
Subject: [PATCH v3, RESEND 07/16] sunrpc: get rpc_pipefs mount point for
rpcb_create[_local] from callers

Posted by [Kirill A. Shutemov](#) on Tue, 08 Feb 2011 18:41:58 GMT

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

Signed-off-by: Kirill A. Shutemov <kas@openvz.org>

Reviewed-by: Rob Landley <rlandley@parallels.com>

include/linux/sunrpc/clnt.h | 4 +--
net/sunrpc/rpcb_clnt.c | 22 ++++++++-----
net/sunrpc/svc.c | 34 ++++++++-----
3 files changed, 35 insertions(+), 25 deletions(-)

diff --git a/include/linux/sunrpc/clnt.h b/include/linux/sunrpc/clnt.h

index dffa...52f6142 100644

--- a/include/linux/sunrpc/clnt.h

+++ b/include/linux/sunrpc/clnt.h

@@ -135,10 +135,10 @@ void rpc_shutdown_client(struct rpc_clnt *);

void rpc_release_client(struct rpc_clnt *);

void rpc_task_release_client(struct rpc_task *);

-int rpcb_register(u32, u32, int, unsigned short);

+int rpcb_register(u32, u32, int, unsigned short, struct vfsmount *);

int rpcb_v4_register(const u32 program, const u32 version,
const struct sockaddr *address,

- const char *netid);

+ const char *netid, struct vfsmount *rpcmount);

void rpcb_getport_async(struct rpc_task *);

void rpc_call_start(struct rpc_task *);

diff --git a/net/sunrpc/rpcb_clnt.c b/net/sunrpc/rpcb_clnt.c

index b059cbe..7fddafa 100644

--- a/net/sunrpc/rpcb_clnt.c

+++ b/net/sunrpc/rpcb_clnt.c

@@ -27,7 +27,6 @@

#include <linux/sunrpc/clnt.h>

#include <linux/sunrpc/sched.h>

#include <linux/sunrpc/xprtsock.h>

-#include <linux/sunrpc/rpc_pipe_fs.h>

#ifdef RPC_DEBUG

define RPCDBG_FACILITY RPCDBG_BIND

@@ -171,7 +170,7 @@ static DEFINE_MUTEX(rpcb_create_local_mutex);

* Returns zero on success, otherwise a negative errno value

* is returned.

*/

-static int rpcb_create_local(void)

+static int rpcb_create_local(struct vfsmount *rpcmount)

```

{
    struct rpc_create_args args = {
        .net = &init_net,
@@ -183,7 +182,7 @@ static int rpcb_create_local(void)
        .version = RPCBVERS_2,
        .authflavor = RPC_AUTH_UNIX,
        .flags = RPC_CLNT_CREATE_NOPING,
-    .rpcmount = init_rpc_pipefs,
+    .rpcmount = rpcmount,
    };
    struct rpc_clnt *clnt, *clnt4;
    int result = 0;
@@ -225,7 +224,8 @@ out:
}

static struct rpc_clnt *rpcb_create(char *hostname, struct sockaddr *srvaddr,
-    size_t salen, int proto, u32 version)
+    size_t salen, int proto, u32 version,
+    struct vfsmount *rpcmount)
{
    struct rpc_create_args args = {
        .net = &init_net,
@@ -238,7 +238,7 @@ static struct rpc_clnt *rpcb_create(char *hostname, struct sockaddr
*srvaddr,
        .authflavor = RPC_AUTH_UNIX,
        .flags = (RPC_CLNT_CREATE_NOPING |
            RPC_CLNT_CREATE_NONPRIVPORT),
-    .rpcmount = init_rpc_pipefs,
+    .rpcmount = rpcmount,
    };

    switch (srvaddr->sa_family) {
@@ -305,7 +305,8 @@ static int rpcb_register_call(struct rpc_clnt *clnt, struct rpc_message
*msg)
    * IN6ADDR_ANY (ie available for all AF_INET and AF_INET6
    * addresses).
    */
-int rpcb_register(u32 prog, u32 vers, int prot, unsigned short port)
+int rpcb_register(u32 prog, u32 vers, int prot, unsigned short port,
+ struct vfsmount *rpcmount)
{
    struct rpcbind_args map = {
        .r_prog = prog,
@@ -318,7 +319,7 @@ int rpcb_register(u32 prog, u32 vers, int prot, unsigned short port)
    };
    int error;

-    error = rpcb_create_local();

