

From: Mike Christie <michael.christie@oracle.com>
To: hch@infradead.org, martin.petersen@oracle.com,
james.bottomley@hansenpartnership.com,
linux-scsi@vger.kernel.org, target-devel@vger.kernel.org
Cc: Mike Christie <michael.christie@oracle.com>
Subject: [PATCH 1/1] scsi: target: Fix WRITE_SAME No Data Buffer crash
Date: Mon, 27 Jun 2022 21:23:25 -0500 [thread overview]
Message-ID: <20220628022325.14627-2-michael.christie@oracle.com> (raw)
In-Reply-To: <20220628022325.14627-1-michael.christie@oracle.com>

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.

Signed-off-by: Mike Christie <michael.christie@oracle.com>

```
---
drivers/target/target_core_file.c | 3 +++
drivers/target/target_core_iblock.c | 4 ++++
drivers/target/target_core_sbc.c | 6 ++++++
3 files changed, 13 insertions(+)
```

```
diff --git a/drivers/target/target_core_file.c b/drivers/target/target_core_file.c
```

```
index e68f1cc8ef98..6c8d8b051bfd 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 378c80313a0f..1ed9381751e6 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 calb2312d6e7..f6132836eb38 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;
}
```

```
+
```

This is an external index of several public inboxes,
see [mirroring instructions](#) on how to clone and mirror
all data and code used by this external index.