
Subject: [PATCH] userns: ptrace: incorporate feedback from Eric

Posted by [serge](#) on Thu, 24 Feb 2011 00:49:01 GMT

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

same_or_ancestore_user_ns() was not an appropriate check to constrain cap_issubset. Rather, cap_issubset() only is meaningful when both capsets are in the same user_ns.

Signed-off-by: Serge E. Hallyn <serge.hallyn@canonical.com>

```
include/linux/user_namespace.h | 9 -----
kernel/user_namespace.c       | 16 -----
security/commoncap.c          | 28 ++++++++-----
3 files changed, 10 insertions(+), 43 deletions(-)
```

diff --git a/include/linux/user_namespace.h b/include/linux/user_namespace.h

index 862fc59..faf4679 100644

--- a/include/linux/user_namespace.h

+++ b/include/linux/user_namespace.h

```
@@ -39,9 +39,6 @@ static inline void put_user_ns(struct user_namespace *ns)
 uid_t user_ns_map_uid(struct user_namespace *to, const struct cred *cred, uid_t uid);
 gid_t user_ns_map_gid(struct user_namespace *to, const struct cred *cred, gid_t gid);
```

```
-int same_or_ancestor_user_ns(struct task_struct *task,
```

```
- struct task_struct *victim);
```

```
-
```

```
#else
```

```
static inline struct user_namespace *get_user_ns(struct user_namespace *ns)
```

```
@@ -69,12 +66,6 @@ static inline gid_t user_ns_map_gid(struct user_namespace *to,
 return gid;
}
```

```
-static inline int same_or_ancestor_user_ns(struct task_struct *task,
```

```
- struct task_struct *victim)
```

```
#{
```

```
- return 1;
```

```
};
```

```
-
```

```
#endif
```

```
#endif /* _LINUX_USER_H */
```

diff --git a/kernel/user_namespace.c b/kernel/user_namespace.c

index 0ef2258..9da289c 100644

--- a/kernel/user_namespace.c

+++ b/kernel/user_namespace.c

```
@@ -129,22 +129,6 @@ gid_t user_ns_map_gid(struct user_namespace *to, const struct cred
 *cred, gid_t
```

```

    return overflowgid;
}

-int same_or_ancestor_user_ns(struct task_struct *task,
- struct task_struct *victim)
-{
- struct user_namespace *u1 = task_cred_xxx(task, user)->user_ns;
- struct user_namespace *u2 = task_cred_xxx(victim, user)->user_ns;
- for (;;) {
- if (u1 == u2)
- return 1;
- if (u1 == &init_user_ns)
- return 0;
- u1 = u1->creator->user_ns;
- }
- /* We never get here */
- return 0;
-}
-
static __init int user_namespaces_init(void)
{
    user_ns_cachep = KMEM_CACHE(user_namespace, SLAB_PANIC);
diff --git a/security/commoncap.c b/security/commoncap.c
index 12ff65c..526106f 100644
--- a/security/commoncap.c
+++ b/security/commoncap.c
@@ -142,19 +142,15 @@ int cap_settime(struct timespec *ts, struct timezone *tz)
int cap_ptrace_access_check(struct task_struct *child, unsigned int mode)
{
    int ret = 0;
- const struct cred *cred, *tcred;
+ const struct cred *cred, *child_cred;

    rcu_read_lock();
    cred = current_cred();
- tcred = __task_cred(child);
- /*
-  * The ancestor user_ns check may be gratuitous, as I think
-  * we've already guaranteed that in kernel/ptrace.c.
-  */
- if (same_or_ancestor_user_ns(current, child) &&
-     cap_issubset(tcred->cap_permitted, cred->cap_permitted))
+ child_cred = __task_cred(child);
+ if (cred->user->user_ns == child_cred->user->user_ns &&
+     cap_issubset(child_cred->cap_permitted, cred->cap_permitted))
    goto out;
- if (ns_capable(tcred->user->user_ns, CAP_SYS_PTRACE))
+ if (ns_capable(child_cred->user->user_ns, CAP_SYS_PTRACE))

```

```

    goto out;
    ret = -EPERM;
out:
@@ -178,19 +174,15 @@ out:
int cap_ptrace_traceme(struct task_struct *parent)
{
    int ret = 0;
- const struct cred *cred, *tcred;
+ const struct cred *cred, *child_cred;

    rcu_read_lock();
    cred = __task_cred(parent);
- tcred = current_cred();
- /*
-  * The ancestor user_ns check may be gratuitous, as I think
-  * we've already guaranteed that in kernel/ptrace.c.
-  */
- if (same_or_ancestor_user_ns(parent, current) &&
-     cap_issubset(tcred->cap_permitted, cred->cap_permitted))
+ child_cred = current_cred();
+ if (cred->user->user_ns == child_cred->user->user_ns &&
+     cap_issubset(child_cred->cap_permitted, cred->cap_permitted))
    goto out;
- if (has_ns_capability(parent, tcred->user->user_ns, CAP_SYS_PTRACE))
+ if (has_ns_capability(parent, child_cred->user->user_ns, CAP_SYS_PTRACE))
    goto out;
    ret = -EPERM;
out:
--
1.7.0.4

```

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
