On Thu, 2011-11-03 at 01:40 -0300, Andrew Vagin wrote:
> process_arg(...) can't parse "__entry->prev_state & (TASK_STATE_MAX-1)",
> because a complicated argument should be within brackets.

No it is fine, the userspace tool is broken.

> > Without this patch "perf report" prints following errors:
> > $ ./perf record -ag -e sched:sched_switch
> > ...
> > $ ./perf report
> > Warning: Error: expected type 5 but read 4
> > Warning: Error: expected type 4 but read 0
> > Fatal: bad op token {
> > > Signed-off-by: Andrew Vagin <avagin@openvz.org>

NACK!

This is a perf userspace bug, not a kernel one. Please fix the userspace
tool instead.

Note, the new version of libparsevent handles this case without issue.
Perf just needs to be updated.

-- Steve

> ---
> include/trace/events/sched.h | 2 +- 1 files changed, 1 insertions(+), 1 deletions(-)
> > diff --git a/include/trace/events/sched.h b/include/trace/events/sched.h
> > index 959ff18..1838237 100644
> > --- a/include/trace/events/sched.h
> > +++ b/include/trace/events/sched.h
> > @@ -140,7 +140,7 @@ TRACE_EVENT(sched_switch,
> > TP_printk("\n> > next_comm=%s next_pid=%d next_prio=%d",
> > __entry->prev_comm, __entry->prev_pid, __entry->prev_prio,}
> > __entry->prev_state & (TASK_STATE_MAX-1) ?
> > - __print_flags(__entry->prev_state & (TASK_STATE_MAX-1), "|",
> > + __print_flags(__entry->prev_state & (TASK_STATE_MAX-1), "|",
> > { 1, "S" }, { 2, "D" }, { 4, "T" }, { 8, "t" },
> > { 16, "Z" }, { 32, "X" }, { 64, "x" },
> { 128, "W" } : "R",

Subject: Re: [PATCH] event: fix TP_printk() argument in sched_switch
Posted by avagin on Thu, 03 Nov 2011 12:57:10 GMT
View Forum Message <> Reply to Message

> NACK!
> This is a perf userspace bug, not a kernel one. Please fix the userspace
> tool instead.
> Note, the new version of libparsevent handles this case without issue.
> Perf just needs to be updated.
I don't understand. I've got
git://git.kernel.org/pub/scm/linux/kernel/git/rostedt/trace-cmd.git
and it reports the same error. Where am I wrong?

# ./trace-cmd report 2> log

<idle>-0  [001] 1516333.292126: sched_switch: [FAILED TO
PARSE] prev_comm=kworker/0:0 prev_pid=0 prev_prio=120 prev_state=0x0
next_comm=trace-cmd next_pid=2900 next_prio=120

# cat log
trace-cmd: No such file or directory
   Error: expected type 5 but read 4
   Error: expected type 4 but read 0
   failed to read event print fmt for sched_switch
trace-cmd: Received SIGINT

Subject: Re: [PATCH] event: fix TP_printk() argument in sched_switch
Posted by Steven Rostedt on Thu, 03 Nov 2011 14:02:11 GMT
View Forum Message <> Reply to Message

On Thu, 2011-11-03 at 16:57 +0400, Andrew Vagin wrote:
>> NACK!
>>
>> This is a perf userspace bug, not a kernel one. Please fix the userspace
>> tool instead.
>>
>> Note, the new version of libparsevent handles this case without issue.
>> Perf just needs to be updated.
>> I don't understand. I've got
git://git.kernel.org/pub/scm/linux/kernel/git/rostedt/trace-cmd.git
>> and it reports the same error. Where am I wrong?
Ah, I tested on a kernel without the updated format field.

But I still stand that the tool is broken and not the kernel.

