| | | |
|---|---|---|
| author | Mike Christie <michael.christie@oracle.com> | 2022-06-27 21:23:25 -0500 |
| committer | Martin K. Petersen <martin.petersen@oracle.com> | 2022-07-07 16:38:14 -0400 |
| commit | ccd3f449052449a917a3e577d8ba0368f43b8f29 (patch) | |
| tree | 92fcfb4b40910560ad306fb846e1733b0dacd4b7 | |
| parent | fce54ed027577517df1e74b7d54dc2b1bd536887 (diff) | |
| download | linux-ccd3f449052449a917a3e577d8ba0368f43b8f29.tar.gz | |

**diff options**

context: 3

space: include

mode: unified

## scsi: target: Fix WRITE_SAME No Data Buffer crash

```
In newer version of the SBC specs, we have a NDOB bit that indicates there
is no data buffer that gets written out. If this bit is set using commands
like "sg_write_same --ndob" we will crash in target_core_iblock/file's
execute_write_same handlers when we go to access the se_cmd->t_data_sg
because its NULL.

This patch adds a check for the NDOB bit in the common WRITE SAME code
because we don't support it. And, it adds a check for zero SG elements in
each handler in case the initiator tries to send a normal WRITE SAME with
no data buffer.

Link: https://lore.kernel.org/r/20220628022325.14627-2-michael.christie@oracle.com
Reviewed-by: Christoph Hellwig <hch@lst.de>
Signed-off-by: Mike Christie <michael.christie@oracle.com>
Signed-off-by: Martin K. Petersen <martin.petersen@oracle.com>
```

**Diffstat**

| | | |
|---|---|---|
| -rw-r--r-- | drivers/target/target_core_file.c | 3 |
| -rw-r--r-- | drivers/target/target_core_iblock.c | 4 |
| -rw-r--r-- | drivers/target/target_core_sbc.c | 6 |

3 files changed, 13 insertions, 0 deletions

```
diff --git a/drivers/target/target_core_file.c b/drivers/target/target_core_file.c
index e68f1cc8ef98bd..6c8d8b051bfd5d 100644
--- a/drivers/target/target_core_file.c
+++ b/drivers/target/target_core_file.c
@@ -448,6 +448,9 @@ fd_execute_write_same(struct se_cmd *cmd)
                return TCM_LOGICAL_UNIT_COMMUNICATION_FAILURE;
        }

+       if (!cmd->t_data_nents)
+               return TCM_INVALID_CDB_FIELD;
+
        if (cmd->t_data_nents > 1 ||
            cmd->t_data_sg[0].length != cmd->se_dev->dev_attrib.block_size) {
                pr_err("WRITE_SAME: Illegal SGL t_data_nents: %u length: %u"

diff --git a/drivers/target/target_core_iblock.c b/drivers/target/target_core_iblock.c
index 378c80313a0f27..1ed9381751e648 100644
--- a/drivers/target/target_core_iblock.c
+++ b/drivers/target/target_core_iblock.c
@@ -494,6 +494,10 @@ iblock_execute_write_same(struct se_cmd *cmd)
                        " backends not supported\n");
                return TCM_LOGICAL_UNIT_COMMUNICATION_FAILURE;
```

```
        }
+
+       if (!cmd->t_data_nents)
+               return TCM_INVALID_CDB_FIELD;
+
        sg = &cmd->t_data_sg[0];

        if (cmd->t_data_nents > 1 ||
```

**diff --git a/drivers/target/target_core_sbc.c b/drivers/target/target_core_sbc.c**
**index ca1b2312d6e7b2..f6132836eb387a 100644**
**--- a/drivers/target/target_core_sbc.c**
**+++ b/drivers/target/target_core_sbc.c**

```
@@ -312,6 +312,12 @@ sbc_setup_write_same(struct se_cmd *cmd, unsigned char flags, struct sbc_ops *op
                pr_warn("WRITE SAME with ANCHOR not supported\n");
                return TCM_INVALID_CDB_FIELD;
        }
+
+       if (flags & 0x01) {
+               pr_warn("WRITE SAME with NDOB not supported\n");
+               return TCM_INVALID_CDB_FIELD;
+       }
+
        /*
         * Special case for WRITE_SAME w/ UNMAP=1 that ends up getting
         * translated into block discard requests within backend code.
```