

---

Subject: [RFC][PATCH 2/3] Replace pid\_t in autofs4 with struct pid reference  
Posted by [Sukadev Bhattiprolu](#) on Thu, 25 Jan 2007 03:54:30 GMT

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

---

ChangeLog:

- Fix Eric Biederman's comments - Use find\_get\_pid() to hold a reference to oz\_pgrp and release while unmounting; separate out changes to autofs and autofs4.
- Also rollback my earlier change to autofs\_wait\_queue (pid and tgid in the wait queue are just used to write to a userspace daemon's pipe).

---

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

Subject: Replace pid\_t in autofs4 with struct pid reference.

Make autofs4 container-friendly by caching struct pid reference rather than pid\_t.

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: containers@lists.osdl.org

---

```
fs/autofs4/autofs_i.h | 6 +++---
fs/autofs4/inode.c   | 29 ++++++-----
fs/autofs4/root.c    | 3 +-
fs/autofs4/waitq.c   | 4 +-
4 files changed, 27 insertions(+), 15 deletions(-)
```

Index: lx26-20-rc4-mm1/fs/autofs4/autofs\_i.h

```
=====
--- lx26-20-rc4-mm1.orig/fs/autofs4/autofs_i.h 2007-01-24 18:39:28.172066416 -0800
```

```
+++ lx26-20-rc4-mm1/fs/autofs4/autofs_i.h 2007-01-24 18:41:56.260553544 -0800
```

```
@@ -35,7 +35,7 @@
```

```
/* #define DEBUG */
```

```
#ifdef DEBUG
```

```
~#define DPRINTK(fmt,args...) do { printk(KERN_DEBUG "pid %d: %s: " fmt "\n" , current->pid ,  
__FUNCTION__ , ##args); } while(0)
```

```
+#define DPRINTK(fmt,args...) do { printk(KERN_DEBUG "pid %d: %s: " fmt "\n" ,  
pid_nr(task_pid(current)), __FUNCTION__ , ##args); } while(0)
```

```
#else
```

```
#define DPRINTK(fmt,args...) do {} while(0)
```

```
#endif
```

```

@@ -96,7 +96,7 @@ struct autofs_sb_info {
    u32 magic;
    int pipefd;
    struct file *pipe;
- pid_t oz_pgrp;
+ struct pid *oz_pgrp;
    int catatonic;
    int version;
    int sub_version;
@@ -127,7 +127,7 @@ static inline struct autofs_info *autofs
    filesystem without "magic".) */

static inline int autofs4_oz_mode(struct autofs_sb_info *sbi) {
- return sbi->catatonic || process_group(current) == sbi->oz_pgrp;
+ return sbi->catatonic || task_pgrp(current) == sbi->oz_pgrp;
}

```

/\* Does a dentry have some pending activity? \*/

Index: lx26-20-rc4-mm1/fs/autofs4/inode.c

=====

--- lx26-20-rc4-mm1.orig/fs/autofs4/inode.c 2007-01-24 18:39:28.173066264 -0800

+++ lx26-20-rc4-mm1/fs/autofs4/inode.c 2007-01-24 18:44:56.311181688 -0800

```

@@ -163,6 +163,8 @@ void autofs4_kill_sb(struct super_block
    if ( !sbi->catatonic )
        autofs4_catatonic_mode(sbi); /* Free wait queues, close pipe */

```

```

+ put_pid(sbi->oz_pgrp);

```

```

+

```

```

/* Clean up and release dangling references */

```

```

autofs4_force_release(sbi);

```

```

@@ -181,7 +183,7 @@ static int autofs4_show_options(struct s
    return 0;

```

```

seq_printf(m, ",fd=%d", sbi->pipefd);
- seq_printf(m, ",pgrp=%d", sbi->oz_pgrp);
+ seq_printf(m, ",pgrp=%d", pid_nr(sbi->oz_pgrp));
seq_printf(m, ",timeout=%lu", sbi->exp_timeout/HZ);
seq_printf(m, ",minproto=%d", sbi->min_proto);
seq_printf(m, ",maxproto=%d", sbi->max_proto);

```

```

@@ -218,16 +220,17 @@ static match_table_t tokens = {
};

```

```

static int parse_options(char *options, int *pipefd, uid_t *uid, gid_t *gid,
- pid_t *pgrp, unsigned int *type,
+ struct pid **pgrp, unsigned int *type,
    int *minproto, int *maxproto)
{

