
Subject: [PATCH] pidns: Place under CONFIG_EXPERIMENTAL
Posted by [ebiederm](#) on Fri, 26 Oct 2007 17:40:59 GMT
[View Forum Message](#) <> [Reply to Message](#)

I finally found a chance to review the pid namespace implementation in detail and currently it is much too easy to find issues where the kernel does the wrong thing outside of the initial pid namespace. At the same time the pid namespace code we have does appear to be a good base to build on.

Therefore until the dust settles and we are certain we have the ABI and the implementation as correct as humanly possible let's hide the availability of process ID namespaces behind CONFIG_EXPERIMENTAL.

Allowing users to avoid bugs, and removing a guarantee of bug compatibility. Allowing any issues that may be found to be fixed properly.

If CONFIG_PID_NS=N this patch will cause copy_pid_ns to unconditionally return -EINVAL removing the availability of multiple pid namespaces.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

```
init/Kconfig | 12 ++++++++
kernel/pid.c |  4 ++++
2 files changed, 16 insertions(+), 0 deletions(-)
```

```
diff --git a/init/Kconfig b/init/Kconfig
```

```
index 8b88d0b..72e37c0 100644
```

```
--- a/init/Kconfig
```

```
+++ b/init/Kconfig
```

```
@ @ -215,6 +215,18 @ @ config USER_NS
```

```
vservers, to use user namespaces to provide different
user info for different servers. If unsure, say N.
```

```
+config PID_NS
```

```
+ bool "PID Namespaces (EXPERIMENTAL)"
```

```
+ default n
```

```
+ depends on EXPERIMENTAL
```

```
+ help
```

```
+ Support process id namespaces. This allows having multiple
```

```
+ process with the same pid as long as they are in different
```

```
+ pid namespaces. This is a building block of containers.
```

```
+
```

```
+ Unless you want to work with an experimental feature
```

```
+ say N here.
```

```
+
config AUDIT
  bool "Auditing support"
  depends on NET
diff --git a/kernel/pid.c b/kernel/pid.c
index d1db36b..8a5637b 100644
--- a/kernel/pid.c
+++ b/kernel/pid.c
@@ -599,6 +599,10 @@ struct pid_namespace *copy_pid_ns(unsigned long flags, struct
pid_namespace *old
  if (flags & CLONE_THREAD)
    goto out_put;

+#ifndef CONFIG_PID_NS
+ goto out_put;
+#endif
+
  new_ns = create_pid_namespace(old_ns->level + 1);
  if (!IS_ERR(new_ns))
    new_ns->parent = get_pid_ns(old_ns);
--
1.5.3.rc6.17.g1911
```

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