Handling User Page Faults from Kernel Tracers

Mathieu Desnoyers, EfficiOS

Tracing & Perf Events Microconference
Linux Plumbers Conference 2024
September 18-20
Vienna, Austria

Use-Cases: Read From Userspace

- Allow kernel tracers to read data from userspace memory:
  - System call entry/exit tracepoints,
  - User events,
  - Stack traces (stack walker backtrace).

Use-Cases: Actions to Perform

- Kernel tracers (e.g. Ftrace, perf, eBPF, LTTng) can use this data for:
  - Copying it to a ring buffer,
  - Perform on-line filtering based on input,
  - Index counters within maps,
  - Determine aggregation quantity for counter maps,
  - Emit trigger notifications with field capture.

Current Limitations

- Specific scenarios lead to **always** unavailable data due to disabled preemption, e.g.:
  - Read a string from program data when a system call is issued immediately after program execve(2) (openat(2) pathname argument).
  - Read any data from program/library data sections which are not yet faulted-in.

Proposal: Handle page faults from system call tracepoints

- Kernel tracepoints currently disable preemption around tracer callback invocation for registration list synchronization,
- Modify system call tracepoints to use Tasks Trace RCU instead, which allows handling page faults.

Tracepoints Patch Series

- [PATCH 0/8] tracing: Allow system call tracepoints to handle page faults
- https://lore.kernel.org/lkml/20240909201652.319406-1-mathieu.desnoyers@efficios.com/

How tracers can take to handle page faults

- Fault all user pages in preparation step before entering preempt-off critical section,
- Copy all user-space data to an area of kernel memory, before an eventual copy to per-CPU ring buffer with preemption disabled,
- Modify data structures (e.g. ring buffers) to allow access with preemption enabled.

Usefulness for seccomp

- Seccomp would benefit from having stable userspace inputs to system calls,
- This could be performed with the copy-to-kernel memory approach,
- This would however require system call implementation to read from kernel copy rather than to re-read userspace data.