

---

Subject: [RFC] [PATCH 3/3] containers: hook /proc/\$\$/container into containerfs  
Posted by [serue](#) on Wed, 20 Dec 2006 06:02:34 GMT

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

---

From: Serge E. Hallyn <serue@us.ibm.com>

Subject: [RFC] [PATCH 3/3] containers: hook /proc/\$\$/container into containerfs

Create a symlink from /proc/\$\$/container into the containerfs.

What works:

1. `ls -l /proc/$$/container`  
shows the full hierarchical name of the container;
2. `mount -t containerfs none /container`  
results in a file tree under /container representing the full container hierarchy
3. `cd /proc/$$/container; ls`  
results in a listing of child containers

What doesn't work:

The /proc/\$\$/container link always appears dead (red in bash on my fedora test system) because it points into a kern\_mounted fs.

Signed-off-by: Serge E. Hallyn <serue@us.ibm.com>

---  
fs/proc/base.c | 18 ++++++  
include/linux/container.h | 1 +  
kernel/container.c | 2 +-  
3 files changed, 20 insertions(+), 1 deletions(-)

diff --git a/fs/proc/base.c b/fs/proc/base.c

index 77a57b5..00c7618 100644

--- a/fs/proc/base.c

+++ b/fs/proc/base.c

@@ -73,6 +73,7 @@ #include <linux/audit.h>

#include <linux/poll.h>

#include <linux/nsproxy.h>

#include <linux/oom.h>

+#include <linux/container.h>

#include "internal.h"

/\* NOTE:

@@ -189,6 +190,21 @@ static int proc\_root\_link(struct inode \*

return result;

}

```

+static int proc_container_link(struct inode *inode, struct dentry **dentry,
+    struct vfsmount **mnt)
+{
+ struct task_struct *task = get_proc_task(inode);
+ int result = -ENOENT;
+
+ if (task) {
+ *dentry = dget(task->nsproxy->container->dentry);
+ *mnt = mntget(containerfs_mount);
+ put_task_struct(task);
+ result = 0;
+ }
+ return result;
+}
+
+#define MAY_PTRACE(task) \
+ (task == current || \
+ (task->parent == current && \
+ @@ -1879,6 +1895,7 @@ #endif
+ #ifdef CONFIG_TASK_IO_ACCOUNTING
+ INF("io", S_IRUGO, pid_io_accounting),
+ #endif
+ LNK("container", container),
+ };

```

```

static int proc_tgid_base_readdir(struct file * filp,
@@ -2157,6 +2174,7 @@ #endif
#ifdef CONFIG_FAULT_INJECTION
REG("make-it-fail", S_IRUGO|S_IWUSR, fault_inject),
#endif
+ LNK("container", container),
+ };

```

```

static int proc_tid_base_readdir(struct file * filp,
diff --git a/include/linux/container.h b/include/linux/container.h
index c224a53..d5d143c 100644
--- a/include/linux/container.h
+++ b/include/linux/container.h
@@ -17,6 +17,7 @@ struct container {
    struct kref ref;
};
extern struct container init_container;
+extern struct vfsmount *containerfs_mount;

```

```

void put_container(struct container *c);
struct container *new_container(struct container *parent,
diff --git a/kernel/container.c b/kernel/container.c
index 6206e72..5ce82b1 100644

```

```
--- a/kernel/container.c
+++ b/kernel/container.c
@@ -26,7 +26,7 @@ #define CONTAINERFS_MAGIC 0xb6663caf
struct nsproxy;

struct container init_container = INIT_CONTAINER(init_container);
-static struct vfsmount *containerfs_mount;
+struct vfsmount *containerfs_mount;
static void containerfs_remove(struct dentry *dentry);
static struct dentry *containerfs_create_dir(struct container *container);

--
1.4.1
```

---

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

---