author	Qu Huang <qu.huang@linux.dev>	2023-02-21 11:35:16 +0000
committer	Greg Kroah-Hartman <gregkh@linuxfoundation.org>	2023-03-22 13:33:54 +0100
commit	d9923e7214a870b312bf61f6a89c7554d0966985 (patch)	
tree	53cd2b68406ef601fa01ca1c7dd0eaf64d3ddc80	
parent	94fd091576b12540924f6316ebc0678e84cb2800 (diff)	
download	linux-d9923e7214a870b312bf61f6a89c7554d0966985.tar.gz	

drm/amdkfd: Fix an illegal memory access

```
[ Upstream commit 4fc8fff378b2f2039f2a666d9f8c570f4e58352c ]

In the kfd_wait_on_events() function, the kfd_event_waiter structure is
allocated by alloc_event_waiters(), but the event field of the waiter
structure is not initialized; When copy_from_user() fails in the
kfd_wait_on_events() function, it will enter exception handling to
release the previously allocated memory of the waiter structure;
Due to the event field of the waiters structure being accessed
in the free_waiters() function, this results in illegal memory access
and system crash, here is the crash log:

localhost kernel: RIP: 0010:native_queued_spin_lock_slowpath+0x185/0x1e0
localhost kernel: RSP: 0018:ffffaa53c362bd60 EFLAGS: 00010082
localhost kernel: RAX: ff3d3d6bff4007cb RBX: 0000000000000282 RCX: 00000000002c0000
localhost kernel: RDX: ffff9e855eeacb80 RSI: 000000000000279c RDI: ffffe7088f6a21d0
localhost kernel: RBP: ffffe7088f6a21d0 R08: 00000000002c0000 R09: ffffaa53c362be64
localhost kernel: R10: ffffaa53c362bbd8 R11: 0000000000000001 R12: 0000000000000002
localhost kernel: R13: ffff9e7ead15d600 R14: 0000000000000000 R15: ffff9e7ead15d698
localhost kernel: FS:  0000152a3d111700(0000) GS:ffff9e855ee80000(0000) knlGS:0000000000000000
localhost kernel: CS:  0010 DS: 0000 ES: 0000 CR0: 0000000080050033
localhost kernel: CR2: 0000152938000010 CR3: 000000044d7a4000 CR4: 00000000003506e0
localhost kernel: Call Trace:
localhost kernel: _raw_spin_lock_irqsave+0x30/0x40
localhost kernel: remove_wait_queue+0x12/0x50
localhost kernel: kfd_wait_on_events+0x1b6/0x490 [hydcu]
localhost kernel: ? ftrace_graph_caller+0xa0/0xa0
localhost kernel: kfd_ioctl+0x38c/0x4a0 [hydcu]
localhost kernel: ? kfd_ioctl_set_trap_handler+0x70/0x70 [hydcu]
localhost kernel: ? kfd_ioctl_create_queue+0x5a0/0x5a0 [hydcu]
localhost kernel: ? ftrace_graph_caller+0xa0/0xa0
localhost kernel: __x64_sys_ioctl+0x8e/0xd0
localhost kernel: ? syscall_trace_enter.isra.18+0x143/0x1b0
localhost kernel: do_syscall_64+0x33/0x80
localhost kernel: entry_SYSCALL_64_after_hwframe+0x44/0xa9
localhost kernel: RIP: 0033:0x152a4dff68d7

Allocate the structure with kcalloc, and remove redundant 0-initialization
and a redundant loop condition check.

Signed-off-by: Qu Huang <qu.huang@linux.dev>
Signed-off-by: Felix Kuehling <Felix.Kuehling@amd.com>
```

Reviewed-by: Felix Kuehling <Felix.Kuehling@amd.com>
Signed-off-by: Alex Deucher <alexander.deucher@amd.com>
Signed-off-by: Sasha Levin <sashal@kernel.org>

Diffstat

-rw-r--r-- drivers/gpu/drm/amd/amdkfd/kfd_events.c 9

1 files changed, 3 insertions, 6 deletions

```
diff --git a/drivers/gpu/drm/amd/amdkfd/kfd_events.c b/drivers/gpu/drm/amd/amdkfd/kfd_events.c
index 729d26d648af3b..2880ed96ac2e33 100644
--- a/drivers/gpu/drm/amd/amdkfd/kfd_events.c
+++ b/drivers/gpu/drm/amd/amdkfd/kfd_events.c
@@ -778,16 +778,13 @@ static struct kfd_event_waiter *alloc_event_waiters(uint32_t num_events)
        struct kfd_event_waiter *event_waiters;
        uint32_t i;

-       event_waiters = kmalloc_array(num_events,
-                                     sizeof(struct kfd_event_waiter),
-                                     GFP_KERNEL);
+       event_waiters = kcalloc(num_events, sizeof(struct kfd_event_waiter),
+                               GFP_KERNEL);
        if (!event_waiters)
                return NULL;

-       for (i = 0; (event_waiters) && (i < num_events) ; i++) {
+       for (i = 0; i < num_events; i++)
                init_wait(&event_waiters[i].wait);
-               event_waiters[i].activated = false;
-       }

        return event_waiters;
 }
```