

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-12/msg02612.html>

- *From:* Trond Myklebust <trond.myklebust@xxxxxxxxxx>
 - *Date:* Fri, 08 Dec 2006 12:00:02 -0500
-

On Fri, 2006-12-08 at 23:01 +1100, NeilBrown wrote:

From: Chuck Lever <chuck.lever@xxxxxxxxxx>
Sometimes we need to create an RPC service but not register it with the local portmapper. NFSv4 delegation callback, for example.

Change the `svc_makesock()` API to allow optionally creating temporary or permanent sockets, optionally registering with the local portmapper, and make it return the ephemeral port of the new socket.

NAK. This one is still buggy.

The NFSv4 callback server should NOT be registering its listening socket on the RPC server 'temporary' list.

Trond

Signed-off-by: Chuck Lever <chuck.lever@xxxxxxxxxx>
Cc: Aurelien Charbon <aurelien.charbon@xxxxxxxxxx>
Signed-off-by: Neil Brown <neilb@xxxxxxxx>

Diffstat output

```
./fs/lockd/svc.c | 26 ++++++-----  
./fs/nfs/callback.c | 20 ++++++-----  
./fs/nfsd/nfssvc.c | 6 ++++--  
./include/linux/sunrpc/svcsock.h | 2 +-  
./net/sunrpc/svcsock.c | 6 ++++--  
5 files changed, 34 insertions(+), 26 deletions(-)
```

```
diff .prev/fs/lockd/svc.c ./fs/lockd/svc.c  
--- .prev/fs/lockd/svc.c 2006-12-08 13:36:33.000000000 +1100  
+++ ./fs/lockd/svc.c 2006-12-08 13:36:33.000000000 +1100  
@@ -223,23 +223,29 @@ static int find_socket(struct svc_serv *
```

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

```
return found;
}

+/*
+ * Make any sockets that are needed but not present.
+ * If nlm_udpport or nlm_tcpport were set as module
+ * options, make those sockets unconditionally
+ */
static int make_socks(struct svc_serv *serv, int proto)
{
- /* Make any sockets that are needed but not present.
- * If nlm_udpport or nlm_tcpport were set as module
- * options, make those sockets unconditionally
- */
- static int warned;
+ static int warned;
int err = 0;
+
if (proto == IPPROTO_UDP || nlm_udpport)
if (!find_socket(serv, IPPROTO_UDP))
- err = svc_makesock(serv, IPPROTO_UDP, nlm_udpport);
- if (err == 0 && (proto == IPPROTO_TCP || nlm_tcpport))
+ err = svc_makesock(serv, IPPROTO_UDP, nlm_udpport,
+ SVC_SOCKET_DEFAULTS);
+ if (err >= 0 && (proto == IPPROTO_TCP || nlm_tcpport))
if (!find_socket(serv, IPPROTO_TCP))
- err= svc_makesock(serv, IPPROTO_TCP, nlm_tcpport);
- if (!err)
+ err = svc_makesock(serv, IPPROTO_TCP, nlm_tcpport,
+ SVC_SOCKET_DEFAULTS);
+
+ if (err >= 0) {
warned = 0;
- else if (warned++ == 0)
+ err = 0;
+ } else if (warned++ == 0)
printk(KERN_WARNING
"lockd_up: makesock failed, error=%d\n", err);
return err;

diff .prev/fs/nfs/callback.c ./fs/nfs/callback.c
--- .prev/fs/nfs/callback.c 2006-12-08 13:36:33.000000000 +1100
+++ ./fs/nfs/callback.c 2006-12-08 13:36:33.000000000 +1100
@@ -106,7 +106,6 @@ static void nfs_callback_svc(struct svc_
int nfs_callback_up(void)
{
struct svc_serv *serv;
- struct svc_sock *svsk;
int ret = 0;

lock_kernel();
```