```

```

+ error = rpcb_create_local(rpcmount);
  if (error)
    return error;

@@ -445,7 +446,8 @@ static int rpcb_unregister_all_protocols(struct rpc_message *msg)
  * advertises the service on all IPv4 and IPv6 addresses.
  */
  int rpcb_v4_register(const u32 program, const u32 version,
-   const struct sockaddr *address, const char *netid)
+   const struct sockaddr *address, const char *netid,
+   struct vfsmount *rpcmount)
  {
    struct rpcbind_args map = {
      .r_prog = program,
@@ -458,7 +460,7 @@ int rpcb_v4_register(const u32 program, const u32 version,
  };
  int error;

- error = rpcb_create_local();
+ error = rpcb_create_local(rpcmount);
  if (error)
    return error;
  if (rpcb_local_clnt4 == NULL)
@@ -594,7 +595,7 @@ void rpcb_getport_async(struct rpc_task *task)
  task->tk_pid, __func__, bind_version);

  rpcb_clnt = rpcb_create(clnt->cl_server, sap, salen, xprt->prot,
-   bind_version);
+   bind_version, clnt->cl_path.mnt);
  if (IS_ERR(rpcb_clnt)) {
    status = PTR_ERR(rpcb_clnt);
    dprintk("RPC: %5u %s: rpcb_create failed, error %ld\n",
diff --git a/net/sunrpc/svc.c b/net/sunrpc/svc.c
index 63abe2b..031f2d4 100644
--- a/net/sunrpc/svc.c
+++ b/net/sunrpc/svc.c
@@ -739,7 +739,8 @@ EXPORT_SYMBOL_GPL(svc_exit_thread);
  */
  static int __svc_rpcb_register4(const u32 program, const u32 version,
    const unsigned short protocol,
-   const unsigned short port)
+   const unsigned short port,
+   struct vfsmount *rpcmount)
  {
    const struct sockaddr_in sin = {
      .sin_family = AF_INET,
@@ -761,14 +762,16 @@ static int __svc_rpcb_register4(const u32 program, const u32 version,
  }

```

```

    error = rpcb_v4_register(program, version,
-   (const struct sockaddr *)&sin, netid);
+   (const struct sockaddr *)&sin, netid,
+   rpcmount);

/*
 * User space didn't support rpcbind v4, so retry this
 * registration request with the legacy rpcbind v2 protocol.
 */
if (error == -EPROTONOSUPPORT)
-   error = rpcb_register(program, version, protocol, port);
+   error = rpcb_register(program, version, protocol, port,
+   rpcmount);

return error;
}
@@ -786,7 +789,8 @@ static int __svc_rpcb_register4(const u32 program, const u32 version,
*/
static int __svc_rpcb_register6(const u32 program, const u32 version,
    const unsigned short protocol,
-   const unsigned short port)
+   const unsigned short port,
+   struct vfsmount *rpcmount)
{
    const struct sockaddr_in6 sin6 = {
        .sin6_family = AF_INET6,
@@ -808,7 +812,8 @@ static int __svc_rpcb_register6(const u32 program, const u32 version,
    }

    error = rpcb_v4_register(program, version,
-   (const struct sockaddr *)&sin6, netid);
+   (const struct sockaddr *)&sin6, netid,
+   rpcmount);

/*
 * User space didn't support rpcbind version 4, so we won't
@@ -831,19 +836,20 @@ static int __svc_register(const char *progname,
    const u32 program, const u32 version,
    const int family,
    const unsigned short protocol,
-   const unsigned short port)
+   const unsigned short port,
+   struct vfsmount *rpcmount)
{
    int error = -EAFNOSUPPORT;

    switch (family) {

```

```

case PF_INET:
    error = __svc_rpcb_register4(program, version,
-    protocol, port);
+    protocol, port, rpcmount);
    break;
#ifdef CONFIG_IPV6 || defined(CONFIG_IPV6_MODULE)
case PF_INET6:
    error = __svc_rpcb_register6(program, version,
-    protocol, port);
+    protocol, port, rpcmount);
#endif /* defined(CONFIG_IPV6) || defined(CONFIG_IPV6_MODULE) */
}

@@ -889,7 +895,8 @@ int svc_register(const struct svc_serv *serv, const int family,
    continue;

    error = __svc_register(progp->pg_name, prog->pg_prog,
-    i, family, proto, port);
+    i, family, proto, port,
+    serv->sv_rpcmount);
    if (error < 0)
        break;
}
@@ -906,18 +913,18 @@ int svc_register(const struct svc_serv *serv, const int family,
    * in this case to clear all existing entries for [program, version].
    */
static void __svc_unregister(const u32 program, const u32 version,
-    const char *progrname)
+    const char *progrname, struct vfsmount *rpcmount)
{
    int error;

-    error = rpcb_v4_register(program, version, NULL, "");
+    error = rpcb_v4_register(program, version, NULL, "", rpcmount);

    /*
     * User space didn't support rpcbind v4, so retry this
     * request with the legacy rpcbind v2 protocol.
     */
    if (error == -EPROTONOSUPPORT)
-    error = rpcb_register(program, version, 0, 0);
+    error = rpcb_register(program, version, 0, 0, rpcmount);

    dprintk("svc: %s(%sv%u), error %d\n",
        __func__, progrname, version, error);
@@ -946,7 +953,8 @@ static void svc_unregister(const struct svc_serv *serv)
    if (progp->pg_vers[i]->vs_hidden)
        continue;

```

```
- __svc_unregister(progp->pg_prog, i, progp->pg_name);
+ __svc_unregister(progp->pg_prog, i, progp->pg_name,
+   serv->sv_rpcmount);
}
```

--

1.7.4

Containers mailing list

Containers@lists.linux-foundation.org

<https://lists.linux-foundation.org/mailman/listinfo/containers>
