Subject: Re: [PATCH] [RFC] nfsd: fix possible dereference of static NULL nfsd_serv pointer
Posted by bfields on Fri, 06 Jul 2012 17:26:30 GMT
View Forum Message <> Reply to Message

On Fri, Jul 06, 2012 at 05:45:56PM +0400, Stanislav Kinsbursky wrote:
> This is a bug fix for 3.5 kernel.
> In case on NFSd service start failure svc_shutdown_net() will call svc_destroy
> callback and zeroize global nfsd_serv pointer, this in turn will lead to Oops
> in svc_destroy().
>
> This patch is marked as RFC, because to many lines were changed. It can be
> easely simplified if requested.
> Moreover, NFSd service shutdown is going to be converted into csomething on
> per-net basis.

Doesn't this leave error paths in e.g. __write_ports_addfd() and
__write_ports_addxprt() unfixed?

I'm inclined to just submit your original fix (split up as in the last
version I sent) for 3.5 if you don't object.

--b.

>
> Signed-off-by: Stanislav Kinsbursky <skinsbursky@parallels.com>
> ---
> fs/nfsd/nfssvc.c | 14 ++++++++------
> 1 files changed, 8 insertions(+), 6 deletions(-)
>
> diff --git a/fs/nfsd/nfssvc.c b/fs/nfsd/nfssvc.c
> index ee709fc..526a4aa 100644
> --- a/fs/nfsd/nfssvc.c
> +++ b/fs/nfsd/nfssvc.c
> @@ -446,6 +446,7 @@ nfsd_svc(unsigned short port, int nrservs)
>   int error;
>   bool nfsd_up_before;
>   struct net *net = &init_net;
> + struct svc_serv *serv = nfsd_serv;
>
>   mutex_lock(&nfsd_mutex);
>   dprintk("nfsd: creating service\n");
> @@ -454,7 +455,7 @@ nfsd_svc(unsigned short port, int nrservs)
>   if (nrservs > NFSD_MAXSERVS)
>   nrservs = NFSD_MAXSERVS;
>   error = 0;
> - if (nrservs == 0 && nfsd_serv == NULL)
> + if (nrservs == 0 && serv == NULL)

```
>    goto out;
>
>   error = nfsd_create_serv();
> @@ -464,23 +465,24 @@ nfsd_svc(unsigned short port, int nrservs)
>   nfsd_up_before = nfsd_up;
>
>   error = nfsd_startup(port, nrservs);
> + error = -EINVAL;
>   if (error)
>    goto out_destroy;
> - error = svc_set_num_threads(nfsd_serv, NULL, nrservs);
> + error = svc_set_num_threads(serv, NULL, nrservs);
>   if (error)
>    goto out_shutdown;
>   /* We are holding a reference to nfsd_serv which
>    * we don't want to count in the return value,
>    * so subtract 1
>    */
> - error = nfsd_serv->sv_nrthreads - 1;
> + error = serv->sv_nrthreads - 1;
>  out_shutdown:
>   if (error < 0 && !nfsd_up_before)
>    nfsd_shutdown();
>  out_destroy:
> - if (nfsd_serv->sv_nrthreads == 1)
> -  svc_shutdown_net(nfsd_serv, net);
> - svc_destroy(nfsd_serv);  /* Release server */
> + if (serv->sv_nrthreads == 1)
> +  svc_shutdown_net(serv, net);
> + svc_destroy(serv);  /* Release server */
>  out:
>   mutex_unlock(&nfsd_mutex);
>   return error;
>
```