Subject: 2.6.20-lxc8: kernel panic with af_unix as module
Posted by Pierre Peiffer on Thu, 03 May 2007 12:13:44 GMT
View Forum Message <> Reply to Message

Hi,

I'm trying to test this kernel with network namespace compiled. But I have a kernel panic (see below) during the boot when loading the module unix.ko (in af_unix_init).

After some search, it seems that specifying a particular section for some variables is incompatible with using them in a module.

In my case, when registering per_net__unix_root_table (in function unix_sysctl_register in file net/unix/sysctl_net_unix.c), it tries to access the child (which is per_net__unix_net_table) at the address 0x00000080 which is the address of this symbol in the section .data.pernet of the module object, but not a virtual address.

When compiling this in the kernel, there is no problem...
As I'm not familiar with the use of some sections in ELF format, I don't know how to correct this....

By the way, just for my knowledge, what is the advantage of specifying a particular section (.data.pernet in this case) for some variables ?

Thanks.

Here is the panic:
INIT: version 2.86 booting
NET: Registered protocol family 1
BUG: unable to handle kernel NULL pointer dereference at virtual address 00000080
  printing eip:
c0123bae
*pde = 00000000
Oops: 0000 [#1]
PREEMPT SMP
Modules linked in: unix ide_disk ide_core ahci libata sd_mod scsi_mod ehci_hcd uhci_hcd usbcore
CPU: 3
EIP: 0060:[<c0123bae>]   Not tainted VLI
EFLAGS: 00010282   (2.6.20-lxc8 #4)
EIP is at sysctl_net_table_fixup+0x7e/0xa0
eax: 37a65e00   ebx: 000000a8   ecx: ffffffff   edx: 00000080
esi: 00000080   edi: 37a65e00   ebp: f7ab1f00   esp: f7ab1ee0
ds: 007b   es: 007b   ss: 0068
Process modprobe (pid: 404, ti=f7ab0000 task=c232aa90 task.ti=f7ab0000)
Stack: f7e2dd08 37a65e00 37a65e00 f7ab1f00 c011d61b f7e2dd08 00000080 37a65e00

f7ab1f28 c0123bab c0329260 00000080 00000000 00000000 00000000 f7e2c0a0
37a65e00 c03c62a0 f7ab1f40 c01241cd f7e2dce0 c03c7f30 37a65e00 08060588
Call Trace:
[<c010414a>] show_trace_log_lvl+0x1a/0x30
[<c0104211>] show_stack_log_lvl+0xb1/0xe0
[<c0104409>] show_registers+0x1c9/0x310
[<c010466c>] die+0x11c/0x250
[<c02c4213>] do_page_fault+0x2c3/0x5c0
[<c02c27c4>] error_code+0x7c/0x84
[<c0123bab>] sysctl_net_table_fixup+0x7b/0xa0
[<c01241cd>] register_net_sysctl_table+0x2d/0x70
[<f885b360>] unix_sysctl_register+0x20/0x30 [unix]
[<f885859c>] unix_net_init+0x2c/0x40 [unix]
[<c0271024>] register_pernet_operations+0x34/0xc0
[<c027129c>] register_pernet_subsys+0x1c/0x30
[<f882b04c>] af_unix_init+0x4c/0x54 [unix]
[<c013bbc1>] sys_init_module+0x91/0x160
[<c010313c>] syscall_call+0x7/0xb
=======================
Code: f8 89 43 e0 89 f8 e8 d2 ee ff ff 8b 53 fc 89 43 f8 89 f8 e8 c5 ee ff ff 85
f6 89 43 fc 74 09 89 f2 89 f8 e8 85 ff ff ff 83 c3 28 <8b>
43 d8 85 c0 75 8b 8b 73 dc 85 f6 75 84 83 c4 14 5b 5e 5f 5d
EIP: [<c0123bae>] sysctl_net_table_fixup+0x7e/0xa0 SS:ESP 0068:f7ab1ee0
Welcome to Fedora Core
Press 'I' to enter interactive startup.

--
Pierre

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

---

Subject: Re: 2.6.20-lxc8: kernel panic with af_unix as module
Posted by ebiederm on Thu, 03 May 2007 23:32:03 GMT
View Forum Message <> Reply to Message

Pierre Peiffer <pierre.peiffer@bull.net> writes:

> Hi,
>
> I'm trying to test this kernel with network namespace compiled. But I have a
> kernel panic (see below) during the boot when loading the module unix.ko (in
> af_unix_init).
>
> After some search, it seems that specifying a particular section for some

> variables is incompatible with using them in a module.

