

Re: rmalloc()/rtfree() problems on CURRENT

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/net/2004-02/0259.html>

From: Gleb Smirnoff (*glebius_at_cell.sick.ru*)

Date: 02/24/04

Date: Tue, 24 Feb 2004 13:21:13 +0300
To: Andre Oppermann <andre@freebsd.org>

Dear Andre,

first, thanks for your feedback.

On Tue, Feb 24, 2004 at 11:14:59AM +0100, Andre Oppermann wrote:

A> Most of the time, if you don't need a cloned route allocated, you can
A> just use `rmalloc_ign(&ro, RTF_CLONING)` to just get a reference to the
A> existing `rtentry`. Not allocating a new cloned route saves work and
A> time for the `rmalloc` code.

Thanks for this hint. So if using `rmalloc_ign()` I do not need to call `rtfree()`?
Why does this situation is not clear in manpage? :)

```
A> > if (ro.ro_rt != NULL) {  
A> > struct rtentry *rt = ro.ro_rt;  
A> >  
A> > /* here some read-only things are done with rt */  
A>  
A> Maybe you do some nasty things to rt here?
```

Really, I don't :) Exact code is given at end of the mail.

```
A> > rtfree(ro.ro_rt);  
A> > }  
A> >  
A> > But on CURRENT rtfree() causes panic. Here is backtrace:  
A> ...  
A> > Does this mean that something is broken in CURRENT or I am doing something wrong?  
A>  
A> rmalloc() and rtfree() are used quite often in the network code and  
A> otherwise work fine. The odds are high that you are doing something  
A> wrong.
```

Here is exact code:

```
bzero((caddr_t)&ro, sizeof(ro));
```

freebsd-net: Re: rttalloc()/rtfree() problems on CURRENT

```
sin = (struct sockaddr_in *)&ro.ro_dst;
sin->sin_len = sizeof(*sin);
sin->sin_family = AF_INET;
sin->sin_addr = fle->r.r_dst;
rtalloc(&ro);
if (ro.ro_rt != NULL) {
    struct rtable *rt = ro.ro_rt;

    /* This is cloned route, use its parent */
    if (ro.ro_rt->rt_flags & RTF_WASCLONED &&
        ro.ro_rt->rt_parent)
        rt = ro.ro_rt->rt_parent;

    fle->o_ifx = rt->rt_ifp->if_index;

    if (rt->rt_flags & RTF_GATEWAY &&
        rt->rt_gateway->sa_family == AF_INET)
        fle->next_hop =
            ((struct sockaddr_in *)(rt->rt_gateway))->sin_addr;

    if (rt_mask(rt))
        fle->dst_mask =
            bit_count(((struct sockaddr_in *)rt_mask(rt))->sin_addr.s_a
ddr);
    else if (rt->rt_flags & RTF_HOST)
        /* Give up. We can't determine mask :( */
        fle->dst_mask = 32;

    rtfree(ro.ro_rt);
}

--
Totus tuus, Glebius.
GLEBIUS-RIPN GLEB-RIPE

-----
freebsd-net@freebsd.org mailing list
http://lists.freebsd.org/mailman/listinfo/freebsd-net
To unsubscribe, send any mail to "freebsd-net-unsubscribe@freebsd.org"
```