author	Vladimir Oltean <vladimir.oltean@nxp.com>	2025-04-15 00:28:50 +0300
committer	Greg Kroah-Hartman <gregkh@linuxfoundation.org>	2025-04-25 10:43:50 +0200
commit	b3c70dfe51f10df60db2646c08cebd24bcdc5247 (patch)	
tree	43ac5bf252cdf21e76057725cf9cb2925d78b29c	
parent	f06b5b4225b869a881e930bf24303adefb98aa78 (diff)	
download	linux-b3c70dfe51f10df60db2646c08cebd24bcdc5247.tar.gz	

net: dsa: mv88e6xxx: avoid unregistering devlink regions which were never registered

```
[ Upstream commit c84f6ce918a9e6f4996597cbc62536bbf2247c96 ]

Russell King reports that a system with mv88e6xxx dereferences a NULL
pointer when unbinding this driver:
https://lore.kernel.org/netdev/Z_lRkMlTJ1KQ0kVX@shell.armlinux.org.uk/

The crash seems to be in devlink_region_destroy(), which is not NULL
tolerant but is given a NULL devlink global region pointer.

At least on some chips, some devlink regions are conditionally registered
since the blamed commit, see mv88e6xxx_setup_devlink_regions_global():

                if (cond && !cond(chip))
                        continue;

These are MV88E6XXX_REGION_STU and MV88E6XXX_REGION_PVT. If the chip
does not have an STU or PVT, it should crash like this.

To fix the issue, avoid unregistering those regions which are NULL, i.e.
were skipped at mv88e6xxx_setup_devlink_regions_global() time.

Fixes: 836021a2d0e0 ("net: dsa: mv88e6xxx: Export cross-chip PVT as devlink region")
Tested-by: Russell King (Oracle) <rmk+kernel@armlinux.org.uk>
Signed-off-by: Vladimir Oltean <vladimir.oltean@nxp.com>
Link: https://patch.msgid.link/20250414212850.2953957-1-vladimir.oltean@nxp.com
Signed-off-by: Jakub Kicinski <kuba@kernel.org>
Signed-off-by: Sasha Levin <sashal@kernel.org>
```

Diffstat

-rw-r--r-- drivers/net/dsa/mv88e6xxx/devlink.c 3

1 files changed, 2 insertions, 1 deletions

```
diff --git a/drivers/net/dsa/mv88e6xxx/devlink.c b/drivers/net/dsa/mv88e6xxx/devlink.c
index 1266eabee0863d..2ab2eb2cb47be2 100644
--- a/drivers/net/dsa/mv88e6xxx/devlink.c
+++ b/drivers/net/dsa/mv88e6xxx/devlink.c
@@ -743,7 +743,8 @@ void mv88e6xxx_teardown_devlink_regions_global(struct dsa_switch *ds)
        int i;

        for (i = 0; i < ARRAY_SIZE(mv88e6xxx_regions); i++)
-               dsa_devlink_region_destroy(chip->regions[i]);
+               if (chip->regions[i])
+                       dsa_devlink_region_destroy(chip->regions[i]);
```

```
+                if (chip->regions[i])
+                        dsa_devlink_region_destroy(chip->regions[i]);
 }

 void mv88e6xxx_teardown_devlink_regions_port(struct dsa_switch *ds, int port)
```