Subject: [PATCH] ncpfs: Ensure we free wdog_pid on parse_option or fill_inode failure.
Posted by ebiederm on Tue, 12 Dec 2006 08:20:01 GMT
View Forum Message <> Reply to Message

This took a little refactoring but now errors are handled cleanly. When
this code used pid_t values this wasn't necessary because you can't
leak a pid_t.

Thanks to Peter Vandrovec <vandrove@vc.cvut.cz> for spotting this.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>
---
fs/ncpfs/inode.c | 23 +++++++++++++++++++--------
1 files changed, 15 insertions(+), 8 deletions(-)

```
diff --git a/fs/ncpfs/inode.c b/fs/ncpfs/inode.c
index 861d950..67a90bf 100644
--- a/fs/ncpfs/inode.c
+++ b/fs/ncpfs/inode.c
@@ -327,6 +327,7 @@ static int ncp_parse_options(struct ncp_mount_data_kernel *data, char
*options)
  char *optarg;
  unsigned long optint;
  int version = 0;
+ int ret;

  data->flags = 0;
  data->int_flags = 0;
@@ -343,8 +344,9 @@ static int ncp_parse_options(struct ncp_mount_data_kernel *data, char
*options)
  data->mounted_vol[0] = 0;

  while ((optval = ncp_getopt("ncpfs", &options, ncp_opts, NULL, &optarg, &optint)) != 0) {
-  if (optval < 0)
-   return optval;
+  ret = optval;
+  if (ret < 0)
+   goto err;
   switch (optval) {
    case 'u':
     data->uid = optint;
@@ -380,18 +382,21 @@ static int ncp_parse_options(struct ncp_mount_data_kernel *data,
char *options)
     data->info_fd = optint;
     break;
    case 'v':
-    if (optint < NCP_MOUNT_VERSION_V4) {
```

```
-     return -ECHRNG;
-    }
-    if (optint > NCP_MOUNT_VERSION_V5) {
-     return -ECHRNG;
-    }
+    ret = -ECHRNG;
+    if (optint < NCP_MOUNT_VERSION_V4)
+     goto err;
+    if (optint > NCP_MOUNT_VERSION_V5)
+     goto err;
     version = optint;
     break;

   }
  }
  return 0;
+err:
+ put_pid(data->wdog_pid);
+ data->wdog_pid = NULL;
+ return ret;
 }

 static int ncp_fill_super(struct super_block *sb, void *raw_data, int silent)
@@ -409,6 +414,7 @@ static int ncp_fill_super(struct super_block *sb, void *raw_data, int silent)
 #endif
  struct ncp_entry_info finfo;

+ data.wdog_pid = NULL;
  server = kzalloc(sizeof(struct ncp_server), GFP_KERNEL);
  if (!server)
   return -ENOMEM;
@@ -679,6 +685,7 @@ out_fput:
   */
  fput(ncp_filp);
 out:
+ put_pid(data.wdog_pid);
  sb->s_fs_info = NULL;
  kfree(server);
  return error;
--
1.4.4.1.g278f
```