I'll update both trace-cmd and perf to handle this breakage. Sad that I still need to update two locations :(

-- Steve

Subject: Re: [PATCH] event: fix TP_printk() argument in sched_switch
Posted by Steven Rostedt on Thu, 03 Nov 2011 14:15:36 GMT
View Forum Message <> Reply to Message

On Thu, 2011-11-03 at 16:57 +0400, Andrew Vagin wrote:
> > NACK!
> >
> > This is a perf userspace bug, not a kernel one. Please fix the userspace
> > tool instead.
> >
> > Note, the new version of libparserevent handles this case without issue.
> > Perf just needs to be updated.
> > I don't understand. I've got
> > git://git.kernel.org/pub/scm/linux/kernel/git/rostedt/trace-cmd.git
> > and it reports the same error. Where am I wrong?
> >
> > # ./trace-cmd report 2> log
> >
> > <idle>-0   [001] 1516333.292126: sched_switch:     [FAILED TO
> > PARSE] prev_comm=kworker/0:0 prev_pid=0 prev_prio=120 prev_state=0x0
> > next_comm=trace-cmd next_pid=2900 next_prio=120
> >
> # cat log
> trace-cmd: No such file or directory
> Error: expected type 5 but read 4
> Error: expected type 4 but read 0
> failed to read event print fmt for sched_switch
> trace-cmd: Received SIGINT

Add the below patch to trace-cmd and see if it fixes the issue. I'll start working on something that fixes perf too.

Thanks,

-- Steve

diff --git a/parse-events.c b/parse-events.c
index 2dbd47b..8ed018d 100644
--- a/parse-events.c
+++ b/parse-events.c
@@ -2167,6 +2167,10 @@ process_flags(struct event_format *event, struct print_arg *arg, char **tok)
    field = alloc_arg();

    type = process_arg(event, field, &token);
+    /* Handle operations in the first argument */
+    while (type == EVENT_OP) {
+        type = process_op(event, field, &token);
+    }
    if (test_type_token(type, token, EVENT_DELIM, ","))
        goto out_free;
    free_token(token);
$ ./perf report
   Warning: Error: expected type 5 but read 4
   Warning: Error: expected type 4 but read 0
   Fatal: bad op token {
>
   Signed-off-by: Andrew Vagin <avagin@openvz.org>
>
   NACK!
>
   This is a perf userspace bug, not a kernel one. Please fix the userspace tool instead.
>
   Note, the new version of libparseevent handles this case without issue.
   Perf just needs to be updated.
>
Can you try this patch on perf. It’s untested (not even compiled tested)

-- Steve

diff --git a/tools/perf/util/trace-event-parse.c b/tools/perf/util/trace-event-parse.c
index 0a7ed5b..6c164dc 100644
--- a/tools/perf/util/trace-event-parse.c
+++ b/tools/perf/util/trace-event-parse.c
@@ -1537,6 +1537,8 @@ process_flags(struct event *event, struct print_arg *arg, char **tok)
    +while (type == EVENT_OP)
    +type = process_op(event, field, &token);
    if (test_type_token(type, token, EVENT_DELIM, ","))
    goto out_free;

Subject: Re: [PATCH] event: fix TP_printk() argument in sched_switch
Posted by avagin on Thu, 03 Nov 2011 21:36:26 GMT
View Forum Message <> Reply to Message

Hi Steve,
>
Can you try this patch on perf. It’s untested (not even compiled tested)
It looks like it works now. Thanks.
>
-- Steve
>
diff --git a/tools/perf/util/trace-event-parse.c b/tools/perf/util/trace-event-parse.c
index 0a7ed5b..6c164dc 100644
process_flags(struct event *event, struct print_arg *arg, char **tok)
    field = malloc_or_die(sizeof(*field));
>
    type = process_arg(event, field,&token);
+while (type == EVENT_OP)
+    type = process_op(event, field,&token);
    if (test_type_token(type, token, EVENT_DELIM, ",",))
        goto out_free;
>
>
--
To unsubscribe from this list: send the line "unsubscribe linux-kernel" in
the body of a message to majordomo@vger.kernel.org
Please read the FAQ at http://www.tux.org/lkml/