



author Vladimir Oltean <vladimir.oltean@nxp.com> 2022-11-14 16:35:51 +0200
committer Jakub Kicinski <kuba@kernel.org> 2022-11-15 20:50:52 -0800
commit [4e0c19fcb8b5323716140fa82b79aa9f60e60407](#) (patch)
tree [92786bc6b6f1fd8bbcdce6d6c56a710f6a025ef](#)
parent [2929cceb2fcf0ded7182562e4888afafece82cce](#) (diff)
download [linux-4e0c19fcb8b5323716140fa82b79aa9f60e60407.tar.gz](#)

diff options

context:
space:
mode:

net: dsa: don't leak tagger-owned storage on switch driver unbind

In the initial commit [dc452a471dba](#) ("net: dsa: introduce tagger-owned storage for private and shared data"), we had a call to `tag_ops->disconnect(dst)` issued from `dsa_tree_free()`, which is called at tree teardown time.

There were problems with connecting to a switch tree as a whole, so this got reworked to connecting to individual switches within the tree. In this process, `tag_ops->disconnect(ds)` was made to be called only from `switch.c` (cross-chip notifiers emitted as a result of dynamic tag protocol changes), but the normal driver teardown code path wasn't replaced with anything.

Solve this problem by adding a function that does the opposite of `dsa_switch_setup_tag_protocol()`, which is called from the equivalent spot in `dsa_switch_teardown()`. The positioning here also ensures that we won't have any use-after-free in tagging protocol (*rcv) ops, since the teardown sequence is as follows:

```
dsa_tree_teardown
-> dsa_tree_teardown_master
    -> dsa_master_teardown
        -> unsets master->dsa_ptr, making no further packets match the
            ETH_P_XDSA packet type handler
-> dsa_tree_teardown_ports
    -> dsa_port_teardown
        -> dsa_slave_destroy
            -> unregisters DSA net devices, there is even a synchronize_net()
                in unregister_netdevice_many()
-> dsa_tree_teardown_switches
    -> dsa_switch_teardown
        -> dsa_switch_teardown_tag_protocol
            -> finally frees the tagger-owned storage
```

Fixes: [7f2973149c22](#) ("net: dsa: make tagging protocols connect to individual switches from a tree")

Signed-off-by: Vladimir Oltean <vladimir.oltean@nxp.com>

Reviewed-by: Saeed Mahameed <saeed@kernel.org>

Reviewed-by: Florian Fainelli <f.fainelli@gmail.com>

Link: <https://lore.kernel.org/r/20221114143551.1906361-1-vladimir.oltean@nxp.com>

Signed-off-by: Jakub Kicinski <kuba@kernel.org>

Diffstat

```
-rw-r--r-- net/dsa/dsa2.c 10
```

1 files changed, 10 insertions, 0 deletions

diff --git a/net/dsa/dsa2.c b/net/dsa/dsa2.c

index e504a18fc12542..5417f7b1187cbf 100644

--- a/net/dsa/dsa2.c

+++ b/net/dsa/dsa2.c

@@ -864,6 +864,14 @@ disconnect:

return err;

}

+static void dsa_switch_tearardown_tag_protocol(struct dsa_switch *ds)

+{

+ const struct dsa_device_ops *tag_ops = ds->dst->tag_ops;

+

+ if (tag_ops->disconnect)

+ tag_ops->disconnect(ds);

+}

+

static int dsa_switch_setup(struct dsa_switch *ds)

{

struct dsa_devlink_priv *dl_priv;

@@ -953,6 +961,8 @@ static void dsa_switch_tearardown(struct dsa_switch *ds)

ds->slave_mii_bus = NULL;

}

+ dsa_switch_tearardown_tag_protocol(ds);

+

if (ds->ops->teardown)

ds->ops->teardown(ds);