
Subject: [PATCH] ncpfs: Use struct pid to track the userspace watchdog process.
Posted by [ebiederm](#) on Mon, 11 Dec 2006 13:22:26 GMT

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

This patch converts the tracking of the user space watchdog process from using a pid_t to use struct pid. This makes us safe from pid wrap around issues and prepares the way for the pid namespace.

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

```
---
fs/ncpfs/inode.c      | 11 ++++++-----
include/linux/ncp_mount.h | 2 +-
2 files changed, 7 insertions(+), 6 deletions(-)

diff --git a/fs/ncpfs/inode.c b/fs/ncpfs/inode.c
index 47462ac..861d950 100644
--- a/fs/ncpfs/inode.c
+++ b/fs/ncpfs/inode.c
@@ -331,7 +331,7 @@ static int ncp_parse_options(struct ncp_mount_data_kernel *data, char
*options)
    data->flags = 0;
    data->int_flags = 0;
    data->mounted_uid = 0;
-   data->wdog_pid = -1;
+   data->wdog_pid = NULL;
    data->ncp_fd = ~0;
    data->time_out = 10;
    data->retry_count = 20;
@@ -371,7 +371,7 @@ static int ncp_parse_options(struct ncp_mount_data_kernel *data, char
*options)
    data->flags = optint;
    break;
    case 'w':
-   data->wdog_pid = optint;
+   data->wdog_pid = find_get_pid(optint);
    break;
    case 'n':
    data->ncp_fd = optint;
@@ -425,7 +425,7 @@ static int ncp_fill_super(struct super_block *sb, void *raw_data, int silent)
    data->flags = md->flags;
    data->int_flags = NCP_IMOUNT_LOGGEDIN_POSSIBLE;
    data->mounted_uid = md->mounted_uid;
-   data->wdog_pid = md->wdog_pid;
+   data->wdog_pid = find_get_pid(md->wdog_pid);
    data->ncp_fd = md->ncp_fd;
    data->time_out = md->time_out;
    data->retry_count = md->retry_count;
@@ -445,7 +445,7 @@ static int ncp_fill_super(struct super_block *sb, void *raw_data, int silent)
```

```

    data.flags = md->flags;
    data.int_flags = 0;
    data.mounted_uid = md->mounted_uid;
-   data.wdog_pid = md->wdog_pid;
+   data.wdog_pid = find_get_pid(md->wdog_pid);
    data.ncp_fd = md->ncp_fd;
    data.time_out = md->time_out;
    data.retry_count = md->retry_count;
@@ -711,7 +711,8 @@ static void ncp_put_super(struct super_block *sb)
    if (server->info_filp)
        fput(server->info_filp);
    fput(server->ncp_filp);
-   kill_proc(server->m.wdog_pid, SIGTERM, 1);
+   kill_pid(server->m.wdog_pid, SIGTERM, 1);
+   put_pid(server->m.wdog_pid);

    kfree(server->priv.data);
    kfree(server->auth.object_name);
diff --git a/include/linux/ncp_mount.h b/include/linux/ncp_mount.h
index f46bddc..a2b549e 100644
--- a/include/linux/ncp_mount.h
+++ b/include/linux/ncp_mount.h
@@ -75,7 +75,7 @@ struct ncp_mount_data_kernel {
    unsigned int  int_flags; /* internal flags */
    #define NCP_IMOUNT_LOGGEDIN_POSSIBLE 0x0001
    __kernel_uid32_t mounted_uid; /* Who may umount() this filesystem? */
-   __kernel_pid_t  wdog_pid; /* Who cares for our watchdog packets? */
+   struct pid      *wdog_pid; /* Who cares for our watchdog packets? */
    unsigned int    ncp_fd; /* The socket to the ncp port */
    unsigned int    time_out; /* How long should I wait after
        sending a NCP request? */
--
1.4.4.1.g278f

```

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