Subject: Re: [PATCH] NFSd: set nfsd_serv to NULL after service destruction
Posted by bfields on Wed, 23 May 2012 21:52:10 GMT
View Forum Message <> Reply to Message

On Fri, May 18, 2012 at 07:26:02PM +0400, Stanislav Kinsbursky wrote:
> Otherwise we will get NULL pointer dereference after
> svc_shutdown_net->sv_shutdown (nfsd_last_thread) call.
> This looks safe, because all such operations (svc_destroy) are performed with
> nfsd_mutex being held.
>
> Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>
> ---
> fs/nfsd/nfsctl.c | 4 ++--
> fs/nfsd/nfsd.h | 6 ++++++
> fs/nfsd/nfssvc.c | 10 ++++++------
> 3 files changed, 12 insertions(+), 8 deletions(-)
>
> diff --git a/fs/nfsd/nfsctl.c b/fs/nfsd/nfsctl.c
> index c55298e..82cca1e 100644
> --- a/fs/nfsd/nfsctl.c
> +++ b/fs/nfsd/nfsctl.c
> @@ -675,7 +675,7 @@ static ssize_t __write_ports_addfd(char *buf)
>   if (err < 0) {
>   if (nfsd_serv->sv_nrthreads == 1)
>   svc_shutdown_net(nfsd_serv, net);
> - svc_destroy(nfsd_serv);
> + nfsd_destroy();

The server could still be running here. We don't want to set nfsd_serv
to NULL.

--b.

>   return err;
>   }
>
> @@ -746,7 +746,7 @@ out_close:
> out_err:
>   if (nfsd_serv->sv_nrthreads == 1)
>   svc_shutdown_net(nfsd_serv, net);
> - svc_destroy(nfsd_serv);
> + nfsd_destroy();
>   return err;
> }
>
> diff --git a/fs/nfsd/nfsd.h b/fs/nfsd/nfsd.h
> index 1671429..35db369 100644
> --- a/fs/nfsd/nfsd.h

```
> +++ b/fs/nfsd/nfsd.h
> @@ -73,6 +73,12 @@ int  nfsd_nrpools(void);
>  int  nfsd_get_nrthreads(int n, int *);
>  int  nfsd_set_nrthreads(int n, int *);
>
> +static inline void nfsd_destroy(void)
> +{
> + svc_destroy(nfsd_serv);
> + nfsd_serv = NULL;
> +}
> +
>  #if defined(CONFIG_NFSD_V2_ACL) || defined(CONFIG_NFSD_V3_ACL)
>  #ifdef CONFIG_NFSD_V2_ACL
>  extern struct svc_version nfsd_acl_version2;
> diff --git a/fs/nfsd/nfssvc.c b/fs/nfsd/nfssvc.c
> index ee709fc..ab89fe0 100644
> --- a/fs/nfsd/nfssvc.c
> +++ b/fs/nfsd/nfssvc.c
> @@ -254,8 +254,6 @@ static void nfsd_shutdown(void)
>
>  static void nfsd_last_thread(struct svc_serv *serv, struct net *net)
>  {
> - /* When last nfsd thread exits we need to do some clean-up */
> - nfsd_serv = NULL;
>   nfsd_shutdown();
>
>   svc_rpcb_cleanup(serv, net);
> @@ -348,7 +346,7 @@ int nfsd_create_serv(void)
>
>   error = svc_bind(nfsd_serv, current->nsproxy->net_ns);
>   if (error < 0) {
> -  svc_destroy(nfsd_serv);
> +  nfsd_destroy();
>    return error;
>   }
>
> @@ -430,7 +428,7 @@ int nfsd_set_nrthreads(int n, int *nthreads)
>
>   if (nfsd_serv->sv_nrthreads == 1)
>    svc_shutdown_net(nfsd_serv, net);
> - svc_destroy(nfsd_serv);
> + nfsd_destroy();
>
>   return err;
>  }
> @@ -480,7 +478,7 @@ out_shutdown:
>  out_destroy:
>   if (nfsd_serv->sv_nrthreads == 1)
```

```
>    svc_shutdown_net(nfsd_serv, net);
> - svc_destroy(nfsd_serv);  /* Release server */
> + nfsd_destroy();  /* Release server */
>  out:
>   mutex_unlock(&nfsd_mutex);
>   return error;
> @@ -684,7 +682,7 @@ int nfsd_pool_stats_release(struct inode *inode, struct file *file)
>   /* this function really, really should have been called svc_put() */
>   if (nfsd_serv->sv_nrthreads == 1)
>    svc_shutdown_net(nfsd_serv, net);
> - svc_destroy(nfsd_serv);
> + nfsd_destroy();
>   mutex_unlock(&nfsd_mutex);
>   return ret;
>  }
>
```