
Subject: Re: [RFC][PATCH 2/5] mqueue namespace: add unshare support
Posted by [Daniel Hokka Zakrisso](#) on Thu, 10 Jul 2008 22:58:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Dave Hansen" <dave@linux.vnet.ibm.com>

>

>

> From: Cedric Le Goater <clg@fr.ibm.com>

>

> This patch includes the mqueue namespace in the nsproxy object. It
> also adds the support of unshare() and clone() with a new clone flag.

^^^

> It's totally harmless for the moment because the current code still
> uses the default mqueue namespace object 'init_mq_ns'

>

> Changes since v6 Mar 05, 2008:

>

> * dropped new clone flag CLONE_NEWMQ and used CLONE_NEWIPC
> instead

Maybe the description above should be updated too?

> Changes since v5 Feb 28, 2008:

>

> * fix typo in create_new_namespaces()

>

> Changes since v4:

>

> * remove CLONE_NEWNS enforcement. stalled user mounts are
> handled in the fs ops.

>

> Signed-off-by: Cedric Le Goater <clg@fr.ibm.com>

> ---

>

> linux-2.6.git-dave/include/linux/init_task.h | 2 +

> linux-2.6.git-dave/include/linux/mq_namespace.h | 1

> linux-2.6.git-dave/include/linux/nsproxy.h | 2 +

> linux-2.6.git-dave/ipc/mq_namespace.c | 36

> ++++++

> linux-2.6.git-dave/kernel/nsproxy.c | 11 ++++++

> 5 files changed, 51 insertions(+), 1 deletion(-)

>

> diff -puN

> include/linux/init_task.h~mq_namespace-add-mq_namespace-to-nsproxy

> include/linux/init_task.h

> ---

>

```

linux-2.6.git/include/linux/init_task.h~mq_namespace-add-mq_namespace-to-nsproxy 2008-06-24
> 12:03:17.000000000 -0700
> +++ linux-2.6.git-dave/include/linux/init_task.h 2008-06-24
> 12:03:17.000000000 -0700
> @@ -10,6 +10,7 @@
> #include <linux/user_namespace.h>
> #include <linux/securebits.h>
> #include <net/net_namespace.h>
> +#include <linux/mq_namespace.h>
>
> extern struct files_struct init_files;
>
> @@ -58,6 +59,7 @@ extern struct nsproxy init_nsproxy;
> INIT_NET_NS(net_ns) \
> INIT_IPC_NS(ipc_ns) \
> .user_ns = &init_user_ns, \
> + INIT_MQ_NS(mq_ns) \
> }
>
> #define INIT_SIGHAND(sighand) { \
> diff -puN
> include/linux/mq_namespace.h~mq_namespace-add-mq_namespace-to-nsproxy
> include/linux/mq_namespace.h
> ---
>
linux-2.6.git/include/linux/mq_namespace.h~mq_namespace-add-mq_namespace-to-nsproxy 2008-06-24
> 12:03:17.000000000 -0700
> +++ linux-2.6.git-dave/include/linux/mq_namespace.h 2008-06-24
> 12:03:17.000000000 -0700
> @@ -2,6 +2,7 @@
> #define _LINUX_MQ_NAMESPACE_H
>
> #include <linux/kref.h>
> +#include <linux/err.h>

```

Why is this necessary (in this patch)?

```

> struct vfsmount;
>
> diff -puN include/linux/nsproxy.h~mq_namespace-add-mq_namespace-to-nsproxy
> include/linux/nsproxy.h
> ---
>
linux-2.6.git/include/linux/nsproxy.h~mq_namespace-add-mq_namespace-to-nsproxy 2008-06-24
> 12:03:17.000000000 -0700
> +++ linux-2.6.git-dave/include/linux/nsproxy.h 2008-06-24
> 12:03:17.000000000 -0700

