
Subject: [PATCH 1/2] SUNRPC: register RPC stats /proc entries in passed network namespace context

Posted by [Stanislav Kinsbursky](#) on Tue, 06 Dec 2011 12:48:03 GMT

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

This patch makes it possible to create NFS program entry ("/proc/net/rpc/nfs") in passed network namespace context instead of hard-coded "init_net".

Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

```
fs/nfs/inode.c      | 6 +++---
include/linux/sunrpc/stats.h | 8 ++++----
net/sunrpc/stats.c   | 15 ++++++++-----
3 files changed, 15 insertions(+), 14 deletions(-)
```

diff --git a/fs/nfs/inode.c b/fs/nfs/inode.c

index 98b2508..5cad6c5 100644

--- a/fs/nfs/inode.c

+++ b/fs/nfs/inode.c

```
@@ -1623,14 +1623,14 @@ static int __init init_nfs_fs(void)
    goto out0;
```

```
#ifdef CONFIG_PROC_FS
```

```
- rpc_proc_register(&nfs_rpcstat);
```

```
+ rpc_proc_register(&init_net, &nfs_rpcstat);
```

```
#endif
```

```
if ((err = register_nfs_fs()) != 0)
```

```
    goto out;
```

```
return 0;
```

```
out:
```

```
#ifdef CONFIG_PROC_FS
```

```
- rpc_proc_unregister("nfs");
```

```
+ rpc_proc_unregister(&init_net, "nfs");
```

```
#endif
```

```
nfs_destroy_directcache();
```

```
out0:
```

```
@@ -1669,7 +1669,7 @@ static void __exit exit_nfs_fs(void)
```

```
    nfs_dns_resolver_destroy();
```

```
    nfs_idmap_quit();
```

```
#ifdef CONFIG_PROC_FS
```

```
- rpc_proc_unregister("nfs");
```

```
+ rpc_proc_unregister(&init_net, "nfs");
```

```
#endif
```

```
nfs_cleanup_cb_ident_idr();
```

```
unregister_nfs_fs();
```

diff --git a/include/linux/sunrpc/stats.h b/include/linux/sunrpc/stats.h

index 680471d..f625b57 100644

```

--- a/include/linux/sunrpc/stats.h
+++ b/include/linux/sunrpc/stats.h
@@ -58,8 +58,8 @@ void  rpc_modcount(struct inode *, int);
#endif

#ifdef CONFIG_PROC_FS
-struct proc_dir_entry * rpc_proc_register(struct rpc_stat *);
-void  rpc_proc_unregister(const char *);
+struct proc_dir_entry * rpc_proc_register(struct net *, struct rpc_stat *);
+void  rpc_proc_unregister(struct net *, const char *);
void  rpc_proc_zero(struct rpc_program *);
struct proc_dir_entry * svc_proc_register(struct svc_stat *,
      const struct file_operations *);
@@ -69,8 +69,8 @@ void  svc_seq_show(struct seq_file *,
      const struct svc_stat *);
#else

-static inline struct proc_dir_entry *rpc_proc_register(struct rpc_stat *s) { return NULL; }
-static inline void rpc_proc_unregister(const char *p) {}
+static inline struct proc_dir_entry *rpc_proc_register(struct net *, struct rpc_stat *s) { return NULL;
}
+static inline void rpc_proc_unregister(struct net *, (const char *p) {}
static inline void rpc_proc_zero(struct rpc_program *p) {}

static inline struct proc_dir_entry *svc_proc_register(struct svc_stat *s,
diff --git a/net/sunrpc/stats.c b/net/sunrpc/stats.c
index 80df89d..f0f6e7c 100644
--- a/net/sunrpc/stats.c
+++ b/net/sunrpc/stats.c
@@ -213,28 +213,29 @@ EXPORT_SYMBOL_GPL(rpc_print_iostats);
 * Register/unregister RPC proc files
 */
static inline struct proc_dir_entry *
-do_register(const char *name, void *data, const struct file_operations *fops)
+do_register(struct net *net, const char *name, void *data,
+    const struct file_operations *fops)
{
    struct sunrpc_net *sn;

    dprintk("RPC:    registering /proc/net/rpc/%s\n", name);
- sn = net_generic(&init_net, sunrpc_net_id);
+ sn = net_generic(net, sunrpc_net_id);
    return proc_create_data(name, 0, sn->proc_net_rpc, fops, data);
}

struct proc_dir_entry *
-rpc_proc_register(struct rpc_stat *statp)
+rpc_proc_register(struct net *net, struct rpc_stat *statp)

