
Subject: [PATCH v2 3/4] SUNRPC: parametrize rpc_pton() by network context
Posted by [Stanislav Kinsbursky](#) on Fri, 13 Jan 2012 09:09:19 GMT
[View Forum Message](#) <> [Reply to Message](#)

Parametrize rpc_pton() 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/nfs/dns_resolve.c      |  4 +++
fs/nfs/nfs4filelayoutdev.c |  2 +-
fs/nfs/nfs4namespace.c    |  2 +-
fs/nfs/super.c            |  4 ++--
fs/nfsd/nfsctl.c          |  2 +-
include/linux/sunrpc/clnt.h |  2 +-
net/sunrpc/addr.c         |  7 ++++---
net/sunrpc/svcauth_unix.c |  2 +-
8 files changed, 13 insertions(+), 12 deletions(-)

diff --git a/fs/nfs/dns_resolve.c b/fs/nfs/dns_resolve.c
index 200eb67..7edc62a 100644
--- a/fs/nfs/dns_resolve.c
+++ b/fs/nfs/dns_resolve.c
@@ -20,7 +20,7 @@ ssize_t nfs_dns_resolve_name(struct net *net, char *name, size_t namelen,
    ip_len = dns_query(NULL, name, namelen, NULL, &ip_addr, NULL);
    if (ip_len > 0)
-   ret = rpc_pton(ip_addr, ip_len, sa, salen);
+   ret = rpc_pton(&init_net, ip_addr, ip_len, sa, salen);
    else
        ret = -ESRCH;
    kfree(ip_addr);
@@ -224,7 +224,7 @@ static int nfs_dns_parse(struct cache_detail *cd, char *buf, int buflen)
    len = qword_get(&buf, buf1, sizeof(buf1));
    if (len <= 0)
        goto out;
-   key.addrlen = rpc_pton(buf1, len,
+   key.addrlen = rpc_pton(&init_net, buf1, len,
        (struct sockaddr *)&key.addr,
        sizeof(key.addr));

diff --git a/fs/nfs/nfs4filelayoutdev.c b/fs/nfs/nfs4filelayoutdev.c
index 8ae9190..0d8b952 100644
--- a/fs/nfs/nfs4filelayoutdev.c
+++ b/fs/nfs/nfs4filelayoutdev.c
@@ -457,7 +457,7 @@ decode_ds_addr(struct xdr_stream *stream, gfp_t gfp_flags)
```

```

INIT_LIST_HEAD(&da->da_node);

- if (!rpc_pton(buf, portstr-buf, (struct sockaddr *)&da->da_addr,
+ if (!rpc_pton(&init_net, buf, portstr-buf, (struct sockaddr *)&da->da_addr,
    sizeof(da->da_addr))) {
    dprintk("%s: error parsing address %s\n", __func__, buf);
    goto out_free_da;
diff --git a/fs/nfs/nfs4namespace.c b/fs/nfs/nfs4namespace.c
index 919a369..48a9acd 100644
--- a/fs/nfs/nfs4namespace.c
+++ b/fs/nfs/nfs4namespace.c
@@ -98,7 +98,7 @@ static size_t nfs_parse_server_name(char *string, size_t len,
{
    ssize_t ret;

