author	Vikash Garodia <quic_vgarodia@quicinc.com>	2025-02-20 22:50:10 +0530
committer	Greg Kroah-Hartman <gregkh@linuxfoundation.org>	2025-05-02 07:39:13 +0200
commit	1b86c1917e16bafbbb08ab90baaff533aa36c62d (patch)	
tree	3551a9f8d62e58bf965486d49f7e428152789cb2	
parent	4dd109038d513b92d4d33524ffc89ba32e02ba48 (diff)	
download	linux-1b86c1917e16bafbbb08ab90baaff533aa36c62d.tar.gz	

diff options

context: 3

space: include

mode: unified

media: venus: hfi: add check to handle incorrect queue size

```
commit 69baf245b23e20efda0079238b27fc63ecf13de1 upstream.

qsize represents size of shared queued between driver and video
firmware. Firmware can modify this value to an invalid large value. In
such situation, empty_space will be bigger than the space actually
available. Since new_wr_idx is not checked, so the following code will
result in an OOB write.
...
qsize = qhdr->q_size

if (wr_idx >= rd_idx)
 empty_space = qsize - (wr_idx - rd_idx)
....
if (new_wr_idx < qsize) {
 memcpy(wr_ptr, packet, dwords << 2) --> OOB write

Add check to ensure qsize is within the allocated size while
reading and writing packets into the queue.

Cc: stable@vger.kernel.org
Fixes: d96d3f30c0f2 ("[media] media: venus: hfi: add Venus HFI files")
Reviewed-by: Bryan O'Donoghue <bryan.odonoghue@linaro.org>
Signed-off-by: Vikash Garodia <quic_vgarodia@quicinc.com>
Signed-off-by: Hans Verkuil <hverkuil@xs4all.nl>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>
```

Diffstat

-rw-r--r-- drivers/media/platform/qcom/venus/hfi_venus.c 6

1 files changed, 6 insertions, 0 deletions

```
diff --git a/drivers/media/platform/qcom/venus/hfi_venus.c b/drivers/media/platform/qcom/venus/hfi_venus.c
index 0b6cf86004fd82..1b37d77bf99856 100644
--- a/drivers/media/platform/qcom/venus/hfi_venus.c
+++ b/drivers/media/platform/qcom/venus/hfi_venus.c
@@ -188,6 +188,9 @@ static int venus_write_queue(struct venus_hfi_device *hdev,
        /* ensure rd/wr indices's are read from memory */
        rmb();

+       if (qsize > IFACEQ_QUEUE_SIZE / 4)
+               return -EINVAL;
+
        if (wr_idx >= rd_idx)
                empty_space = qsize - (wr_idx - rd_idx);
        else
@@ -256,6 +259,9 @@ static int venus_read_queue(struct venus_hfi_device *hdev,
        wr_idx = qhdr->write_idx;
        qsize = qhdr->q_size;
```

```
+       if (qsize > IFACEQ_QUEUE_SIZE / 4)
+               return -EINVAL;
+
        /* make sure data is valid before using it */
        rmb();
```