```

```

{
- return do_register(statp->program->name, statp, &rpc_proc_fops);
+ return do_register(net, statp->program->name, statp, &rpc_proc_fops);
}
EXPORT_SYMBOL_GPL(rpc_proc_register);

void
-rpc_proc_unregister(const char *name)
+rpc_proc_unregister(struct net *net, const char *name)
{
    struct sunrpc_net *sn;

- sn = net_generic(&init_net, sunrpc_net_id);
+ sn = net_generic(net, sunrpc_net_id);
    remove_proc_entry(name, sn->proc_net_rpc);
}
EXPORT_SYMBOL_GPL(rpc_proc_unregister);
@@ -242,7 +243,7 @@ EXPORT_SYMBOL_GPL(rpc_proc_unregister);
struct proc_dir_entry *
svc_proc_register(struct svc_stat *statp, const struct file_operations *fops)
{
- return do_register(statp->program->pg_name, statp, fops);
+ return do_register(&init_net, statp->program->pg_name, statp, fops);
}
EXPORT_SYMBOL_GPL(svc_proc_register);

```

Subject: Re: [PATCH 1/2] SUNRPC: register RPC stats /proc entries in passed network namespace context

Posted by [bfields](#) on Tue, 06 Dec 2011 21:33:22 GMT

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

On Tue, Dec 06, 2011 at 04:42:40PM +0300, Stanislav Kinsbursky wrote:

> This patch makes it possible to create NFS program entry ("/proc/net/rpc/nfs")
> in passed network namespace context instead of hard-coded "init_net".

Looks straightforward to me.

Acked-by: J. Bruce Fields <bfields@redhat.com>

--b.

```

>
> Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>
>
> ---
> fs/nfs/inode.c          | 6 +++---
> include/linux/sunrpc/stats.h | 8 ++++----

```

```

> net/sunrpc/stats.c      | 15 ++++++-----
> 3 files changed, 15 insertions(+), 14 deletions(-)
>
> diff --git a/fs/nfs/inode.c b/fs/nfs/inode.c
> index 98b2508..5cad6c5 100644
> --- a/fs/nfs/inode.c
> +++ b/fs/nfs/inode.c
> @@ -1623,14 +1623,14 @@ static int __init init_nfs_fs(void)
>  goto out0;
>
> #ifdef CONFIG_PROC_FS
> - rpc_proc_register(&nfs_rpcstat);
> + rpc_proc_register(&init_net, &nfs_rpcstat);
> #endif
> if ((err = register_nfs_fs()) != 0)
>  goto out;
> return 0;
> out:
> #ifdef CONFIG_PROC_FS
> - rpc_proc_unregister("nfs");
> + rpc_proc_unregister(&init_net, "nfs");
> #endif
> nfs_destroy_directcache();
> out0:
> @@ -1669,7 +1669,7 @@ static void __exit exit_nfs_fs(void)
>  nfs_dns_resolver_destroy();
>  nfs_idmap_quit();
> #ifdef CONFIG_PROC_FS
> - rpc_proc_unregister("nfs");
> + rpc_proc_unregister(&init_net, "nfs");
> #endif
> nfs_cleanup_cb_ident_idr();
> unregister_nfs_fs();
> diff --git a/include/linux/sunrpc/stats.h b/include/linux/sunrpc/stats.h
> index 680471d..f625b57 100644
> --- a/include/linux/sunrpc/stats.h
> +++ b/include/linux/sunrpc/stats.h
> @@ -58,8 +58,8 @@ void  rpc_modcount(struct inode *, int);
> #endif
>
> #ifdef CONFIG_PROC_FS
> -struct proc_dir_entry * rpc_proc_register(struct rpc_stat *);
> -void  rpc_proc_unregister(const char *);
> +struct proc_dir_entry * rpc_proc_register(struct net *, struct rpc_stat *);
> +void  rpc_proc_unregister(struct net *, const char *);
> void  rpc_proc_zero(struct rpc_program *);
> struct proc_dir_entry * svc_proc_register(struct svc_stat *,
>      const struct file_operations *);

