



index : kernel/git/stable/linux.git

master



switch

Linux kernel stable tree

Stable Group

[about](#) [summary](#) [refs](#) [log](#) [tree](#) [commit](#) [diff](#) [stats](#)

log msg



search

author Shang XiaoJing <shangxiaojing@huawei.com> 2022-11-18 10:15:34 +0900
committer Masami Hiramatsu (Google) <mhiramat@kernel.org> 2022-11-18 10:15:34 +0900
commit [22ea4ca9631eb137e64e5ab899e9c89cb6670959](#) (patch)
tree [83f1d765e7010b1b1b6f07bc9cf2fef678dbbbb2](#)
parent [e0d75267f59d7084e0468bd68beeb1bf9c71d7c0](#) (diff)
download [linux-22ea4ca9631eb137e64e5ab899e9c89cb6670959.tar.gz](#)

diff options

context:

3

space:

include

mode:

unified

tracing: kprobe: Fix potential null-ptr-deref on trace_array in kprobe_event_gen_test_exit()

When test_gen_kprobe_cmd() failed after kprobe_event_gen_cmd_end(), it will goto delete, which will call kprobe_event_delete() and release the corresponding resource. However, the trace_array in gen_kretprobe_test will point to the invalid resource. Set gen_kretprobe_test to NULL after called kprobe_event_delete() to prevent null-ptr-deref.

BUG: kernel NULL pointer dereference, address: 0000000000000070

PGD 0 P4D 0

Oops: 0000 [#1] SMP PTI

CPU: 0 PID: 246 Comm: modprobe Tainted: G W

6.1.0-rc1-00174-g9522dc5c87da-dirty #248

Hardware name: QEMU Standard PC (i440FX + PIIX, 1996), BIOS

rel-1.15.0-0-g2dd4b9b3f840-prebuilt.qemu.org 04/01/2014

RIP: 0010:__ftrace_set_clr_event_nolock+0x53/0x1b0

Code: e8 82 26 fc ff 49 8b 1e c7 44 24 0c ea ff ff ff 49 39 de 0f 84 3c

01 00 00 c7 44 24 18 00 00 00 00 e8 61 26 fc ff 48 8b 6b 10 <44> 8b 65

70 4c 8b 6d 18 41 f7 c4 00 02 00 00 75 2f

RSP: 0018:ffffc9000159fe00 EFLAGS: 00010293

RAX: 0000000000000000 RBX: ffff88810971d268 RCX: 0000000000000000

RDX: ffff8881080be600 RSI: ffffffff811b48ff RDI: ffff88810971d058

RBP: 0000000000000000 R08: 0000000000000000 R09: 0000000000000001

R10: fffffc9000159fe58 R11: 0000000000000001 R12: ffffffff80001064

R13: ffffffff8000106c R14: ffff88810971d238 R15: 0000000000000000

FS: 00007f89eef6540(0000) GS: ffff88813b600000(0000)

knlgS:0000000000000000

CS: 0010 DS: 0000 ES: 0000 CR0: 0000000080050033

CR2: 0000000000000070 CR3: 000000010599e004 CR4: 000000000330ef0

DR0: 0000000000000000 DR1: 0000000000000000 DR2: 0000000000000000

DR3: 0000000000000000 DR6: 00000000fffe0ff0 DR7: 0000000000000400

Call Trace:

<TASK>

__ftrace_set_clr_event+0x3e/0x60

trace_array_set_clr_event+0x35/0x50

? 0xffffffff80000000

kprobe_event_gen_test_exit+0xcd/0x10b [kprobe_event_gen_test]

__x64_sys_delete_module+0x206/0x380

? lockdep_hardirqs_on_prepare+0xd8/0x190

? syscall_enter_from_user_mode+0x1c/0x50

do_syscall_64+0x3f/0x90

entry_SYSCALL_64_after_hwframe+0x63/0xcd

RIP: 0033:0x7f89eeb061b7

Link: <https://lore.kernel.org/all/20221108015130.28326-3-shangxiaojing@huawei.com/>

Fixes: 64836248dda2 ("tracing: Add kprobe event command generation test module")

Signed-off-by: Shang XiaoJing <shangxiaojing@huawei.com>

Cc: stable@vger.kernel.org

Acked-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>

Signed-off-by: Masami Hiramatsu (Google) <mhiramat@kernel.org>

Diffstat

```
-rw-r--r-- kernel/trace/kprobe_event_gen_test.c 4
```

1 files changed, 4 insertions, 0 deletions

diff --git a/kernel/trace/kprobe_event_gen_test.c b/kernel/trace/kprobe_event_gen_test.c

index 1c98fafcf33347..c736487fc0e48a 100644

--- a/kernel/trace/kprobe_event_gen_test.c

+++ b/kernel/trace/kprobe_event_gen_test.c

@@ -143,6 +143,8 @@ static int __init test_gen_kprobe_cmd(void)

kfree(buf);

return ret;

delete:

+ if (trace_event_file_is_valid(gen_kprobe_test))

+ gen_kprobe_test = NULL;

/* We got an error after creating the event, delete it */

ret = kprobe_event_delete("gen_kprobe_test");

goto out;

@@ -206,6 +208,8 @@ static int __init test_gen_kretprobe_cmd(void)

kfree(buf);

return ret;

delete:

+ if (trace_event_file_is_valid(gen_kretprobe_test))

+ gen_kretprobe_test = NULL;

/* We got an error after creating the event, delete it */

ret = kprobe_event_delete("gen_kretprobe_test");

goto out;