
Subject: [PATCH v2 4/4] SUNRPC: parametrize rpc_uaddr2sockaddr() by network context

Posted by [Stanislav Kinsbursky](#) on Fri, 13 Jan 2012 09:09:27 GMT

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

Parametrize rpc_uaddr2sockaddr() by network context and thus force it's callers to pass in network context instead of using hard-coded "init_net".

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

```
---
fs/nfsd/nfs4state.c      |  2 +-
include/linux/sunrpc/clnt.h |  2 +-
net/sunrpc/addr.c        |  8 ++++++---
net/sunrpc/rpcb_clnt.c    |  2 +-
4 files changed, 8 insertions(+), 6 deletions(-)

diff --git a/fs/nfsd/nfs4state.c b/fs/nfsd/nfs4state.c
index 9ca16dc..56ad8ca 100644
--- a/fs/nfsd/nfs4state.c
+++ b/fs/nfsd/nfs4state.c
@@ -1301,7 +1301,7 @@ gen_callback(struct nfs4_client *clp, struct nfsd4_setclientid *se, struct
svc_r
    else
        goto out_err;

- conn->cb_addrlen = rpc_uaddr2sockaddr(se->se_callback_addr_val,
+ conn->cb_addrlen = rpc_uaddr2sockaddr(&init_net, se->se_callback_addr_val,
    se->se_callback_addr_len,
    (struct sockaddr *)&conn->cb_addr,
    sizeof(conn->cb_addr));
diff --git a/include/linux/sunrpc/clnt.h b/include/linux/sunrpc/clnt.h
index 7bd114f..9e754e3 100644
--- a/include/linux/sunrpc/clnt.h
+++ b/include/linux/sunrpc/clnt.h
@@ -166,7 +166,7 @@ size_t rpc_ntop(const struct sockaddr *, char *, const size_t);
size_t rpc_pton(struct net *, const char *, const size_t,
    struct sockaddr *, const size_t);
char * rpc_sockaddr2uaddr(const struct sockaddr *, gfp_t);
-size_t rpc_uaddr2sockaddr(const char *, const size_t,
+size_t rpc_uaddr2sockaddr(struct net *, const char *, const size_t,
    struct sockaddr *, const size_t);

static inline unsigned short rpc_get_port(const struct sockaddr *sap)
diff --git a/net/sunrpc/addr.c b/net/sunrpc/addr.c
index 82b06b7..d11418f 100644
--- a/net/sunrpc/addr.c
+++ b/net/sunrpc/addr.c
```

```

@@ -297,6 +297,7 @@ char *rpc_sockaddr2uaddr(const struct sockaddr *sap, gfp_t gfp_flags)

/**
 * rpc_uaddr2sockaddr - convert a universal address to a socket address.
+ * @net: applicable network namespace
 * @uaddr: C string containing universal address to convert
 * @uaddr_len: length of universal address string
 * @sap: buffer into which to plant socket address
@@ -308,8 +309,9 @@ char *rpc_sockaddr2uaddr(const struct sockaddr *sap, gfp_t gfp_flags)
 * Returns the size of the socket address if successful; otherwise
 * zero is returned.
 */
-size_t rpc_uaddr2sockaddr(const char *uaddr, const size_t uaddr_len,
- struct sockaddr *sap, const size_t salen)
+size_t rpc_uaddr2sockaddr(struct net *net, const char *uaddr,
+ const size_t uaddr_len, struct sockaddr *sap,
+ const size_t salen)
{
    char *c, buf[RPCBIND_MAXUADDRLEN + sizeof('\0')];
    unsigned long portlo, porthi;
@@ -341,7 +343,7 @@ size_t rpc_uaddr2sockaddr(const char *uaddr, const size_t uaddr_len,
    port = (unsigned short)((porthi << 8) | portlo);

    *c = '\0';
- if (rpc_pton(&init_net, buf, strlen(buf), sap, salen) == 0)
+ if (rpc_pton(net, buf, strlen(buf), sap, salen) == 0)
    return 0;

    switch (sap->sa_family) {
diff --git a/net/sunrpc/rpcb_clnt.c b/net/sunrpc/rpcb_clnt.c
index 6d6a84f..7c6fac0 100644
--- a/net/sunrpc/rpcb_clnt.c
+++ b/net/sunrpc/rpcb_clnt.c
@@ -934,7 +934,7 @@ static int rpcb_dec_getaddr(struct rpc_rqst *req, struct xdr_stream *xdr,
    dprintk("RPC: %5u RPCB_%s reply: %s\n", task->tk_pid,
        task->tk_msg.rpc_proc->p_name, (char *)p);

- if (rpc_uaddr2sockaddr((char *)p, len, sap, sizeof(address)) == 0)
+ if (rpc_uaddr2sockaddr(&init_net, (char *)p, len, sap, sizeof(address)) == 0)
    goto out_fail;
    rpcb->r_port = rpc_get_port(sap);

```
