Subject: [PATCH 2/3] Replace pid_t in autofs with struct pid reference
Posted by Sukadev Bhattiprolu on Thu, 01 Mar 2007 04:00:59 GMT
View Forum Message <> Reply to Message

From: Sukadev Bhattiprolu <sukadev@us.ibm.com>
Subject: [PATCH 2/3] Replace pid_t in autofs with struct pid reference.

Make autofs container-friendly by caching struct pid reference rather
than pid_t and using pid_nr() to retreive a task's pid_t.

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.
 - Fix Cedric's comments: retain old prototype of parse_options()
   and move necessary change to its caller.

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/autofs/autofs_i.h | 4 ++--
 fs/autofs/inode.c | 20 ++++++++++++++++----
 fs/autofs/root.c | 6 ++++--
 3 files changed, 22 insertions(+), 8 deletions(-)

Index: lx26-20-mm2c/fs/autofs/autofs_i.h
===
--- lx26-20-mm2c.orig/fs/autofs/autofs_i.h 2007-02-28 14:48:35.000000000 -0800
+++ lx26-20-mm2c/fs/autofs/autofs_i.h 2007-02-28 15:47:11.000000000 -0800
@@ -101,7 +101,7 @@ struct autofs_symlink {
 struct autofs_sb_info {
  u32 magic;
  struct file *pipe;
- pid_t oz_pgrp;
+ struct pid *oz_pgrp;
  int catatonic;
  struct super_block *sb;
  unsigned long exp_timeout;
@@ -122,7 +122,7 @@ static inline struct autofs_sb_info *aut
    filesystem without "magic".) */

 static inline int autofs_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;
 }

 /* Hash operations */
Index: lx26-20-mm2c/fs/autofs/inode.c
===================================================================
--- lx26-20-mm2c.orig/fs/autofs/inode.c 2007-02-28 15:47:09.000000000 -0800
+++ lx26-20-mm2c/fs/autofs/inode.c 2007-02-28 15:47:11.000000000 -0800
@@ -37,6 +37,8 @@ void autofs_kill_sb(struct super_block *
  if (!sbi->catatonic)
   autofs_catatonic_mode(sbi); /* Free wait queues, close pipe */

+ put_pid(sbi->oz_pgrp);
+
  autofs_hash_nuke(sbi);
  for (n = 0 ; n < AUTOFS_MAX_SYMLINKS ; n++) {
   if (test_bit(n, sbi->symlink_bitmap))
@@ -139,6 +141,7 @@ int autofs_fill_super(struct super_block
  int pipefd;
  struct autofs_sb_info *sbi;
  int minproto, maxproto;
+ pid_t pgid;

  sbi = kzalloc(sizeof(*sbi), GFP_KERNEL);
  if (!sbi)
@@ -150,7 +153,6 @@ int autofs_fill_super(struct super_block
  sbi->pipe = NULL;
  sbi->catatonic = 1;
  sbi->exp_timeout = 0;
- sbi->oz_pgrp = process_group(current);
  autofs_initialize_hash(&sbi->dirhash);
  sbi->queues = NULL;
  memset(sbi->symlink_bitmap, 0, sizeof(long)*AUTOFS_SYMLINK_BITMAP_LEN);
@@ -171,7 +173,7 @@ int autofs_fill_super(struct super_block

  /* Can this call block?  - WTF cares? s is locked. */
  if (parse_options(data, &pipefd, &root_inode->i_uid,
-    &root_inode->i_gid, &sbi->oz_pgrp, &minproto,
+    &root_inode->i_gid, &pgid, &minproto,
     &maxproto)) {
   printk("autofs: called with bogus options\n");
   goto fail_dput;
@@ -184,13 +186,21 @@ int autofs_fill_super(struct super_block
   goto fail_dput;
  }

- DPRINTK(("autofs: pipe fd = %d, pgrp = %u\n", pipefd, sbi->oz_pgrp));
+ DPRINTK(("autofs: pipe fd = %d, pgrp = %u\n", pipefd, pgid));
```

```
+ sbi->oz_pgrp = find_get_pid(pgid);
+
+ if (!sbi->oz_pgrp) {
+  printk("autofs: could not find process group %d\n", pgid);
+  goto fail_dput;
+ }
+
  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;
  sbi->pipe = pipe;
@@ -205,6 +215,8 @@ int autofs_fill_super(struct super_block
 fail_fput:
  printk("autofs: pipe file descriptor does not contain proper ops\n");
  fput(pipe);
+fail_put_pid:
+ put_pid(sbi->oz_pgrp);
 fail_dput:
  dput(root);
  goto fail_free;
Index: lx26-20-mm2c/fs/autofs/root.c
===================================================================
--- lx26-20-mm2c.orig/fs/autofs/root.c 2007-02-28 15:47:09.000000000 -0800
+++ lx26-20-mm2c/fs/autofs/root.c 2007-02-28 15:47:11.000000000 -0800
@@ -213,8 +213,10 @@ static struct dentry *autofs_root_lookup
  sbi = autofs_sbi(dir->i_sb);

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

  /*
   * Mark the dentry incomplete, but add it. This is needed so
```

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

Subject: Re: [PATCH 2/3] Replace pid_t in autofs with struct pid reference
Posted by ebiederm on Fri, 09 Mar 2007 17:28:50 GMT
View Forum Message <> Reply to Message

sukadev@us.ibm.com writes:

> From: Sukadev Bhattiprolu <sukadev@us.ibm.com>
> Subject: [PATCH 2/3] Replace pid_t in autofs with struct pid reference.
>
> Make autofs container-friendly by caching struct pid reference rather
> than pid_t and using pid_nr() to retreive a task's pid_t.
>
> 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.
> - Fix Cedric's comments: retain old prototype of parse_options()
>   and move necessary change to its caller.

Maybe I'm blind but I don't see any obvious problems here.

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

Eric

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