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

```
@@ -119,17 +118,14 @@ int nfs_callback_up(void)
ret = -ENOMEM;
if (!serv)
goto out_err;
- /* FIXME: We don't want to register this socket with the portmapper */
- ret = svc_makesock(serv, IPPROTO_TCP, nfs_callback_set_tcpport);
- if (ret < 0)
+
+ ret = svc_makesock(serv, IPPROTO_TCP, nfs_callback_set_tcpport,
+ (SVC_SOCKET_ANONYMOUS | SVC_SOCKET_TEMPORARY));
+ if (ret <= 0)
goto out_destroy;
- if (!list_empty(&serv->sv_permsocks)) {
- svsk = list_entry(serv->sv_permsocks.next,
- struct svc_sock, sk_list);
- nfs_callback_tcpport = ntohs(inet_sk(svsk->sk_sk)->sport);
- dprintk("Callback port = 0x%x\n", nfs_callback_tcpport);
- } else
- BUG();
+ nfs_callback_tcpport = ret;
+ dprintk("Callback port = 0x%x\n", nfs_callback_tcpport);
+
ret = svc_create_thread(nfs_callback_svc, serv);
if (ret < 0)
goto out_destroy;
@@ -140,6 +136,8 @@ out:
unlock_kernel();
return ret;
out_destroy:
+ dprintk("Couldn't create callback socket or server thread; err = %d\n",
+ ret);
svc_destroy(serv);
out_err:
nfs_callback_info.users--;

diff .prev/fs/nfsd/nfssvc.c ./fs/nfsd/nfssvc.c
--- .prev/fs/nfsd/nfssvc.c 2006-12-08 13:36:33.000000000 +1100
+++ ./fs/nfsd/nfssvc.c 2006-12-08 13:36:33.000000000 +1100
@@ -235,7 +235,8 @@ static int nfsd_init_socks(int port)

error = lockd_up(IPPROTO_UDP);
if (error >= 0) {
- error = svc_makesock(nfsd_serv, IPPROTO_UDP, port);
+ error = svc_makesock(nfsd_serv, IPPROTO_UDP, port,
+ SVC_SOCKET_DEFAULTS);
if (error < 0)
lockd_down();
}
@@ -245,7 +246,8 @@ static int nfsd_init_socks(int port)
#ifdef CONFIG_NFSD_TCP
error = lockd_up(IPPROTO_TCP);
```

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

```
if (error >= 0) {
- error = svc_makesock(nfsd_serv, IPPROTO_TCP, port);
+ error = svc_makesock(nfsd_serv, IPPROTO_TCP, port,
+ SVC SOCK_DEFAULTS);
if (error < 0)
lockd_down();
}
```

```
diff .prev/include/linux/sunrpc/svcsock.h ./include/linux/sunrpc/svcsock.h
--- .prev/include/linux/sunrpc/svcsock.h 2006-12-08 13:35:43.000000000 +1100
+++ ./include/linux/sunrpc/svcsock.h 2006-12-08 13:36:33.000000000 +1100
@@ -62,7 +62,7 @@ struct svc_sock {
/*
* Function prototypes.
*/
-int svc_makesock(struct svc_serv *, int, unsigned short);
+int svc_makesock(struct svc_serv *, int, unsigned short, int flags);
void svc_delete_socket(struct svc_sock *);
int svc_recv(struct svc_rqst *, long);
int svc_send(struct svc_rqst *);
```

```
diff .prev/net/sunrpc/svcsock.c ./net/sunrpc/svcsock.c
--- .prev/net/sunrpc/svcsock.c 2006-12-08 13:35:43.000000000 +1100
+++ ./net/sunrpc/svcsock.c 2006-12-08 13:36:33.000000000 +1100
@@ -1659,9 +1659,11 @@ svc_delete_socket(struct svc_sock *svsk)
* @serv: RPC server structure
* @protocol: transport protocol to use
* @port: port to use
+ * @flags: requested socket characteristics
*
*/
-int svc_makesock(struct svc_serv *serv, int protocol, unsigned short port)
+int svc_makesock(struct svc_serv *serv, int protocol, unsigned short port,
+ int flags)
{
struct sockaddr_in sin = {
.sin_family = AF_INET,
@@ -1670,7 +1672,7 @@ int svc_makesock(struct svc_serv *serv,
};

dprintk("svc: creating socket proto = %d\n", protocol);
- return svc_create_socket(serv, protocol, &sin, SVC SOCK_DEFAULTS);
+ return svc_create_socket(serv, protocol, &sin, flags);
}

/*
```

Take Surveys. Earn Cash. Influence the Future of IT
Join SourceForge.net's Techsay panel and you'll get the chance to share your
opinions on IT & business topics through brief surveys – and earn cash

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper

<http://www.techsay.com/default.php?page=join.php&p=sourceforge&CID=DEVDEV>

NFS maillist – NFS@xxxxxxxxxxxxxxxxxxxxxxxxxxxx

<https://lists.sourceforge.net/lists/listinfo/nfs>

–

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>

Re: [NFS] [PATCH 002 of 13] knfsd: SUNRPC: allow creating an RPC service without registering with portmapper