author	Pavel Begunkov <asml.silence@gmail.com>	2025-03-27 09:57:27 +0000
committer	Greg Kroah-Hartman <gregkh@linuxfoundation.org>	2025-04-20 10:15:40 +0200
commit	b7c6d081c19a5e11bbd77bb97a62cff2b6b21cb5 (patch)	
tree	4936eb5bc74843217bd5d321948cf02d41c5332b	
parent	0828d6e9add67ce4f893c5fbb31ec6c23f34426c (diff)	
download	linux-b7c6d081c19a5e11bbd77bb97a62cff2b6b21cb5.tar.gz	

io_uring/net: fix io_req_post_cqe abuse by send bundle

commit 6889ae1b4df1579bcdffef023e2ea9a982565dff upstream.

```
[  114.987980][ T5313] WARNING: CPU: 6 PID: 5313 at io_uring/io_uring.c:872 io_req_post_cqe+0x12e/0x4f0
[  114.991597][ T5313] RIP: 0010:io_req_post_cqe+0x12e/0x4f0
[  115.001880][ T5313] Call Trace:
[  115.002222][ T5313]  <TASK>
[  115.007813][ T5313]  io_send+0x4fe/0x10f0
[  115.009317][ T5313]  io_issue_sqe+0x1a6/0x1740
[  115.012094][ T5313]  io_wq_submit_work+0x38b/0xed0
[  115.013223][ T5313]  io_worker_handle_work+0x62a/0x1600
[  115.013876][ T5313]  io_wq_worker+0x34f/0xdf0
```

As the comment states, io_req_post_cqe() should only be used by multishot requests, i.e. REQ_F_APOLL_MULTISHOT, which bundled sends are not. Add a flag signifying whether a request wants to post multiple CQEs. Eventually REQ_F_APOLL_MULTISHOT should imply the new flag, but that's left out for simplicity.

Cc: stable@vger.kernel.org
Fixes: a05d1f625c7aa ("io_uring/net: support bundles for send")
Signed-off-by: Pavel Begunkov <asml.silence@gmail.com>
Link: https://lore.kernel.org/r/8b611dbb54d1cd47a88681f5d38c84d0c02bc563.1743067183.git.asml.silence@gmail.com
Signed-off-by: Jens Axboe <axboe@kernel.dk>
Signed-off-by: Greg Kroah-Hartman <gregkh@linuxfoundation.org>

Diffstat

-rw-r--r--	include/linux/io_uring_types.h	3
-rw-r--r--	io_uring/io_uring.c	4
-rw-r--r--	io_uring/net.c	1

3 files changed, 6 insertions, 2 deletions

```
diff --git a/include/linux/io_uring_types.h b/include/linux/io_uring_types.h
index 4b9ba523978d20..5ce332fc6ff507 100644
--- a/include/linux/io_uring_types.h
+++ b/include/linux/io_uring_types.h
@@ -457,6 +457,7 @@ enum {
        REQ_F_SKIP_LINK_CQES_BIT,
        REQ_F_SINGLE_POLL_BIT,
        REQ_F_DOUBLE_POLL_BIT,
+       REQ_F_MULTISHOT_BIT,
        REQ_F_APOLL_MULTISHOT_BIT,
        REQ_F_CLEAR_POLLIN_BIT,
        REQ_F_HASH_LOCKED_BIT,
@@ -530,6 +531,8 @@ enum {
        REQ_F_SINGLE_POLL       = IO_REQ_FLAG(REQ_F_SINGLE_POLL_BIT),
        /* double poll may active */
        REQ_F_DOUBLE_POLL       = IO_REQ_FLAG(REQ_F_DOUBLE_POLL_BIT),
+       /* request posts multiple completions, should be set at prep time */
+       REQ_F_MULTISHOT         = IO_REQ_FLAG(REQ_F_MULTISHOT_BIT),
```

```
        /* fast poll multishot mode */
        REQ_F_APOLL_MULTISHOT   = IO_REQ_FLAG(REQ_F_APOLL_MULTISHOT_BIT),
        /* recvmsg special flag, clear EPOLLIN */
```

```
diff --git a/io_uring/io_uring.c b/io_uring/io_uring.c
index cf28d29fffbf0e..19de7129ae0b35 100644
--- a/io_uring/io_uring.c
+++ b/io_uring/io_uring.c
@@ -1821,7 +1821,7 @@ fail:
         * Don't allow any multishot execution from io-wq. It's more restrictive
         * than necessary and also cleaner.
         */
-       if (req->flags & REQ_F_APOLL_MULTISHOT) {
+       if (req->flags & (REQ_F_MULTISHOT|REQ_F_APOLL_MULTISHOT)) {
                err = -EBADFD;
                if (!io_file_can_poll(req))
                        goto fail;
@@ -1832,7 +1832,7 @@ fail:
                                goto fail;
                        return;
                } else {
-                       req->flags &= ~REQ_F_APOLL_MULTISHOT;
+                       req->flags &= ~(REQ_F_APOLL_MULTISHOT|REQ_F_MULTISHOT);
                }
        }
```

```
diff --git a/io_uring/net.c b/io_uring/net.c
index 8ef96c4427fb79..384915d931b72c 100644
--- a/io_uring/net.c
+++ b/io_uring/net.c
@@ -435,6 +435,7 @@ int io_sendmsg_prep(struct io_kiocb *req, const struct io_uring_sqe *sqe)
                sr->msg_flags |= MSG_WAITALL;
                sr->buf_group = req->buf_index;
                req->buf_list = NULL;
+               req->flags |= REQ_F_MULTISHOT;
        }

 #ifdef CONFIG_COMPAT
```