
Subject: [PATCH RFC 22/31] net: Add network namespace clone support.
Posted by [ebiederm](#) on Thu, 25 Jan 2007 19:00:24 GMT
[View Forum Message](#) <> [Reply to Message](#)

From: Eric W. Biederman <ebiederm@xmission.com> - unquoted

This patch allows you to create a new network namespace
using `sys_clone(...)`.

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

```
include/linux/sched.h | 1 +
kernel/nsproxy.c      | 11 ++++++++
net/core/net_namespace.c | 38 ++++++++++++++++++++++++++++++++++++++
3 files changed, 50 insertions(+), 0 deletions(-)
```

```
diff --git a/include/linux/sched.h b/include/linux/sched.h
index 4463735..9e0f91a 100644
```

```
--- a/include/linux/sched.h
+++ b/include/linux/sched.h
```

```
@@ -26,6 +26,7 @@
```

```
#define CLONE_STOPPED 0x02000000 /* Start in stopped state */
#define CLONE_NEWUTS 0x04000000 /* New utsname group? */
#define CLONE_NEWIPC 0x08000000 /* New ipcs */
+#define CLONE_NEWNET 0x20000000 /* New network namespace */
```

```
/*
```

```
 * Scheduling policies
```

```
diff --git a/kernel/nsproxy.c b/kernel/nsproxy.c
index 4f3c95a..7861c4c 100644
```

```
--- a/kernel/nsproxy.c
+++ b/kernel/nsproxy.c
```

```
@@ -20,6 +20,7 @@
```

```
#include <linux/mnt_namespace.h>
#include <linux/utsname.h>
#include <linux/pid_namespace.h>
+#include <net/net_namespace.h>
```

```
struct nsproxy init_nsproxy = INIT_NS_PROXY(init_nsproxy);
EXPORT_SYMBOL_GPL(init_nsproxy);
@@ -70,6 +71,7 @@ struct nsproxy *dup_namespaces(struct nsproxy *orig)
    get_ipc_ns(ns->ipc_ns);
    if (ns->pid_ns)
        get_pid_ns(ns->pid_ns);
+   get_net(ns->net_ns);
}
```

```
return ns;
```

```

@@ -117,10 +119,18 @@ int copy_namespaces(int flags, struct task_struct *tsk)
    if (err)
        goto out_pid;

+ err = copy_net(flags, tsk);
+ if (err)
+ goto out_net;
+
    out:
        put_nsproxy(old_ns);
        return err;

+out_net:
+ if (new_ns->pid_ns)
+ put_pid_ns(new_ns->pid_ns);
+
    out_pid:
        if (new_ns->ipc_ns)
            put_ipc_ns(new_ns->ipc_ns);
@@ -146,5 +156,6 @@ void free_nsproxy(struct nsproxy *ns)
    put_ipc_ns(ns->ipc_ns);
    if (ns->pid_ns)
        put_pid_ns(ns->pid_ns);
+ put_net(ns->net_ns);
    kfree(ns);
}
diff --git a/net/core/net_namespace.c b/net/core/net_namespace.c
index 93e3879..cc56105 100644
--- a/net/core/net_namespace.c
+++ b/net/core/net_namespace.c
@@ -175,6 +175,44 @@ out_undo:
    goto out;
}

+int copy_net(int flags, struct task_struct *tsk)
+{
+ net_t old_net = tsk->nsproxy->net_ns;
+ net_t new_net;
+ int err;
+
+ get_net(old_net);
+
+ if (!(flags & CLONE_NEWNET))
+ return 0;
+
+ err = -EPERM;
+ if (!capable(CAP_SYS_ADMIN))
+ goto out;

```

```

+
+ err = -ENOMEM;
+ new_net = net_alloc();
+ if (null_net(new_net))
+ goto out;
+
+ mutex_lock(&net_mutex);
+ err = setup_net(new_net);
+ if (err)
+ goto out_unlock;
+
+ net_lock();
+ net_list_append(new_net);
+ net_unlock();
+
+ tsk->nsproxy->net_ns = new_net;
+
+out_unlock:
+ mutex_unlock(&net_mutex);
+out:
+ put_net(old_net);
+ return err;
+}
+
void pernet_modcopy(void *pnetdst, const void *src, unsigned long size)
{
    net_t net;
}
--
1.4.4.1.g278f

```

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