Subject: RE: [PATCH v2 2/4] NFS: release per-net clients lock before calling PipeFS dentries creation
Posted by David Laight on Mon, 27 Feb 2012 15:59:23 GMT
View Forum Message <> Reply to Message

```
>   spin_lock(&nn->nfs_client_lock);
> - list_for_each_entry(clp, &nn->nfs_client_list, cl_share_link) {
> + list_for_each_entry_safe(clp, tmp, &nn->nfs_client_list,
cl_share_link) {
>    if (clp->rpc_ops != &nfs_v4_clientops)
>     continue;
> +  atomic_inc(&clp->cl_count);
> +  spin_unlock(&nn->nfs_client_lock);
>    error = __rpc_pipefs_event(clp, event, sb);
> +  nfs_put_client(clp);
>    if (error)
>     break;
> +  spin_lock(&nn->nfs_client_lock);
>   }
>   spin_unlock(&nn->nfs_client_lock);
>   return error;
```

The locking doesn't look right if the loop breaks on error.
(Same applied to patch v2 1/4)

Although list_fo_each_entry_safe() allows the current entry
to be freed, I don't believe it allows the 'next' to be freed.
I doubt there is protection against that happening.

Do you need to use an atomic_inc() for cl_count.
I'd guess the nfs_client_lock is usually held?

David