 3 files changed, 31 insertions(+)

diff -puN include/linux/init_task.h~statically-initialize-struct-pid-for-swapper include/linux/init_task.h
--- a/include/linux/init_task.h~statically-initialize-struct-pid-for-swapper
+++ a/include/linux/init_task.h
@@ -88,6 +88,28 @@ extern struct nsproxy init_nsproxy;

 extern struct group_info init_groups;

+#define INIT_STRUCT_PID {      \
+ .count   = ATOMIC_INIT(1),    \
+ .nr  = 0,       \
+ /* Don't put this struct pid in pid_hash */   \
+ .pid_chain = { .next = NULL, .pprev = NULL },  \
+ .tasks  = {      \
+  { .first = &init_task.pids[PIDTYPE_PID].node },  \
+  { .first = &init_task.pids[PIDTYPE_PGID].node }, \
+  { .first = &init_task.pids[PIDTYPE_SID].node },  \
+ },       \
+ .rcu  = RCU_HEAD_INIT,    \
+}
+
+#define INIT_PID_LINK(type)      \
+{       \
```

```
+ .node = {      \
+  .next = NULL,     \
+  .pprev = &init_struct_pid.tasks[type].first, \
+ },       \
+ .pid = &init_struct_pid,    \
+}
+
 /*
  *  INIT_TASK is used to set up the first task table, touch at
  * your own risk!. Base=0, limit=0x1fffff (=2MB)
@@ -139,6 +161,11 @@ extern struct group_info init_groups;
  .cpu_timers = INIT_CPU_TIMERS(tsk.cpu_timers),  \
  .fs_excl = ATOMIC_INIT(0),    \
  .pi_lock = __SPIN_LOCK_UNLOCKED(tsk.pi_lock),  \
+ .pids = {       \
+  [PIDTYPE_PID]  = INIT_PID_LINK(PIDTYPE_PID),  \
+  [PIDTYPE_PGID] = INIT_PID_LINK(PIDTYPE_PGID),  \
+  [PIDTYPE_SID]  = INIT_PID_LINK(PIDTYPE_SID),  \
+ },       \
  INIT_TRACE_IRQFLAGS      \
  INIT_LOCKDEP       \
 }
diff -puN include/linux/pid.h~statically-initialize-struct-pid-for-swapper include/linux/pid.h
--- a/include/linux/pid.h~statically-initialize-struct-pid-for-swapper
+++ a/include/linux/pid.h
@@ -51,6 +51,8 @@ struct pid
  struct rcu_head rcu;
 };

+extern struct pid init_struct_pid;
+
 struct pid_link
 {
  struct hlist_node node;
diff -puN kernel/pid.c~statically-initialize-struct-pid-for-swapper kernel/pid.c
--- a/kernel/pid.c~statically-initialize-struct-pid-for-swapper
+++ a/kernel/pid.c
@@ -27,11 +27,13 @@
 #include <linux/bootmem.h>
 #include <linux/hash.h>
 #include <linux/pid_namespace.h>
+#include <linux/init_task.h>

 #define pid_hashfn(nr) hash_long((unsigned long)nr, pidhash_shift)
 static struct hlist_head *pid_hash;
 static int pidhash_shift;
 static struct kmem_cache *pid_cachep;
+struct pid init_struct_pid = INIT_STRUCT_PID;
```

int pid_max = PID_MAX_DEFAULT;

–

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