```

```

char *p;
substring_t args[MAX_OPT_ARGS];
int option;
+ pid_t pgid;

*uid = current->uid;
*gid = current->gid;
- *pgrp = process_group(current);
+ pgid = process_group(current);

*minproto = AUTOFS_MIN_PROTO_VERSION;
*maxproto = AUTOFS_MAX_PROTO_VERSION;
@@ -261,7 +264,7 @@ static int parse_options(char *options,
case Opt_pgrp:
if (match_int(args, &option))
return 1;
- *pgrp = option;
+ pgid = option;
break;
case Opt_minproto:
if (match_int(args, &option))
@@ -286,7 +289,13 @@ static int parse_options(char *options,
return 1;
}
}
- return (*pipefd < 0);
+
+ if (*pipefd < 0)
+ return 1;
+
+ *pgrp = find_get_pid(pgid);
+
+ return (*pgrp == NULL);
}

static struct autofs_info *autofs4_mkroot(struct autofs_sb_info *sbi)
@@ -326,7 +335,7 @@ int autofs4_fill_super(struct super_bloc
sbi->pipe = NULL;
sbi->catatonic = 1;
sbi->exp_timeout = 0;
- sbi->oz_pgrp = process_group(current);
+ sbi->oz_pgrp = NULL;
sbi->sb = s;
sbi->version = 0;
sbi->sub_version = 0;
@@ -381,7 +390,7 @@ int autofs4_fill_super(struct super_bloc
"daemon (%d, %d) kernel (%d, %d)\n",
sbi->min_proto, sbi->max_proto,

```

```

    AUTOFS_MIN_PROTO_VERSION, AUTOFS_MAX_PROTO_VERSION);
- goto fail_dput;
+ goto fail_put_pid;
}

```

```

/* Establish highest kernel protocol version */
@@ -391,12 +400,12 @@ int autofs4_fill_super(struct super_bloc
    sbi->version = sbi->max_proto;
    sbi->sub_version = AUTOFS_PROTO_SUBVERSION;

- DPRINTK("pipe fd = %d, pgrp = %u", pipefd, sbi->oz_pgrp);
+ DPRINTK("pipe fd = %d, pgrp = %u", pipefd, pid_nr(sbi->oz_pgrp));
    pipe = fget(pipefd);

```

```

if ( !pipe ) {
    printk("autofs: could not open pipe file descriptor\n");
- goto fail_dput;
+ goto fail_put_pid;
}
if ( !pipe->f_op || !pipe->f_op->write )
    goto fail_fput;
@@ -417,6 +426,8 @@ fail_fput:
    printk("autofs: pipe file descriptor does not contain proper ops\n");
    fput(pipe);
/* fall through */
+fail_put_pid:
+ put_pid(sbi->oz_pgrp);
fail_dput:
    dput(root);
    goto fail_free;

```

Index: lx26-20-rc4-mm1/fs/autofs4/root.c

```

=====
--- lx26-20-rc4-mm1.orig/fs/autofs4/root.c 2007-01-24 18:39:28.173066264 -0800
+++ lx26-20-rc4-mm1/fs/autofs4/root.c 2007-01-24 18:44:56.310181840 -0800
@@ -495,7 +495,8 @@ static struct dentry *autofs4_lookup(str
    oz_mode = autofs4_oz_mode(sbi);

```

```

    DPRINTK("pid = %u, pgrp = %u, catatonic = %d, oz_mode = %d",
- current->pid, process_group(current), sbi->catatonic, oz_mode);
+ pid_nr(task_pid(current)), process_group(current),
+ sbi->catatonic, oz_mode);

```

```

/*
 * Mark the dentry incomplete, but add it. This is needed so
Index: lx26-20-rc4-mm1/fs/autofs4/waitq.c

```

```

=====
--- lx26-20-rc4-mm1.orig/fs/autofs4/waitq.c 2007-01-24 18:39:28.173066264 -0800
+++ lx26-20-rc4-mm1/fs/autofs4/waitq.c 2007-01-24 18:41:56.262553240 -0800

```

```
@@ -292,8 +292,8 @@ int autofs4_wait(struct autofs_sb_info *
    wq->ino = autofs4_get_ino(sbi);
    wq->uid = current->uid;
    wq->gid = current->gid;
-   wq->pid = current->pid;
-   wq->tgid = current->tgid;
+   wq->pid = pid_nr(task_pid(current));
+   wq->tgid = pid_nr(task_tgid(current));
    wq->status = -EINTR; /* Status return if interrupted */
    atomic_set(&wq->wait_ctr, 2);
    mutex_unlock(&sbi->wq_mutex);
```

---

Containers mailing list

Containers@lists.osdl.org

<https://lists.osdl.org/mailman/listinfo/containers>

---