```

```

> @@ -8,6 +8,7 @@ struct mnt_namespace;
> struct uts_namespace;
> struct ipc_namespace;
> struct pid_namespace;
> +struct mq_namespace;
>
> /*
>  * A structure to contain pointers to all per-process
> @@ -29,6 +30,7 @@ struct nsproxy {
> struct pid_namespace *pid_ns;
> struct user_namespace *user_ns;
> struct net      *net_ns;
> + struct mq_namespace *mq_ns;
> };
> extern struct nsproxy init_nsproxy;
>
> diff -puN ipc/mq_namespace.c~mq_namespace-add-mq_namespace-to-nsproxy
> ipc/mq_namespace.c
> ---
> linux-2.6.git/ipc/mq_namespace.c~mq_namespace-add-mq_namespace-to-nsproxy 2008-06-24
> 12:03:17.000000000 -0700
> +++ linux-2.6.git-dave/ipc/mq_namespace.c 2008-06-24 12:03:17.000000000
> -0700
> @@ -10,14 +10,48 @@
>  */
>
> #include <linux/mq_namespace.h>
> +#include <linux/slab.h>
> +#include <linux/sched.h>
> +#include <linux/err.h>
> +
> +static struct mq_namespace *clone_mq_ns(struct mq_namespace *old_ns)
> +{
> + struct mq_namespace *mq_ns;
> +
> + mq_ns = kmalloc(sizeof(struct mq_namespace), GFP_KERNEL);
> + if (!mq_ns)
> + return ERR_PTR(-ENOMEM);
> +
> + kref_init(&mq_ns->kref);
> + mq_ns->queues_count = 0;
> + mq_ns->queues_max = DFLT_QUEUESMAX;
> + mq_ns->msg_max = DFLT_MSGMAX;
> + mq_ns->msgsize_max = DFLT_MSGSIZEMAX;
> + mq_ns->mnt = NULL;
> + return mq_ns;
> +}
>

```

```

> struct mq_namespace *copy_mq_ns(unsigned long clone_flags,
> struct mq_namespace *old_ns)
> {
> + struct mq_namespace *mq_ns;
> +
> BUG_ON(!old_ns);
> - return get_mq_ns(old_ns);
> + get_mq_ns(old_ns);
> +
> + if (!(clone_flags & CLONE_NEWIPC))
> + return old_ns;
> +
> + mq_ns = clone_mq_ns(old_ns);
> +
> + put_mq_ns(old_ns);
> + return mq_ns;
> }
>
> void free_mq_ns(struct kref *kref)
> {
> + struct mq_namespace *ns;
> +
> + ns = container_of(kref, struct mq_namespace, kref);
> + kfree(ns);
> }
> diff -puN kernel/nsproxy.c~mq_namespace-add-mq_namespace-to-nsproxy
> kernel/nsproxy.c
> ---
> linux-2.6.git/kernel/nsproxy.c~mq_namespace-add-mq_namespace-to-nsproxy 2008-06-24
> 12:03:17.000000000 -0700
> +++ linux-2.6.git-dave/kernel/nsproxy.c 2008-06-24 12:03:17.000000000
> -0700
> @@ -93,8 +93,17 @@ static struct nsproxy *create_new_namesp
> goto out_net;
> }
>
> + new_nsp->mq_ns = copy_mq_ns(flags, tsk->nsproxy->mq_ns);
> + if (IS_ERR(new_nsp->mq_ns)) {
> + err = PTR_ERR(new_nsp->mq_ns);
> + goto out_mq;
> + }
> +
> return new_nsp;
>
> +out_mq:
> + if (new_nsp->net_ns)
> + put_net(new_nsp->net_ns);
> out_net:

```

```
> if (new_nsp->user_ns)
>   put_user_ns(new_nsp->user_ns);
> @@ -182,6 +191,8 @@ void free_nsproxy(struct nsproxy *ns)
>   put_pid_ns(ns->pid_ns);
>   if (ns->user_ns)
>     put_user_ns(ns->user_ns);
> + if (ns->mq_ns)
> +   put_mq_ns(ns->mq_ns);
>   put_net(ns->net_ns);
>   kmem_cache_free(nsproxy_cachep, ns);
> }
> _
```

--

Daniel Hokka Zakrisson

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>
