
Subject: [PATCH 01/16] appletalk: In notifier handlers convert the void pointer to a netdevice

Posted by [ebiederm](#) on Sat, 08 Sep 2007 21:09:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

This slightly improves code safety and clarity.

Later network namespace patches touch this code so this is a preliminary cleanup.

Signed-off-by: Eric W. Biederman <ebiederm@xmission.com>

```
net/appletalk/aarp.c | 7 ++++---
net/appletalk/ddp.c | 4 +++-
2 files changed, 7 insertions(+), 4 deletions(-)
```

```
diff --git a/net/appletalk/aarp.c b/net/appletalk/aarp.c
```

```
index 3d1655f..80b5414 100644
```

```
--- a/net/appletalk/aarp.c
```

```
+++ b/net/appletalk/aarp.c
```

```
@@ -330,15 +330,16 @@ static void aarp_expire_timeout(unsigned long unused)
static int aarp_device_event(struct notifier_block *this, unsigned long event,
void *ptr)
```

```
{
+ struct net_device *dev = ptr;
int ct;
```

```
if (event == NETDEV_DOWN) {
write_lock_bh(&aarp_lock);
```

```
for (ct = 0; ct < AARP_HASH_SIZE; ct++) {
- __aarp_expire_device(&resolved[ct], ptr);
- __aarp_expire_device(&unresolved[ct], ptr);
- __aarp_expire_device(&proxies[ct], ptr);
+ __aarp_expire_device(&resolved[ct], dev);
+ __aarp_expire_device(&unresolved[ct], dev);
+ __aarp_expire_device(&proxies[ct], dev);
}
```

```
write_unlock_bh(&aarp_lock);
```

```
diff --git a/net/appletalk/ddp.c b/net/appletalk/ddp.c
```

```
index fbdfb12..594b597 100644
```

```
--- a/net/appletalk/ddp.c
```

```
+++ b/net/appletalk/ddp.c
```

```
@@ -647,9 +647,11 @@ static inline void atalk_dev_down(struct net_device *dev)
static int ddp_device_event(struct notifier_block *this, unsigned long event,
void *ptr)
```

```
{
```

```
+ struct net_device *dev = ptr;
+
+   if (event == NETDEV_DOWN)
+     /* Discard any use of this */
-   atalk_dev_down(ptr);
+   atalk_dev_down(dev);

    return NOTIFY_DONE;
}
--
1.5.3.rc6.17.g1911
```

Containers mailing list
Containers@lists.linux-foundation.org
<https://lists.linux-foundation.org/mailman/listinfo/containers>
