author	Octavian Purdila <tavip@google.com>	2025-04-07 13:24:08 -0700
committer	Greg Kroah-Hartman <gregkh@linuxfoundation.org>	2025-04-20 10:17:32 +0200
commit	5e5e1fcc1b8ed57f902c424c5d9b328a3a19073d (patch)	
tree	08b5a9f434c6704e37765f538bc87b2128d7fd84	
parent	64022a26a3866acd4da7313205bb4756d44c2b53 (diff)	
download	linux-5e5e1fcc1b8ed57f902c424c5d9b328a3a19073d.tar.gz	

net_sched: sch_sfq: move the limit validation

```
[ Upstream commit b3bf8f63e6179076b57c9de660c9f80b5abefe70 ]

It is not sufficient to directly validate the limit on the data that
the user passes as it can be updated based on how the other parameters
are changed.

Move the check at the end of the configuration update process to also
catch scenarios where the limit is indirectly updated, for example
with the following configurations:

tc qdisc add dev dummy0 handle 1: root sfq limit 2 flows 1 depth 1
tc qdisc add dev dummy0 handle 1: root sfq limit 2 flows 1 divisor 1

This fixes the following syzkaller reported crash:

------------[ cut here ]------------
UBSAN: array-index-out-of-bounds in net/sched/sch_sfq.c:203:6
index 65535 is out of range for type 'struct sfq_head[128]'
CPU: 1 UID: 0 PID: 3037 Comm: syz.2.16 Not tainted 6.14.0-rc2-syzkaller #0
Hardware name: Google Google Compute Engine/Google Compute Engine, BIOS Google 12/27/2024
Call Trace:
 <TASK>
 __dump_stack lib/dump_stack.c:94 [inline]
 dump_stack_lvl+0x201/0x300 lib/dump_stack.c:120
 ubsan_epilogue lib/ubsan.c:231 [inline]
 __ubsan_handle_out_of_bounds+0xf5/0x120 lib/ubsan.c:429
 sfq_link net/sched/sch_sfq.c:203 [inline]
 sfq_dec+0x53c/0x610 net/sched/sch_sfq.c:231
 sfq_dequeue+0x34e/0x8c0 net/sched/sch_sfq.c:493
 sfq_reset+0x17/0x60 net/sched/sch_sfq.c:518
 qdisc_reset+0x12e/0x600 net/sched/sch_generic.c:1035
 tbf_reset+0x41/0x110 net/sched/sch_tbf.c:339
 qdisc_reset+0x12e/0x600 net/sched/sch_generic.c:1035
 dev_reset_queue+0x100/0x1b0 net/sched/sch_generic.c:1311
 netdev_for_each_tx_queue include/linux/netdevice.h:2590 [inline]
 dev_deactivate_many+0x7e5/0xe70 net/sched/sch_generic.c:1375

Reported-by: syzbot <syzkaller@googlegroups.com>
Fixes: 10685681bafc ("net_sched: sch_sfq: don't allow 1 packet limit")
Signed-off-by: Octavian Purdila <tavip@google.com>
Acked-by: Cong Wang <xiyou.wangcong@gmail.com>
```

Signed-off-by: David S. Miller <davem@davemloft.net>
Signed-off-by: Sasha Levin <sashal@kernel.org>

Diffstat

-rw-r--r-- net/sched/sch_sfq.c 10

1 files changed, 6 insertions, 4 deletions

```
diff --git a/net/sched/sch_sfq.c b/net/sched/sch_sfq.c
index 7714ae94e05217..58b42dcf8f2013 100644
--- a/net/sched/sch_sfq.c
+++ b/net/sched/sch_sfq.c
@@ -661,10 +661,6 @@ static int sfq_change(struct Qdisc *sch, struct nlattr *opt,
                if (!p)
                        return -ENOMEM;
        }
-       if (ctl->limit == 1) {
-               NL_SET_ERR_MSG_MOD(extack, "invalid limit");
-               return -EINVAL;
-       }

        sch_tree_lock(sch);

@@ -705,6 +701,12 @@ static int sfq_change(struct Qdisc *sch, struct nlattr *opt,
                limit = min_t(u32, ctl->limit, maxdepth * maxflows);
                maxflows = min_t(u32, maxflows, limit);
        }
+       if (limit == 1) {
+               sch_tree_unlock(sch);
+               kfree(p);
+               NL_SET_ERR_MSG_MOD(extack, "invalid limit");
+               return -EINVAL;
+       }

        /* commit configuration */
        q->limit = limit;
```