- ret = rpc_pton(string, len, sa, salen);
+ ret = rpc_pton(&init_net, string, len, sa, salen);
    if (ret == 0) {
        ret = nfs_dns_resolve_name(server->client->cl_xprt->xprt_net,
            string, len, sa, salen);
diff --git a/fs/nfs/super.c b/fs/nfs/super.c
index e45feb0..b79f2a1 100644
--- a/fs/nfs/super.c
+++ b/fs/nfs/super.c
@@ -1408,7 +1408,7 @@ static int nfs_parse_mount_options(char *raw,
    if (string == NULL)
        goto out_nomem;
    mnt->nfs_server.addrlen =
-    rpc_pton(string, strlen(string),
+    rpc_pton(&init_net, string, strlen(string),
        (struct sockaddr *)
            &mnt->nfs_server.address,
            sizeof(mnt->nfs_server.address));
@@ -1430,7 +1430,7 @@ static int nfs_parse_mount_options(char *raw,
    if (string == NULL)
        goto out_nomem;
    mnt->mount_server.addrlen =
-    rpc_pton(string, strlen(string),
+    rpc_pton(&init_net, string, strlen(string),
        (struct sockaddr *)
            &mnt->mount_server.address,
            sizeof(mnt->mount_server.address));
diff --git a/fs/nfsd/nfsctl.c b/fs/nfsd/nfsctl.c
index bb4a11d..3bc8242 100644
--- a/fs/nfsd/nfsctl.c
+++ b/fs/nfsd/nfsctl.c
@@ -222,7 +222,7 @@ static ssize_t write_unlock_ip(struct file *file, char *buf, size_t size)
    if (qword_get(&buf, fo_path, size) < 0)

```

```

return -EINVAL;

- if (rpc_pton(fo_path, size, sap, salen) == 0)
+ if (rpc_pton(&init_net, fo_path, size, sap, salen) == 0)
    return -EINVAL;

    return nlmsvc_unlock_all_by_ip(sap);
diff --git a/include/linux/sunrpc/clnt.h b/include/linux/sunrpc/clnt.h
index e891a8a..7bd114f 100644
--- a/include/linux/sunrpc/clnt.h
+++ b/include/linux/sunrpc/clnt.h
@@ -163,7 +163,7 @@ size_t rpc_peeraddr(struct rpc_clnt *, struct sockaddr *, size_t);
const char *rpc_peeraddr2str(struct rpc_clnt *, enum rpc_display_format_t);

size_t rpc_ntop(const struct sockaddr *, char *, const size_t);
-size_t rpc_pton(const 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,
diff --git a/net/sunrpc/addr.c b/net/sunrpc/addr.c
index 18c40a2..82b06b7 100644
--- a/net/sunrpc/addr.c
+++ b/net/sunrpc/addr.c
@@ -230,6 +230,7 @@ static size_t rpc_pton6(struct net *net, const char *buf, const size_t
buflen,

/**
 * rpc_pton - Construct a sockaddr in @sap
+ * @net: applicable network namespace
 * @buf: C string containing presentation format IP address
 * @buflen: length of presentation address in bytes
 * @sap: buffer into which to plant socket address
@@ -242,14 +243,14 @@ static size_t rpc_pton6(struct net *net, const char *buf, const size_t
buflen,
 * socket address, if successful. Returns zero if an error
 * occurred.
 */
-size_t rpc_pton(const char *buf, const size_t buflen,
+size_t rpc_pton(struct net *net, const char *buf, const size_t buflen,
    struct sockaddr *sap, const size_t salen)
{
    unsigned int i;

    for (i = 0; i < buflen; i++)
        if (buf[i] == ':')
-    return rpc_pton6(&init_net, buf, buflen, sap, salen);
+    return rpc_pton6(net, buf, buflen, sap, salen);

```

```

    return rpc_pton4(buf, buflen, sap, salen);
}
EXPORT_SYMBOL_GPL(rpc_pton);
@@ -340,7 +341,7 @@ size_t rpc_uaddr2sockaddr(const char *uaddr, const size_t uaddr_len,
    port = (unsigned short)((porthi << 8) | portlo);

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

    switch (sap->sa_family) {
diff --git a/net/sunrpc/svcauth_unix.c b/net/sunrpc/svcauth_unix.c
index 01153ea..2f8c426 100644
--- a/net/sunrpc/svcauth_unix.c
+++ b/net/sunrpc/svcauth_unix.c
@@ -211,7 +211,7 @@ static int ip_map_parse(struct cache_detail *cd,
    len = qword_get(&mesg, buf, mlen);
    if (len <= 0) return -EINVAL;

- if (rpc_pton(buf, len, &address.sa, sizeof(address)) == 0)
+ if (rpc_pton(&init_net, buf, len, &address.sa, sizeof(address)) == 0)
    return -EINVAL;
    switch (address.sa.sa_family) {
    case AF_INET:

```
