Subject: [PATCH v2 4/4] SUNRPC: optimize net_ns dereferencing in rpcbind registering calls
Posted by Stanislav Kinsbursky on Tue, 25 Oct 2011 13:59:48 GMT
View Forum Message <> Reply to Message

Static rpcbind registering functions can be parametrized by network namespace
pointer, calculated only once, instead of using init_net pointer (or taking it
from current when virtualization will be comleted) in many places.

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

---
 net/sunrpc/rpcb_clnt.c | 18 +++++++++---------
 1 files changed, 9 insertions(+), 9 deletions(-)

diff --git a/net/sunrpc/rpcb_clnt.c b/net/sunrpc/rpcb_clnt.c
index ef37c55..92ed74f 100644
--- a/net/sunrpc/rpcb_clnt.c
+++ b/net/sunrpc/rpcb_clnt.c
@@ -452,14 +452,14 @@ int rpcb_register(u32 prog, u32 vers, int prot, unsigned short port)
 /*
  * Fill in AF_INET family-specific arguments to register
  */
-static int rpcb_register_inet4(const struct sockaddr *sap,
+static int rpcb_register_inet4(struct sunrpc_net *sn,
+   const struct sockaddr *sap,
          struct rpc_message *msg)
 {
  const struct sockaddr_in *sin = (const struct sockaddr_in *)sap;
  struct rpcbind_args *map = msg->rpc_argp;
  unsigned short port = ntohs(sin->sin_port);
  int result;
- struct sunrpc_net *sn = net_generic(&init_net, sunrpc_net_id);

  map->r_addr = rpc_sockaddr2uaddr(sap);

@@ -480,14 +480,14 @@ static int rpcb_register_inet4(const struct sockaddr *sap,
 /*
  * Fill in AF_INET6 family-specific arguments to register
  */
-static int rpcb_register_inet6(const struct sockaddr *sap,
+static int rpcb_register_inet6(struct sunrpc_net *sn,
+   const struct sockaddr *sap,
          struct rpc_message *msg)
 {
  const struct sockaddr_in6 *sin6 = (const struct sockaddr_in6 *)sap;
  struct rpcbind_args *map = msg->rpc_argp;
  unsigned short port = ntohs(sin6->sin6_port);

```
  int result;
- struct sunrpc_net *sn = net_generic(&init_net, sunrpc_net_id);

  map->r_addr = rpc_sockaddr2uaddr(sap);

@@ -505,10 +505,10 @@ static int rpcb_register_inet6(const struct sockaddr *sap,
  return result;
 }

-static int rpcb_unregister_all_protofamilies(struct rpc_message *msg)
+static int rpcb_unregister_all_protofamilies(struct sunrpc_net *sn,
+          struct rpc_message *msg)
 {
  struct rpcbind_args *map = msg->rpc_argp;
- struct sunrpc_net *sn = net_generic(&init_net, sunrpc_net_id);

  dprintk("RPC:       unregistering [%u, %u, '%s'] with "
   "local rpcbind\n",
@@ -581,13 +581,13 @@ int rpcb_v4_register(const u32 program, const u32 version,
   return -EPROTONOSUPPORT;

  if (address == NULL)
-  return rpcb_unregister_all_protofamilies(&msg);
+  return rpcb_unregister_all_protofamilies(sn, &msg);

  switch (address->sa_family) {
  case AF_INET:
-  return rpcb_register_inet4(address, &msg);
+  return rpcb_register_inet4(sn, address, &msg);
  case AF_INET6:
-  return rpcb_register_inet6(address, &msg);
+  return rpcb_register_inet6(sn, address, &msg);
  }

  return -EAFNOSUPPORT;
```