



author Richard Weinberger <richard@nod.at> 2025-04-18 10:02:50 +0200
committer Greg Kroah-Hartman <gregkh@linuxfoundation.org> 2025-05-02 08:01:38 +0200
commit [83c00860a37b3fcba8026cb344101f1b8af547cf](#) (patch)
tree [2ade084821bfe5f941b249091f0785425790c71d](#)
parent [65e9a6fc1e7210cef41aa162ead86b0e423659de](#) (diff)
download [linux-83c00860a37b3fcba8026cb344101f1b8af547cf.tar.gz](#)

diff options

context:

3 ▼

space:

include ▼

mode:

unified ▼

nvmet: fix out-of-bounds access in nvmet_enable_port

[Upstream commit 3d7aa0c7b4e96cd460826d932e44710cdeb3378b]

When trying to enable a port that has no transport configured yet, nvmet_enable_port() uses NVMF_TRTYPE_MAX (255) to query the transports array, causing an out-of-bounds access:

```
[ 106.058694] BUG: KASAN: global-out-of-bounds in nvmet_enable_port+0x42/0x1da
[ 106.058719] Read of size 8 at addr ffffffff89dafa58 by task ln/632
[...]
[ 106.076026] nvmet: transport type 255 not supported
```

Since commit 200adac75888, NVMF_TRTYPE_MAX is the default state as configured by nvmet_ports_make().

Avoid this by checking for NVMF_TRTYPE_MAX before proceeding.

Fixes: 200adac75888 ("nvme: Add PCI transport type")

Signed-off-by: Richard Weinberger <richard@nod.at>

Reviewed-by: Sagi Grimberg <sagi@grimberg.me>

Reviewed-by: Chaitanya Kulkarni <kch@nvidia.com>

Reviewed-by: Damien Le Moal <dlemoal@kernel.org>

Signed-off-by: Sasha Levin <sashal@kernel.org>

Diffstat

```
-rw-r--r-- drivers/nvme/target/core.c 3
```

1 files changed, 3 insertions, 0 deletions

```
diff --git a/drivers/nvme/target/core.c b/drivers/nvme/target/core.c
index 2e741696f3712e..6ccce0ee515735 100644
--- a/drivers/nvme/target/core.c
+++ b/drivers/nvme/target/core.c
@@ -324,6 +324,9 @@ int nvmet_enable_port(struct nvmet_port *port)

    lockdep_assert_held(&nvmet_config_sem);

+   if (port->disc_addr.trtype == NVMF_TRTYPE_MAX)
+       return -EINVAL;
+
    ops = nvmet_transports[port->disc_addr.trtype];
    if (!ops) {
        up_write(&nvmet_config_sem);
```