
Subject: [PATCH] SUNRPC: search for service transports in network namespace context

Posted by [Stanislav Kinsbursky](#) on Fri, 20 Jan 2012 12:50:53 GMT

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

Service transports are parametrized by network namespace. And thus lookup of transport instance have to take network namespace into account.

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

```
---
fs/lockd/svc.c          | 2 +-
fs/nfsd/nfsctl.c        | 4 ++--
include/linux/sunrpc/svc_xprt.h | 3 ++-
net/sunrpc/svc_xprt.c    | 6 +++++-
4 files changed, 10 insertions(+), 5 deletions(-)

diff --git a/fs/lockd/svc.c b/fs/lockd/svc.c
index c061b9a..ff379ff 100644
--- a/fs/lockd/svc.c
+++ b/fs/lockd/svc.c
@@ -193,7 +193,7 @@ static int create_lockd_listener(struct svc_serv *serv, const char *name,
{
    struct svc_xprt *xprt;

- xprt = svc_find_xprt(serv, name, family, 0);
+ xprt = svc_find_xprt(serv, name, &init_net, family, 0);
    if (xprt == NULL)
        return svc_create_xprt(serv, name, &init_net, family, port,
                                SVC_SOCKET_DEFAULTS);
diff --git a/fs/nfsd/nfsctl.c b/fs/nfsd/nfsctl.c
index 330352d..64c24af 100644
--- a/fs/nfsd/nfsctl.c
+++ b/fs/nfsd/nfsctl.c
@@ -722,7 +722,7 @@ static ssize_t __write_ports_addxprt(char *buf)
    nfsd_serv->sv_nrthreads--;
    return 0;
out_close:
- xprt = svc_find_xprt(nfsd_serv, transport, PF_INET, port);
+ xprt = svc_find_xprt(nfsd_serv, transport, &init_net, PF_INET, port);
    if (xprt != NULL) {
        svc_close_xprt(xprt);
        svc_xprt_put(xprt);
@@ -748,7 +748,7 @@ static ssize_t __write_ports_delxprt(char *buf)
    if (port < 1 || port > USHRT_MAX || nfsd_serv == NULL)
        return -EINVAL;

- xprt = svc_find_xprt(nfsd_serv, transport, AF_UNSPEC, port);
```

```
+ xprt = svc_find_xprt(nfsd_serv, transport, &init_net, AF_UNSPEC, port);
  if (xprt == NULL)
    return -ENOTCONN;
```

```
diff --git a/include/linux/sunrpc/svc_xprt.h b/include/linux/sunrpc/svc_xprt.h
index dfa9009..b3f64b1 100644
```

```
--- a/include/linux/sunrpc/svc_xprt.h
+++ b/include/linux/sunrpc/svc_xprt.h
@@ -121,7 +121,8 @@ void svc_close_xprt(struct svc_xprt *xprt);
int svc_port_is_privileged(struct sockaddr *sin);
int svc_print_xprts(char *buf, int maxlen);
struct svc_xprt *svc_find_xprt(struct svc_serv *serv, const char *xcl_name,
- const sa_family_t af, const unsigned short port);
+ struct net *net, const sa_family_t af,
+ const unsigned short port);
int svc_xprt_names(struct svc_serv *serv, char *buf, const int buflen);
```

```
static inline void svc_xprt_get(struct svc_xprt *xprt)
diff --git a/net/sunrpc/svc_xprt.c b/net/sunrpc/svc_xprt.c
index 74cb0d8..de1e1a8 100644
```

```
--- a/net/sunrpc/svc_xprt.c
+++ b/net/sunrpc/svc_xprt.c
@@ -1089,6 +1089,7 @@ static struct svc_deferred_req *svc_deferred_dequeue(struct svc_xprt
*xprt)
 * svc_find_xprt - find an RPC transport instance
 * @serv: pointer to svc_serv to search
 * @xcl_name: C string containing transport's class name
+ * @net: owner net pointer
 * @af: Address family of transport's local address
 * @port: transport's IP port number
 *
@@ -1101,7 +1102,8 @@ static struct svc_deferred_req *svc_deferred_dequeue(struct svc_xprt
*xprt)
 * service's list that has a matching class name.
 */
```

```
struct svc_xprt *svc_find_xprt(struct svc_serv *serv, const char *xcl_name,
- const sa_family_t af, const unsigned short port)
+ struct net *net, const sa_family_t af,
+ const unsigned short port)
{
```

```
    struct svc_xprt *xprt;
    struct svc_xprt *found = NULL;
@@ -1112,6 +1114,8 @@ struct svc_xprt *svc_find_xprt(struct svc_serv *serv, const char
*xcl_name,
```

```
    spin_lock_bh(&serv->sv_lock);
    list_for_each_entry(xprt, &serv->sv_permsocks, xprt_list) {
+ if (xprt->xprt_net != net)
```

```
+ continue;  
  if (strcmp(xprt->xpt_class->xcl_name, xcl_name))  
    continue;  
  if (af != AF_UNSPEC && af != xprt->xpt_local.ss_family)
```
