



author Cong Wang <xiyou.wangcong@gmail.com> 2025-04-17 11:47:30 -0700
committer Jakub Kicinski <kuba@kernel.org> 2025-04-23 17:16:50 -0700
commit [3df275ef0a6ae181e8428a6589ef5d5231e58b5c](#) (patch)
tree [802ae4ab4417f65d95d77e7a2c49f81147403673](#)
parent [d861a5dbb960c5619ea78ae9f121b9b460685429](#) (diff)
download [linux-3df275ef0a6ae181e8428a6589ef5d5231e58b5c.tar.gz](#)

diff options

context: ▼
space: ▼
mode: ▼

net_sched: hfsc: Fix a UAF vulnerability in class handling

This patch fixes a Use-After-Free vulnerability in the HFSC qdisc class handling. The issue occurs due to a time-of-check/time-of-use condition in `hfsc_change_class()` when working with certain child qdiscs like `netem` or `codel`.

The vulnerability works as follows:

1. `hfsc_change_class()` checks if a class has packets (`q qlen != 0`)
2. It then calls `qdisc_peek_len()`, which for certain qdiscs (e.g., `codel`, `netem`) might drop packets and empty the queue
3. The code continues assuming the queue is still non-empty, adding the class to `vttree`
4. This breaks HFSC scheduler assumptions that only non-empty classes are in `vttree`
5. Later, when the class is destroyed, this can lead to a Use-After-Free

The fix adds a second queue length check after `qdisc_peek_len()` to verify the queue wasn't emptied.

Fixes: [21f4d5cc25ec](#) ("net_sched/hfsc: fix curve activation in `hfsc_change_class()`")

Reported-by: Gerrard Tai <gerrard.tai@starlabs.sg>

Reviewed-by: Konstantin Khlebnikov <koct9i@gmail.com>

Signed-off-by: Cong Wang <xiyou.wangcong@gmail.com>

Reviewed-by: Jamal Hadi Salim <jhs@mojatatu.com>

Link: <https://patch.msgid.link/20250417184732.943057-2-xiyou.wangcong@gmail.com>

Signed-off-by: Jakub Kicinski <kuba@kernel.org>

Diffstat

```
-rw-r--r-- net/sched/sch_hfsc.c 9
```

1 files changed, 7 insertions, 2 deletions

diff --git a/net/sched/sch_hfsc.c b/net/sched/sch_hfsc.c

index ce5045eea06568..b368ac0595d5c0 100644

--- a/net/sched/sch_hfsc.c

+++ b/net/sched/sch_hfsc.c

@@ -961,6 +961,7 @@ hfsc_change_class(struct Qdisc *sch, u32 classid, u32 parentid,

```
    if (cl != NULL) {  
        int old_flags;  
+       int len = 0;
```

```
    if (parentid) {
```

```

        if (cl->cl_parent &&
@@ -991,9 +992,13 @@ hfsc_change_class(struct Qdisc *sch, u32 classid, u32 parentid,
            if (usc != NULL)
                hfsc_change_usc(cl, usc, cur_time);

+
+        if (cl->qdisc->q.qlen != 0)
+            len = qdisc_peek_len(cl->qdisc);
+        /* Check queue length again since some qdisc implementations
+        * (e.g., netem/codel) might empty the queue during the peek
+        * operation.
+        */
        if (cl->qdisc->q.qlen != 0) {
-            int len = qdisc_peek_len(cl->qdisc);
-
            if (cl->cl_flags & HFSC_RSC) {
                if (old_flags & HFSC_RSC)
                    update_ed(cl, len);

```