



author Chen Zhongjin <chenzhongjin@huawei.com> 2022-10-26 10:03:21 +0800
committer Greg Kroah-Hartman <gregkh@linuxfoundation.org> 2022-11-10 18:17:16 +0100
commit [4d2024b138d9f7b02ae13ee997fd3a71e9e46254](#) (patch)
tree [e2789e24357f89054227fb9abf81dcafd113b9c](#)
parent [db333ae981fb8843c383aa7dbf62cc682597d401](#) (diff)
download [linux-4d2024b138d9f7b02ae13ee997fd3a71e9e46254.tar.gz](#)

diff options

context:

3 ▼

space:

include ▼

mode:

unified ▼

net: dsa: Fix possible memory leaks in dsa_loop_init()

[Upstream commit 633efc8b3dc96f56f5a57f2a49764853a2fa3f50]

kmemleak reported memory leaks in dsa_loop_init():

kmemleak: 12 new suspected memory leaks

unreferenced object 0xfffff8880138ce000 (size 2048):

comm "modprobe", pid 390, jiffies 4295040478 (age 238.976s)

backtrace:

```
[<0000000006a94f1d5>] kcalloc_trace+0x26/0x60
[<000000000a9c44622>] phy_device_create+0x5d/0x970
[<000000000d0ee2afc>] get_phy_device+0xf3/0x2b0
[<000000000dca0c71f>] __fixed_phy_register.part.0+0x92/0x4e0
[<0000000008a834798>] fixed_phy_register+0x84/0xb0
[<00000000055223fcb>] dsa_loop_init+0xa9/0x116 [dsa_loop]
```

...

There are two reasons for memleak in dsa_loop_init().

First, fixed_phy_register() create and register phy_device:

```
fixed_phy_register()
    get_phy_device()
        phy_device_create() # freed by phy_device_free()
        phy_device_register() # freed by phy_device_remove()
```

But fixed_phy_unregister() only calls phy_device_remove().
So the memory allocated in phy_device_create() is leaked.

Second, when mdio_driver_register() fail in dsa_loop_init(),
it just returns and there is no cleanup for phydevs.

Fix the problems by catching the error of mdio_driver_register()
in dsa_loop_init(), then calling both fixed_phy_unregister() and
phy_device_free() to release phydevs.

Also add a function for phydevs cleanup to avoid duplciate.

Fixes: [98cd1552ea27](#) ("net: dsa: Mock-up driver")

Signed-off-by: Chen Zhongjin <chenzhongjin@huawei.com>

Signed-off-by: David S. Miller <davem@davemloft.net>

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

Diffstat

```
-rw-r--r-- drivers/net/dsa/dsa_loop.c 25
```

1 files changed, 18 insertions, 7 deletions

diff --git a/drivers/net/dsa/dsa_loop.c b/drivers/net/dsa/dsa_loop.c

index 263e41191c2926..0d20ebafbd03a7 100644

--- a/drivers/net/dsa/dsa_loop.c

+++ b/drivers/net/dsa/dsa_loop.c

@@ -378,6 +378,17 @@ static struct mdio_driver dsa_loop_drv = {

#define NUM_FIXED_PHYS (DSA_LOOP_NUM_PORTS - 2)

+static void dsa_loop_phydevs_unregister(void)

+{
+ unsigned int i;
+ for (i = 0; i < NUM_FIXED_PHYS; i++)
+ if (!IS_ERR(phydevs[i])) {
+ fixed_phy_unregister(phydevs[i]);
+ phy_device_free(phydevs[i]);
+ }
+}

+static int __init dsa_loop_init(void)
{

@@ -385,23 +396,23 @@ static int __init dsa_loop_init(void)
 .speed = SPEED_100,
 .duplex = DUPLEX_FULL,

};
- unsigned int i;
+ unsigned int i, ret;

for (i = 0; i < NUM_FIXED_PHYS; i++)
 phydevs[i] = fixed_phy_register(PHY_POLL, &status, NULL);

- return mdio_driver_register(&dsa_loop_drv);

+ ret = mdio_driver_register(&dsa_loop_drv);

+ if (ret)

+ dsa_loop_phydevs_unregister();

+ return ret;

}
module_init(dsa_loop_init);

static void __exit dsa_loop_exit(void)
{

- unsigned int i;

- mdio_driver_unregister(&dsa_loop_drv);

- for (i = 0; i < NUM_FIXED_PHYS; i++)

- if (!IS_ERR(phydevs[i]))

- fixed_phy_unregister(phydevs[i]);

+ dsa_loop_phydevs_unregister();

}
module_exit(dsa_loop_exit);