Bother. I thought I had that all implemented properly but it looks like
I have missed something.

> In my case, when registering per_net__unix_root_table (in function
> unix_sysctl_register in file net/unix/sysctl_net_unix.c), it tries to access the
> child (which is per_net__unix_net_table) at the address 0x00000080 which is the
> address of this symbol in the section .data.pernet of the module object, but not
> a virtual address.

Hmm. It should be a global offset in the .data.pernet section. If it
is just for that module I clearly have a problem.

> When compiling this in the kernel, there is no problem...
> As I'm not familiar with the use of some sections in ELF format, I don't know
> how to correct this....
>
> By the way, just for my knowledge, what is the advantage of specifying a
> particular section (.data.pernet in this case) for some variables ?

So you have one instance for each network namespace. Currently I think
I'm using the concept a little too much requiring a large page
allocation for all of the network namespace state. But otherwise it
seems to work fairly well.

Thanks. I will go back and look but I don't plan on back porting
anything for 2.6.20. I'm lazy and do not have enough hours in the
day. :)

Eric

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers

---

Subject: Re: 2.6.20-lxc8: kernel panic with af_unix as module
Posted by Pierre Peiffer on Fri, 04 May 2007 08:26:24 GMT
View Forum Message <> Reply to Message

>
>> In my case, when registering per_net__unix_root_table (in function
>> unix_sysctl_register in file net/unix/sysctl_net_unix.c), it tries to access the
>> child (which is per_net__unix_net_table) at the address 0x00000080 which is the
>> address of this symbol in the section .data.pernet of the module object, but not

>> a virtual address.
>
> Hmm. It should be a global offset in the .data.pernet section.

Yes, indeed, it is.

If I do some printk (in unix_sysctl_register) just before the panic, I get:

per_net(unix_root_table, net) = 0xf7e2dce0
per_net(unix_root_table, net)->child = 0x00000080   <= cause the panic

=> should be equal to this:

__per_net_base(unix_net_table)= 0xc03c7f40

And:
==========
$ objdump -D -j .data.pernet net/unix/unix.ko

net/unix/unix.ko: file format elf32-i386

Disassembly of section .data.pernet:

00000000 <per_net__sysctl_unix_max_dgram_qlen>:
   0:   0a 00   or   (%eax),%al
        ...

00000020 <per_net__unix_root_table>:
  20:   03 00   add   (%eax),%eax
[...]

00000080 <per_net__unix_net_table>:
  80:   04 00   add $0x0,%al
  82:   00 00   add   %al,(%eax)
...
==========

The init of per_net(unix_root_table, net)->child is done with the offset of
__per_net_base(unix_net_table) in the module, and never translated to its
virtual address.

It looks like the module loader translates/relocates correctly the address of
__per_net_base(unix_net_table), but the not the address (value of)
per_net(unix_root_table, net)->child ???

But if I do:
$ objdump -r -j .data.pernet net/unix/unix.ko

net/unix/unix.ko: file format elf32-i386

RELOCATION RECORDS FOR [.data.pernet]:
OFFSET   TYPE   VALUE
00000024 R_386_32   .rodata.str1.1
00000034 R_386_32   .data.pernet <== per_net__unix_root_table->child (*)
00000070 R_386_32   .data.pernet
00000084 R_386_32   .rodata.str1.1
00000094 R_386_32   .data.pernet <== per_net__unix_net_table>->child (?)
000000e4 R_386_32   .rodata.str1.1
000000e8 R_386_32   per_net__sysctl_unix_max_dgram_qlen
000000f8 R_386_32   proc_dointvec

(*) If I well read/understand, this relocation entry should correspond to
per_net__unix_root_table->child and should be translated at load time, but it
isn't ?

>
> Thanks. I will go back and look but I don't plan on back porting
> anything for 2.6.20. I'm lazy and do not have enough hours in the
> day. :)

No problem.
Thanks.

--
Pierre

_____
Containers mailing list
Containers@lists.linux-foundation.org
https://lists.linux-foundation.org/mailman/listinfo/containers