

---

Subject: Re: Broken "Make ip6\_frags per namespace" patch  
Posted by [Alexey Dobriyan](#) on Fri, 18 Jan 2008 13:03:05 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

On Thu, Jan 17, 2008 at 01:01:11PM +0100, Daniel Lezcano wrote:

> Alexey Dobriyan wrote:

> > On Thu, Jan 17, 2008 at 11:40:42AM +0100, Daniel Lezcano wrote:

> > > Alexey Dobriyan wrote:

> > > > commit c064c4811b3e87ff8202f5a966ff4eea0bc54575

> > > > Author: Daniel Lezcano <dlezcano@fr.ibm.com>

> > > > Date: Thu Jan 10 02:56:03 2008 -0800

> > > >

> > > > [NETNS][IPV6]: Make ip6\_frags per namespace.

> > > >

> > > > The ip6\_frags is moved to the network namespace structure. Because

> > > > there can be multiple instances of the network namespaces, and the

> > > > ip6\_frags is no longer a global static variable, a helper function

> > > > has

> > > > been added to facilitate the initialization of the variables.

> > > >

> > > > Until the ipv6 protocol is not per namespace, the variables are

> > > > accessed relatively from the initial network namespace.

> > > > --- a/include/net/netns/ipv6.h

> > > > +++ b/include/net/netns/ipv6.h

> > > > @@ -11,6 +13,7 @@ struct netns\_sysctl\_ipv6 {

> > > > #ifdef CONFIG\_SYSCTL

> > > > struct ctl\_table\_header \*table;

> > > > #endif

> > > > + struct inet\_frags\_ctl frags;

> > > > --- a/net/ipv6/reassembly.c

> > > > +++ b/net/ipv6/reassembly.c

> > > > @@ -632,6 +625,11 @@ static struct inet6\_protocol frag\_protocol =

> > > > .flags = INET6\_PROTO\_NOPOLICY,

> > > > };

> > > >

> > > > +void ipv6\_frag\_sysctl\_init(struct net \*net)

> > > > +{

> > > > + ipv6\_frags\_ctl = &net->ipv6.sysctl.frags;

> > > > +}

> > > \_This\_ can't work. ip6frags is only one and ->ctl pointer is flipped

> > > onto per-netns data. Changelog is also misleading: ip6\_frags\_ctl is

> > > moved to netns not all ip6\_frags.

> > >

> > > Oopsing place below -- f->ctl dereference in preparation of mod\_timer()

> > > call.

> > >

> > >

> > >

```

> >>>BUG: unable to handle kernel paging request at virtual address f5da8fc8
> >>>printing eip: c11d868a *pdpt = 0000000000003001 *pde = 0000000001728067
> >>>*pte = 0000000035da8000 Oops: 0000 [#1] PREEMPT SMP DEBUG_PAGEALLOC
> >>>Modules linked in: ebt_ip ebt_dnat ebt_arpreply ebt_arp ebt_among
> >>>ebtable_nat ip6t_REJECT ip6table_filter ip6_tables ebtable_filter
> >>>ebtable_broute ebt_802_3 ebtables des_generic nf_conntrack_netbios_ns
> >>>nf_conntrack_ipv4 xt_state nf_conntrack xt_tcpudp ipt_REJECT
> >>>iptable_filter ip_tables deflate zlib_deflate zlib_inflate cryptomgr
> >>>crypto_hash cpufreq_stats cpufreq_ondemand cdrom cbc bridge llc
> >>>blkcipher crypto_algapi arpt_mangle arptable_filter arp_tables x_tables
> >>>ah6 af_packet ipv6
> >>>
> >>>Pid: 0, comm: swapper Not tainted (2.6.24-rc7-net-2.6.25-nf-sysfs-n #30)
> >>>EIP: 0060:[<c11d868a>] EFLAGS: 00010246 CPU: 1
> >>>EIP is at inet_frag_secret_rebuild+0xaa/0xd0
> >>>EAX: f5da8fbc EBX: 00000000 ECX: c1310000 EDX: 00000100
> >>>ESI: f7cba000 EDI: f898f7a0 EBP: 00000040 ESP: c1310f90
> >>>DS: 007b ES: 007b FS: 00d8 GS: 0000 SS: 0068
> >>>Process swapper (pid: 0, ti=c1310000 task=f7c9a580 task.ti=f7c9b000)
> >>>Stack: f898f7a8 f898f8a8 000ddcbd f898f7a0 f7cba000 c1310fc4 00000100
> >>>c1026d60 00000002 00000001 c1191183 c4779ddc c11d85e0 f898c860
> >>> f898c860 c12c4a88 00000001 c1308da0 0000000a c1023477 00000001
> >>> c130b640 c130b640 f7c9bf34 Call Trace:
> >>>[<c1026d60>] run_timer_softirq+0x120/0x190
> >>>[<c1191183>] net_rx_action+0x53/0x220
> >>>[<c11d85e0>] inet_frag_secret_rebuild+0x0/0xd0
> >>>[<c1023477>] __do_softirq+0x87/0x100
> >>>[<c10059cf>] do_softirq+0xaf/0x110
> >>>[<c10233e3>] irq_exit+0x83/0x90
> >>>[<c1010ce7>] smp_apic_timer_interrupt+0x57/0x90
> >>>[<c10036e1>] apic_timer_interrupt+0x29/0x38
> >>>[<c10036eb>] apic_timer_interrupt+0x33/0x38
> >>>[<c1001460>] default_idle+0x0/0x60
> >>>[<c10014a0>] default_idle+0x40/0x60
> >>>[<c1000ea3>] cpu_idle+0x73/0xb0
> >>>=====
> >>>Code: 8b 10 85 d2 89 13 74 03 89 5a 04 89 18 89 43 04 85 f6 89 f3 75 bb
> >>>45 83 fd 40 75 a5 8b 44 24 04 e8 4c 3f 01 00 8b 87 50 01 00 00 <8b> 50
> >>>0c 01 54 24 08 8d 87 38 01 00 00 8b 54 24 08 83 c4 0c 5b EIP:
> >>>[<c11d868a>] inet_frag_secret_rebuild+0xaa/0xd0 SS:ESP 0068:c1310f90
> >>>Kernel panic - not syncing: Fatal exception in interrupt
> >>Hi Alexey,
> >>
> >>does it happen after unsharing the network ?
> >
> >Yep. clone(CLONE_NEWNET) in a loop and sooner or later you'll see this.
>
> Thanks.

```

>

> The network namespace is not yet complete, this is normal that you have  
> not ip6\_frag per namespace. Pavel is doing that.

it would be nice to not oops meanwhile :)

> Perhaps, I should disable ip6\_frag\_sysctl\_init when not in the init\_net  
> and enable it again when Pavel send its fragment patchset ?

Pavel told me (he won't be in office for couple of days) that this  
place was the biggest PITA during netnsization of fragments.  
FWIW, I'm currently running with

```
- ip6_fragments.ctl = &net->ipv6.sysctl.fragments;  
+ ip6_fragments.ctl = &init_net.ipv6.sysctl.fragments;
```

---