```

```

> @@ -69,8 +69,8 @@ void  svc_seq_show(struct seq_file *,
>      const struct svc_stat *);
> #else
>
> -static inline struct proc_dir_entry *rpc_proc_register(struct rpc_stat *s) { return NULL; }
> -static inline void rpc_proc_unregister(const char *p) {}
> +static inline struct proc_dir_entry *rpc_proc_register(struct net *, struct rpc_stat *s) { return
NULL; }
> +static inline void rpc_proc_unregister(struct net *, (const char *p) {}
> static inline void rpc_proc_zero(struct rpc_program *p) {}
>
> static inline struct proc_dir_entry *svc_proc_register(struct svc_stat *s,
> diff --git a/net/sunrpc/stats.c b/net/sunrpc/stats.c
> index 80df89d..f0f6e7c 100644
> --- a/net/sunrpc/stats.c
> +++ b/net/sunrpc/stats.c
> @@ -213,28 +213,29 @@ EXPORT_SYMBOL_GPL(rpc_print_iostats);
>  * Register/unregister RPC proc files
>  */
> static inline struct proc_dir_entry *
> -do_register(const char *name, void *data, const struct file_operations *fops)
> +do_register(struct net *net, const char *name, void *data,
> +    const struct file_operations *fops)
> {
>     struct sunrpc_net *sn;
>
>     dprintk("RPC:    registering /proc/net/rpc/%s\n", name);
> - sn = net_generic(&init_net, sunrpc_net_id);
> + sn = net_generic(net, sunrpc_net_id);
>     return proc_create_data(name, 0, sn->proc_net_rpc, fops, data);
> }
>
> struct proc_dir_entry *
> -rpc_proc_register(struct rpc_stat *statp)
> +rpc_proc_register(struct net *net, struct rpc_stat *statp)
> {
> - return do_register(statp->program->name, statp, &rpc_proc_fops);
> + return do_register(net, statp->program->name, statp, &rpc_proc_fops);
> }
> EXPORT_SYMBOL_GPL(rpc_proc_register);
>
> void
> -rpc_proc_unregister(const char *name)
> +rpc_proc_unregister(struct net *net, const char *name)
> {
>     struct sunrpc_net *sn;
>
> - sn = net_generic(&init_net, sunrpc_net_id);

```

```
> + sn = net_generic(net, sunrpc_net_id);
> remove_proc_entry(name, sn->proc_net_rpc);
> }
> EXPORT_SYMBOL_GPL(rpc_proc_unregister);
> @@ -242,7 +243,7 @@ EXPORT_SYMBOL_GPL(rpc_proc_unregister);
> struct proc_dir_entry *
> svc_proc_register(struct svc_stat *statp, const struct file_operations *fops)
> {
> - return do_register(statp->program->pg_name, statp, fops);
> + return do_register(&init_net, statp->program->pg_name, statp, fops);
> }
> EXPORT_SYMBOL_GPL(svc_proc_register);
>
>
```

Subject: Re: [PATCH 1/2] SUNRPC: register RPC stats /proc entries in passed network namespace context

Posted by [Myklebust, Trond](#) on Mon, 30 Jan 2012 20:49:49 GMT

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

On Mon, 2012-01-30 at 13:29 -0500, Paul Gortmaker wrote:

```
> On Tue, Dec 6, 2011 at 8:42 AM, Stanislav Kinsbursky
> <skinsbursky@parallels.com> wrote:
> > This patch makes it possible to create NFS program entry ("/proc/net/rpc/nfs")
> > in passed network namespace context instead of hard-coded "init_net".
>
> Hi Trond,
>
> Is this patch going to be dropped and re-submitted? It has obvious
> typos that have been causing reported build failures[1] in linux-next
> for a week now. In addition to that, it has several whitespace errors
> (long lines, no space after comma, etc.) that I bet checkpatch.pl
> would have found, if it was used.
>
> Commit 4494c750240bab1bb83bd8f4d5a268663c37c9b3 in next-20120130.
```

I've applied Randy's fix to my tree. Sorry it took so long, but I was travelling last week...

--

Trond Myklebust
Linux NFS client maintainer

NetApp
Trond.Myklebust@netapp.com
www.netapp.com
