
Subject: [PATCH] SUNRPC: make SUNRPC clients list per network namespace context

Posted by [Stanislav Kinsbursky](#) on Wed, 14 Dec 2011 08:20:34 GMT

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

This patch moves static SUNRPC clients list and it's lock to sunrpc_net structure.

Currently this list is used only for debug purposes. But later it will be used also for selecting clients by networks namespace on PipeFS mount/umount events. Per-network namespace lists will make this faster and simpler.

Note: client list is taken from "init_net" network namespace context in rpc_show_tasks(). This will be changed with making SUNRPC sysctl's per network namespace context.

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

```
include/linux/sunrpc/sched.h | 3 +-
net/sunrpc/clnt.c            | 26 ++++++++-----
net/sunrpc/netns.h           | 3 +++
net/sunrpc/sunrpc_syms.c     | 3 +++
net/sunrpc/sysctl.c          | 4 +++-
5 files changed, 26 insertions(+), 13 deletions(-)
```

```
diff --git a/include/linux/sunrpc/sched.h b/include/linux/sunrpc/sched.h
```

```
index e775689..b16243a 100644
```

```
--- a/include/linux/sunrpc/sched.h
```

```
+++ b/include/linux/sunrpc/sched.h
```

```
@@ -244,7 +244,8 @@ int rpciod_up(void);
```

```
void rpciod_down(void);
```

```
int __rpc_wait_for_completion_task(struct rpc_task *task, int (*)(void *));
```

```
#ifdef RPC_DEBUG
```

```
-void rpc_show_tasks(void);
```

```
+struct net;
```

```
+void rpc_show_tasks(struct net *);
```

```
#endif
```

```
int rpc_init_mempool(void);
```

```
void rpc_destroy_mempool(void);
```

```
diff --git a/net/sunrpc/clnt.c b/net/sunrpc/clnt.c
```

```
index f0268ea..c5f04aa 100644
```

```
--- a/net/sunrpc/clnt.c
```

```
+++ b/net/sunrpc/clnt.c
```

```
@@ -38,6 +38,7 @@
```

```
#include <linux/sunrpc/bc_xprt.h>
```

```
#include "sunrpc.h"
```

```
+#include "netns.h"
```

```

#ifdef RPC_DEBUG
# define RPCDBG_FACILITY RPCDBG_CALL
@@ -50,8 +51,6 @@
/*
 * All RPC clients are linked into this list
 */
-static LIST_HEAD(all_clients);
-static DEFINE_SPINLOCK(rpc_client_lock);

static DECLARE_WAIT_QUEUE_HEAD(destroy_wait);

@@ -81,16 +80,20 @@ static int rpc_ping(struct rpc_clnt *clnt);

static void rpc_register_client(struct rpc_clnt *clnt)
{
- spin_lock(&rpc_client_lock);
- list_add(&clnt->cl_clients, &all_clients);
- spin_unlock(&rpc_client_lock);
+ struct sunrpc_net *sn = net_generic(clnt->cl_xprt->xprt_net, sunrpc_net_id);
+
+ spin_lock(&sn->rpc_client_lock);
+ list_add(&clnt->cl_clients, &sn->all_clients);
+ spin_unlock(&sn->rpc_client_lock);
}

static void rpc_unregister_client(struct rpc_clnt *clnt)
{
- spin_lock(&rpc_client_lock);
+ struct sunrpc_net *sn = net_generic(clnt->cl_xprt->xprt_net, sunrpc_net_id);
+
+ spin_lock(&sn->rpc_client_lock);
  list_del(&clnt->cl_clients);
- spin_unlock(&rpc_client_lock);
+ spin_unlock(&sn->rpc_client_lock);
}

static int
@@ -1852,14 +1855,15 @@ static void rpc_show_task(const struct rpc_clnt *clnt,
  task->tk_action, rpc_waitq);
}

-void rpc_show_tasks(void)
+void rpc_show_tasks(struct net *net)
{
  struct rpc_clnt *clnt;
  struct rpc_task *task;
  int header = 0;

```

```

+ struct sunrpc_net *sn = net_generic(net, sunrpc_net_id);

- spin_lock(&rpc_client_lock);
- list_for_each_entry(clnt, &all_clients, cl_clients) {
+ spin_lock(&sn->rpc_client_lock);
+ list_for_each_entry(clnt, &sn->all_clients, cl_clients) {
    spin_lock(&clnt->cl_lock);
    list_for_each_entry(task, &clnt->cl_tasks, tk_task) {
        if (!header) {
@@ -1870,6 +1874,6 @@ void rpc_show_tasks(void)
    }
    spin_unlock(&clnt->cl_lock);
}
- spin_unlock(&rpc_client_lock);
+ spin_unlock(&sn->rpc_client_lock);
}
#endif
diff --git a/net/sunrpc/netns.h b/net/sunrpc/netns.h
index d013bf2..6010c46 100644
--- a/net/sunrpc/netns.h
+++ b/net/sunrpc/netns.h
@@ -9,6 +9,9 @@ struct cache_detail;
struct sunrpc_net {
    struct proc_dir_entry *proc_net_rpc;
    struct cache_detail *ip_map_cache;
+
+ struct list_head all_clients;
+ spinlock_t rpc_client_lock;
};

extern int sunrpc_net_id;
diff --git a/net/sunrpc/sunrpc_syms.c b/net/sunrpc/sunrpc_syms.c
index 8ec9778..e57aa10 100644
--- a/net/sunrpc/sunrpc_syms.c
+++ b/net/sunrpc/sunrpc_syms.c
@@ -29,6 +29,7 @@ int sunrpc_net_id;
static __net_init int sunrpc_init_net(struct net *net)
{
    int err;
+ struct sunrpc_net *sn = net_generic(net, sunrpc_net_id);

    err = rpc_proc_init(net);
    if (err)
@@ -38,6 +39,8 @@ static __net_init int sunrpc_init_net(struct net *net)
    if (err)
        goto err_ipmap;

+ INIT_LIST_HEAD(&sn->all_clients);

```

```

+ spin_lock_init(&sn->rpc_client_lock);
  return 0;

err_ipmap:
diff --git a/net/sunrpc/sysctl.c b/net/sunrpc/sysctl.c
index e65dcc6..af7d339 100644
--- a/net/sunrpc/sysctl.c
+++ b/net/sunrpc/sysctl.c
@@ -20,6 +20,8 @@
#include <linux/sunrpc/stats.h>
#include <linux/sunrpc/svc_xprt.h>

+#include "netns.h"
+
+/*
+ * Declare the debug flags here
+ */
@@ -110,7 +112,7 @@ proc_dodebug(ctl_table *table, int write,
  *(unsigned int *) table->data = value;
  /* Display the RPC tasks on writing to rpc_debug */
  if (strcmp(table->procname, "rpc_debug") == 0)
-   rpc_show_tasks();
+   rpc_show_tasks(&init_net);
  } else {
    if (!access_ok(VERIFY_WRITE, buffer, left))
      return -EFAULT;

```

Subject: Re: [PATCH] SUNRPC: make SUNRPC clients list per network namespace context

Posted by [Myklebust, Trond](#) on Sun, 25 Dec 2011 12:48:15 GMT

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

On Wed, 2011-12-14 at 12:19 +0300, Stanislav Kinsbursky wrote:

```

> This patch moves static SUNRPC clients list and it's lock to sunrpc_net
> structure.
> Currently this list is used only for debug purposes. But later it will be used
> also for selecting clients by networks namespace on PipeFS mount/umount events.
> Per-network namespace lists will make this faster and simpler.
>
> Note: client list is taken from "init_net" network namespace context in
> rpc_show_tasks(). This will be changed with making SUNRPC sysctl's per network
> namespace context.
>
> Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>

```

This doesn't seem to integrate with any of your patch series, and conflicts with everything that touches netns.h, for instance...

--

Trond Myklebust
Linux NFS client maintainer

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

Subject: Re: [PATCH] SUNRPC: make SUNRPC clients list per network namespace context

Posted by [Stanislav Kinsbursky](#) on Mon, 26 Dec 2011 12:25:06 GMT

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

> On Wed, 2011-12-14 at 12:19 +0300, Stanislav Kinsbursky wrote:
>> This patch moves static SUNRPC clients list and it's lock to sunrpc_net
>> structure.
>> Currently this list is used only for debug purposes. But later it will be used
>> also for selecting clients by networks namespace on PipeFS mount/umount events.
>> Per-network namespace lists will make this faster and simpler.
>>
>> Note: client list is taken from "init_net" network namespace context in
>> rpc_show_tasks(). This will be changed with making SUNRPC sysctl's per network
>> namespace context.
>>
>> Signed-off-by: Stanislav Kinsbursky<skinsbursky@parallels.com>
>
> This doesn't seem to integrate with any of your patch series, and
> conflicts with everything that touches netns.h, for instance...
>

Yep, you are right. Currently this path doesn't apply, if you'll try to apply patches in the same order I've sent them.
This is caused by different development branches.
Anyway, most of the patches will not apply to current tree.
And I'll be glad to rebase and resent them on request.

--

Best regards,
Stanislav Kinsbursky
